

Министерство образования Республики Беларусь
БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ

Кафедра «Системы автоматизированного проектирования»

А.В. Бородуля
А.В. Василевский
В.А. Кочуров

ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ИНТЕГРИРОВАННЫХ САПР

Методическое пособие

В 3 частях

Часть 1

Минск
БНТУ
2010

УДК 681.518(075.8)

ББК 32.965я7

Б 83

Рецензенты:

В.В. Напрасников, Ю.Е. Лившиц

Бородуля, А.В.

Б 83 Техническое обеспечение интегрированных САПР: методическое пособие: в 3 ч. / А.В. Бородуля, А.В. Василевский, В.А. Кочуров. – Минск: БНТУ, 2010. – Ч. 1. – 55 с.

ISBN 978-985-525-353-3 (Ч. 1).

Рассмотрена модель конечного автомата как фундамента ЭВМ на простых примерах электроавтоматики с применением электрических контактов. Разбираются вопросы минимизации логических функций и логических схем, синхронизации цифровых устройств, информационных потоков. В конце приведено пять практических работ. Предназначено для тех, кто изучает основы автоматического управления.

УДК 681.518(075.8)

ББК 32.965я7

ISBN 978-985-525-353-3 (Ч. 1)

ISBN 978-985-525-354-0

© Бородуля А.В.,
Василевский А.В.,
Кочуров В.А., 2010
© БНТУ, 2010

СОДЕРЖАНИЕ

1. КОНЕЧНЫЙ АВТОМАТ	4
1.1. Электромагнитное реле	4
1.2. Построение схем электроавтоматики	7
1.3. Гонки и меры по их устранению	16
2. МИНИМИЗАЦИЯ ЛОГИЧЕСКИХ ФУНКЦИЙ, ПОСТРОЕНИЕ ЛОГИЧЕСКИХ СХЕМ	21
2.1. Минимизация с помощью матрицы Вейча–Карно	25
2.2. Метод Квайна–МакКласки	28
2.3. Метод свертки таблицы истинности	32
2.4. Построение и анализ работы логической схемы	35
3. СХЕМА СИНХРОНИЗАЦИИ	38
4. ПОСТРОЕНИЕ СХЕМ ПО ТАБЛИЦЕ ПЕРЕХОДОВ	40
5. СХЕМА ИНФОРМАЦИОННЫХ ПОТОКОВ	42
Лабораторная работа № 1 Минимизация логических схем и функций	44
Лабораторная работа № 2 Анализ работы логических функций и схем	45
Лабораторная работа № 3 Схема синхронизации	45
Лабораторная работа № 4 Схема информационных потоков	46
СПИСОК ИСТОЧНИКОВ	47
ПРИЛОЖЕНИЕ	48

1. КОНЕЧНЫЙ АВТОМАТ

Конечный автомат представляет собой математическую модель, которая отображает поведение дискретных устройств автоматики, в том числе электроавтоматики, ЭВМ, их программ и т.д. Ее знание полезно программистам. Основы этой модели удобно изучать, используя простые устройства электроавтоматики на основе традиционных электромагнитных реле.

1.1. Электромагнитное реле

Электромагнитное реле – очень распространенный элемент электроавтоматики, созданный более 100 лет назад. Оно надежно, имеет простую конструкцию, но работает медленно. Схематично конструкция электромагнитного реле показана на рис. 1 слева.

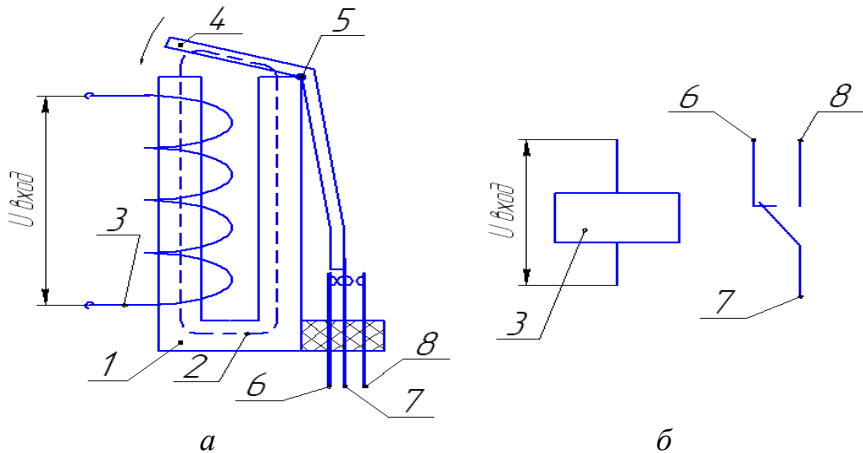


Рис. 1. Электромагнитное реле:
а – схема конструкции; б – условное обозначение

Электромагнитное реле имеет обмотку 3 (катушку), помещенную на магнитопровод 1. При подаче на обмотку входного

напряжения по ней идет ток, создающий в магнитопроводе магнитный поток \mathcal{L} , стремящийся сократить длину магнитных линий (по закону физики). Это приводит к повороту якоря 4 вокруг шарнира 5 против часовой стрелки. Далее рычаг на правом конце якоря перемещает вправо средний контакт 7 , находящийся на конце плоской пружинки, которая закреплена в изоляторе. В исходном положении этот средний контакт 7 был прижат к левому контакту 6 , но в результате срабатывания реле он прижимается к правому контакту 8 . Такой контакт называется еще переключающим (перекидным).

Справа на рис. 1 изображено условное графическое обозначение электромагнитного реле. Обмотка электромагнита представлена прямоугольником, правее изображены контакты. При срабатывании реле в результате подачи напряжения U входное на катушку левая пара контактов размыкается. Они называются еще нормально замкнутыми или размыкающими. Правая пара называется соответственно нормально разомкнутыми или замыкающими. В одном реле при одном якоре может быть много различных контактов, но все они обозначаются одним именем этого реле, поскольку представляют собой одну двоичную переменную. При этом размыкающие контакты являются инверсией от замыкающих. Имея одно и тоже обозначение, контакты одного реле могут располагаться в различных местах электрической принципиальной схемы для удобства вычерчивания и по другим причинам.

Существует много вариантов конструкции электромагнитного реле в зависимости от назначения. Например, реле для включения электродвигателей, называемые пускателями, имеют мощные контакты для силовых цепей, хотя при этом у пускателя могут быть и маломощные контакты для сигнальных цепей. Условное графическое обозначение при этом почти не изменяется, например замыкающий контакт для силовой цепи обозначается как на рис. 2.

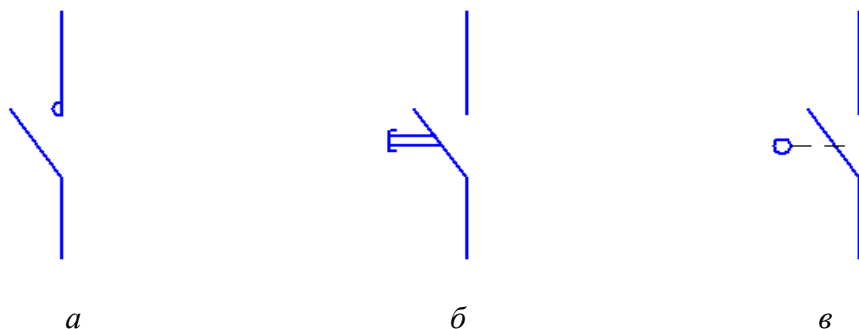


Рис. 2. Условное обозначение различных контактов:
а – сильноточный; *б* – кнопка; *в* – путевой переключатель

Электрические кнопки также широко распространены в электроаппаратуре, они служат для ручного управления электрическими цепями и часто, как и электромагнитные реле, содержат электрические контакты таких же типов. Перемещение контактов в кнопках осуществляется от руки человека, а не от электромагнита. При управлении движениями механизмов контакты могут перемещаться этими механизмами. Такие переключатели называются путевыми, их частный случай – конечные выключатели по краям перемещения.

Электрические контакты создают двоичные сигналы, так как могут либо полностью замыкать цепь (давать очень малое сопротивление), либо полностью ее размыкать (давать очень большое сопротивление). Промежуточных значений электрического сопротивления они практически не создают. Они просты, надежны и имеют малую цену, что обеспечило им широкое практическое применение, если не нужно высокое быстродействие. В то же самое время электрические контакты удобны для учебных целей.

При работе электрических контактов совместно с электронными элементами отрицательное влияние может оказывать так называемый дребезг контактов. Прежде, чем контакт замкнется, он дает несколько импульсов тока, длительностью около одной миллисекунды. Подобно мячику, отскакивающе-

му от земли, две половинки контакта несколько раз отскакивают друг от друга. На работу электромагнитных реле это явление обычно не влияет по причине их инерционности.

1.2. Построение схем электроавтоматики

При изучении логических основ ЭВМ в [1] использовалась очень простая и распространенная схема электроавтоматики, позволяющая включить электродвигатель кнопкой «Пуск» и в нужное время отключить его кнопкой «Стоп». Можно было бы обойтись даже одной кнопкой, которая при нажиме включала бы электродвигатель, а при отпускании отключала. Это самая минимальная по числу элементов (самая простая) схема (рис. 3, *а*). Здесь кнопка пуска *К* включает на сравнительно невысокое напряжение от 12 до 36 вольт для сигнальных цепей, обозначенное «+ ->», катушку пускателя *П*, который своими мощными контактами включает электродвигатель, часто на стандартное трехфазное напряжение 380/220 вольт. Кнопку *К* нужно нажимать, пока включен электродвигатель, однако это неудобно, если двигатель должен работать долго.

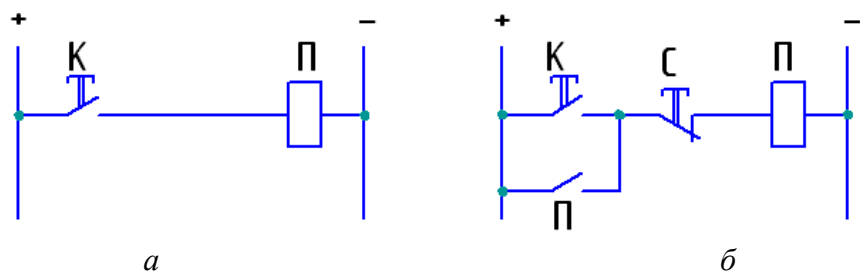


Рис. 3. Схемы включения электродвигателя:
а – простейшая схема; *б* – схема с памятью

Чтобы руки были свободны, применяется схема рис. 3, *б*. Здесь нужно только кратковременно нажать кнопку *К*, после чего пускатель *П* включит, как и ранее, электродвигатель

мощными контактами, а еще одним контактом включит сам себя по параллельной кнопке К цепи. При отпускании К пускатель останется включенным. Такая параллельная цепь называется иногда «самопитанием или самоблокировкой», она обеспечивает запоминание того события, что кнопка К была нажата. Для отключения электродвигателя нажимаем на кнопку С, которая своим размыкающим контактом разрывает цепь электромагнита пускателя П, и он отключается.

Используем простую схему на рис. 3, б для изучения фундаментальных закономерностей работы ЭВМ, так как эта схема также относится к дискретным двоичным устройствам. Предположим, что нужно, помимо включения электродвигателя, включать иногда электрическую лампочку над рабочим местом, если потребуется, например, в темное время суток. Конечно можно добавить еще одну такую же схему отдельно для лампочки, но лучше поступить более экономно и использовать уже имеющиеся кнопки К и С.

Обозначим нажатое состояние кнопки как «1», а отпущенное состояние – «0», это общепринято для любого двоичного элемента. Здесь существует 4 различных случая, приведенных в табл. 1.

Таблица 1

Различные значения для двух двоичных переменных К и С

№ п/п	К	С	Функция
0	0	0	Исходное состояние, двигатель отключен
1	0	1	Останов двигателя
2	1	0	Пуск двигателя
3	1	1	Не использовано

Неиспользованный случай (см. табл. 1, строка 3) применим для включения освещения в схеме на рис. 4 посредством пускателя Л с контактом самопитания.

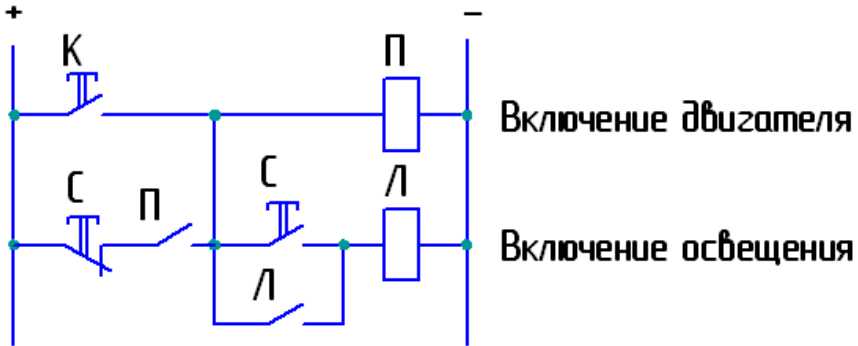


Рис. 4. Схема включения электродвигателя и освещения

Аналогично предыдущей табл. 1 составим табл. 2 возможных состояний пускателей П и Л.

Таблица 2

Различные возможные состояния пускателей по рис. 4

№ п/п	П	Л	Функция
0	0	0	Исходное состояние, двигатель и освещение отключены
1	0	1	Не использовано (недостижимо в схеме рис. 4)
2	1	0	Двигатель включен
3	1	1	Двигатель и освещение включены

Схема по рис. 4 работает следующим образом. В исходном состоянии все отключено. Нажимаем на кнопку пуска электродвигателя К, если нужно включить освещение, то не отпуская К нажимаем еще на кнопку останова С – включается освещение. Далее отпускаем кнопку С, а затем отпускаем кнопку К. Если нужно отключить двигатель и освещение, то нажимаем на кнопку С. Предполагается, что освещение без двигателя включать не нужно.

Если схема по рис. 3, б достаточно проста, чтобы, не имея опыта, представить ее работу, то схему по рис. 4 проанализиру-

ем с помощью направленного графа на рис. 7, но первоначально-но научимся его строить на более простой схеме по рис. 3, б.

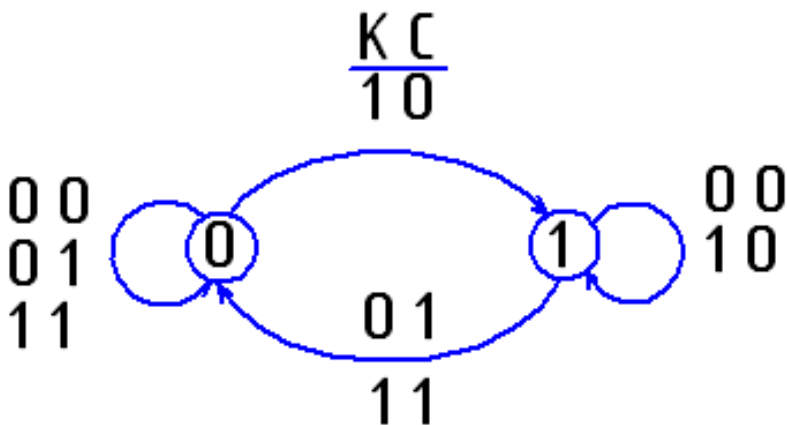


Рис. 5. Диаграмма переходов схемы (рис. 3, б) как конечного автомата

Схема на рис. 3, б может быть в двух состояниях: «пускатель П отключен» (исходное) и «пускатель П включен», соответственно на рис. 5 первое состояние обозначено вершиной графа «0», а второе – вершиной графа «1» (по значениям двоичной переменной П). Ребра направленного графа изображены стрелочками, на которых нанесена надпись, соответствующая кнопкам. Например, переход из отключенного состояния во включенное вызывается нажатием на К без нажима на С. Переход из включенного состояния в отключенное вызывается нажатием на кнопку С. Слева от вершины «0» и справа от вершины «1» показаны значения кнопок КС, при которых состояния сохраняются (переходят сами в себя).

Схему на рис. 3, б можно представить согласно рис. 6 в виде прямоугольника с двумя входами К и С и одним выходом П. В таких схемах слева представлены входы – независимые аргументы (причины), а справа – выходы – зависимые величины (следствия).



Рис. 6. Представление цифровой схемы как конечного автомата:
a – для схемы на рис. 3, *б* – общее представление

На этой схеме есть еще одна переменная величина $S(t)$, которая обозначает текущее состояние конечного автомата. Как было показано на рис. 5, состояние представлено вершинами графа, а значения входов при этом – надписями на ребрах. Изменение состояния конечного автомата можно представить также при помощи табл. 3, называемой таблицей переходов. На этой таблице все возможные значения входов КС показаны в верхней строке, причем соседние столбцы отличаются только по значению одной переменной (единичное изменение). Это позволяет использовать в дальнейшем минимизацию с помощью матрицы Вейча–Карно.

Таблица 3

Таблица переходов конечного автомата по схеме рис. 3, б

S \ КС	00	01	11	10
0	0	0	-	1
1	1	0	-	1

Слева столбец S содержит значения состояния (всего два) по значению переменной Π . Правые нижние 8 клеточек задают зависимость последующего состояния $S(t + 1)$ от предыдущего состояния $S(t)$ и значения входов $x(t)$ по рис. 6. Предположим, что конечный автомат по схеме 3, б находился в исходном состоянии, когда пускатель выключен, а на обоих

входах КС – «00». Это соответствует верхней левой клеточке на пересечении столбца входов «00» и строки состояний «0». В клеточки таблицы записываются значения последующего состояния $S(t + 1)$, но в данном случае последующее состояние равно предыдущему, то есть автомат продолжает находиться в том же состоянии сколь угодно долго. Иначе говоря, это состояние устойчиво при данном входе, а в клеточке записано то же самое состояние «0». Если дать на вход «10», т.е. нажать на кнопку К, то пускатель включится, автомат перейдет в состояние «1». Поэтому на пересечении столбца «10» и строки «0» в клеточке записано «1». Это значит, что мы переходим на строку состояния «1», то есть на следующую строку в клеточку ниже, а там записано «1», что соответствует устойчивому состоянию при заданном входе «10».

Если автомат находился в исходном состоянии «0», а на входы поступило «01», то есть мы нажали на кнопку С (стоп), то состояние автомата не изменится. В столбце «11» пока стоят прочерки по той причине, что от значения входов «00» практически невозможно сразу дать на входы «11», всегда одна из кнопок окажется нажатой раньше, хотя бы на очень короткий интервал времени. Однако представим, что на входе было «01» (нажата кнопка С), и мы нажали еще на кнопку пуск К, тогда на вход поступит значение «11», эта ситуация показана в табл. 4, где в клеточку записан «0», т.е. исходное состояние «0» сохраняется.

Таблица 4

Модифицированная табл. 3

КС \ S	00	01	11	10
0	0	0	0	1
1	1	0	0	1

Предположим, что из исходного состояния схема перешла в состояние «1» после нажима на К, после чего мы нажали еще на кнопку С. В табл. 4 в соответствующую клеточку на пересечении $S = 1, X = 11$ записан «0», то есть автомат перейдет в исходное состояние $S = 0$, пускатель П отключится. Здесь можно еще сказать, что кнопке останова С дается предпочтение перед кнопкой пуска К (кнопка С пересиливает кнопку К). Однако для схемы рис. 4 такая ситуация не годится, так как при нажиге на обе кнопки К и С нужно включить освещение не отключая пускатель двигателя. По этой причине потребуется таблица переходов 5, где в клеточку на пересечении столбца «11» и строки «1» нужно записать «1». В клеточку на пересечении столбца «11» и строки «0» также нужно записать «1». Это означает, что будучи в исходном состоянии после нажима на С мы нажимаем еще на пуск К (на входе «11»), и автомат переходит в состояние «1» – включается пускатель. В данном случае предпочтение отдается кнопке пуска К (кнопка К пересиливает кнопку С).

Таблица 5

Таблица переходов с предпочтением кнопке К (пуск)

S \ КС	00	01	11	10
0	0	0	1	1
1	1	0	1	1

Перемещение по таблице переходов для включения освещения показано в табл. 6.

Однако возникает определенная трудность при отпускании обеих кнопок, то есть при переходе значения входа с «11» на «00». Может оказаться, что мы отпустили раньше кнопку К, а кнопка С осталась еще нажатой, то есть на входе произойдет

«11»—«01»—«00». Как следует из таблицы переходов (табл. 7, столбец «01»), пускатель отключается, и нам не удастся не только включить освещение, но и сохранить включенным пускатель и двигатель.

Таблица 6

Перемещение текущего состояния

S \ KC	00	01	11	10
0	0	0	1	1
1	1	0	1	1

↓
на последующее состояние

Таблица 7

Трудности при смене значения входа с «11» на «00»

S \ KC	00	01	11	10
0	0	0	1	1
1	1	0	1	1

Как было сказано ранее, нужно не отпуская кнопку К отпустить кнопку С. Иначе говоря, нельзя оставлять включенной кнопку С при отключенной кнопке К. Допустимый путь перехода состояния показан в табл. 8.

Таблица 8

Допустимый путь перехода состояния схемы на рис. 3, б

	КС	00	01	11	10
S					
0		0	0	1	1
1		1	0	1	1

Построим граф переходов (рис. 7) и таблицу переходов (табл. 9) для схемы с двумя пускателями на рис. 4. У этой схемы может быть 4 различных состояния, как показано ранее в табл. 2.

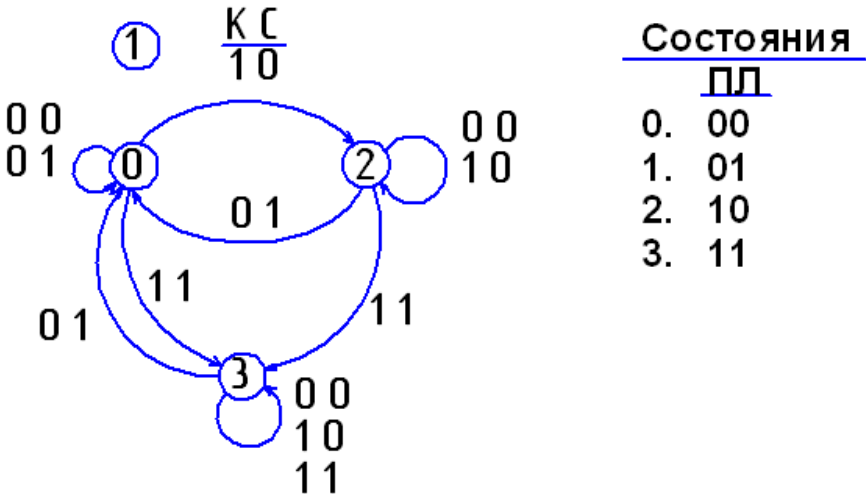


Рис. 7. Диаграмма переходов для схемы на рис. 4

Таблица переходов для схемы на рис. 4

S \ КС	00	01	11	10
0	0	0	3	2
1	1	1	1	1
2	2	0	3	2
3	3	0	3	3

В табл. 9 состояние «1» (включен только пускатель освещения) изолировано от других состояний, в него схема не может перейти. Переходы состояния при включении сразу пускателей П и Л (двигателя и лампочки) и последующем отключении показаны пунктирными стрелками. Изменение значения входов КС «00»–«11», происходящее через «10» («00»–«10»–«11») или «01» («00»–«01»–«11») здесь ничего не меняет – включение П и Л происходит однозначно, чего нельзя сказать об отключении. Отключать К и С «11»–«00» с сохранением включенными П и Л нужно только через «10», иначе отключатся и П, и Л. Такая неоднозначность при изменении значения двух и более двоичных переменных называется «состязаниями» или «гонками» (races) [2].

1.3. Гонки и меры по их устранению

Изменение нескольких независимых двоичных переменных не может произойти идеально одновременно из-за неодинаковых физических условий для каждой переменной, например,

различная жесткость пружин в различных кнопках (схемы на рис. 3, б и 4), различная длина соединяющих проводов в быстродействующих электронных устройствах (ЭВМ) и т.д. Это означает, что даже для двух переменных между двумя значениями «00» и «11» появляются кратковременно значения либо «01», либо «10», что может привести к различным результатам, иногда неприемлемым, как например, отключение и двигателя, и освещения в схеме на рис. 4. Тем более описанная ситуация испортит работу 64-разрядного процессора, у которого каждый разряд – двоичная переменная. Однако на заре развития компьютеров применили очень эффективное средство против гонок – синхронизацию. Суть синхронизации можно пояснить графиком (рис. 8).

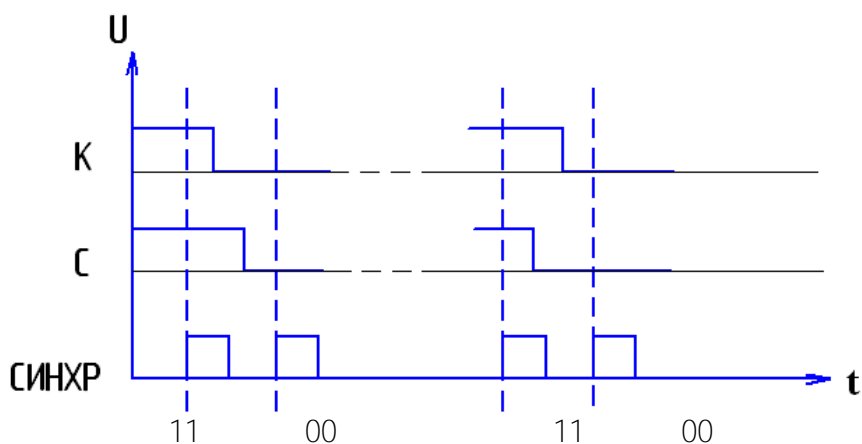


Рис. 8. График, поясняющий синхронизацию

Слева на этом графике показан переход значений переменных КС «11»–«01»–«00», который неприемлем, а справа – «11»–«10»–«00», обеспечивающий нормальную работу. Идея синхронизации состоит в том, чтобы исключить из рассмотрения промежуточные значения «01» и «10», а рассматривать только «11»–«00». Моменты времени, когда рассматриваются

значения переменных, задаются импульсами синхронизации СИНХР, точнее их передними фронтами. Для выполнения такой задачи был специально создан D -триггер, в который записываются данные только в момент скачка напряжения на входе синхронизации [1, 3].

Цифровые устройства, в которых применена синхронизация, называются синхронными. В них импульсы синхронизации создаются с помощью тактового генератора, эти импульсы называются еще «тактовыми» или «стробами» (strobe). Они обозначаются на схемах через C (clock – время, часы). Сложные электронные цифровые схемы такие, как ЭВМ, всегда выполняются синхронными, так как иначе обеспечить их устойчивую работу в условиях массового производства практически невозможно. Цифровые устройства, где не использована синхронизация, называются асинхронными. В последующем будет рассматриваться асинхронная цикловая схема управления производственным механизмом, где гонки не проявляются (см. приложение).

Для такой простой схемы, как рис. 4 на практике целесообразнее применить более простое решение, нежели синхронизацию, – ввести специальную кнопку включения освещения, обозначенную на рис. 9 буквой В, что исключает гонки. В этой схеме использовано 3 кнопки: включение электродвигателя К, включение освещения В и общее отключение того и другого С.

В данном разделе преследовалось несколько целей при рассмотрении простых схем электроавтоматики, построенных на электромагнитных реле.

1. Обучение собственно электроавтоматике на реле, которые получили и в настоящее время большое распространение.
2. Изучение сути синхронизации цифровых схем.
3. Изучение модели конечного автомата на примере простых схем, в частности, графа переходов и таблицы переходов конечного автомата. Следует отметить, что более наглядной является диаграмма переходов, зато таблица переходов обес-

печивает полноту рассмотрения, так как требует заполнения всех клеточек, помимо этого по таблице переходов легко строить логические функции и схемы, как будет видно из дальнейшего (см. раздел 4).

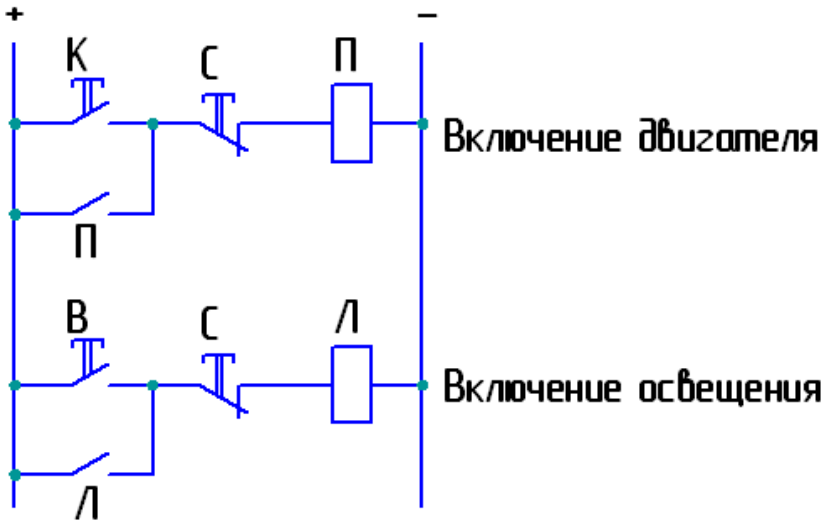


Рис. 9. Схема, свободная от гонок.

Модель конечного автомата лежит в основе дискретной электроавтоматики, дискретной пневмо- и гидроавтоматики, цифровых схем, в том числе ЭВМ. Компьютерная программа также является конечным автоматом, поэтому его изучение полезно программистам, например для представления алгоритмов.

Конечный автомат имеет входы $X(t)$, выходы $Z(t)$ и внутренние состояния $S(t)$, которые могут изменяться во времени t . Все эти величины дискретные, как в случае релейных элементов, рассмотренных ранее. Время тоже разбито на отдельные интервалы (такты), разделенные сменой значения X , куда относятся также синхросигналы. Слово «конечный» указывает на конечное число значений S в противоположность беско-

нечному, которое может быть при строгих математических рассуждениях. Практические устройства имеют конечное число состояний, иногда очень большое, как ЭВМ. Конечный автомат реагирует на входные воздействия (например, нажим на кнопки) изменением своего внутреннего состояния и/или значения выхода. При этом для него характерны однозначные зависимости

$$S(t+1) = F[X(t), S(t)];$$

$$Z(t) = f[X(t), S(t)],$$

где $X(t)$ – значение входа в текущем такте;

$Z(t)$ – значение выхода в текущем такте;

$S(t)$ – состояние в текущем такте;

$S(t+1)$ – состояние в следующем такте.

На практике находят применение как схемы с одним внутренним состоянием (без памяти), так и с несколькими внутренними состояниями, хранящимися в памяти. Примером первых может служить кодовый замок, который можно открыть, задав одну единственную правильную комбинацию входных кнопок. Такие схемы называют комбинационными (combinational), также логическими, также одноктактными. Примером вторых может служить кодовый замок, для открытия которого нужно правильно задать последовательность комбинаций входных кнопок, для запоминания которых нужна память. Такие устройства называются последовательностными (sequential), также многотактными. ЭВМ можно отнести к многотактному синхронному устройству.

2. МИНИМИЗАЦИЯ ЛОГИЧЕСКИХ ФУНКЦИЙ, ПОСТРОЕНИЕ ЛОГИЧЕСКИХ СХЕМ

Еще до появления ЭВМ при анализе работы схем электроавтоматики и их построении (синтезе) использовалась Булева алгебра [2]. При этом логические функции, выполняемые схемами электроавтоматики, старались минимизировать, чтобы свести к минимуму количество элементов в схеме. В последующем эта минимизация потребовалась при разработке электронных схем, в том числе ЭВМ, и при составлении программ ЭВМ с целью уменьшения числа команд при одной и той же функциональности (алгоритме).

Логическую функцию можно рассматривать как частный случай конечного автомата с одним состоянием (рис. 6, б). Зависимость выхода от входов $Z(t) = f[X(t)]$ для логической функции и для логического устройства часто представляют в виде таблицы истинности (от названия «истина–ложь» значений Булевой переменной). Очень простой пример логической функции (и соответственно логического устройства) приведен в схеме электроавтоматики [1], где нужно было обеспечить включение электропилы только двумя кнопками К1 и К2, разнесенными на расстояние около 0,5 м. Это делалось для того, чтобы исключить попадание одной руки рабочего в пилу при ее включении, то есть пилу можно включить только двумя руками, одной рукой (от одной кнопки) она никак не включится.

На этой схеме (рис. 10) электрическая цепь катушки пускателя П замыкается только, если замкнуты контакты и К1, и К2. Рассматривая каждую кнопку как двоичную переменную, можно сказать, что $K1 = 1$, и $K2 = 1$ при включении пускателя П. Если нам нужно замыкать электрическую цепь, то последовательное включение контактов выполняет логическую функцию «И». Если нужно размыкать цепь, то последовательное соединение контактов выполняет логическую функцию «ИЛИ», однако этот случай далее не рассматривается и предлагается для самостоятельного изучения.

Параллельное соединение контактов обеспечивает логическую функцию «ИЛИ», если нужно замыкать цепь, то есть логическое значение «ИСТИНА» соответствует замкнутой цепи. Это показано на рис. 11, где под каждой цепью приведена таблица истинности логической функции проводимости между точками *a* и *b*. Логическое «И» обозначают точкой, как умножение, а логическое «ИЛИ» обозначают, как сложение, знаком «+».

Все возможные случаи цепей «И» при двух переменных X_1 и X_0 с учетом их инверсии показаны на рис. 12. Инверсия отмечена черточкой сверху и соответствует размыкающему (нормально замкнутому) контакту (см. рис. 1, б).

Часто составление логических функций и соответственно построение электрических цепей (логических схем) начинается с таблицы истинности. Предположим, задана исходная табл. 10.

Таблица 10

Таблица истинности примера на рис. 13

X_1	X_2	Z
0	0	0
0	1	0
1	0	1
1	1	1

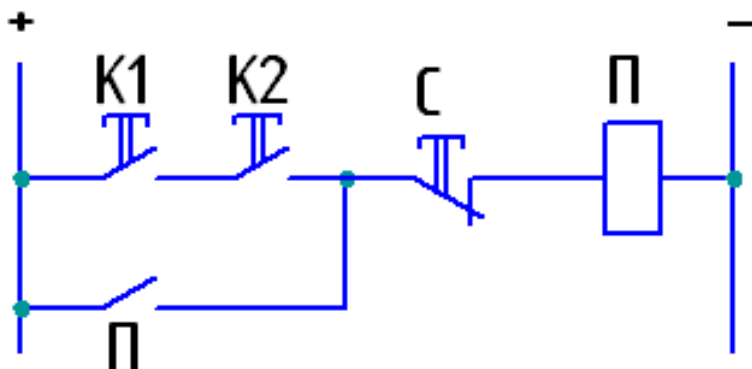
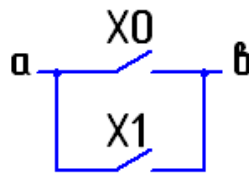
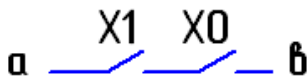


Рис. 10. Включение пускателя только двумя кнопками K1 и K2



входы		выход
X1	X0	И ·
0	0	0
<u>0</u>	1	0
1	<u>0</u>	0
<u>1</u>	<u>1</u>	1

X1	X0	ИЛИ +
<u>0</u>	<u>0</u>	0
0	1	1
1	<u>0</u>	1
<u>1</u>	1	1

Рис. 11. Электрические цепи контактов и их логические функции

Электрич. цепь	Логическая функция "И"	Цепь замкнута, функция = 1 при X1 X0
	$\overline{X1} \cdot \overline{X0}$	0 0
	$\overline{X1} \cdot X0$	0 1
	$X1 \cdot \overline{X0}$	1 0
	$X1 \cdot X0$	1 1

Рис. 12. Все возможные случаи цепей «И» при двух переменных X1 и X0

Исходя из табл. 10, требуется обеспечить проводимость в последних двух строках, соответственно по двум нижним строкам рис. 12 составляем схему, приведенную на рис. 13. Слева показана исходная схема, она составляется достаточно просто – каждой строке таблицы истинности с $Z = 1$ соответствует своя последовательная цепь контактов, «1» соответствует замыкающему контакту, «0» – размыкающему. Далее приступаем к минимизации схемы и функции. В любом случае будет замкнут либо верхний, либо нижний контакт $X0$, поэтому его можно исключить из схемы.

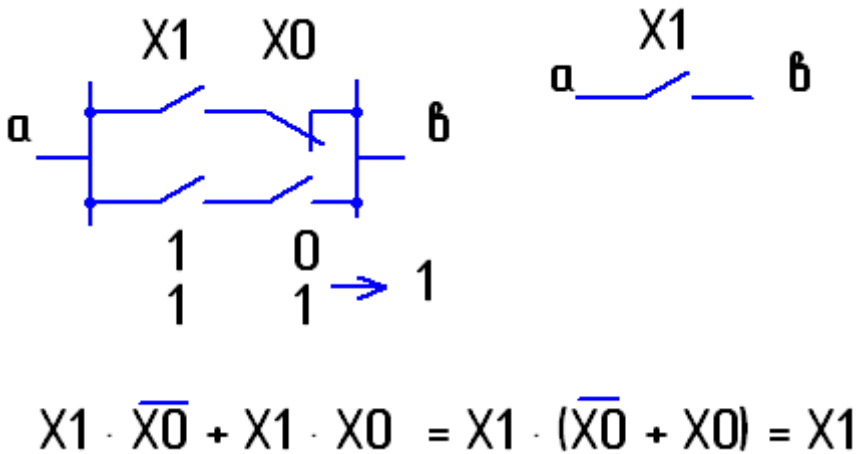


Рис. 13. Электрическая схема по табл. 10 и ее минимизация

Таким образом, удастся четыре контакта заменить одним, а функцию с четырьмя символами и знаком «+» заменить одним символом.

Получается простая схема с одним контактом $X1$. Эта закономерность лежит в основе двух методов минимизации, рассмотренных далее. Функция, приведенная на рис. 13 внизу, представляет собой логическую сумму двух логических произведений. На математическом языке логическая операция

«И» называется конъюнкцией, а операция «ИЛИ» – дизъюнкцией. Рассматриваемая форма функции называется дизъюнктивной нормальной формой (ДНФ).

2.1. Минимизация с помощью матрицы Вейча–Карно

Рассмотрим логическую функцию от 4 аргументов, все возможные значения ее входов в количестве $2^4 = 16$ представлены в табл. 11.

Таблица 11

Таблица истинности для 4 аргументов X_0 – X_3

N	X_3	X_2	X_1	X_0	Z
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	
3	0	0	1	1	
4	0	1	0	0	
5	0	1	0	1	
6	0	1	1	0	
7	0	1	1	1	
8	1	0	0	0	
9	1	0	0	1	
10	1	0	1	0	
11	1	0	1	1	
12	1	1	0	0	
13	1	1	0	1	
14	1	1	1	0	
15	1	1	1	1	

В левом столбце табл. 11 записан по порядку натуральный ряд чисел, а в столбцах X – соответствующие 4-разрядные двоичные числа. В столбце Z может быть записана «1», если функция для данной строки дает значение «истина», либо «0»,

если функция дает «ложь». Для логической схемы по умолчанию «1» соответствует проводимости.

Таблицу истинности можно более компактно представить в форме квадратной матрицы 4×4 (табл. 12).

Таблица 12

Матрица для 4 аргументов X_3 – X_0

X 1,0 3,2	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

В 16 клеточек справа внизу записаны значения N , из табл. 11, которые соответствуют разрядам двоичного числа X_3, X_2, X_1, X_0 . Необходимо отметить, что **соседние столбцы и соседние строки матрицы отличаются только значением одной переменной X** . В клеточки матрицы записываются значения $Z = 0$ или $Z = 1$, для экономии времени можно записывать только 1, а клеточки с 0 оставить пустыми. Представленная матрица позволяет достаточно просто проводить минимизацию и называется «матрицей Вейча или картой Карно» по имени ее изобретателей. Рассмотрим на этой матрице простейший случай, показанный на рис. 13. Здесь имеется только 2 аргумента X_1 и X_0 , а матрица будет 2×2 (табл. 13).

Таблица 13

Матрица для 2 аргументов X_1, X_0

X 0 1	0	1
0	0	0
1	1	1

Минимизация на матрице проводится путем объединения соседних клеточек в одну область с одинаковым значением функции. В данном случае мы объединяем две соседние клеточки внизу справа, они отмечены затемнением. Аргумент, который в одной области имеет значения и 0, и 1 удаляется как несущественный, то есть удаляется X_0 . Оставляются только те аргументы, которые сохраняют свое значение. В данном случае оставляем только аргумент X_1 . Получается очень простая функция $F = X_1$, что соответствует рис. 13.

Рассмотрим пример минимизации функции, которая равна 1 при значениях аргументов 1, 6, 7, 12, 14, 15. Матрица для нее представлена в табл. 14 (см. также табл. 12).

Таблица 14

Матрица для примера

X 1,0 3,2	00	01	11	10
00		1		
01			1	1
11	1		1	1
10				

Соседние клеточки 6, 7, 14, 15 объединяются в одну область, для которой получается одно слагаемое $X_1 \cdot X_2$, аргумент X_0 для этой области имеет сначала значение 1, а затем 0, поэтому он удаляется. Аналогично удаляется аргумент X_3 . Клеточку 12 можно объединить с клеточкой 14, так как матрица мыслится непрерывной по горизонтали и по вертикали. Это объединение даст слагаемое не $X_0 \cdot X_2 \cdot X_3$. Клеточку 1 не удастся ни с чем объединить, она дает самое длинное слагаемое $X_0 \cdot \text{не}X_1 \cdot \text{не}X_2 \cdot \text{не}X_3$. Таким образом, получили $Z = X_1 \cdot X_2 + \text{не}X_0 \cdot X_2 \cdot X_3 + X_0 \cdot \text{не}X_1 \cdot \text{не}X_2 \cdot \text{не}X_3$. Чем объединение крупнее, тем слагаемое короче, поэтому стремимся

объединить как можно больше клеточек. Объединения могут накладываться друг на друга, как, например, на клеточке 14.

Проверим корректность полученной функции, например, для клеточки 0, для нее все аргументы имеют значение 0. Первое слагаемое имеет значение 0, так как $X_1 = 0$, умножение на 0 сразу дает 0. Для второго слагаемого $X_0 = 0$, но инверсия $\neg X_0 = 1$, однако далее $X_2 = 0$, поэтому второе слагаемое также равно 0. Для третьего слагаемого $X_0 = 0$, поэтому оно также равно 0. Логическая сумма нулей равна нулю, то есть для данного значения аргументов функция корректна. Можно проверить корректность функции для всех 16 значений. Эта работа достаточно трудоемка и однообразна.

2.2. Метод Квайна–МакКласки

Предыдущий метод позволяет минимизировать функции с четырьмя и менее аргументами, если же их больше, то используется данный метод. Он основан на том же явлении, которое иллюстрировано на рис. 13. Рассмотрим характерный пример, представленный вначале с помощью матрицы в табл. 15.

Таблица 15

Пример

X 1,0 3,2	00	01	11	10
00	1	1	1	1
01	0	0	0	1
11	0	0	0	1
10	0	0	0	0

Сначала в таблице истинности оставляют только ту часть, которая соответствует $Z = 1$. Исходная таблица истинности для примера представлена в табл. 16

Таблица 16

Исходная таблица истинности для примера

N	X_3	X_2	X_1	X_0	Z
0	0	0	0	0	1
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	1
6	0	1	1	0	1
14	1	1	1	0	1

Выполняем поиск пар значений аргументов, которые отличаются только **одной переменной**, например строка 0 и строка 1 отличаются только значением $X_0 = 0$ и $X_0 = 1$. В результате X_0 удаляется, это действие иногда называется «склеиванием». Получаем табл. 17. Чтобы не пропустить возможное склеивание, сравниваем пары строк в следующем порядке: первую строку сравниваем со всеми остальными ниже ее, далее вторую строку сравниваем со всеми ниже ее, третью строку сравниваем со всеми ниже ее и т. д. Получаем 6 строк, которые наново нумеруем, как показано в правом столбце.

Таблица 17

Результат 1-го «склеивания» (табл. 16)

Пары	X_3	X_2	X_1	X_0	№
0,1	0	0	0	–	0
0,2	0	0	–	0	1
1,3	0	0	–	1	2
2,3	0	0	1	–	3
2,6	0	–	1	0	4
6,14	–	1	1	0	5

Продолжаем склеивание результатов первого склеивания, здесь можно склеивать строки с одинаковым местом прочерка, получаем табл. 18.

Таблица 18

Результат 2-го «склеивания» (табл. 17)

№ пар	X3	X2	X1	X0	№
0,3	0	0	–	–	0
1,2	0	0	–	–	1

Дальнейшее «склеивание» невозможно. Полученные результаты записываем в табл. 19, в ней содержатся все строки исходной табл. 16 или результаты их склеивания, повторы исключаются. Еще здесь может быть применено правило «поглощения», когда строка с большим числом прочерков исключает строку с меньшим числом прочерков, например, «00--» поглощает «00-1», остается только «00--».

Таблица 19

Результаты склеивания и поглощения строк

X3	X2	X1	X0	№
0	0	–	–	0
0	–	1	0	1
–	1	1	0	2

Далее остается выбрать такой набор полученных элементов, называемых импликантами, который выполнил бы все строки исходной табл. 16 и одновременно был бы минимальным. Для этого можно составить так называемую матрицу Квайна, представленную в табл. 20. В этой матрице каждый столбец изображает результирующий элемент (импликант), указанный номером из табл. 19. Этот элемент является слагаемым результирующей логической суммы, которая представляет собой искомую функцию. Каждая строка матрицы Квайна изображает строку из исходной табл. 16, также указанную номером.

Таблица 20

Получение минимального набора слагаемых (матрица Квайна)

	0	1	2
0	1		
1	1		
2	1	1	
3	1		
6		1	1
14			1

Строки 0, 1, 3 выполняются только слагаемым 0, поэтому его нужно оставить. Строка 14 выполняется только слагаемым 2, поэтому его тоже нужно оставить. Слагаемое 1 можно опустить, так как без него все строки табл. 20 будут выполнены, таким образом это слагаемое избыточно. В результате двух этапов склеивания и поиска минимального покрытия исходной таблицы истинности получена следующая функция:

$$Z = \text{не}X_2 \cdot \text{не}X_3 + \text{не}X_0 \cdot X_1 \cdot X_2.$$

Действия, выполненные в данном разделе по методу Квайна–МакКласки, хорошо согласуются с матрицей Вейча–Карно. Из последней хорошо видно, что из двух объединений 2-6 и 6-14 (см. также табл. 12) первое является избыточным с точки зрения обязательного покрытия всех клеточек матрицы, так как клеточка 2 покрывалась объединением 0-3.

Рассмотренные методы минимизации логических функций и логических схем (Вейча–Карно и Квайна–МакКласки) широко известны. Они дают так называемую дизъюнктивную нормальную форму функции (ДНФ), сумму произведений. Такая функция соответствует двухуровневой логической схеме: первый уровень от входов – логические ячейки «И» (фильтры), второй уровень – логическая ячейка «ИЛИ» (сборка по высокой единице) (рис. 15).

2.3. Метод свертки таблицы истинности

Данный метод более трудоемкий, однако он имеет преимущества перед двумя предыдущими методами минимизации логических функций:

- 1) не ограничен формой представления функции (не обязательно ДНФ);
- 2) дает более минимальные результаты;
- 3) может работать с не полностью определенными таблицами истинности, где для некоторых значений аргументов значение функции не определено или безразлично, как, например, в табл. 3, столбце «11».

Число аргументов не ограничено, как и в методе Квайна–МакКласки.

Рассмотрим этот метод на примере функции, которая имеет 5 аргументов. Дает 1 при следующих значениях: 1, 12, 16, 20, 21 и дает 0: при 2, 3, 8, 10, 22, 23, 25, 27, 29, 30, 31. Это соответствует не полностью определенной таблице истинности, которая занимает 6 левых столбцов табл. 21 и отмечена сверху затемнением. Верхние 5 строк соответствуют значению функции 1, ниже выделена затемнением граница, под которой записаны аргументы, соответствующие значению функции 0.

Данный метод основывается на определении влияния каждого аргумента на значение функции. Рассмотрим рис. 11 с этой точки зрения, исключим на таблице истинности схемы «И» левый столбец X_1 (временно закроем). Останется только столбец переменной X_0 , и вторую строку таблицы нельзя будет отличить от четвертой, так как в ней значение $X_0 = 1$, то есть одинаково. Таким образом схема «И» не сможет различить, где давать на выходе 0 и где 1 без X_1 . Таким образом, X_1 необходим для различения второй и четвертой строк таблицы истинности, отмечаем это подчеркиванием. В табл. 21 это отмечается жирными символами.

Таблица 21

Решение примера минимизации сверткой таблицы истинности

граница	гра	гра	гра	гра	гра										
Аргументы $X \rightarrow$	4	3	2	1	0	5		3	1	5	6		1	6	
1	0	0	0	0	1	0		0	0	0	1		0	1	
12	0	1	1	0	0	1		1	0	1	1				
16	1	0	0	0	0	0									
20	1	0	1	0	0	1		0	0	1	1				
21	1	0	1	0	1	0									
граница	гра	гра	гра	гра	гра	гра	гра	гра	гра	гра	гра	гра	гра	гра	гра
2	0	0	0	1	0	0		0	1	0	1		1	1	
3	0	0	0	1	1	0									
8	0	1	0	0	0	0		1	0	0	0		0	0	
10	0	1	0	1	0	0		1	1	0	0		1	0	
22	1	0	1	1	0	1		0	1	1	1				
23	1	0	1	1	1	0		0	1	0	1				
25	1	1	0	0	1	0									
27	1	1	0	1	1	0									
29	1	1	1	0	1	0									
30	1	1	1	1	0	1		1	1	1	1				
31	1	1	1	1	1	0									
Число совпадений	0	3	1	5	1			1	3	1			1	1	

Возвращаясь к рис. 11, временно исключим $X0$ и отметим подчеркиванием те места, где необходимо $X0$, аналогично $X1$. Для схемы «И» характерно выделение нескольких «1» в одной строке из области «1» на выходе. Аналогично для схемы «ИЛИ» характерно выделение нескольких «0» в одной строке из области «0» на выходе, как показано на рис. 11.

Для табл. 21 в столбцах 4, 3, 2, 1, 0 аналогично исключим временно по очереди каждый столбец и отметим жирным шрифтом те места, где нужен аргумент для отличия области единиц от области нулей (ниже границы). Отметим число

совпадений с одной и с другой стороны границы и запишем это число в нижней строке.

Начнем рассмотрение столбцов с минимального числа совпадений. Окажется, что столбец 4 вовсе не нужен, так как в нем нет ни одного совпадения, то есть он избыточен с точки зрения различимости подмножеств аргументов, дающих результат «1» и дающих «0». Эти подмножества разделены затемненной границей. **Исключим первый столбец.**

Далее рассмотрим столбцы 2 и 0 с одним совпадением, в одной строке 12 у них имеется 1 и 0 в области «1» на выходе (отмечено затемнением). Это соответствует «И», если взять инверсию от X_0 . Вводим промежуточную переменную $X_5 = X_2 \cdot \text{не}X_0$ и в столбце 5 записываем ее значение. Она равна «1» при $X_2, X_0 = 1, 0$, во всех остальных случаях она равна «0». Поскольку вновь введенная переменная X_5 взяла в себя информацию от X_2 и X_0 , то рассмотрим возможность полной их замены. Различимость областей «1» и «0» в тех строках, где были необходимы X_2 и X_0 , обеспечивается переменной X_5 . Таким образом, в таблице истинности остаются переменные X_3, X_1, X_5 . Эта новая таблица расположена правее после пустого столбца, у нее уменьшилось число столбцов и число строк за счет исключения повторений, то есть таблица заметно свернулась.

Аналогично поступаем с новой таблицей истинности и вводим переменную $X_6 = \text{не}X_3 + X_5$. Здесь в одной строке для столбцов с минимальным числом совпадений оказались $X_3 = 1$ и $X_5 = 0$ в области «0» (нижней). Для получения 00, как требуется для «ИЛИ», пришлось взять инверсию от X_3 . Получаем новую совсем свернутую таблицу истинности со столбцами X_1 и X_6 .

Для новой таблицы истинности в одной строке оказались $X_1 = 0$ и $X_6 = 1$ в области «1», что соответствует заключительной подстановке $Z = \text{не}X_1 \cdot X_6$. Последовательно подставляя в функцию промежуточные переменные, получаем:

$$Z = \text{не}X1 \cdot X6 = \text{не}X1 \cdot (\text{не}X3 + X5) = \text{не}X1 \cdot (\text{не}X3 + \text{не}X0 \cdot X2).$$

Проверим корректность результата, решив ту же задачу с помощью матрицы Вейча–Карно (табл. 22). Здесь сразу нельзя применить эту матрицу, так как число переменных равно 5, но после того, как была удалена переменная $X4$, их осталось 4: $X0$ – $X3$.

Таблица 22

Матрица для проверки правильности результата

X 1,0 3,2	00	01	11	10
00	1	1	0	0
01	1	1	0	0
11	1	0	0	0
10	0	0	0	0

Имеем на матрице две объединенных области: 0, 1, 4, 5 и 4, 12 и соответственно 2 слагаемых: $Z = \text{не}X1 \cdot \text{не} X3 + \text{не}X0 \times \times \text{не}X1 \cdot X2$; такой же результат дает метод Квайна–Мак-Класки. Легко заметить, что данное выражение совпадает по результату с выражением, полученным с помощью свертки таблицы истинности, в котором $\text{не}X1$ вынесено за скобку, однако последнее короче.

2.4. Построение и анализ работы логической схемы

Каждая подстановка промежуточной переменной в методе свертки таблицы истинности соответствует логической ячейке.

В приведенном выше примере использовались логические ячейки «И», «ИЛИ», «НЕ». Полученная логическая схема изображена на рис. 14.

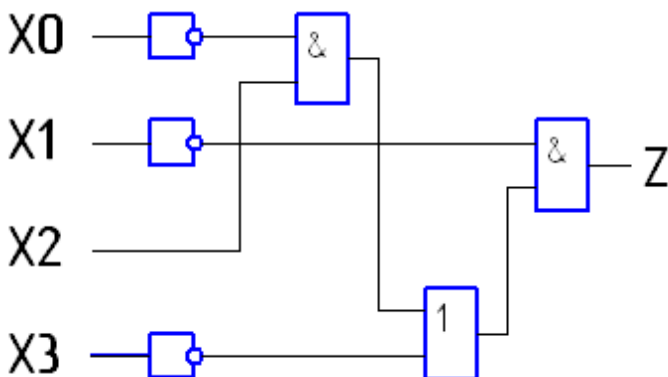


Рис. 14. Логическая схема,
полученная методом свертки таблицы истинности

Для сравнения ниже на рис. 15 приведена логическая схема, полученная по матрице истинности табл. 22.

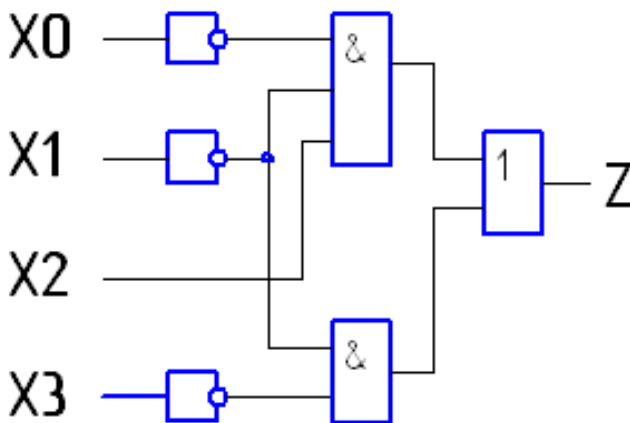


Рис. 15. Логическая схема по форме ДНФ,
полученная методами Вейча–Карно, либо Квайна–МакКласки

Сравнивая обе схемы, можно заключить, что первая схема проще, так одна из логических ячеек во второй схеме имеет 3 входа, а у первой все ячейки на 2 входа. Соответственно при

программировании логических операторов у первой схемы все операторы будут иметь 2 аргумента. Отметим, что метод свертки таблицы истинности позволяет получить результат, одинаковый со второй схемой, так как можно было бы на табл. 21 по выделению в строке 12, столбцах 2, 1, 0 сразу подставить $X_2 \cdot \text{не}X_1 \cdot \text{не}X_0$.

Для анализа правильности схемы и функции можно проверить работу схемы, как показано на рис. 16. Здесь можно руководствоваться следующими правилами:

- 1) в ячейке «И» низкий уровень (0 по умолчанию) с любого входа сразу проходит на выход;
- 2) в ячейке «ИЛИ» высокий уровень (1 по умолчанию) с любого входа сразу проходит на выход.

Для некоторых значений входов, указанных над рисунком в порядке X_3, X_2, X_1, X_0 , показано распространение сигнала слева направо, при этом высокий уровень обозначен толстой линией. Следует отметить, что выделенные жирным значения входов соответствуют отмеченным значениям в табл. 21 и являются критичными для выхода – их изменение на противоположное меняет значение выхода.

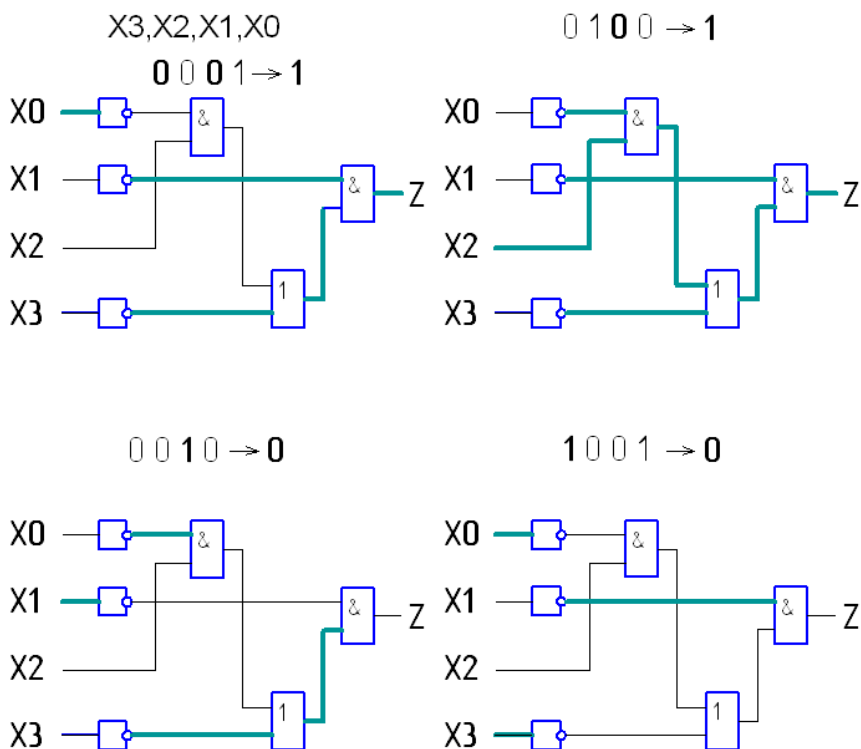


Рис. 16. Распространение сигналов в логической схеме по рис. 14

3. СХЕМА СИНХРОНИЗАЦИИ

В качестве примера практического использования полученной логической схемы рассмотрим схему синхронизации цифрового устройства, приведенную на рис. 17. Это устройство имеет цикл работы в 16 тактов от 0 до 15. Необходимо обеспечить синхросигнал Z высокого уровня в тактах 0, 1, 4, 5, 12, в остальных тактах синхросигнал должен быть низким. Для этого подходит полученная ранее логическая схема на

рис. 14, на ее входы подаются сигналы Q от счетчика импульсов СТ2, который получает входные импульсы C от тактового генератора ТГ. На рис. 17 приведена диаграмма сигналов схемы синхронизации. Краткую выборочную проверку схемы можно провести по сигналу $Q1$ – там, где он высокий, Z должен быть низким, так как после инвертора он низкий и сразу проходит на выход ячейки $\&$.

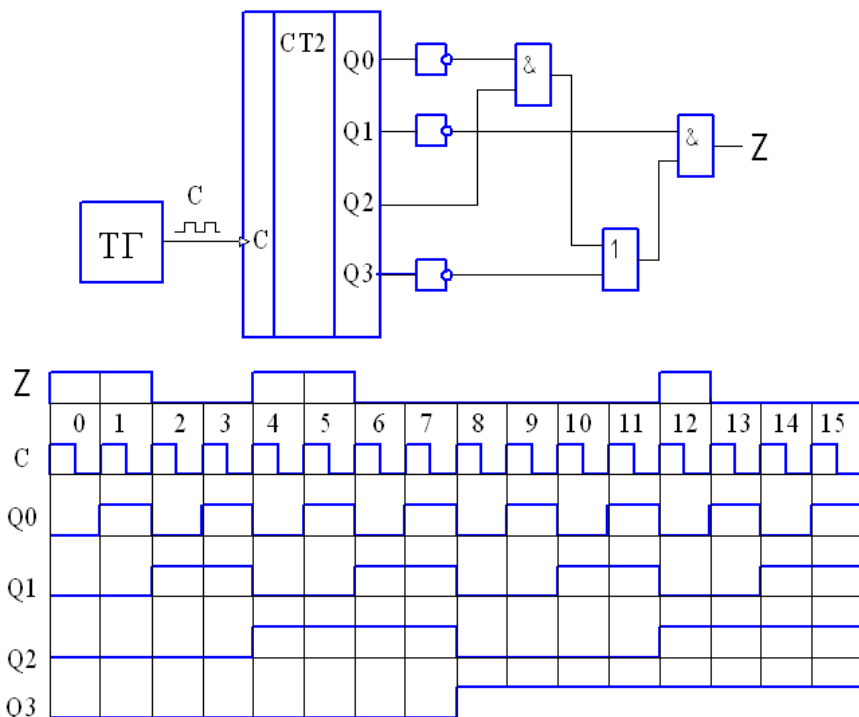


Рис. 17. Схема синхронизации с использованием логической схемы рис. 16

Для получения синхросигнала Z можно использовать другую схему синхронизации, приведенную на рис. 18. Она дороже предыдущей, так как использует более дорогие элементы (дешифратор и триггеры), но она позволяет легче изменять

синхросигнал (сдвигать его по тактам). Часто это требуется при первоначальной наладке устройства путем переключения входов логической схемы на другие выходы дешифратора.

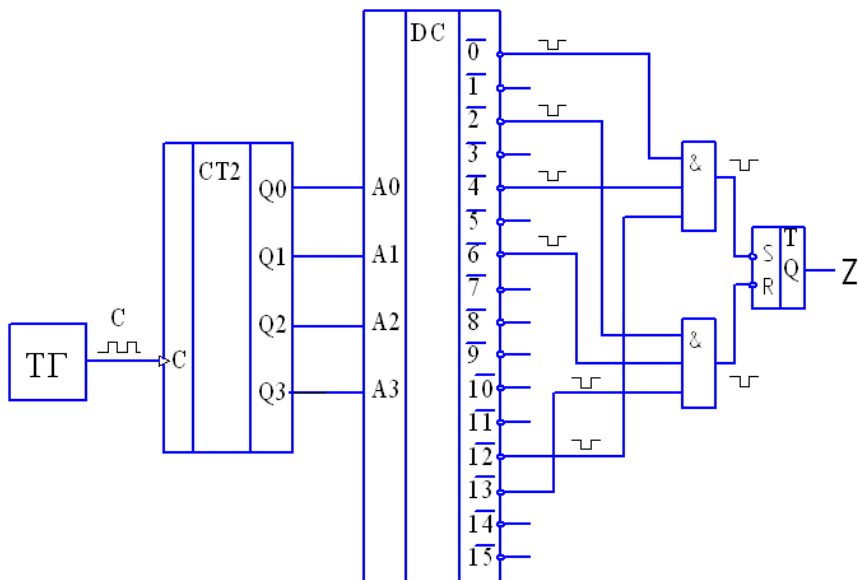


Рис. 18. Схема синхронизации с дешифратором

4. ПОСТРОЕНИЕ СХЕМ ПО ТАБЛИЦЕ ПЕРЕХОДОВ

Как было сказано ранее, в таблице переходов соседние столбцы соответствуют единичным изменениям аргументов (входов) с целью применения матрицы Вейча–Карно. Применим эту матрицу к табл. 4, где можно воспользоваться двумя объединениями клеточек с 1: две правых клеточки дают слагаемое $(K \cdot \text{не}C)$ и правая нижняя с левой нижней клеточкой дают $(\Pi \cdot \text{не}C)$, получаем функцию $\Pi = K \cdot \text{не}C + \Pi \cdot \text{не}C$. Аналогично по табл. 5 получается функция $\Pi = K + \Pi \cdot \text{не}C$.

Применим свертку таблицы истинности для получения функции по табл. 4.

Таблица 23

Свертка таблицы истинности по табл. 4

К	С	П	f		c	f
0	0	1	1		0	1
1	0	0	1			
1	0	1	1			
0	0	0	0		0	0
0	1	0	0		1	0
1	1	0	1		1	1
1	1	1	1			
0	1	1	1			
1	3	1			1	1

$f = K + П$; $П = неС \cdot f = неС \cdot (K + П)$. Это соответствует выражению, полученному ранее по табл. 4, но с вынесением неС за скобку.

Интересно применить свертку таблицы истинности к не полностью определенной матрице по табл. 3. Начинаем с таблицы истинности 24.

Таблица 24

Не полностью определенная таблица истинности по табл. 3

К	С	П	d		К	d			С	e
0	0	1	1		0	1			0	1
1	0	0	0		1	0				
1	0	1	1		1	1				
0	0	0	0		0	0			0	0
0	1	0	0						1	0
0	1	1	0						1	1

На табл. 24 приведено два варианта решения задачи. Если сделать первую подстановку $d = неС \cdot П$, то далее $П = К + d = К + неС \cdot П$, что уже было получено, здесь отдано предпочтение переменной $К$. Если же сделать первую подстановку $e = К + П$, то, как показано справа в табл. 24, $П = неС \cdot e = неС \cdot (К + П)$. Здесь отдано предпочтение переменной $С$.

5. СХЕМА ИНФОРМАЦИОННЫХ ПОТОКОВ

Многие информационные системы представляют собой сети линий передачи информации (каналов) между различными устройствами, преобразующими эту информацию. Один пример такой сети – логические схемы, рассмотренные на рис. 16. В такой сети по каждой линии передается значение одной двоичной переменной, это минимальное количество информации – один бит. В других сетях по линиям (каналам связи) передается значительно большее количество информации. В качестве примера можно взять микропроцессор $BC1$, изучавшийся ранее [4]. При выполнении каждой команды в этом устройстве используются различные элементы и линии связи. Ниже на рис. 19 представлены потоки информации в микропроцессоре $BC1$ при выполнении команды $Q = 2 \cdot (D + B)$. Отдельные элементы микропроцессора связаны четырехбитными линиями (четырьмя проводами, передающими по одному биту). При выполнении команд используется лишь часть всех имеющихся элементов системы и соответственно часть линий связи. Рассматриваются потоки информации, которые непрерывны, то есть исправный канал передачи информации должен передать ее без потерь. На рис. 19 для указанной вверху операции потоки информации показаны утолщенными линиями на структурной схеме микропроцессора. Слагаемое D поступает с внешнего одноименного входа D на селектор входов

АЛУ и далее на вход АЛУ R . Второе слагаемое B читается из регистра общего назначения (РОН) по адресу B в Рег. B и далее проходит через селектор входов АЛУ на вход АЛУ S . Полученная сумма F поступает на сдвигатель Q , где выполняется сдвиг на один разряд вправо, что равносильно умножению на 2, и далее записывается в регистр Q .

$$Q = 2 \cdot (D+B)$$

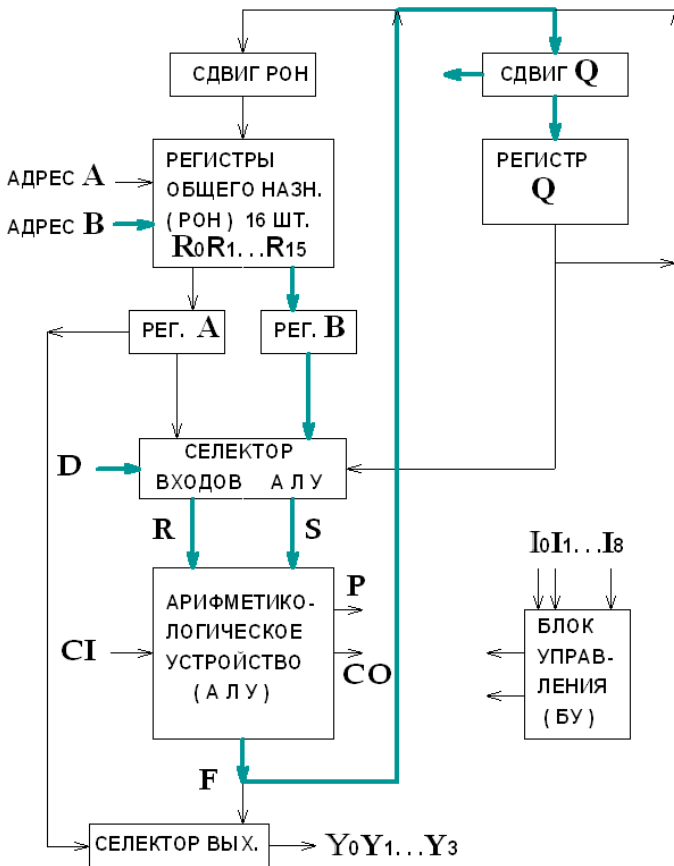


Рис. 19. Потоки информации в микропроцессоре К 1804 BC1

Лабораторная работа № 1

Минимизация логических схем и функций

Логическую (комбинационную) схему можно представить как одноканальный преобразователь входной информации по входам A, B, C, D, \dots в выходную на Z , аналогично рис. 6, б. Этот преобразователь информации выполняет логическую функцию $Z(A, B, C, D, \dots)$, которая задается таблицей истинности или матрицей Вейча–Карно, если число входных переменных не более 4. Различные варианты задания приведены в таблице заданий, каждому номеру варианта соответствует 5 десятичных чисел, которые следует преобразовать в 4-разрядные двоичные числа. В формате A, B, C, D они соответствуют значениям входов, которые дают на выходе $Z = 1$. Все остальные значения входов, которые не указаны, соответствуют $Z = 0$.

Таблица заданий

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

Отчет о работе должен содержать:

- 1) минимизацию логической функции с помощью матрицы Вейча–Карно;
- 2) минимизацию логической функции методом Квайна–МакКласки;
- 3) минимизацию логической функции методом свертки таблицы истинности.

Все результаты должны соответствовать одной и той же логической функции, но по п. 3 она должна быть короче.

Лабораторная работа № 2 **Анализ работы логических функций и схем**

Чтобы проверить корректность полученной функции, нужно подставить в нее значения входных величин, содержащихся в задании, и убедиться, что получается правильный результат. Аналогично проверяется правильность работы логической схемы, полученной на основании логической функции.

Отчет о работе должен содержать:

- 1) 4 картины логической схемы, аналогично рис. 16, при значении $Z = 1$;
- 2) 4 картины логической схемы, аналогично рис. 16, при значении $Z = 0$.

Необходимо обратить внимание на выделенные значения входных переменных по методу свертки таблицы истинности. В этом случае изменение значения только одной выделенной переменной вызывает изменение значения выхода.

Лабораторная работа № 3 **Схема синхронизации**

По варианту задания из лабораторной работы № 1 и соответствующей логической схеме построить схему синхронизации и диаграмму сигналов в соответствии с рис. 17. Объяс-

нить работу логической схемы по графику сигналов счетчика импульсов, выборочно проверить правильность логической схемы по наиболее влиятельному сигналу (переменной).

Отчет о работе должен содержать в соответствии с вариантом задания:

1) схему синхронизации с логической схемой и диаграмму сигналов;

2) схему синхронизации с дешифратором, аналогично рис. 18.

Лабораторная работа № 4 **Схема информационных потоков**

По варианту задания, приведенному ниже в таблице, построить схему информационных потоков в микропроцессоре К 1804 ВС1, аналогично рис. 19. В таблице заданий каждому варианту соответствует своя команда микропроцессора.

Таблица заданий

№ вар.	Операция	№ вар.	Операция
0	$A = 2(Q - B)$	14	$A = 2(B - D)$
1	$A = 2(Q + B)$	15	$A = 1/2(D - B)$
2	$A = 2(B - Q)$	16	$A = 1/2(D + B)$
3	$A = 1/2(Q - B)$	17	$A = 1/2(B - D)$
4	$A = 1/2(Q + B)$	18	$B = 2(D - A)$
5	$A = 1/2(B - Q)$	19	$B = 2(D + A)$
6	$B = 2(Q - A)$	20	$B = 2(A - D)$
7	$B = 2(Q + A + 1)$	21	$B = 1/2(D - A)$
8	$B = 2(A - Q)$	22	$B = 1/2(D + A)$
9	$B = 1/2(Q - A)$	23	$B = 1/2(A - D)$
10	$B = 1/2(Q + A)$	24	$Q = 2(Q - B)$
11	$B = 1/2(A - Q)$	25	$Q = 2(Q + B)$
12	$A = 2(D - B)$	26	$Q = 2(B - Q)$
13	$A = 2(D + B + 1)$	27	$Q = 1/2(Q - B)$

Окончание таблицы

№ вар.	Операция	№ вар.	Операция
28	$Q = 1/2(Q + B + 1)$	38	$A = 2(B - A)$
29	$Q = 1/2(B - Q)$	39	$A = 1/2(A - B)$
30	$Q = 2(Q - A)$	40	$A = 1/2(A + B)$
31	$Q = 2(Q + A)$	41	$A = 1/2(B - A)$
32	$Q = 2(A - Q)$	42	$B = 2(B - A)$
33	$Q = 1/2(Q - A)$	43	$B = 2(B + A + 1)$
34	$Q = 1/2(Q + A)$	44	$B = 2(B - Q)$
35	$Q = 1/2(A - Q)$	45	$B = 1/2(B - A)$
36	$A = 2(A - B)$	46	$B = 1/2(B + A)$
37	$A = 2(A + B)$	47	$B = 1/2(B - Q)$

Использованные операции микропроцессора:

$$F = R + S + Cl;$$

$$F = R - S - 1 + Cl;$$

$$F = S - R - 1 + Cl.$$

Список источников

1. Василевский, А.В. Организация и функционирование ЭВМ: метод. пособие / А.В. Василевский. – Минск: БНТУ, 2009. – 64 с.: ил.
2. Колдуэлл, С. Логический синтез релейных устройств / С. Колдуэлл. – М.: ИЛ, 1962. – 737с.: ил.
3. Василевский, А.В. Цифровые функционально-логические элементы: практикум / А.В. Василевский. – Минск: БНТУ, 2009. – 35с.: ил.
4. Василевский, А.В. Организация и функционирование ЭВМ: практикум / А.В. Василевский. – Минск: БНТУ, 2009. – 31с.: ил.

ПРИЛОЖЕНИЕ

Проектирование цикловой схемы управления производственным механизмом

Производственный механизм (рис. П1) состоит из стойки E , каретки F и поворотного стола G , на котором установлен инструмент в точке T . Движения механизма рассматриваются в системе координат XOY , неподвижной относительно основания. Стойка перемещается вдоль оси X вперед (увеличение X , движение E) и назад (движение $-E$) с помощью винтовой передачи от асинхронного электродвигателя ME . На стойке E установлен асинхронный двигатель MF , который с помощью винтовой передачи перемещает каретку F вдоль оси Y вверх (движение F) и вниз (движение $-F$). На каретке F установлен асинхронный двигатель MG , который с помощью зубчатой передачи поворачивает стол G по часовой стрелке (движение G) и против (движение $-G$). Посредством указанных движений нужно осуществить перемещение инструмента по определенной траектории (точка T), исходя из требований технологического процесса, в соответствии с циклограммой на рис. П2, где стрелки с номерами показывают порядок выполнения отдельных движений во времени. Далее рассматривается выполнение работы на примере следующих движений по указанным номерам:

1	2	3	4	5	6
E	F	G	$-E$	$-F$	$-G$

Исходное положение механизма: стойка E в крайнем левом положении (X_{\min} , флажок введен в конечный выключатель $K1$, и он включен), каретка в крайнем нижнем положении (Y_{\min} , включен $K4$), поворотный стол в крайнем положении против часовой стрелки, включен $K7$. Из указанного исходного поло-

жения по сигналу «ПУСК» (кнопка П) начинается движение 1, в нашем случае E , при определенном значении X с помощью конечного, а вернее путевого, выключателя К2 подается сигнал начала движения 2 (F). Оба движения выполняются одновременно, причем первое ограничено конечным выключателем К3, а второе – конечным выключателем К5. Выключатели К3 и К5 могут сработать в разное время, однако следующее движение 3 (G) начинается только тогда, когда сработают и К3, и К5 (конъюнкция). По окончании движения 3 в соответствии с циклограммой на рис. П2 начинается сразу 2 движения: 4 и 5 ($-E$ и $-F$), причем движение 4 противоположно по направлению движению 1. Вторая часть движения 4 происходит только в том случае, если закончилось движение 5. Если движение 5 закончилось раньше, чем первая часть движения 4, тогда движение 4 происходит без остановки до конца (К1), иначе происходит временная остановка при срабатывании К2. По окончании движения 4 ($-E$) происходит движение 6 ($-G$), завершающее цикл всех движений механизма, он возвращается в исходное положение и может снова повторять указанный цикл движений неограниченное число раз.

В табл. П1 приведены исходные данные для различных вариантов работ – номер варианта указан в столбце 1, для которого задана своя последовательность движений в порядке возрастания номеров на стрелках рис. П2. Может быть 4 варианта исходного положения.

1. Стойка находится в крайнем левом положении X_{\min} , работает К1; каретка – в крайнем нижнем положении Y_{\min} , работает К4, это соответствует рассматриваемому примеру.

2. Стойка – в крайнем правом положении X_{\max} ; каретка – в крайнем нижнем положении Y_{\min} .

3. Стойка – в крайнем левом положении X_{\min} ; каретка – в крайнем верхнем положении Y_{\max} , работает К5.

4. Стойка – в крайнем правом положении X_{\max} ; каретка – в крайнем верхнем положении Y_{\max} . Обозначения конечных вы-

ключателей могут быть иными.

Таблица П1

Варианты работы

№ вар.	Исходные X_{\min}, Y_{\min}	№ вар.	Исходные X_{\max}, Y_{\min}	№ вар.	Исходные X_{\min}, Y_{\max}
0	<i>EF G-E-F-G</i>	18	<i>G-EF-GE-F</i>	36	<i>E-FG-EF-G</i>
1	<i>EGF-E-G-F</i>	19	<i>G-EF-G-FE</i>	37	<i>EG-F-E-GF</i>
2	<i>EGF-E-F-G</i>	20	<i>GF-E-GE-F</i>	38	<i>EG-F-EF-G</i>
3	<i>EF G-E-G-F</i>	21	<i>GF-E-G-FE</i>	39	<i>E-FG-E-GF</i>
4	<i>EF-F-EG-G</i>	22	<i>G-EE-GF-F</i>	40	<i>E-FF-EG-G</i>
5	<i>EG-G-EF-F</i>	23	<i>GF-F-G-EE</i>	41	<i>EG-G-E-FF</i>
6	<i>FEG-F-E-G</i>	24	<i>-EFG E-F-G</i>	42	<i>-FEG F-E-G</i>
7	<i>FGE-F-G-E</i>	25	<i>-EGFE-G-F</i>	43	<i>-FGE F-G-E</i>
8	<i>FGE-F-E-G</i>	26	<i>-EGFE-F-G</i>	44	<i>-FGE F-E-G</i>
9	<i>FEG-F-G-E</i>	27	<i>-EFG E-G-F</i>	45	<i>-FEG F-G-E</i>
10	<i>FE-E-FG-G</i>	28	<i>EF-F-EG-G</i>	46	<i>-FE-EFG-G</i>
11	<i>FG-G-FE-E</i>	29	<i>-EG-GEF-F</i>	47	<i>-FG-GFE-E</i>
12	<i>GEF-G-E-F</i>	30	<i>F-EG-FE-G</i>	48	<i>GE-F-G-EF</i>
13	<i>GEF-G-F-E</i>	31	<i>FG-E-F-GE</i>	49	<i>GE-F-GF-E</i>
14	<i>GFE-G-E-F</i>	32	<i>FG-E-FE-G</i>	50	<i>G-FE-G-EF</i>
15	<i>GFE-G-F-E</i>	33	<i>F-EG-F-GE</i>	51	<i>G-FE-GF-E</i>
16	<i>GE-E-GF-F</i>	34	<i>F-EE-FG-G</i>	52	<i>GE-E-G-FF</i>
17	<i>GF-F-GE-E</i>	35	<i>FG-G-F-EE</i>	53	<i>G-FF-GE-E</i>

Каждое отдельное движение механизма (*E, F, G*) ограничено двумя конечными выключателями, например К4 – К5 для движения *F*, рис. П1, оно может выполняться независимо от других движений с помощью своего привода (асинхронного двигателя) и своей передачи. Для получения сигнала в середине движения применяется дополнительный путевой выключатель, например К2 для движения *E* на рис. П1. Заданная для каждого варианта в таблице П1 последовательность движений и исходное положение, а также взаимное расположение движений в пространстве (рис. П1) и взаимосвязь движений (рис. П2) однозначно определяют траекторию движения инструмента (точки *T*, на рис. П1). Эта траектория для рассмат-

риваемого примера представлена на рис. П3. Из исходного положения X_{\min} , Y_{\min} , точка 0 выполняется движение E , далее добавляется движение F в точке a , точке c соответствует момент, когда завершились движения E , F . Движение F может завершиться раньше, чем движение E , крайний в этом отношении случай представлен точкой b' . Аналогично, движение E может окончиться раньше движения F , крайний случай представлен здесь точкой b , область возможных траекторий от a до c заштрихована. Движение G представлено дугой cd на угол $\Delta\alpha$, далее происходят обратные движения аналогично описанным выше, которые заканчиваются выходом в исходное положение 0.

Отчет по первой части работы должен содержать:

- 1) кинематическую схему механизма, аналогичную рис. П1, с расстановкой и обозначением конечных и путевого переключателей;
- 2) циклограмму работы механизма, аналогичную рис. П2, с обозначениями движений вместо их номеров, а также конечных и путевого переключателей вместо букв траектории;
- 3) построенную траекторию движения точки T с обозначениями конечных и путевого выключателей в узловых точках;
- 4) поясняющий текст.

Вторая часть работы заключается в построении цикловой схемы автоматического управления механизмом по циклограмме, приведенной на рис. П2, которая должна обеспечить построенную в первой части траекторию движения инструмента. Поскольку все отдельные движения выполняются в прямом и обратном направлениях, то используется реверсивная схема включения асинхронных электродвигателей привода, приведенная на рис. П4 для двигателя ME . Включение пускателя E дает вращение двигателя в прямом направлении, включение пускателя $-E$ – в обратном направлении. Для предотвращения одновременного включения обоих пускате-

лей, приводящего к двойному короткому замыканию между фазами B и C , должна использоваться электрическая блокировка этой пары пускателей путем включения катушки каждого из пускателей через нормально замкнутый контакт другого пускателя. В этом случае включение одного из пускателей разрывает цепь катушки другого пускателя. Два остальных электродвигателя MF и MG включаются аналогично соответственно пускателями $F, -F, G, -G$ с описанной выше блокировкой от междупазного короткого замыкания.

Последовательность сигналов и движений механизма при его автоматической работе представлена на рис. П5. В исходном положении для нашего примера включены конечные выключатели $K1, K4, K7$, все остальное выключено. В качестве конечных и путевых переключателей (выключателей) используются бесконтактные выключатели типа БВК-24, остальные элементы схемы управления контактные, их устанавливают в шкаф управления, где они работают достаточно надежно. Кнопка «Пуск» включает своим контактом Π пускатель E , включающий электродвигатель ME , начинается движение E , стойка идет вперед. Пускатель E может включиться только тогда, когда включен $K1$, так как его нормально открытый (замыкающий) контакт включен последовательно с Π , рис. П6. После этого кнопка Π может разомкнуться (палец на нажатой кнопке держать не нужно), да и $K1$ также разомкнется, так как движение стойки вперед выведет флажок из $K1$. Чтобы при этом не отключился двигатель ME , пускатель E включает свою катушку своим же контактом E на самопитание (самоблокировку) через нормально замкнутый контакт $K3$ для последующего отключения двигателя ME , но пока он включен. Самопитание показано на рис. П5 дужкой со стрелкой. По достижении флажком путевого переключателя $K2$ он срабатывает и включает пускатель двигателя F , который также включается на самопитание, так как $K2$ вскоре отключается, поскольку дальнейшее движение стойки E выводит флажок из $K2$.

Движение каретки F вверх отключает $K4$ и продолжается до срабатывания $K5$, который отключает пускатель F (отключение обозначено чертой сверху на рис. П5). На схеме управления (рис. П6) это осуществляется размыканием нормально замкнутого контакта $K5$ в цепи самоудержания катушки F . Аналогично этому срабатывание $K3$ отключает пускатель E и движение E (стойка вперед). Пускатель G включается только в том случае, когда работают и $K3$, и $K5$, его отключение происходит при срабатывании $K6$, благодаря нормально замкнутому контакту $K6$ в цепи G , рис. П6. Катушки всех пускателей включены через нормально замкнутые (размыкающие) контакты противоположных по направлению движения механизма пускателей с целью блокировки от междуфазного короткого замыкания при одновременном включении в случае неисправности схемы управления. Срабатывание $K6$, завершающее движение G (по часовой стрелке), включает также нормально открытым (закрывающим) контактом пускатели $-E$ и $-F$, начиная оба эти движения одновременно. Движение $-F$ завершается срабатыванием $K4$, нормально открытый контакт $K4$ при этом шунтирует нормально закрытый контакт $K2$ в цепи $-E$, и последующее срабатывание $K2$ в этом случае не отключает $-E$, это происходит при срабатывании $K1$. Нормально открытым контактом $K1$ включается пускатель $-G$, осуществляющий поворот стола против часовой стрелки до срабатывания $K7$, это завершает цикл работы.

Отчет по второй части должен содержать:

- 1) схему включения электродвигателя, аналогичную рис. П4;
- 2) последовательность сигналов и движений (схему сигналов), аналогичную рис. П5;
- 3) электрическую принципиальную схему системы управления, аналогичную рис. П6;
- 4) поясняющий текст.

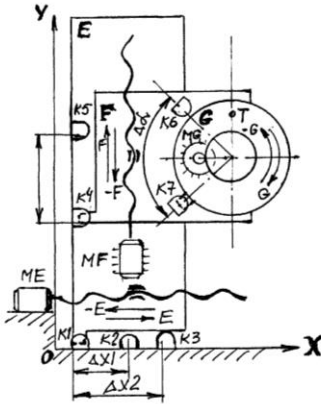


Рис. П1. Кинематическая схема механизма

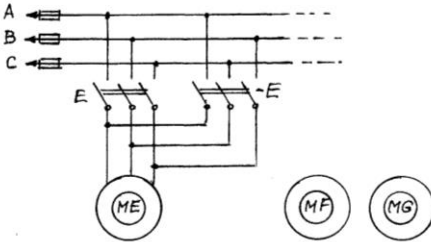
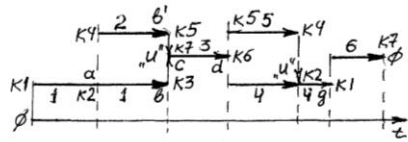


Рис. П4. Схема включения электродвигателей



Рис. П5. Последовательность сигналов и движений цикла / схема сигналов/



E, F, G, -E, -F, -G

Рис. П2. Циклограмма работы механизма

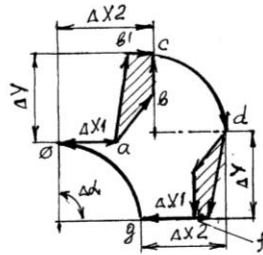


Рис. П3. Траектория движения точки T

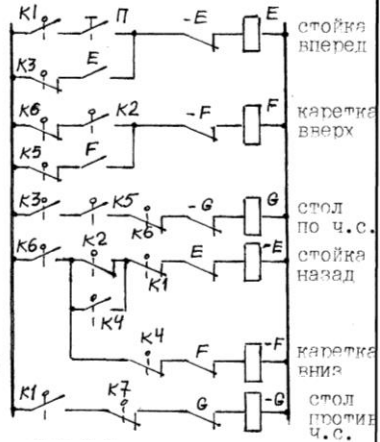


Рис. П6. Схема управления

Учебное издание

БОРОДУЛЯ Алексей Валентинович
ВАСИЛЕВСКИЙ Александр Витальевич
КОЧУРОВ Вадим Александрович

ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ
ИНТЕГРИРОВАННЫХ САПР

Методическое пособие

В 3 частях

Часть 1

Редактор И.Ю. Никитенко
Компьютерная верстка Д.А. Исаева

Подписано в печать 22.12.2010.

Формат 60×84 ¹/₁₆. Бумага офсетная.

Отпечатано на ризографе. Гарнитура Таймс.

Усл. печ. л. 3,20. Уч.-изд. л. 2,50. Тираж 150. Заказ 146.

Издатель и полиграфическое исполнение:
Белорусский национальный технический университет.

ЛИ № 02330/0494349 от 16.03.2009.

Проспект Независимости, 65. 220013, Минск.