

Г. Цеманек
Н. Зетанек

Австрия

ПОСЛЕДОВАТЕЛЬНОСТНАЯ АСИНХРОННАЯ ЛОГИКА

1. Введение

По асинхронной, или не зависящей от времени, логике до сих пор опубликовано очень мало работ; большинство из них принадлежит Д. Мюллеру, профессору прикладной математики Университета Иллинойса. Настоящая статья также основана на его работе. Она представляет собой компромисс между чистой теорией [1] и практическим применением [2]. В ней используется та форма логики одноктактных и многотактных (последовательностных) схем, которая описана в обзорной работе автора [3]. Целью статьи является изложение основных проблем и свойств асинхронной логики и переключательной алгебры; однако не предполагается рассматривать практические логико-технические задачи асинхронной техники.

2. Время в логике

Время играет важную роль в приложениях логики к процессам переработки информации: эти процессы всегда представляют собой временные последовательности. Однако возможны различные способы трактовки времени.

1. Время можно рассматривать отдельно, на особом уровне. В этом случае проблему можно ограничить логикой одноктактных схем. Теоретически для осуществления такой логики достаточно одного вида элементарного блока: функции Шеффера либо функции Пирса. Однако на практике обычно используются три вида блоков: конъюнкция (элемент И), дизъюнкция (элемент ИЛИ) и отрицание.

2. Другая возможность заключается в том, чтобы строго квантовать время. В этом случае имеется множество равноотстоящих моментов времени, в которые такт за тактом выдаются результаты. Это — синхронная последовательностная логика с фиксированной частотой часов. Здесь

используются те же блоки, что и в логике одноктактных схем, и, кроме того, блок задержки с единичной задержкой.

3. В третьем и наиболее общем случае моменты работы не зависят от частоты каких-либо часов: следующий такт может начаться, как только текущий такт выдал свой результат, и, следовательно, закончился. Никакой синхронизации не нужно, длина такта может зависеть от локальных условий. Между тем в синхронной системе единица времени должна быть больше максимального времени релаксации. Будет показано, что асинхронную логику можно реализовать на блоках последовательностной логики, но с произвольным временем задержки (или временем реакции).

За преимущества, связанные с возможностью использования произвольного времени реакции, приходится расплачиваться. Прежде всего, уже невозможно представлять одно из двух значений истинности неактивным состоянием физических элементов; оба значения истинности должны быть представлены активными состояниями. Во-вторых, сигнал о завершении каждого такта должен подаваться на вход блока и служить началом следующего такта. Аналогично, сигнал о завершении выполнения команды должен подаваться на вход и пропускать следующую команду.

Должно обрабатываться все большее количество информации, и это ведет к увеличению используемых материалов. Более быстрое функционирование и больший расход материалов характерны для параллельной системы (в то же время последовательные системы используют меньше материалов за счет уменьшения быстродействия); асинхронная или не зависящая от времени логика лучше соответствует параллельным системам и, вообще говоря, не согласуется с последовательными системами.

3. Структура асинхронной логики

Три основных элемента информации асинхронной логики:

неактивный

$$0) X = (z)$$

отсутствие информации, пауза, время ожидания;

активные

$$1) X = 0$$

значение истинности 0;

$$2) X = 1$$

значение истинности 1.

Это может навести на мысль о трехзначной логике. Однако, во-первых, автор не верит в трехзначную логику; во-вторых, активные и неактивные элементы информации не связаны никакой операцией; в-третьих, в логике наиболее часто выбор происходит между двумя возможностями. Пара двузначных переменных дает вполне пригодное решение проблемы. Это решение, кроме того, содержит четвертую, часто полезную, комбинацию — значение «безразлично» (и 0 и 1 удовлетворяют логическим условиям):

Временной
 Подписано
 9.11.1979
 Назначено
 Синхронизация

неактивный элемент

0) $X = (0,0)$

пауза;

активные элементы

1) $X = (1,0)$

значение истинности 0;

2) $X = (0,1)$

значение истинности 1;

3) $X = (1,1)$

значение истинности «безразлично».

Там, где это нужно, компоненты переменных обозначаются индексами 0 и 1:

$$X_i = (X_{i0}, X_{i1}).$$

Асинхронные элементы информации в общем случае появляются в различные моменты времени. Необходимо, следовательно, сочетать все логические функции с некоторой памятью. Однако возможны исключения.

Возможны два основных вида памяти (рис. 1):

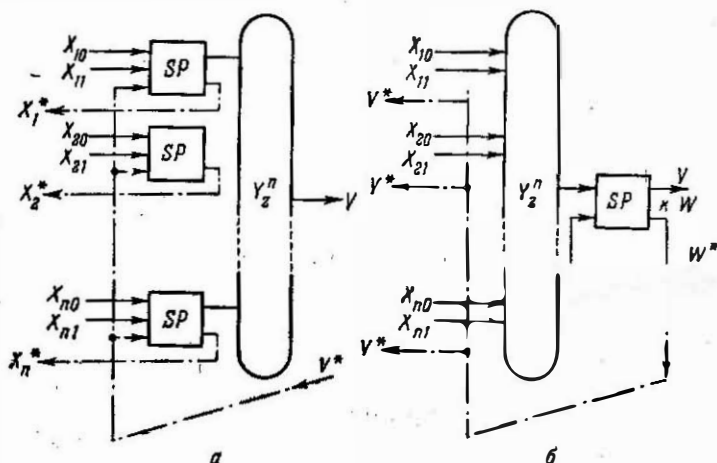


Рис. 1. Память и сигнал завершения

1) память получает входной сигнал (не зависящий от других входов) и отпирает предшествующий элемент, начиная тем самым новый такт (входная память, рис. 1, а);

2) память ждет, пока не поступят все входные сигналы, выдает результат и помнит его до тех пор, пока следующий блок не сигнализирует о приеме (выходная память, рис. 1, б).

При прочих равных условиях лучше пользоваться памятью второго вида, так как этот вид требует одной, максимум двух ячеек памяти на блок, тогда как память первого вида требует одной или двух ячеек на каждую входную переменную.

завершение такта в асинхр. устройстве

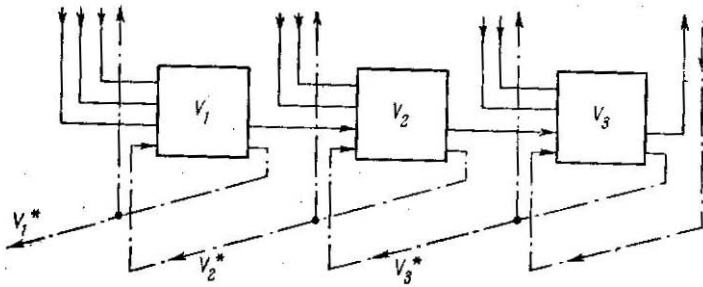


Рис. 2. Структура асинхронной логики

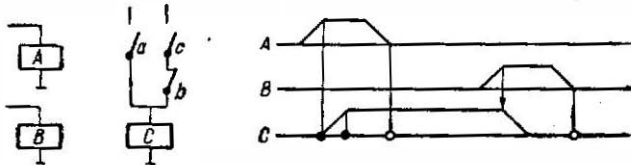


Рис. 3. Временная диаграмма переключений, предложенная Виткелем

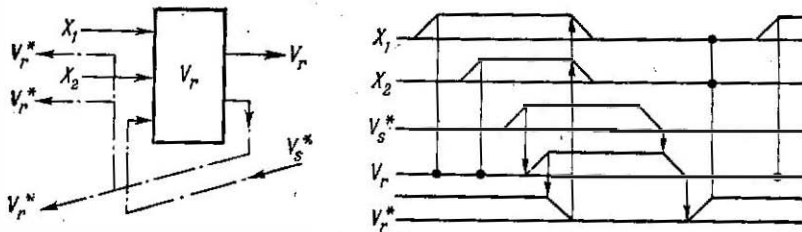
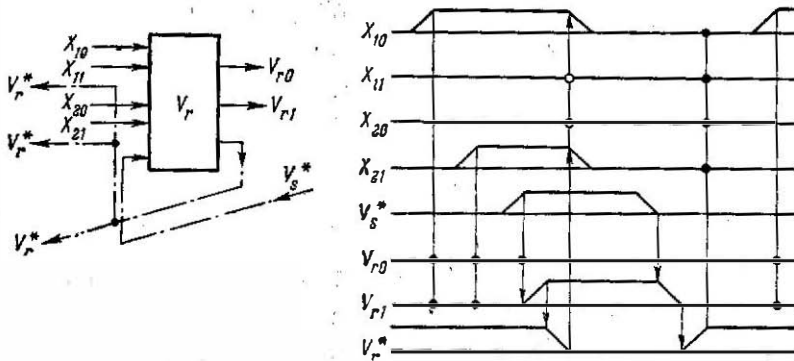


Рис. 4. Применение диаграммы к блоку асинхронной логики (внизу — упрощенное представление)

Обратная связь с предшествующими блоками может иметь только два значения: «завершено» и «не завершено». Эти значения должны формироваться в соответствии с содержанием памяти и посылаются в направлении, противоположном потоку логической информации. Принципиальная структура цепи блоков дана на рис. 2. Очевидно, что разветвления (т. е. подачу выхода блока V_r на вход нескольких последующих блоков) нельзя производить без специального блока разветвления, так как различные значения V_r должны быть согласованы (выходную память можно устранить, только когда все последующие блоки сработали; ср. блок разветвления в разд. 5).

Между прочим, память асинхронного блока и сигнал завершения — это свойства, которые могут иметь место и в синхронных элементах; в некоторых динамических схемах удобно применять запоминающие свойства реактивных сопротивлений и полупроводников (это верно для ограниченного периода времени); сигнал синхронизации во многих отношениях можно сравнить с сигналом завершения.

4. Представление временной структуры

Временная структура асинхронной логики довольно сложна. Мы будем пользоваться особым видом ее представления с помощью диаграммы, которая была предложена Винкелем [4,5] для телефонных схем, т. е. для типичной асинхронной логики. На диаграмме (рис. 3) устойчивым состояниям соответствуют значения 0 и 1 (величины соответствующих напряжений или тока не указываются), а переходы изображаются наклонными линиями; в течение этого времени выход определяется предшествующим значением истинности. Горизонтальные оси указывают время, причем масштаб времени не обязательно фиксирован (представление может быть даже нелинейным). Взаимодействие различных переменных указано стрелками, готовящиеся изменения обозначены малыми кружками: черными — переходы от 0 к 1 и белыми — переходы от 1 к 0.

Этот вид представления можно использовать без каких-либо изменений для асинхронных логических функций. Переменные можно изображать либо отдельно — X_{i0} и X_{i1} , либо вместе — X_i (это обозначение может относиться к большему числу переменных или даже ко всем сразу). В последнем случае функциональное поведение следует рассматривать при построении и чтении диаграммы. Отдельная функция изображена на рис. 4, цепь функций — на рис. 5. Начало и конец цепи здесь не указаны; ими могут являться ячейки памяти (ср. разд. 6).

На рис. 5 показано, как задержки, содержащиеся в начале, в конце или где-нибудь внутри цепи, останавливают работу в ожидании всей необходимой информации. В остальных случаях работа происходит с максимальной скоростью.

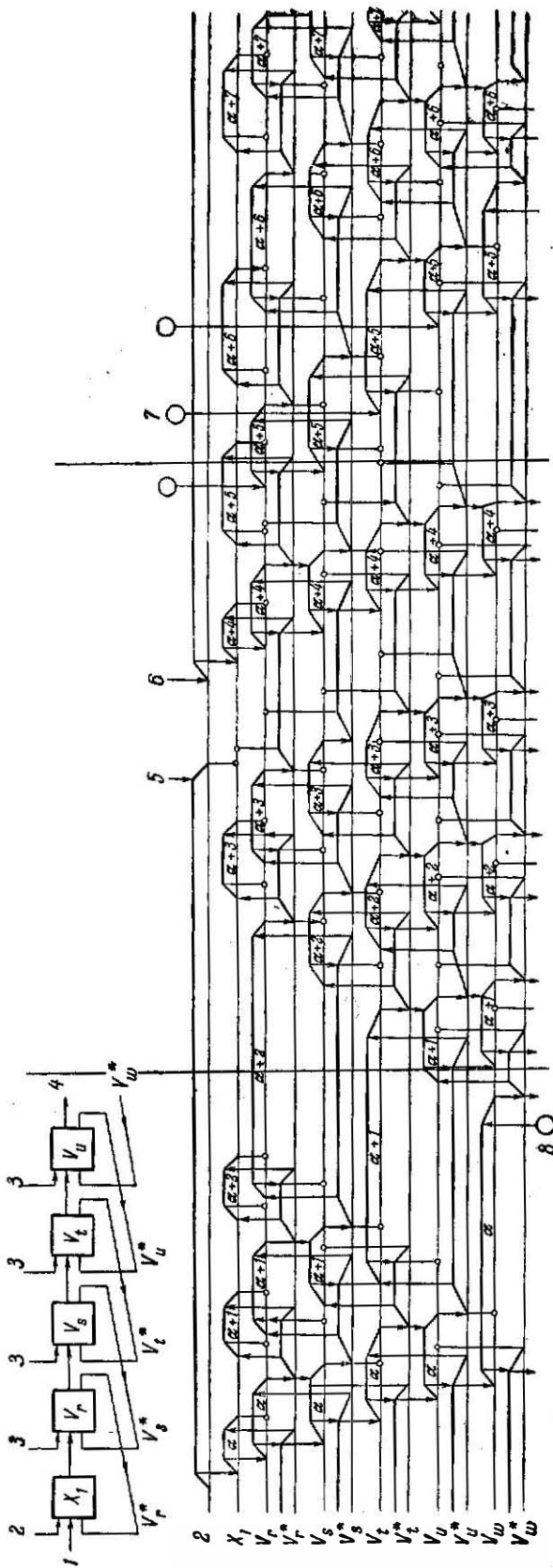


Рис. 5. Применение диаграммы к цепи блоков

1 — память, среда и т. д.; 2 — пуск; 3 — другие входы; 4 — другие входы; 5 — команда останова; 6 — задерживаемые значения; 7 — выход не завершен

5. Основные блоки асинхронной логики

Логические блоки — это обычные блоки двузначной логики. Они используются в форме, в которой можно допустить произвольную задержку, не нарушая логической корректности процесса.

Отрицание. В блоке отрицания нет необходимости: достаточно поменять местами две компоненты логической переменной:

$$X_i = (X_{i0}, X_{i1}); \quad \bar{X}_i = (X_{i1}, X_{i0}).$$

C-функция. Так как все функции асинхронной логики должны иметь ячейки памяти, то необходима основная ячейка памяти, полученная из классической (последовательно-синхронной) ячейки памяти (ср. [3], стр. 140) и учитывающая специфику записи и стирания в асинхронной логике.

Эта ячейка называется C-элементом; ее состояние изменяется с 0 на 1, если все входы равны 1, и возвращается в 0, если все входы равны 0. На рис. 6 показан C-элемент для 3 переменных; для другого числа переменных существенных отличий нет.

Функция n переменных. Обычная функция Y_z^n n переменных, не обладающая особыми запоминающими свойствами, требует

- 1) построения 2^n членов совершенной нормальной формы;
- 2) построения функций $Y_{z_0}^n$ и $Y_{z_1}^n$ на выходе этих членов;
- 3) двух C-элементов для запоминания результата и
- 4) построения функции завершения X^* для входов X .

Соответствующая структура показана на рис. 7; вследствие свойств симметрии различных функций n переменных требуют не 2^{2^n} различных блоков, а только 2^{n-1} , так как на входах и на выходе можно использовать все комбинации сигналов с отрицаниями и без.

Для двух переменных достаточно, таким образом, только двух блоков или даже одного с переключателем (рис. 8), чтобы иметь возможность реализовать все нетривиальные функции. Переключатель выбирает между двумя группами: дизъюнкции — конъюнкции и антивалентность — эквивалентность; число членов этих групп — 8 и 2 соответственно; член определяется выбором нулей и единиц на входах и на выходе. Соединения для функций X_0 и X_1 и $X_0 \equiv X_1$ соответственно показаны на рис. 8.

Вспомогательные блоки. На рис. 9 изображены некоторые вспомогательные блоки, предложенные Мюллером, — блок начала линии (а), блок ее конца (б), блоки задержки для отдельного провода (в) и для двойного провода (г) и единичный (д) и двойной (е) блоки разветвления.

6. Специальные блоки асинхронной логики

Триггер. В асинхронной логике триггер несколько усложняется. Мюллер [2] дал пример с несколькими элементами (рис. 10). Переменные B и D включаются попеременно после четного и нечетного числа входных

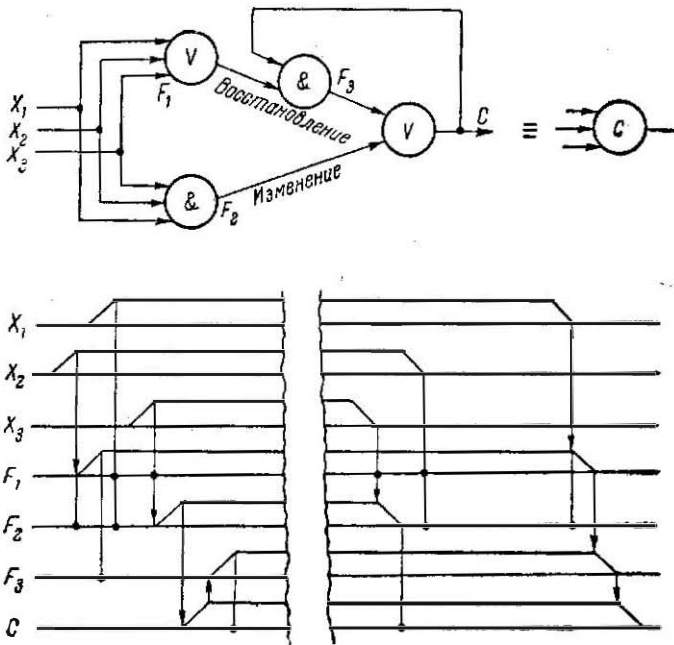


Рис. 6. С-элемент

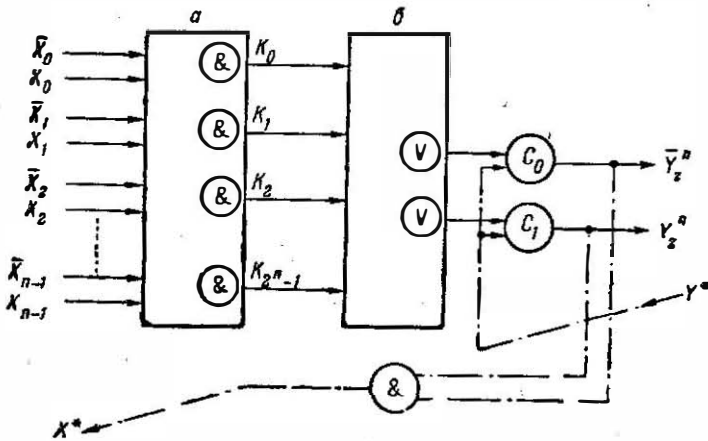


Рис. 7. Функция n переменных

a — построение 2^n членов совершенной нормальной формы; b — построение функций Y_z^n

импульсов X ; переменные A и C подают эти импульсы попеременно на B и D . Выход Y имеет примерно ту же длительность, что и импульсы X . В этом случае сигнал завершения не нужен; если же это необходимо, следует добавить два C -элемента, и сигнал завершения должен идти обратно на вход X .

Триггер на рис. 10 имеет только один вход. Другой вид триггера имеет два входа — один для перехода от 0 к 1, другой для перехода от 1 к 0 (рис. 11). В этом случае нужна логика считывания и сигнал завершения.

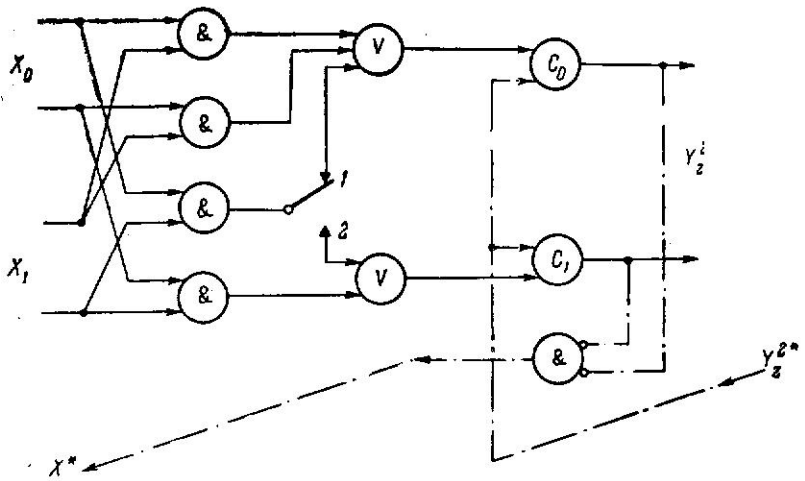


Рис. 8. Функция двух переменных

1 — дизъюнкции-конъюнкции; 2 — антивалентность-эквивалентность

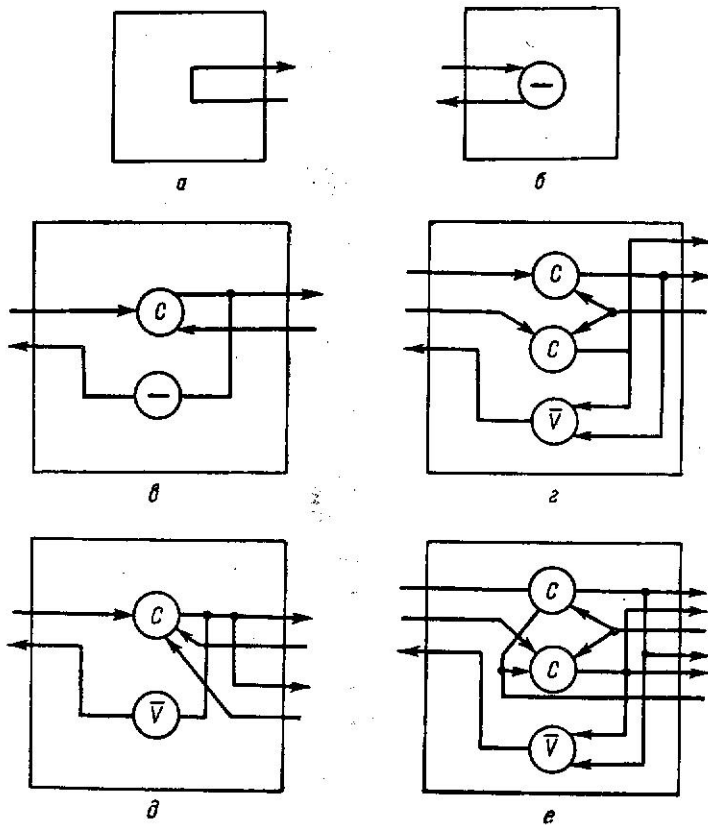


Рис. 9. Вспомогательные блоки

Ячейка памяти асинхронной логики отличается от обычной. Наиболее хорошо согласуется с принципами асинхронных схем схема с двумя частичными блоками памяти для запоминания отдельно нулей и единиц (рис. 12); можно, конечно, свести ее к одному блоку памяти. На рис. 12

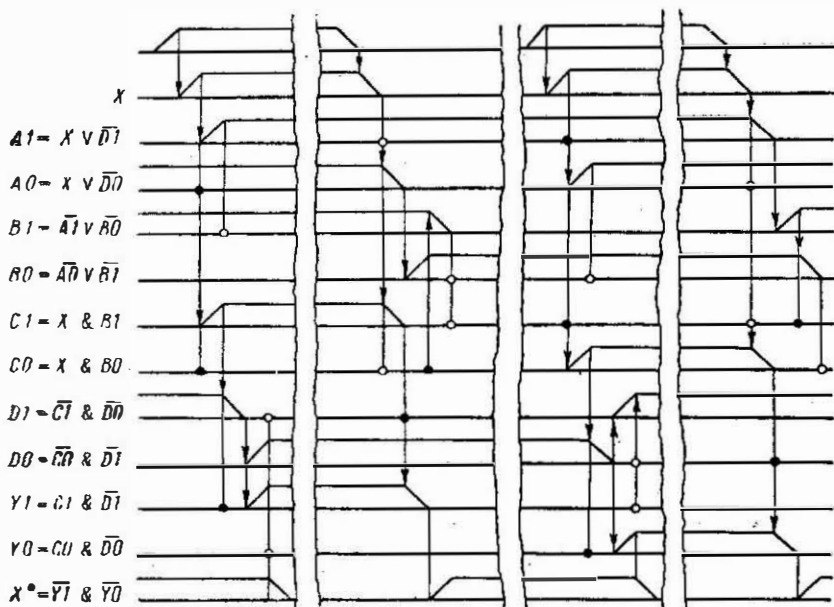
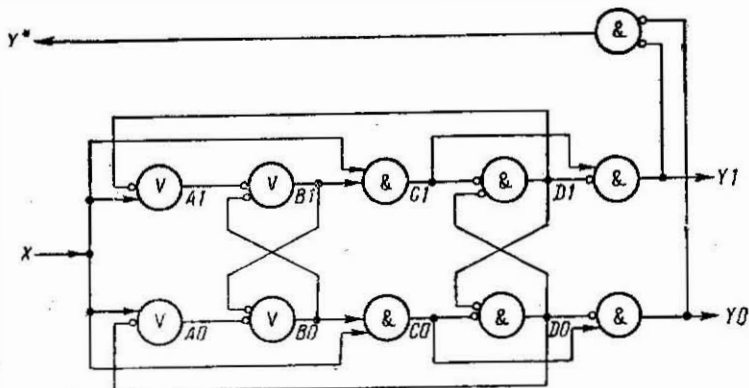


Рис. 10. Триггер с одним входом

предполагается, что имеются шины для чтения и записи и что указанная ячейка памяти возбуждается по специальному управляющему входу. При специально построенной временной структуре сигнал стирания может появляться раньше любой операции записи (так что нет необходимости в особой команде стирания).

Можно отметить, что физическая реализация ячеек памяти, соответствующих данной логической структуре, может оказаться тяжелой и до-

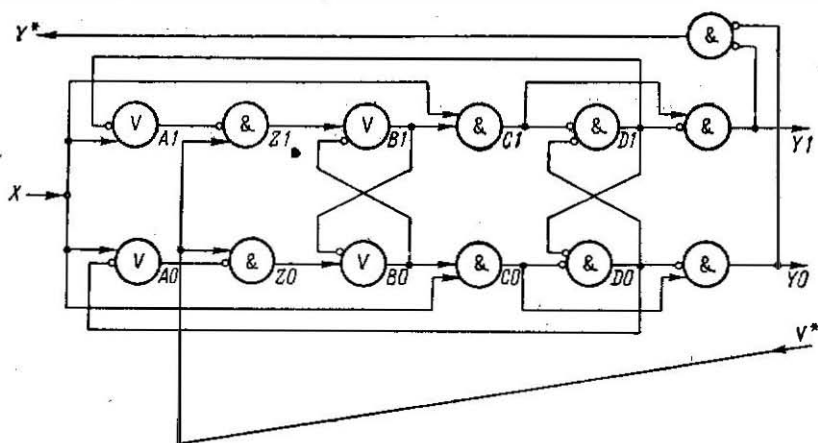


Рис. 11

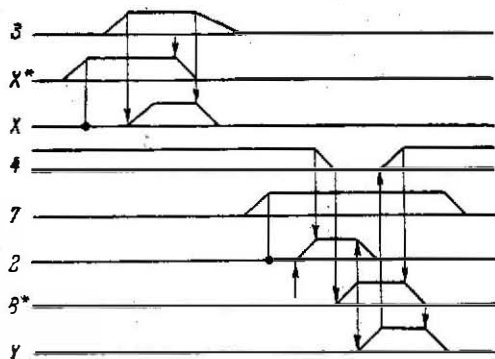
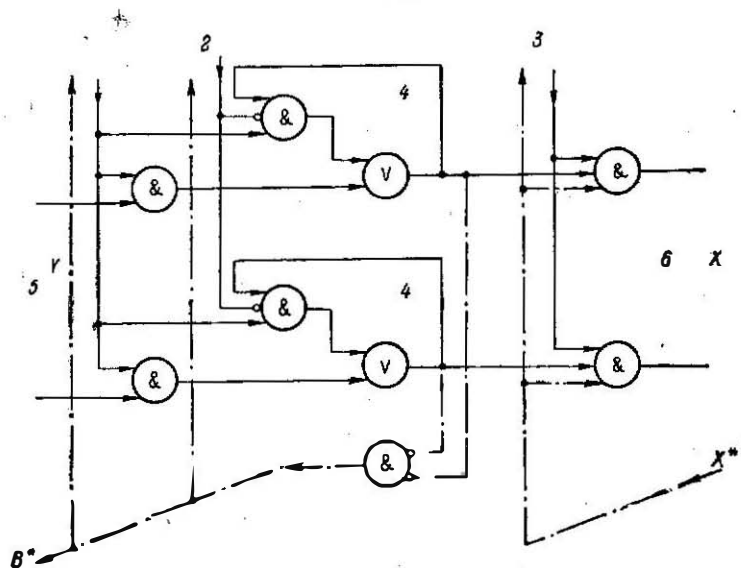


Рис. 12. Ячейка памяти

1 — адрес записи и стирания; 2 — стирание; 3 — адрес считывания; 4 — ячейка памяти; 5 — сигнал записи; 6 — сигнал считывания; 7 — адрес записи и считывания

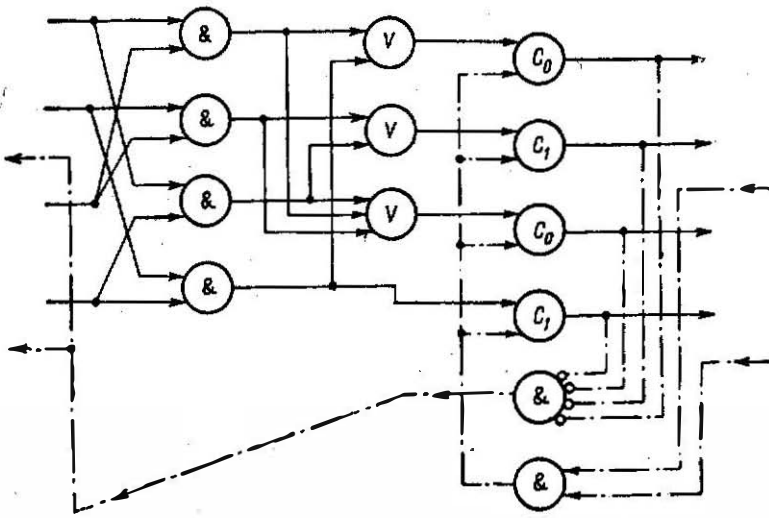


Рис. 13. Двоичный полусумматор

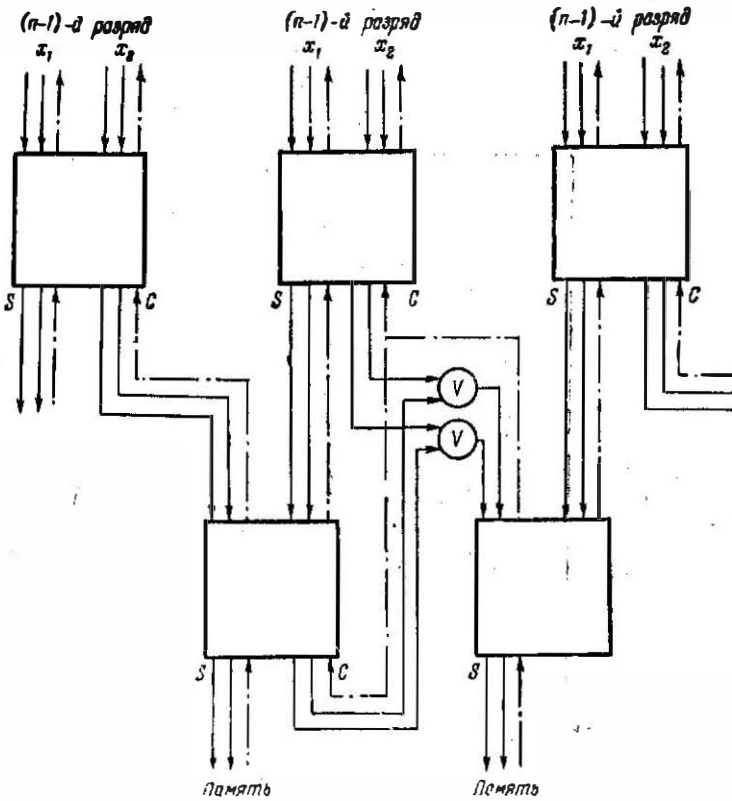


Рис. 14. Двоичный сумматор, состоящий из двух полусумматоров

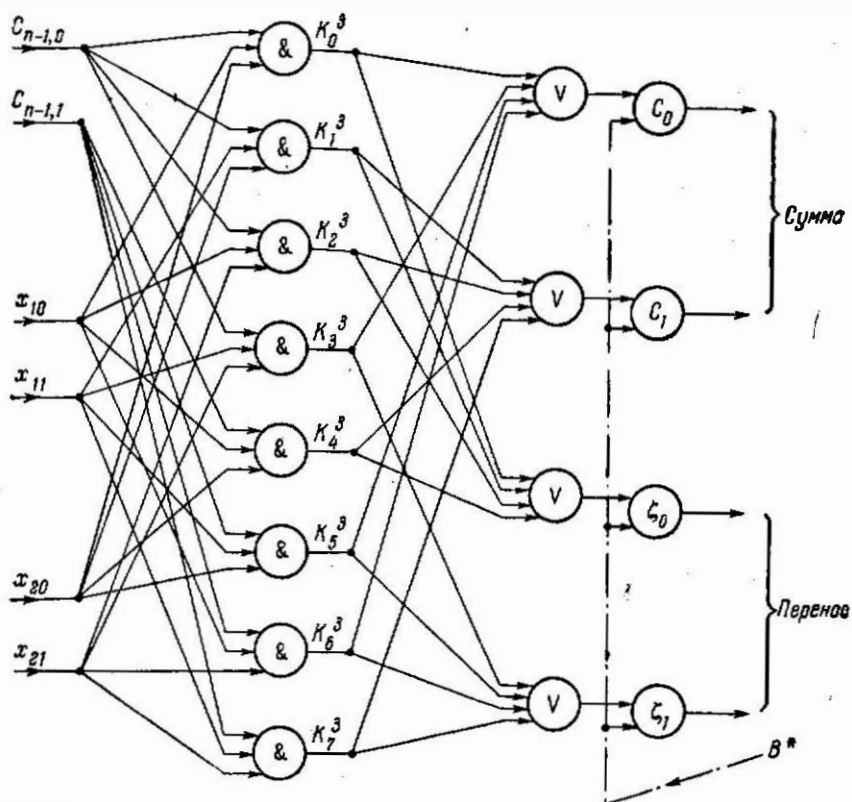


Рис. 15. Двоичный сумматор как двойная функция трех переменных

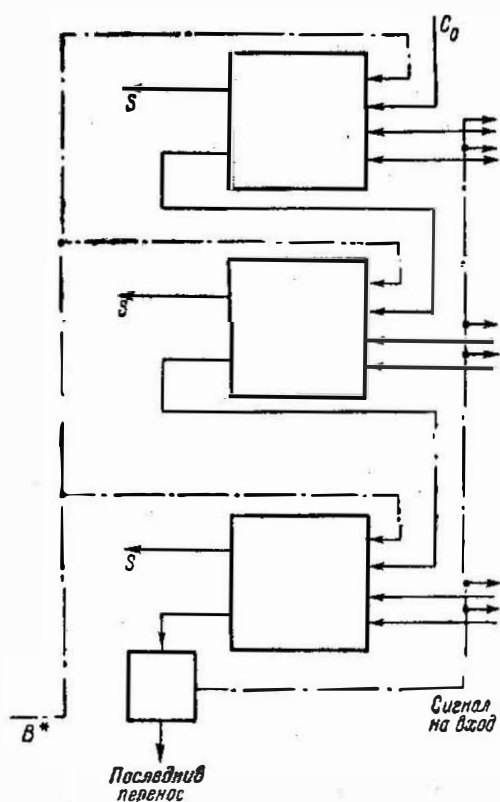


Рис. 16. Двоичный сумматор с кольцевым переносом

рогостоящей. Классические элементы типа магнитных сердечников здесь неприменимы вследствие их сравнительно больших временных параметров.

Сумматоры. В этой работе нет возможности детально рассматривать асинхронные системы. В качестве иллюстраций приведем лишь примеры двоичных полусумматоров и сумматоров, а также сумматора с кольцевым переносом.

Полусумматор использует две функции двух переменных (рис. 13) и легко может быть получен из рис. 8.

Сумматор можно получить объединением двух полусумматоров (рис. 14) или как двойную функцию трех переменных (рис. 15).

Сумматор с кольцевым переносом (рис. 16) можно получить из релейного сумматора Цузи (Zuse). Из этой схемы видно, что при определенных условиях ограничения быстродействие определяется уже не временными параметрами элементов, а скоростью передачи сигнала, которая не может превышать скорости света. Наносекунду можно интерпретировать (по аналогии со световым годом в астрономии) как световой фут.

7. Заключительные замечания

Асинхронная логика, описанная в этой работе, имеет те же свойства принципиальной реализуемости и универсальности, что и обычная последовательностная логика, с той лишь разницей, что скорость работы соответствует локальным условиям системы и что поток информации управляется не постоянной синхронизирующей частотой, а сигналами завершения отдельных событий.

Это не единственный способ синтеза схем асинхронной логики. Предлагались теоретические системы различной природы [6]; утверждалось, что они применимы также и для недетерминистских соотношений. Мы не будем обсуждать, возможно ли это и имеет ли это смысл; во всяком случае, система, описанная в этой работе, ограничена детерминистскими соотношениями.

ЛИТЕРАТУРА

1. D. E. Muller, W. S. Bartky. A theory of asynchronous circuits. — Proc. Intern. Symposium on the theory of switching. Cambridge, 1957, p. 1. Cambridge, Harvard Univ. Press, 1959, p. 204.
2. D. E. Muller. Asynchronous logics and application to information processing. — Proc. Symposium on the Application of Switching Theory in Space Technology. Sunnyvale, 1962, in press.
3. H. Zemanek. Logische Algebra und Theorie der Schaltnetzwerke. — Taschenbuch der Nachrichtenverarbeitung, T. 4. Berlin, Springer Verlag. 1962.
4. E. Winkel. Die Darstellung der Schaltvorgänge in einer Relais — Wöhlanlage nach dem Schaltzeitplan. — Zeitschrift für Fernmeldetechnik. Werk — und Gerätebau, 1939, 20, № 8, S. 117; № 9, S. 138.
5. E. Winkel. Einführung in die Wähltechnik. München und Berlin. R. Oldenbourg, 1942, p. 115.
6. C. A. Petri. Fundamentals of a theory of asynchronous information flow. IFIP Congress. Munich, 1962» Session II (Automata Theory). Amsterdam. North Holland Publishing Co. 1963.