



ЦИФРОВЫЕ СХЕМЫ

УЧЕНИЦА СТАРШЕЙ ШКОЛЫ ПО ИМЕНИ АИСАКА ЦУМО, ЛЮБЯЩАЯ РАЗБИРАТЬ ЭЛЕКТРОННЫЕ ПРИБОРЫ, ПЫТАЕТСЯ УСТРОИТЬСЯ НА РАБОТУ В КОМИССИОННЫЙ МАГАЗИН БЫТОВОЙ ТЕХНИКИ И ИЗ СТАРЫХ НЕНУЖНЫХ ДЕТАЛЕЙ ДЕЛАЕТ ЭЛЕКТРОННЫЙ КУБИК С ГОРЯЩИМИ ЛАМПОЧКАМИ.

КНИГА ПОЗНАКОМИТ ЧИТАТЕЛЯ С ОСОБЕННОСТЯМИ И ТИПАМИ ЦИФРОВЫХ СХЕМ, БАЗОВЫМИ МЕТОДАМИ ЛОГИЧЕСКОГО ПРОЕКТИРОВАНИЯ И ЛЕЖАЩИМИ В ИХ ОСНОВЕ ИДЕЯМИ. ФОРМАТ МАНГИ НАИЛУЧШИМ ОБРАЗОМ ПОДХОДИТ ДЛЯ ОПИСАНИЯ МИРА ЦИФРОВЫХ СХЕМ, НАПОЛНЕННОГО ГРАФИКОЙ: ТАБЛИЦАМИ ИСТИННОСТИ, ЧЕРТЕЖАМИ ЛОГИЧЕСКИХ СХЕМ, КАРТАМИ КАРНО, ДИАГРАММАМИ СОСТОЯНИЙ. В ЗАКЛЮЧЕНИЕ ВЫ УЗНАЕТЕ О НОВЕЙШИХ МЕТОДАХ ПРОЕКТИРОВАНИЯ СХЕМ: ЯЗЫКАХ ОПИСАНИЯ АППАРАТУРЫ И ВЫСОКОУРОВНЕВОМ СИНТЕЗЕ.

МАНГА БУДЕТ ИНТЕРЕСНА ШКОЛЬНИКАМ СТАРШИХ КЛАССОВ И СТУДЕНТАМ, А ТАКЖЕ ВСЕМ, КТО УВЛЕКАЕТСЯ ЭЛЕКТРОНИКОЙ И СХЕМОТЕХНИКОЙ.

Интернет-магазин: www.dmkpress.com

Книга-почтой: orders@aliants-kniga.ru

Оптовая продажа: "Альянс-книга".
(499)782-3889. books@aliants-kniga.ru



ISBN 978-5-97060-661-2



9 785970 606612 >

ЦИФРОВЫЕ СХЕМЫ

Аmano Хидэхару
Мэгуро Кодзи

ЗАНИМАТЕЛЬНАЯ ЭЛЕКТРОНИКА
ЦИФРОВЫЕ СХЕМЫ

DMK

Аmano Хидэхару
Мэгуро Кодзи
Office sawa



Ohmsha



Занимательная электроника
Цифровые схемы
Манга

ОБРАЗОВАТЕЛЬНАЯ МАНГА

ЗАНИМАТЕЛЬНАЯ ЭЛЕКТРОНИКА ЦИФРОВЫЕ СХЕМЫ

Аmano Хидэхару
Мэгуро Кодзи

Перевод

А. Б. Клионского

Научный редактор

С. А. Сенченков



Ohmsha



Москва
DMK Пресс, 2018

УДК 530.145

ББК 22.314

A21

Аmano Хидэхару

A21 Занимательная электроника. Цифровые схемы. Манга / Аmano Хидэхару (автор), Мэгуро Кодзи (худож.); пер. с яп. А. Б. Клионского, научн. ред. С. А. Сенченков — М.: ДМК Пресс, 2018. — 224 с. : ил. — (Серия «Образовательная манга»). — Доп. тит. л. яп.

ISBN 978-5-97060-661-2

Ученица старшей школы по имени Аисака Ицумо, любящая разбирать электронные приборы, пытается устроиться на работу в комиссионный магазин бытовой техники. Книга познакомит читателя с особенностями и типами цифровых схем, причинами их широкого распространения, базовыми методами логического проектирования и лежащими в их основе идеями.

УДК 530.145

ББК 22.314

Original Japanese edition
Manga de wakarū Dijitaru Kairo (The Manga Guide to Digital Circuit)
By Amano Hideharu (Author), Meguro Koji (Illustrator) and
Office sawa (Producer)
Published by Ohmsha, Ltd.
Russian language edition copyright © 2018 by DMK Press

Все права защищены. Никакая часть этого издания не может быть воспроизведена в любой форме или любыми средствами, электронными или механическими, включая фотографирование, ксерокопирование или иные средства копирования или сохранения информации, без письменного разрешения издательства.

ISBN 978-4-274-06958-1 (яп.)
ISBN 978-5-97060-661-2 (рус.)

Copyright © 2013 by Amano Hideharu and Office sawa
© Перевод, оформление, издание, ДМК Пресс, 2018

ПРЕДИСЛОВИЕ

Нас окружают разнообразные цифровые устройства: компьютеры, смартфоны, планшеты и т. п. И «последний оплот» аналоговой техники — телевидение, тоже недавно перешло на цифровой формат. Почему же цифровые схемы получили такое развитие? Каковы принципы их действия и как их проектируют? В этой книге наглядно, без сложных формул алгебры логики, изложены основы проектирования цифровых схем.

Цифровые схемы очень просты. Это — мир, в котором нет ничего, кроме 0 и 1. Также просто устроены их «кирпичики» — логические элементы. В главе 1 я познакомлю вас с цифровыми схемами, окружающими нас в повседневной жизни, в главе 2 объясню, почему они так быстро развились и вытеснили аналоговую технику, в главе 3 опишу простые и доступные каждому методы проектирования схем комбинационной логики, в которых состояние выходов определяется только текущим состоянием входов. Глава 4 посвящена упрощению схем. Преобразование схем с помощью алгебры логики — это самый сложный раздел университетского курса, но здесь я познакомлю вас с интуитивно понятными методами, доступными даже неискущённым в математике читателям. В главе 5 речь пойдёт о проектировании элементов памяти и схем последовательной логики, а в заключение вы узнаете о новейших методах проектирования схем: языках описания аппаратуры, о высокоуровневом синтезе и др.

Формат манги лучше, чем обычная книга, подходит для описания мира цифровых схем, наполненного графикой: таблицами истинности, чертежами логических схем, картами Карно, диаграммами состояний и т. п. Материал изложен по возможности простым языком, но при этом описано и внутреннее устройство логических схем на основе множества примеров проектирования, а разделы «Дополнительная информация» содержат также знания довольно высокого уровня о современных технологиях проектирования и др. Надеюсь, что эта манга будет полезной как для интересующихся электроникой, так и для людей с другой специализацией, занятых в этой области.

В работе над книгой мне очень помогли художник Мэгуро Кодзи и ответственная за оформление г-жа Савада Савако, предложив такие наглядные рисунки и глубокие примеры, каких ещё не было в учебниках по цифровой технике. Хочу поблагодарить также коллектив издательства Ohmsha за предоставленную возможность взяться за перо и за ценные советы, которые очень помогли мне в работе.

СОДЕРЖАНИЕ

Глава 1

ЧТО ТАКОЕ ЦИФРОВЫЕ СХЕМЫ? 1

- Цифровые интегральные схемы 21
- Программируемая пользователем вентильная матрица 22
- Объяснение терминов главы 1 24

Глава 2

ЦИФРОВЫЕ И АНАЛОГОВЫЕ СХЕМЫ 25

1. Цифровая и аналоговая техника 26

- От аналоговых схем к цифровым 26
- Представление об аналоговой и цифровой технике 28

2. Почему цифровые схемы победили? 33

- Что такое логические схемы? 33
- Логические вентили работают просто 35
- Цифровые схемы побеждают числом! 37
- В чём преимущество цифровых схем? 39
- Проектировать цифровые схемы — просто! 44
- Что такое алгебра логики? 49
- Объяснение терминов главы 2 52



Глава 3

СТРОИМ КОМБИНАЦИОННУЮ СХЕМУ 53

1. **Таблица истинности, нотация MIL** 54
 - Мажоритарные цифровые схемы 54
 - Строим таблицу истинности из L и H 58
 - Что такое нотация MIL? 64
 - Активный низкий и активный высокий 68
 - Законы де Моргана 72
 - Базовые логические вентили в нотации MIL 75
2. **Строим мажоритарную схему** 78
 - Строим схему по таблице истинности (порядок проектирования ДНФ) 79
 - Что такое КМОП? 85
 - Принцип работы полевого МОП-транзистора 88
 - Объяснение терминов главы 3 91

Глава 4

КАК УПРОСТИТЬ СХЕМУ? 93

1. **Используем карты Карно** 94
 - Нерациональные схемы 94
 - Исследуем карту Карно 98
 - Выделяем группы единиц 102
 - Важные моменты выделения групп 106
 - Усложняем задачу выбора ресторана 108
2. **Безразличное состояние** 111
 - Идентифицируем длинные месяцы 111
 - Десятичные и двоичные числа 114
 - Проектируем схему идентификации длинных месяцев 118
 - Безразличное состояние 120

3. А если выходов несколько?	124
• Объединяем выходы	125
• Проектируем схему индикатора электронного кубика	129
• Сумматоры и вычитатели	137
• Объяснение терминов главы 4	143

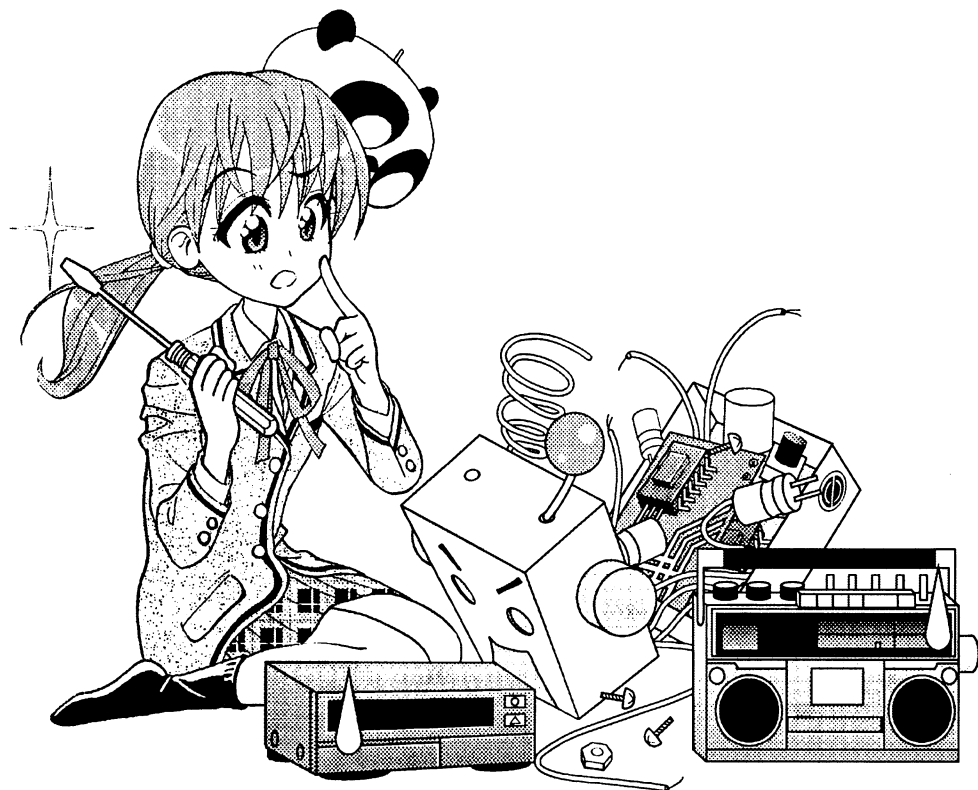
Глава 5

СТРОИМ ПОСЛЕДОВАТЕЛЬНЫЕ СХЕМЫ

1. Что такое последовательные схемы?	146
• У последовательных схем есть память	148
2. D-триггеры	153
• Триггер — это качели	153
• D-триггер и тактовый сигнал	155
• Что такое регистр?	167
3. Проектируем электронный кубик	171
• Электронный кубик — последовательная схема	171
• (1) Чертим диаграмму состояний	174
• (2) Назначаем состояниям двоичные числа	177
• (3) Проектируем комбинационную схему	181
• Устройство D-триггера	187
• Различные триггеры	190
• Объяснение терминов главы 5	193
Послесловие. Проектирование цифровых схем с помощью HDL	208
Список литературы	211
Предметный указатель	212

ГЛАВА 1

ЧТО ТАКОЕ ЦИФРОВЫЕ СХЕМЫ?



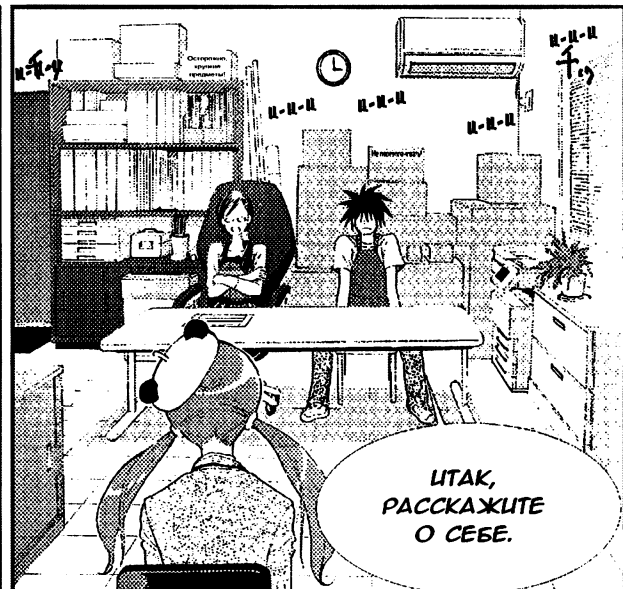
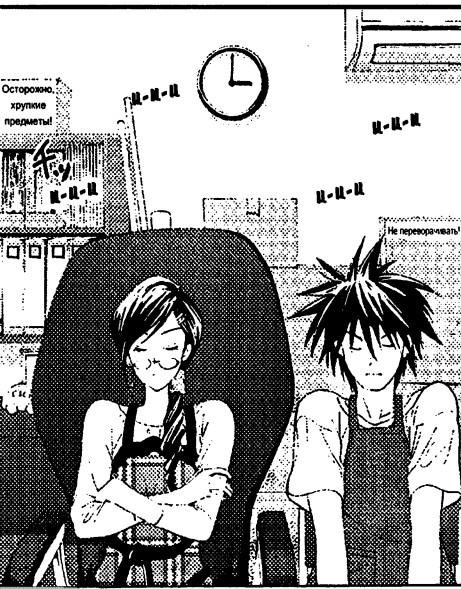
Где-то
в Токио

бытовой
ТЕХНИКИ

Комиссионный магазин
WOZ

WOZ

Добро пожаловать!
Купим дорого!



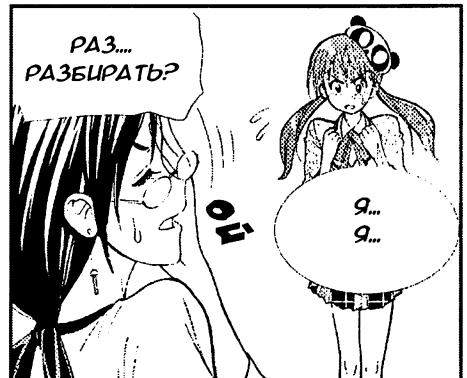


* Примерно соответствует 11-му классу в России.



* Шапка в форме панды.





...ОБОЖАЮ
ОТКРУЧИВАТЬ
ТУГЕ ВИНТЫ...

...НЕ СЛИШКОМ
СТАРЫХ,
КРАСИВЫХ
ЭЛЕКТРОПРИБОРОВ...

...ОТКРЫВАТЬ
КОРПУС И
СМОТРЕТЬ, ЧТО
ВНУТРИ

У СЕБЯ
ДОМА
ПОЧТИ
ВСЁ
РАЗОБРАЛА...

ПОДОЖДИ...
ЗНАЧИТ, ТЫ
У СЕБЯ ДОМА
ЧИНИШЬ
И СОБИРАЕШЬ
РАЗЛИЧНЫЕ
ЭЛЕКТРО-
ПРИБОРЫ?

ЧИНЮ?

СОБИРАЮ?

АА
НЕТ ЖЕ!

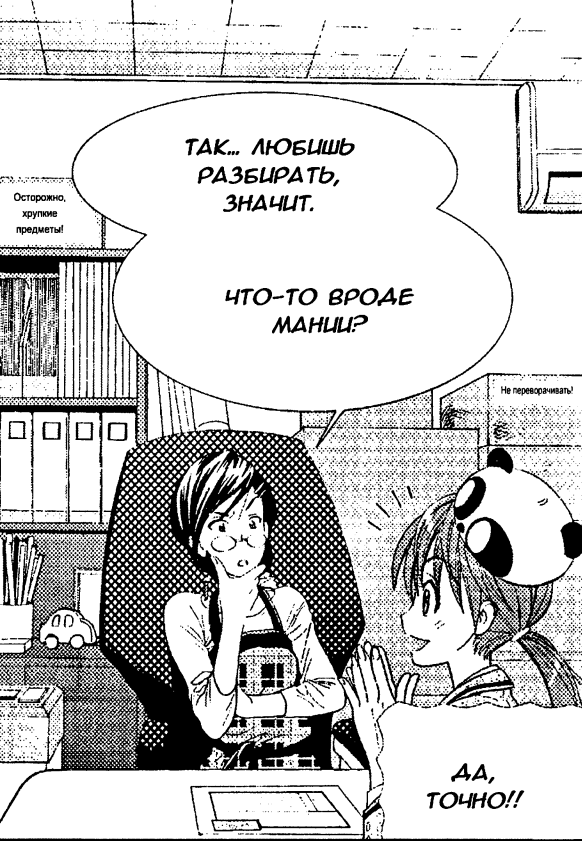
Я ЛЮБЛЮ
РАЗОБРАТЬ И
СМОТРЕТЬ НА
ТО, ЧТО
ОСТАЛОСЬ. ♡

ТИПА... ВОТ
КАКИЕ ОНИ
ТЕПЕРЬ...

МОЖЕТ, ТУТ
БОЛЬШЕ
ПОАХОДИТ
СЛОВО
"ЛОМАТЬ"?!

ДИРЕКТОР,
ЭТА ДЕВУШКА
ОПАСНА!
ОТКАЖЕМ ЕЁ!!

КАК БЫСТРО
ТЫ МНЕНИЕ
ИЗМЕНИЛ!



ТАК... ЛЮБИШЬ
РАЗБИРАТЬ,
ЗНАЧИТ.

ЧТО-ТО ВРОДЕ
МАНЦУР

Не переворачивай!

АА,
ТОЧНО!!



ОТВЁРТКУ
ВСЕГДА
С СОБОЙ
НОШУ.

А КОГДА РАЗБИРАЮ
ЧТО-НИБУДЬ,
СЕРАЦЕ ОТ
РАДОСТИ СКАЧЕТ!!!



СТРАННАЯ,
ОДНАКО...

НО...

...В ПОСЛЕДНЕЕ
ВРЕМЯ МНЕ ЭТОГО
УЖЕ МАЛО.
ВЕДЬ РАЗОБРАТЬ МОЖНО
ТОЛЬКО ДО КАКОГО-ТО
ПРЕДЕЛА.

ТАМ МНОГО КАКИХ-ТО
ЧЁРНЫХ КВАДРАТИКОВ НА
ЗЕЛЁНЫХ ПЛАСТИНАХ...

И МНЕ ХОЧЕТСЯ
УЗНАТЬ, ЧТО ЭТО ТАКОЕ,
ЧТО У НИХ ВНУТРИ.



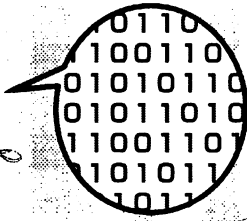
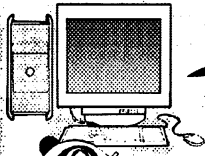
КАКОЕ-ТО
СТРАННОЕ
ЧУВСТВО
ОБУРЕВАЕТ.

ВОТ
Я РАЗЫЦАЮ,
СМОТРЮ, ЧТО
ВНУТРИ,
НО ВСЁ РАВНО
НИЧЕГО
НЕ ПОНИМАЮ...

ЧТОБЫ ПОНЯТЬ,
ЧТО ВНУТРИ
ЭЛЕКТРОННЫХ
УСТРОЙСТВ,
ТЕБЕ НАДО ИЗУЧИТЬ
ЦИФРОВЫЕ
СХЕМЫ.

ЦИФРОВЫЕ...
СХЕМЫ?

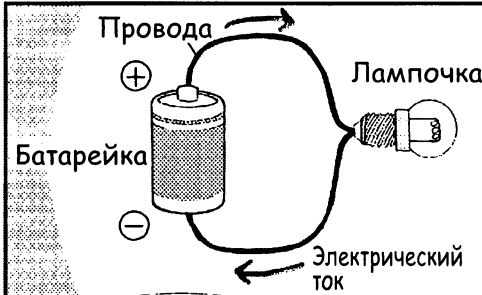




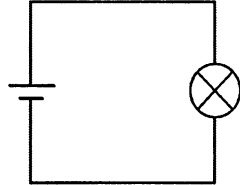
Ничего, кроме 0 и 1!

ТАК... КОМПЬЮТЕР - ЦИФРОВОЕ УСТРОЙСТВО, ТО ЕСТЬ МИР НУЛЕЙ И ЕДИНИЦ?

МЫСЛИШЬ ПРАВИЛЬНО.



Принципиальная схема электрической цепи очень проста!



ПРО ЭЛЕКТРИЧЕСКУЮ ЦЕПЬ НАМ ОБЪЯСНЯЛИ В ШКОЛЕ. ЭТО ПУТЬ ПРОТЕКАНИЯ ТОКА ЧЕРЕЗ РАЗЛИЧНЫЕ УСТРОЙСТВА?

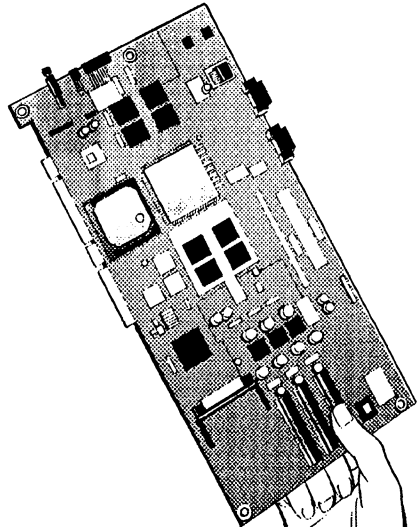
НО КАК ЭТО СВЯЗАНО С "ЦИФРОВЫМИ СХЕМАМИ"?

ТАКАБА, ТЫ ВЕДЬ ЭТО В ИНСТИТУТЕ ИЗУЧАЕШЬ, НЕ ТАК ЛИ?

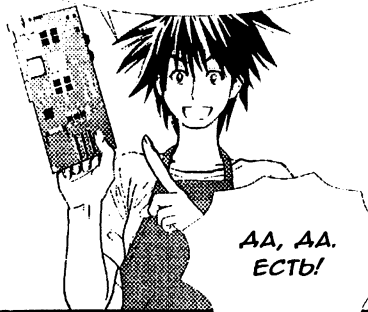
НУ-КА, РАССКАЖИ.

ЧТО?! Я?!

НУ, В ОБЩЕМ...

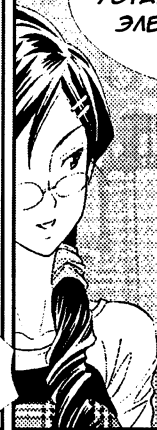


ПОГОВОРИМ О КОНКРЕТНЫХ
ИЗДЕЛИЯХ. ВНУТРИ КОМПЬЮ-
ТЕРОВ, БЫТОВОЙ ЭЛЕКТРО-
НИКИ ЕСТЬ "ПЕЧАТНЫЕ
ПЛАТЫ".



АА, АА.
ЕСТЬ!

А НА ПЛАТАХ
УСТАНОВЛЕНЫ РАЗЛИЧНЫЕ
ЭЛЕКТРОННЫЕ ДЕТАЛИ.

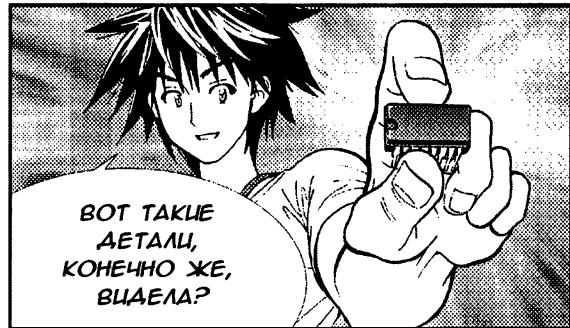


И ОНИ ПОДКЛЮЧЕНЫ
К ПРОВОДИКАМ,
ПО КОТОРЫМ
ТЕЧЁТ ТОК.

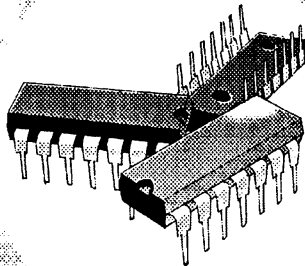
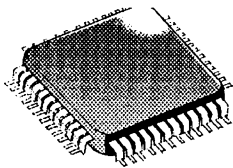
ГМ... АА,
И ВСЁ ЭТО
ВЫГЛЯДИТ ТАК
ЗАПУТАННО.



ВОТ ТАКИЕ
ДЕТАЛИ,
КОНЕЧНО ЖЕ,
ВИДЕЛА?



ВНУТРИ НОВЕЙШИХ УСТРОЙСТВ
ОНИ ВОТ ТАКИЕ - ТОНКИЕ И
МАЛЕНЬКИЕ.



А ВНУТРИ
УСТАРЕВШИХ - ВОТ ТАКИЕ.

АА. Я ЧАСТО ИХ ВИДЕЛА И
УДИВЛЯЛАСЬ!!
КВАДРАТЫ И
ПРЯМОУГОЛЬНИКИ, ПОХОЖИЕ
НА ЧЁРНЫХ ЖУКОВ
С СЕРЕБРИСТЫМИ
НОЖКАМИ!!



ЭТО!
ЭТО!

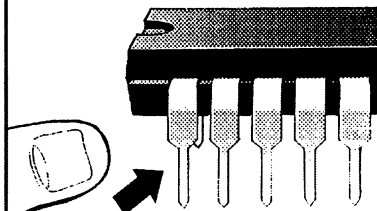
ЧЁРНЫХ
ЖУКОВ
.....?

ЭТО - ОЧЕНЬ ВАЖНЫЕ ЭЛЕКТРОННЫЕ ДЕТАЛИ, КОТОРЫЕ НАЗЫВАЮТ ИС (ИНТЕГРАЛЬНЫМИ СХЕМАМИ)*.



ДЛЯ НАЧАЛА - О ПРОСТЕЙШИХ, БАЗОВЫХ ИС, КОТОРЫЕ СУЩЕСТВУЮТ С ДАВНИХ ПОР.

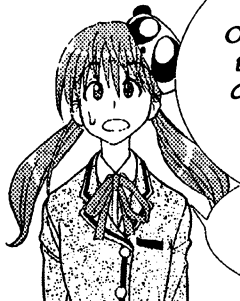
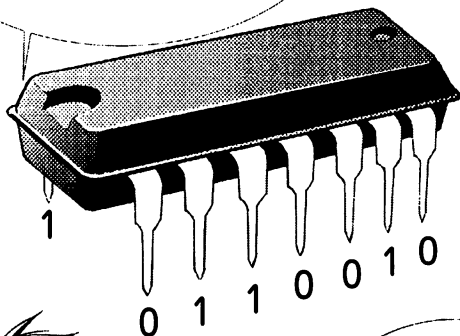
ЭТИ СЕРЕБРИСТЫЕ НОЖКИ НАЗЫВАЮТСЯ ВЫВОДАМИ



Выводы

* Сравнительно большие интегральные схемы иногда называют БИС.

ПО КАЖДОМУ ИЗ НИХ ТЕЧЁТ ТОК, ВВОДЯ ИЛИ ВЫВОДЯ ЦИФРОВУЮ ИНФОРМАЦИЮ В ВИДЕ 0 ИЛИ 1.



КАК, ОКАЗЫВАЕТСЯ, ВАЖНЫ ЭТИ СЕРЕБРИСТЫЕ НОЖКИ.

А Я И НЕ ЗНАЛА...

А ЭТО ЧЁРНОЕ - ЧТО?



ИС БЫВАЮТ РАЗНЫЕ, НО ВНУТРИ БАЗОВЫХ ЦИФРОВЫХ ИС...



...ЕСТЬ ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ (ВЕНТИЛИ), ВЫПОЛНЯЮЩИЕ ЛОГИЧЕСКИЕ ОПЕРАЦИИ.

Ток-шоу «Споры до утра»*



ЛО... ЛОГИЧЕСКИЕ?!



ОБЪЯСНЯЙСЯ ЛОГИЧНЕЕ!

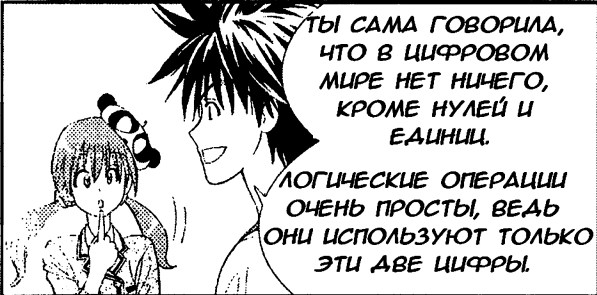
ЭТО ТЫ НЕЛОГИЧЕН!

ШУМ-ГАМ

ЧТО-ТО ВРОДЕ ЭТОГО?

ВЫБРОСЬ ИЗ ГОЛОВЫ ЭТУ ЧЕПУХУ!

* Непереводаемая игра слов с названием телепередачи и словом «логический». — прим. перев.



ТЫ САМА ГОВОРИЛА, ЧТО В ЦИФРОВОМ МИРЕ НЕТ НИЧЕГО, КРОМЕ НУЛЕЙ И ЕДИНИЦ.

ЛОГИЧЕСКИЕ ОПЕРАЦИИ ОЧЕНЬ ПРОСТЫ, ВЕДЬ ОНИ ИСПОЛЬЗУЮТ ТОЛЬКО ЭТИ ДВЕ ЦИФРЫ.



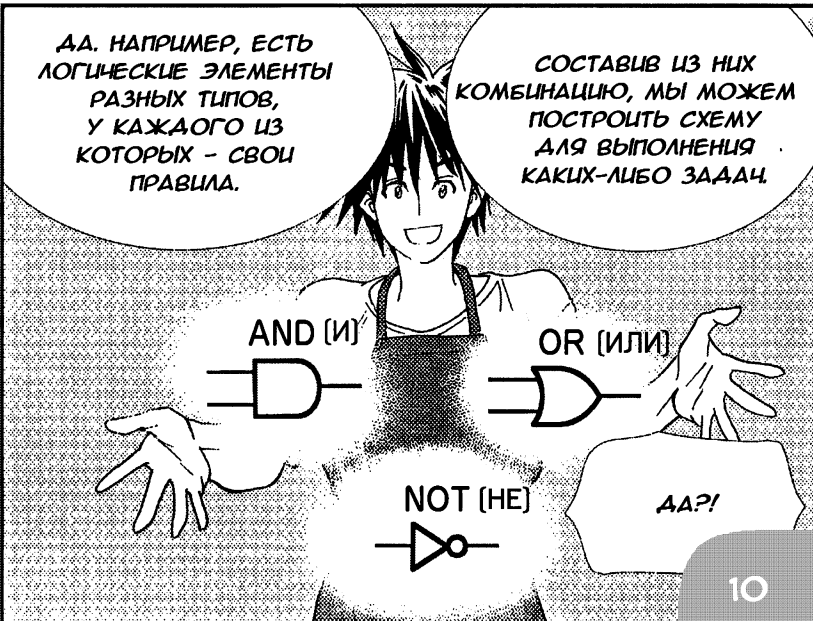
ЭТО ЧТО-ТО ВРОДЕ КОНСТРУКТОРА ИЗ ПРАВИЛ.



КОНСТРУКТОРА...?

ИЗ ПРАВИЛ...?

ого...



АА. НАПРИМЕР, ЕСТЬ ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ РАЗНЫХ ТИПОВ, У КАЖДОГО ИЗ КОТОРЫХ - СВОИ ПРАВИЛА.

СОСТАВИВ ИЗ НИХ КОМБИНАЦИЮ, МЫ МОЖЕМ ПОСТРОИТЬ СХЕМУ ДЛЯ ВЫПОЛНЕНИЯ КАКИХ-ЛИБО ЗАДАЧ.

AND (И)

OR (ИЛИ)

NOT (НЕ)

АА?!

СПРОЕКТИРОВАТЬ СХЕМУ -
ЗНАЧИТ СОЗДАТЬ
ЭЛЕКТРОННОЕ
УСТРОЙСТВО,
РАБОТАЮЩЕЕ ТАК,
КАК ТЫ ЗАДУМАЛА.



СОЗДАНИЕ ЭЛЕКТРО-
ПРИБОРОВ, КОТОРЫМИ
МЫ ПОЛЬЗУЕМСЯ,
ТОЖЕ НАЧИНАЕТСЯ
С ПРОЕКТИРОВАНИЯ
СХЕМ.

ВСЁ ЭТО
ИЗУЧАЕТСЯ
В РАЗДЕЛЕ
ЭЛЕКТРОНИКИ...

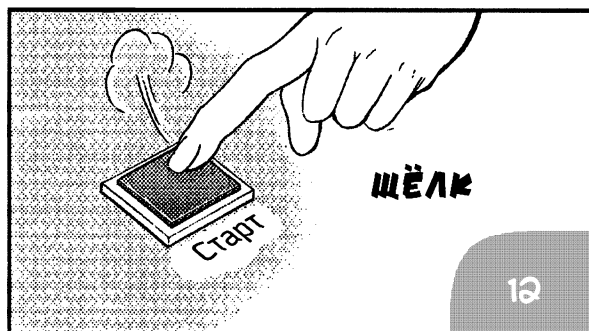
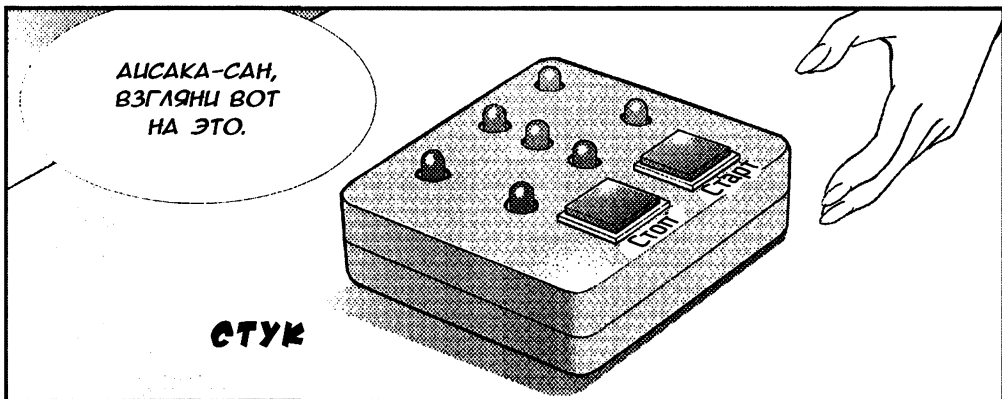
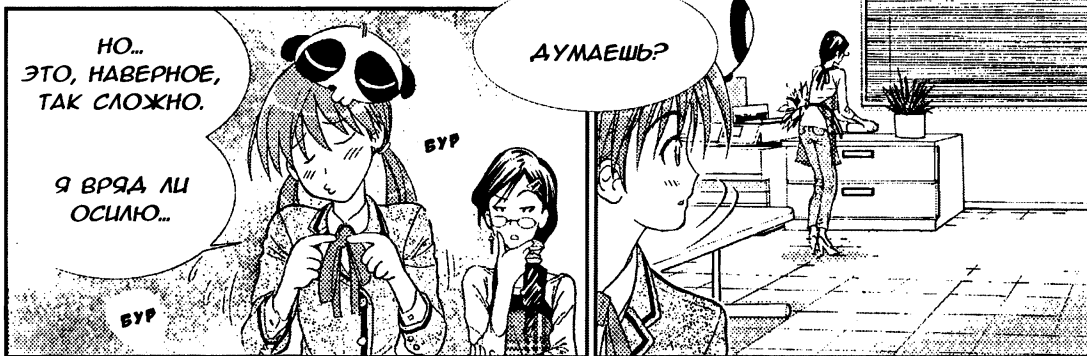


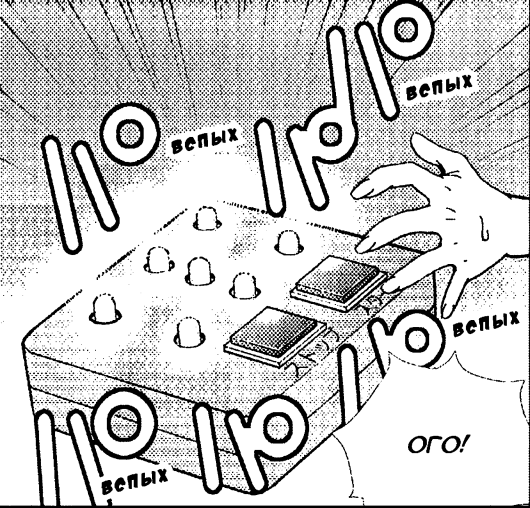
КАК ИНТЕРЕСНО!

НАУКА О
ЭЛЕКТРОННЫХ
УСТРОЙСТВАХ,
КОТОРЫЕ ЕСТЬ
ПОВСЮДУ!

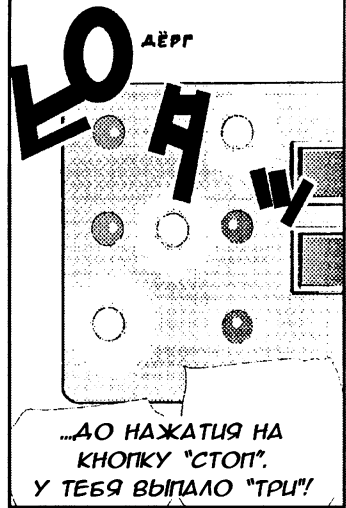
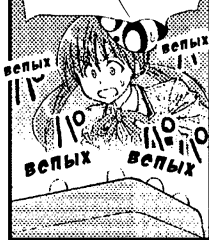


...ПОД НАЗВАНИЕМ
"ЦИФРОВЫЕ СХЕМЫ"!

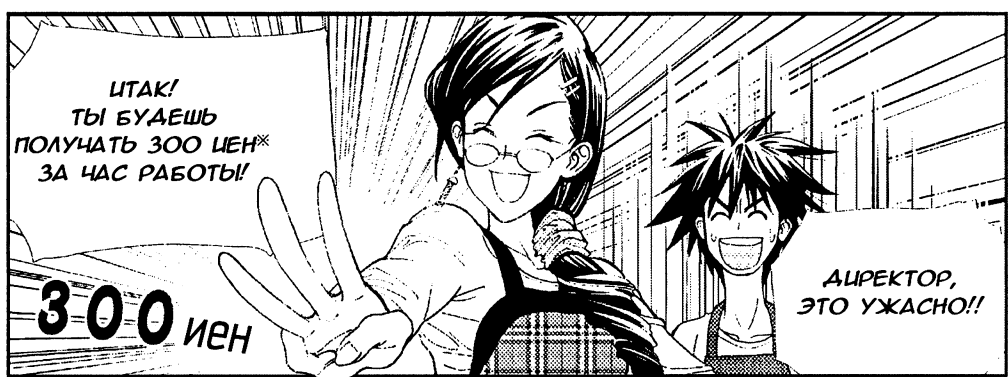




ЛАМПЫ
БЫСТРО
ЗАГОРАЮТСЯ
И ГАСНУТ...



...ДО НАЖАТИЯ НА
КНОПКУ "СТОП".
У ТЕБЯ ВЫПАЛО "ТРИ"!



ИТАК!
ТЫ БУДЕШЬ
ПОЛУЧАТЬ 300 ИЕН*
ЗА ЧАС РАБОТЫ!

300 ИЕН

ДИРЕКТОР,
ЭТО УЖАСНО!!

* Около 150 рублей.



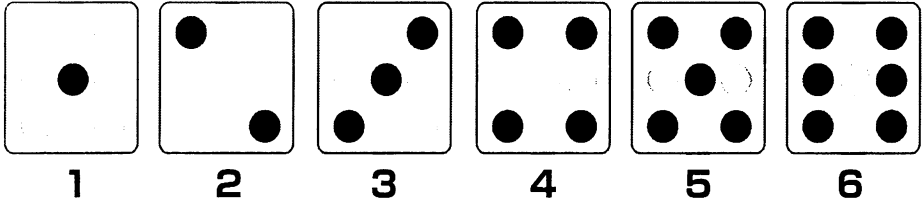
ТРИСТА...
ИЕН...

УСПОКОЙСЯ,
ЭТО ШУТКА!



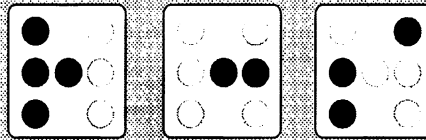
КОНЕЧНО,
Я ПОШУТИЛА.
ОБРАТИ ВНИМАНИЕ,
ЧТО КОМБИНАЦИИ
ГОРЯЩИХ ЛАМПОЧЕК
ВСЕГДА БУДУТ
СООТВЕТСТВОВАТЬ
ГРАНЯМ НАСТОЯЩЕГО
КУБИКА.

Допустимые комбинации светодиодов



ВОТ ТАКОГО
НЕ БУДЕТ
НИКОГДА.

Недопустимые комбинации светодиодов



?

АА,
НА НАСТОЯЩЕМ
КУБИКЕ ТАКОГО
ВЫПАСТЬ
НЕ МОЖЕТ.

НО ЕСЛИ БЫ
ВАРУГ ВЫПАЛО,
ТО Я БЫ
ИСПУГАЛАСЬ.

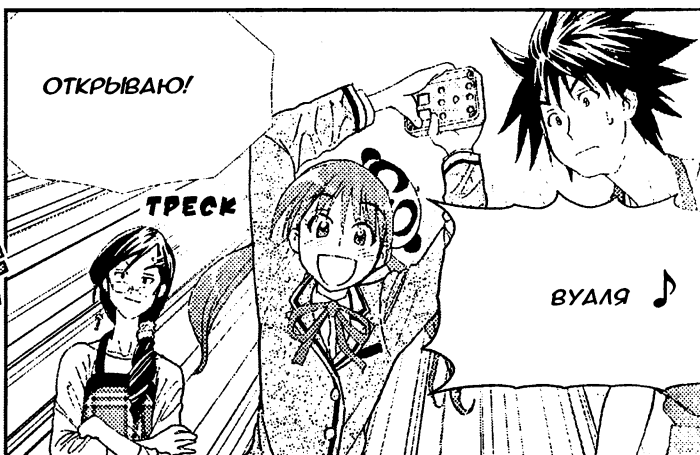
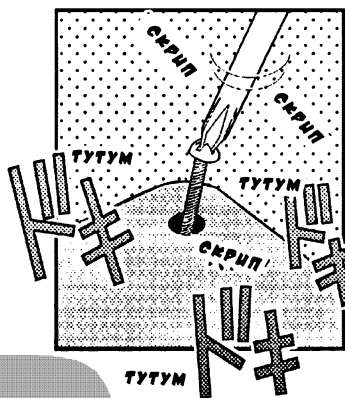
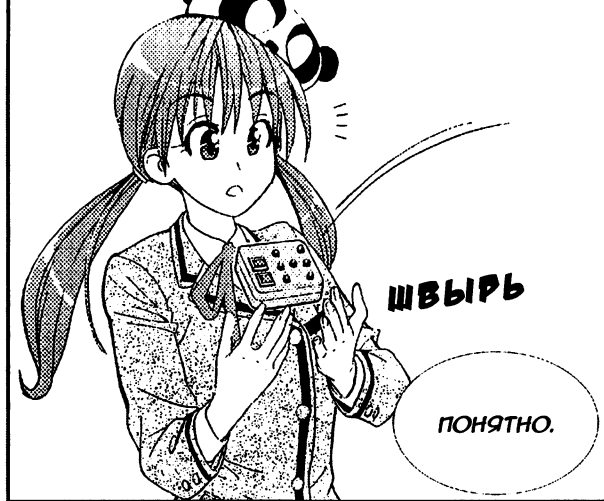
ЗНАЕШЬ,
ПОЧЕМУ
ЗАГОРАЮТСЯ
ТОЛЬКО ПРАВИЛЬНЫЕ
КОМБИНАЦИИ?

ПОТОМУ,
ЧТО...

...ТАК
СПРОЕКТИРОВАНА
СХЕМА!!!

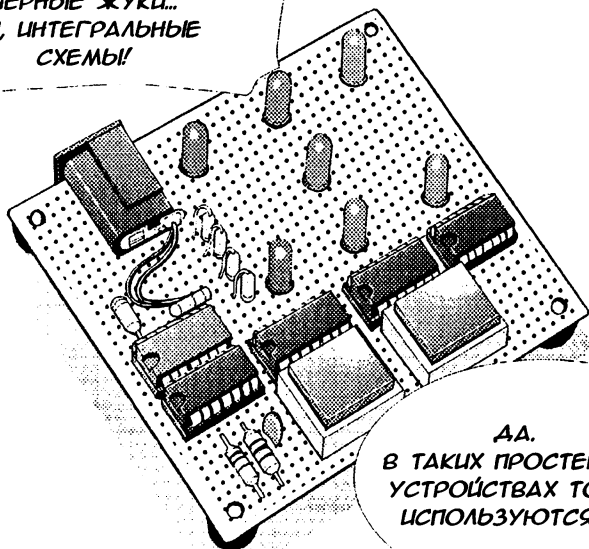
ВЕРНО!

ТАААА!



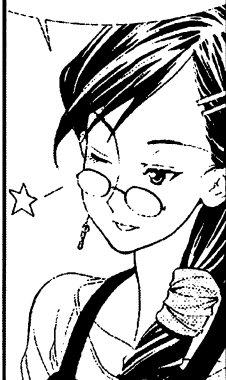
ОГО!

ОПЯТЬ ЭТИ
ЧЁРНЫЕ ЖУКИ...
ОЙ, ИНТЕГРАЛЬНЫЕ
СХЕМЫ!



АА.
В ТАКИХ ПРОСТЕЙШИХ
УСТРОЙСТВАХ ТОЖЕ
ИСПОЛЬЗУЮТСЯ ИС.

ОНИ НУЖНЫ
ДЛЯ
РЕАЛИЗАЦИИ
СХЕМЫ,
КОТОРУЮ Я
ПРЕДУМАЛА.



ВОТ КАК?
ДАЖЕ ТАКАЯ
ПРОСТАЯ ИГРУШКА
НЕ ОБХОДИТСЯ
БЕЗ МИКРОСХЕМ.



МОЖЕТ БЫТЬ,
ВСЕ ЭТИ ИС, ИЛИ,
КАК ИХ ТАМ,
ЛОГИЧЕСКИЕ
ВЕНТИЛИ..



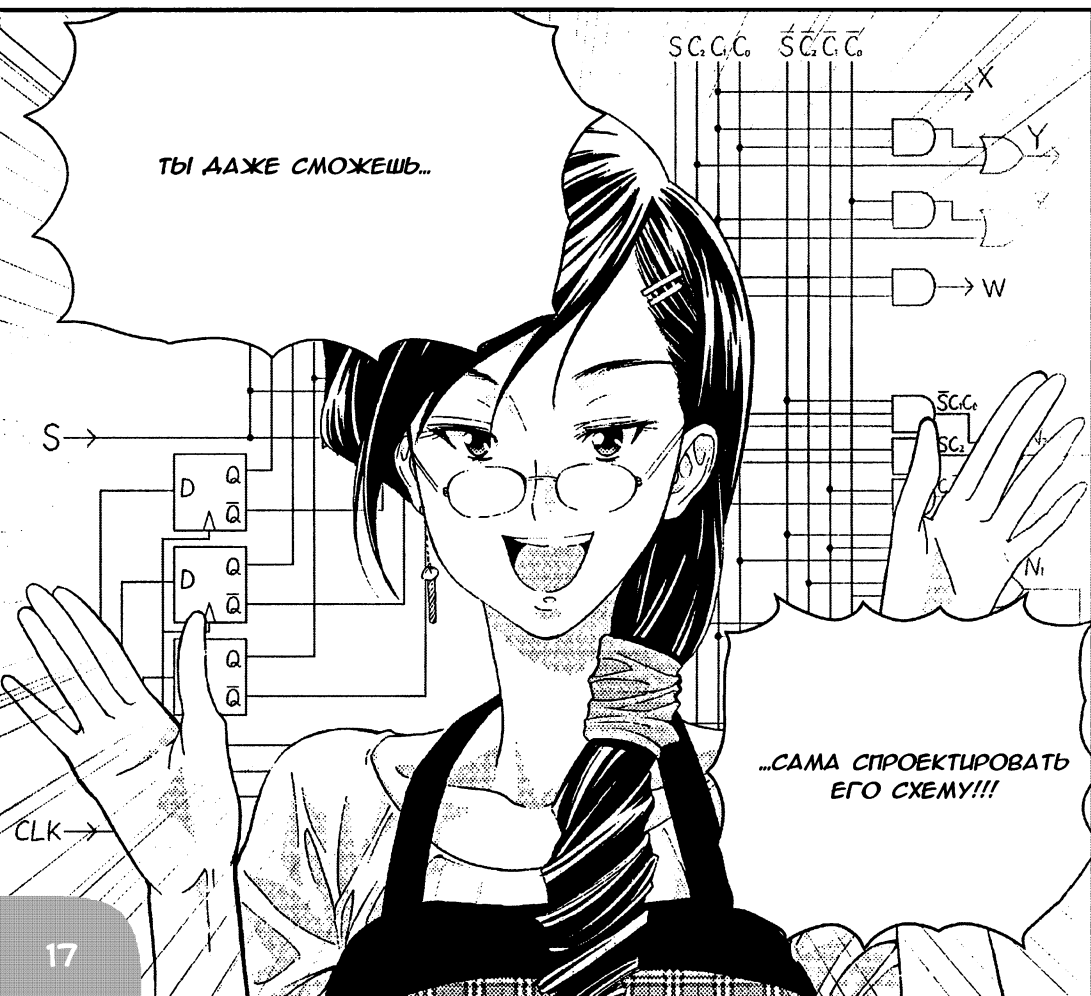
...НА САМОМ ДЕЛЕ
ОЧЕНЬ ПРОСТЫ!



ВЕРНО!

ЕСЛИ РЕЧЬ ИДЁТ НЕ
О КАКОМ-ТО ЭЛЕКТРОННОМ
УСТРОЙСТВЕ СО
СЛОЖНЫМИ ФУНКЦИЯМИ,
А О ТАКОМ ЭЛЕКТРОННОМ
КУБИКЕ...

...ТО И НОВИЧОК
ПРИ ЖЕЛАНИИ СМОЖЕТ
РАЗОБРАТЬСЯ.



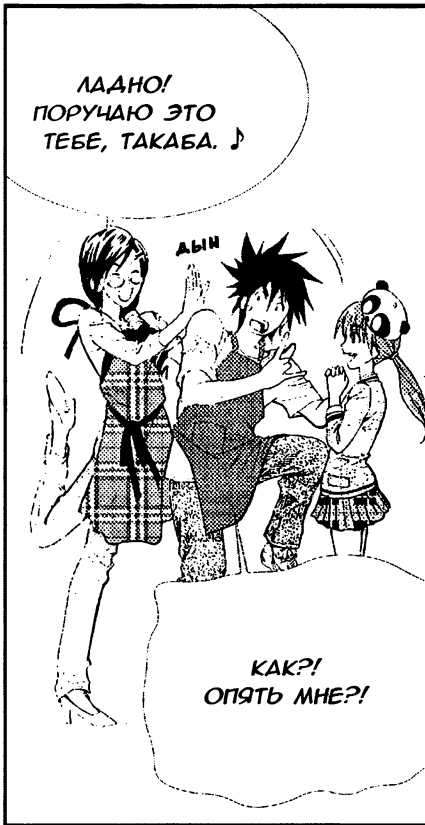
ТЫ ДАЖЕ СМОЖЕШЬ...

...САМА СПРОЕКТИРОВАТЬ
ЕГО СХЕМУ!!!



ПРАВДА?!
Я САМА?!

РАССКАЖИ,
КАК ЭТО СДЕЛАТЬ,
ПОЖАЛУЙСТА!!!



ЛАДНО!
ПОРУЧАЮ ЭТО
ТЕБЕ, ТАКАБА. ♪

АЫН

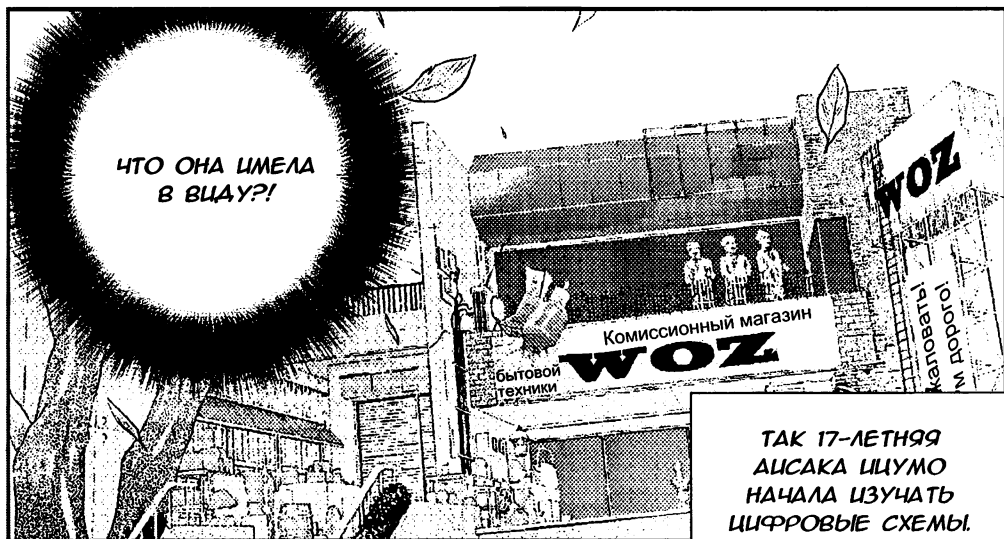
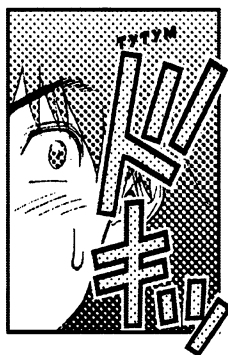
КАК?!
ОПЯТЬ МНЕ?!



ТАКАБА-САН, НУ
ПОЖАЛУЙСТА!!

ОБЕЩАЮ
ПРИЛЕЖНО ИЗУЧАТЬ
ЦИФРОВЫЕ СХЕМЫ!!!!





Интегральные схемы (ИС), называемые ещё микросхемами или чипами, — это полупроводниковые приборы, состоящие из большого числа транзисторов. Многие ИС являются цифровыми устройствами.

Сложность микросхемы определяется степенью интеграции — количеством содержащихся в ней транзисторов. ИС, содержащие более 1 млн транзисторов, иногда называют «большими» (БИС, англ. LSI: Large Scale Circuit). Современные чипы выпускаются в корпусах для поверхностного монтажа с миниатюрными выводами. Поверхностный монтаж (SMD: Surface Mounted Device) — это установка электронной детали на поверхность печатной платы без сквозных отверстий. Разобрав цифровое устройство, например мобильный телефон, вы увидите на его миниатюрной печатной плате подобные микросхемы, но число их будет невелико, ведь высокая степень интеграции современных схем позволяет создавать устройства даже на базе одного-единственного чипа.

В электронном кубике, который разобрала Аисака-сан, использовались ИС низкой степени интеграции в корпусах с двухрядным расположением выводов (DIP: Dual Inline Package), похожие на многоножек. Радиолюбители 70-х, 80-х годов собирали на универсальных платах цифровые устройства из таких микросхем, соединяя их проводами, но в наши дни подобные ИС используются в основном в образовательных целях, а на практике применяются программируемые логические интегральные схемы (ПЛИС), логику работы которых задаёт проектировщик. Программируемая пользователем вентиляционная матрица (ППВМ) — это разновидность ПЛИС, позволяющая легко и с минимальными затратами создавать большие и быстродействующие цифровые схемы. Продаются и специальные тестовые платы, на которые ППВМ устанавливаются без пайки соединений. Однако, хотя в наши дни для создания цифровых схем удобнее использовать ППВМ, такие устройства невозможно «пощупать руками», как это сделала Аисака-сан с электронным кубиком в манге, так как вся электронная схема будет размещаться внутри единственного чипа, поэтому здесь будут рассмотрены устройства на базе обычных микросхем. Так как основные принципы цифровых схем совершенно не зависят от того, используется ли современная ППВМ или же микросхемы низкой степени интеграции, освоив эти принципы, вы сможете создавать любые цифровые схемы с помощью САД (Computer Aided Design: система автоматизированного проектирования, или САПР) и ППВМ.

Программируемая пользователем вентиляльная матрица

ППВМ (программируемая пользователем вентиляльная матрица, англ. FPGA: Field Programmable Gate Array) — это программируемая микросхема, состоящая из большого количества блоков логических элементов, запоминающих элементов (триггеров), соединительных проводников и маленьких переключателей. Устройство классической ППВМ показано на рис. 1.

Заполнив таблицы истинности блоков элементов значениями и правильно установив переключатели, мы сможем создать нужную нам цифровую схему.

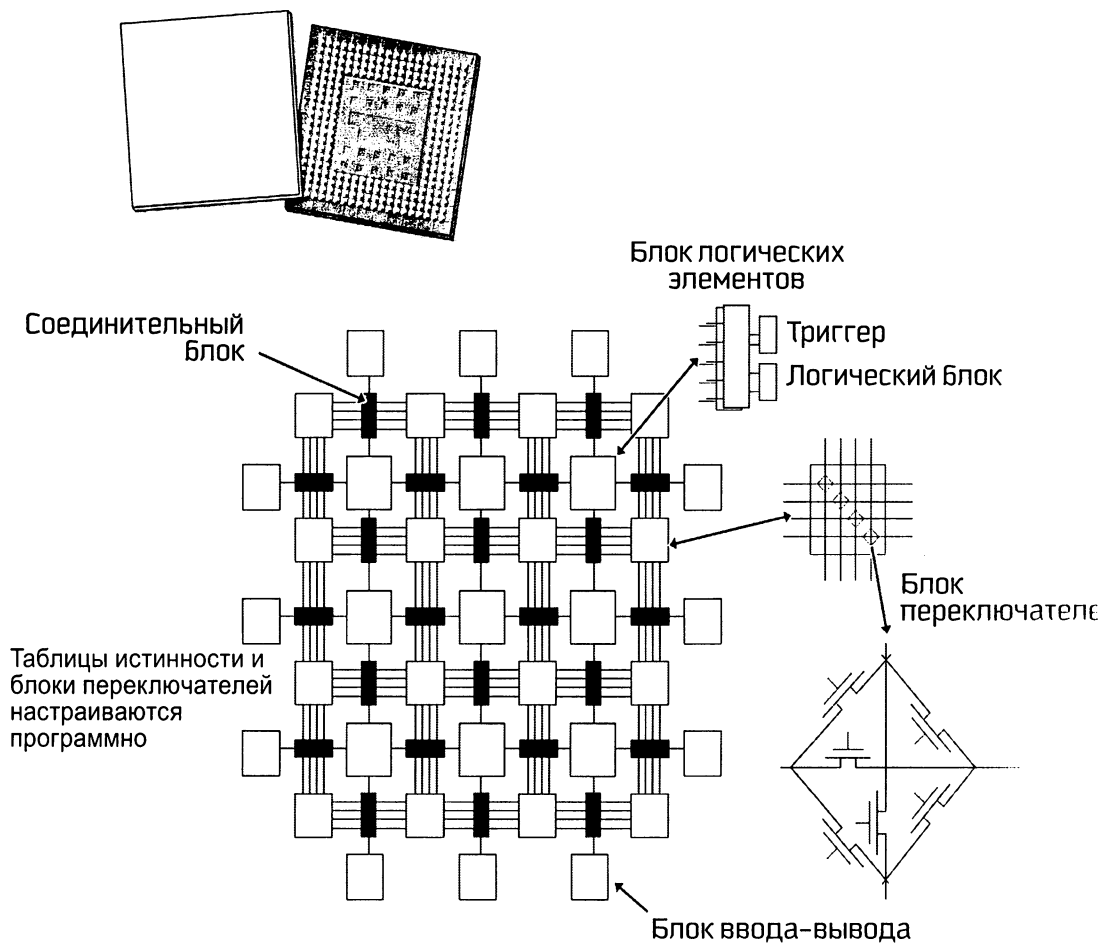


Рис. 1. Устройство ППВМ



Проектирование цифровых схем с помощью ППВМ сводится к написанию компьютерных программ на языках описания аппаратуры (HDL: Hardware Description Language), о которых будет рассказано в разделе «Дополнительная информация» главы 5.

Схема соединений, которая будет записана в ППВМ, создаётся на компьютере средствами САПР. На момент написания этой книги было два знаменитых производителя ППВМ — Xilinx и Altera*, и оба они бесплатно предоставляют программное обеспечение САПР в образовательных целях, а недорогую тестовую плату для ППВМ можно приобрести всего за несколько сотен рублей.

Вышеуказанная методика существенно облегчает проектирование и тестирование цифровых схем: микропроцессоров, систем управления роботами или игрушками и т. п. Проводятся даже конкурсы самодельных устройств на базе ППВМ, например электронных игрушек.

* В настоящее время компания Altera поглощена корпорацией Intel, однако поддержка старых устройств ещё сохраняется. — *прим. ред.*

◆ Altera

<https://www.altera.com/products.html>

http://www.alterawiki.com/wiki/Main_Page

◆ Xilinx

<https://www.xilinx.com/products/silicon-devices/fpga.html>

Объяснение терминов главы 1

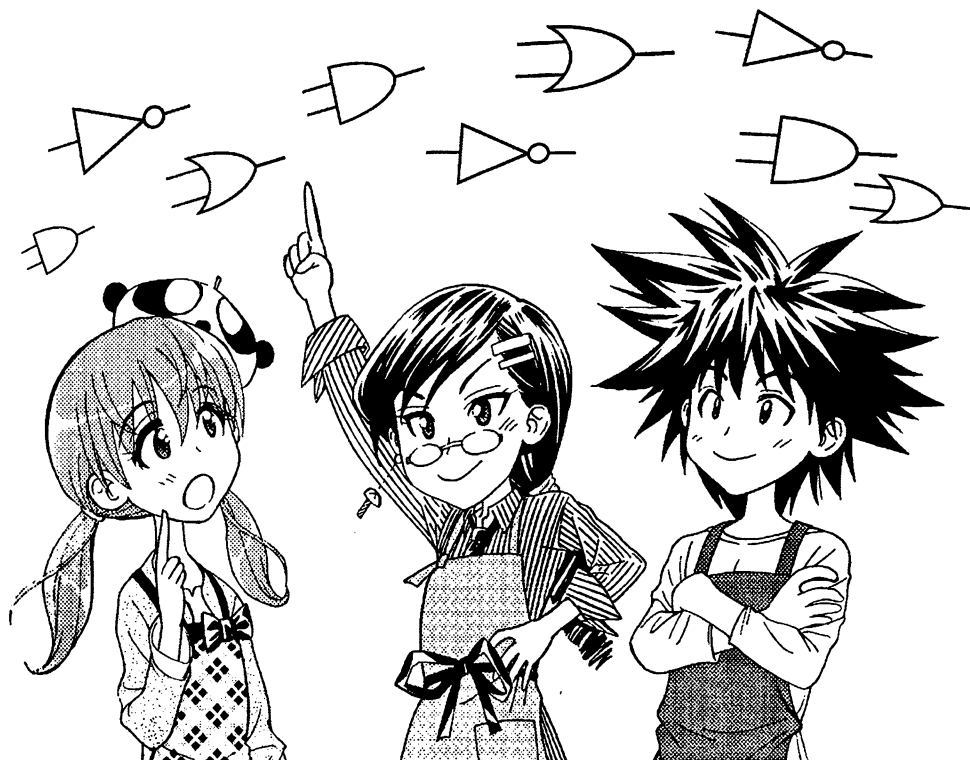
- **ИС (интегральная схема, микросхема, чип)** — полупроводниковый прибор, в котором интегрировано множество транзисторов. ИС могут быть цифровыми или аналоговыми, но в книге мы будем изучать только цифровые микросхемы, используемые в цифровых устройствах. ИС высокой степени интеграции называют также БИС (большими интегральными схемами).
- **Логические схемы** — схемы, выполняющие логические операции, например AND (И), OR (ИЛИ), NOT (НЕ) и т. п. над значениями 1 и 0. Можно считать синонимом термина «цифровые схемы».
- **Логический вентиль** — базовый элемент, выполняющий логические операции: AND (И, конъюнкция, логическое умножение), OR (ИЛИ, дизъюнкция, логическое сложение), NOT (НЕ, отрицание, инверсия), NAND (И-НЕ, штрих Шеффера) и XOR (исключающее ИЛИ, логическое вычитание). В главе 2 описывается устаревший чип стандартной логики SN7432^{※1} из нескольких логических вентилях, довольно распространённый в прошлом.
- **Алгебра логики (булева логика^{※2}, двоичная логика)** — математические правила, лежащие в основе проектирования цифровых схем. Подробнее см. раздел «Дополнительная информация» главы 2.

※1 Отечественный аналог — микросхема К155ЛЛ1. — прим.перев.

※2 Не путать с «булевой алгеброй». — прим.перев.

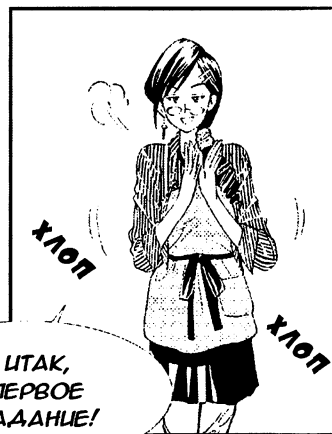
ГЛАВА 2

ЦИФРОВЫЕ И АНАЛОГОВЫЕ СХЕМЫ



1. ЦИФРОВАЯ И АНАЛОГОВАЯ ТЕХНИКА

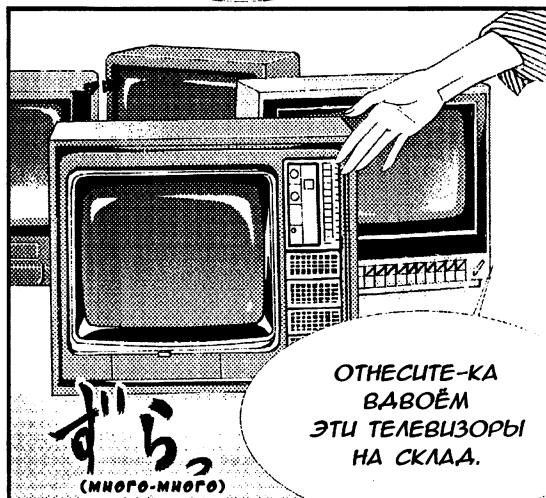
От аналоговых схем к цифровым



ИТАК,
ПЕРВОЕ
ЗАДАНИЕ!

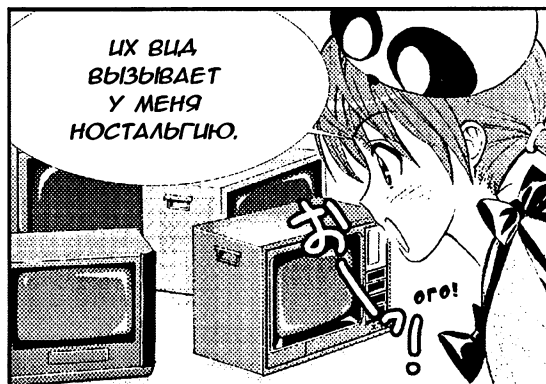


Я ГОТОВА!!



ОТНЕСИТЕ-КА
ВАМ
ЭТИ ТЕЛЕВИЗОРЫ
НА СКЛАД.

ずら
(МНОГО-МНОГО)



ИХ ВИД
ВЫЗВАЕТ
У МЕНЯ
НОСТАЛЬГИЮ.

АА. РАНЬШЕ
ТАКИЕ АНАЛОГОВЫЕ
ТЕЛЕВИЗОРЫ БЫЛИ
ОБЫЧНЫМ ДЕЛОМ...



АА?

...НО СЕЙЧАС УЖЕ
ПОЛНОСТЬЮ
НА ЦИФРОВЫЕ
ПЕРЕШЛИ.

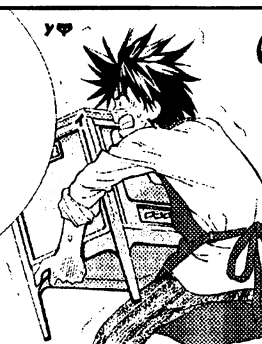
КАК ЛЕТИТ ВРЕМЯ!
ВОТ И МОЛОДОСТЬ
ПРОШЛА...



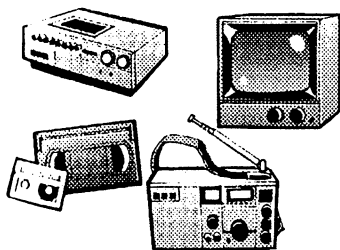
АХ...

ЭЙ, ХВАТИТ ВИТАТЬ
В ОБЛАКАХ!
ПОМОГЛА БЫ МНЕ
ЛУЧШЕ!

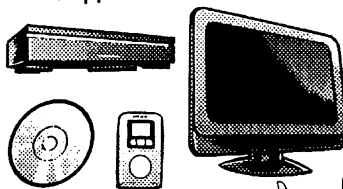
КСТАТИ, ЭТИ ЦИФРОВЫЕ
СХЕМЫ, КОТОРЫЕ Я
СОБРАЛАСЬ ИЗУЧАТЬ,
КАК-ТО СВЯЗАНЫ
С ТЕЛЕВИЗОРАМИ?



Изделия на базе
аналоговых схем



Изделия на базе
цифровых схем



ОГО!
КОНЕЧНО!
РАНЬШЕ НЕ ТОЛЬКО
ТЕЛЕВИЗОРЫ, НО И
ВИДЕОМАГНИТОФНЫ,
РАДИОПРИЁМНИКИ И ДРУГОЕ
ЗВУКОВОЕ ОБОРУДОВАНИЕ -
ВСЁ СОЗДАВАЛОСЬ
НА БАЗЕ АНАЛОГОВЫХ СХЕМ.

ОГО!



А СЕЙЧАС ИХ
СОЗДАЮТ
НА БАЗЕ
ЦИФРОВЫХ СХЕМ.





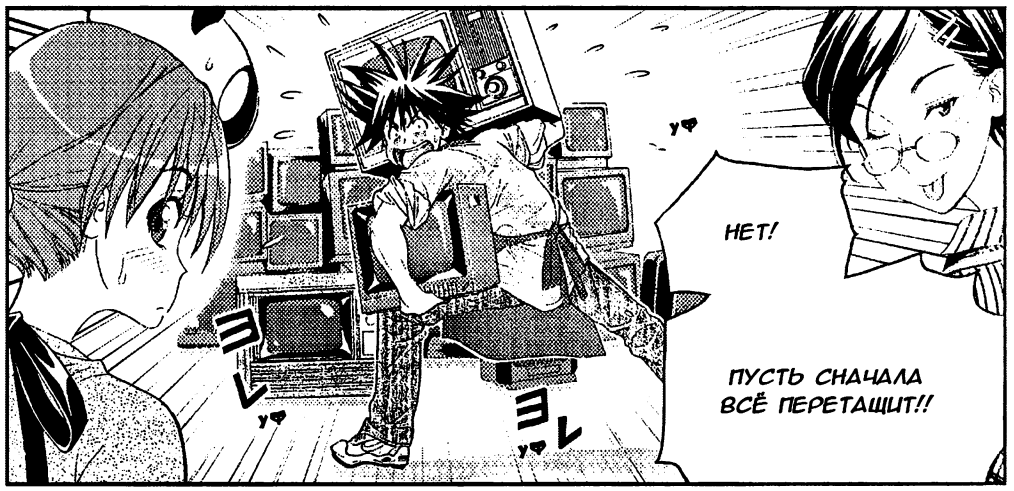
ММ...

ЗАЧЕМ
ПОНАДОБИЛОСЬ
ПЕРЕХОДИТЬ
С АНАЛОГОВЫХ СХЕМ
НА ЦИФРОВЫЕ?

НЕПОНЯТНО!

РАССКАЖИ,
ПОЖАЛУЙСТА!!!

АХ...



НЕТ!

ПУСТЬ СНАЧАЛА
ВСЁ ПЕРЕТАЩИТ!!

Представление об аналоговой и цифровой технике



ТАК...
ОТАВШАВШИСЬ
НЕМНОГО...

УФУФ

ШАТЬ

ШАТЬ

УФУФ

...ДАМ ТЕБЕ
ОБЩЕЕ
ПРЕДСТАВЛЕНИЕ
ОБ АНАЛОГОВЫХ
И ЦИФРОВЫХ
УСТРОЙСТВАХ.

УФУФ

В ОБЩЕМ,
В ЭТИХ ПРИБОРАХ
СТРЕЛКА
УКАЗЫВАЕТ
НА ТОЧКУ
НА ШКАЛЕ.



...А ЭТИ
ОТБРАЖАЮТ
ЦИФРЫ
НА ТАБЛО.

Я НЕМНОГО
ПОПРАВЛАСЬ?
ТАК...



Аналоговые

Я
ПОПРАВЛАСЬ!!
АБЭ!



Цифровые

АГА!
СТРЕЛКА АНАЛОГОВЫХ
ВЕСОВ МОЖЕТ НАХОДИТЬСЯ
ГДЕ-ТО МЕЖДУ ДЕЛЕНИЯМИ,
А ЦИФРОВЫЕ ВЕСЫ
ПОКАЗЫВАЮТ ЧИСЛО, НЕ
ОСТАВЛЯЮЩЕЕ СОМНЕНИЙ.



ДА, ДА.
ОТЛИЧНЫЙ
ПРИМЕР
ИЗ ЖЕНСКОЙ
ПСИХОЛОГИИ.



ИМЕННО ЭТИМ
И ОТЛИЧАЮТСЯ
ЦИФРОВЫЕ СХЕМЫ
ОТ АНАЛОГОВЫХ.



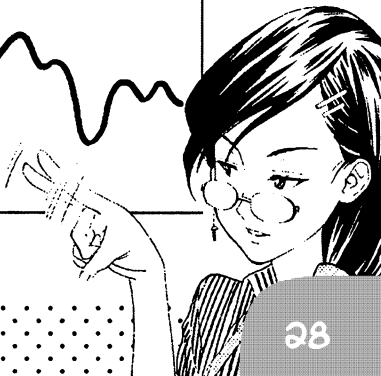
ТАК ЧЕМ ЖЕ?

ДАВАЙ ПОСМОТРИМ НА
ИЗМЕНЕНИЕ НАПРЯЖЕНИЯ*
В СХЕМАХ.
В АНАЛОГОВЫХ СХЕМАХ
СУЩЕСТВЕННЫ ДАЖЕ
МАЛЫЕ ИЗМЕНЕНИЯ
НАПРЯЖЕНИЯ.
НАПРИМЕР, УРОВНИ
НАПРЯЖЕНИЯ, РАВНЫЕ 0.1 В И
0.2 В, ВОСПРИНИМАЮТСЯ
СХЕМОЙ ПО РАЗНОМУ,
И ВПОЛНЕ ВОЗМОЖНА СХЕМА,
ГДЕ СУЩЕСТВЕННА РАЗНИЦА
МЕЖДУ 0.1 И 0.1001 ВОЛЬТАМИ.

↓ График изменения напряжения во времени



* Напряжение подобно уровню воды, удерживаемому плотиной, в то время как электрический ток — это поток воды через трубу внизу плотины. Напряжение измеряется в вольтах (В), ток — в амперах (А).



ЗНАЧИТ,
НАПРЯЖЕНИЕ
ВСЁ ВРЕМЯ
ПОНЕМНУ
ИЗМЕНЯЕТСЯ,
ОБРАЗУЯ ВОЛНУ
СЛОЖНОЙ
ФОРМЫ?

КАК



КАК

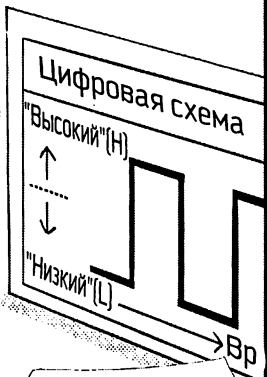
НО ТЫ УДИВИШЬСЯ, УЗНАВ,
ЧТО В ЦИФРОВЫХ СХЕМАХ
УРОВНЕЙ НАПРЯЖЕНИЯ
ВСЕГО ДВА:
"ВЫСОКИЙ" (H)
И "НИЗКИЙ" (L)!!



ОТ АНГЛИЙСКИХ
СЛОВ
HIGH (ВЫСОКИЙ)
И LOW (НИЗКИЙ)!



ОГО!!
В ЭТИХ ЦИФРОВЫХ
СХЕМАХ
НАПРЯЖЕНИЕ
ИЗМЕНЯЕТСЯ
КАК-ТО... КВАДРАТНО,
ЧТО ЛИ?



КАКИЕ
КРАЙНОСТИ!!!

ДА,
В САМОМ ДЕЛЕ.
ПРОСТО, КАК КИРПИЦ!



ОДНАКО ЦИФРОВЫЕ
СХЕМЫ ПОЛУЧИЛИ ТАКОЕ
РАЗВИТИЕ ИМЕННО
БЛАГОДАРЯ ПРОСТОТЕ
ИСПОЛЬЗОВАНИЯ
ВСЕГО ДВУХ
УРОВНЕЙ*.

* Всего два уровня напряжения позволяют соединять, или, как говорят, сопрягать между собой различные устройства, что оказалось очень удобно.

ММ...

КАК-ТО
НЕПОНЯТНО...



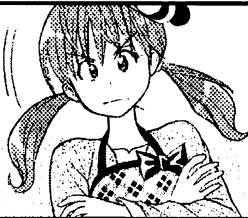
КАК ТЫ САМА ГОВОРИЛА,
КОМПЬЮТЕР -
ЭТО МИР НУЛЕЙ
И ЕДИНИЦ (СМ. СТР. 8).

ЭТИ НУЛИ И ЕДИНИЦЫ И ЕСТЬ
УРОВНИ НАПРЯЖЕНИЯ
"ВЫСОКИЙ" (H)
И "НИЗКИЙ" (L).



ВОТ КАК?

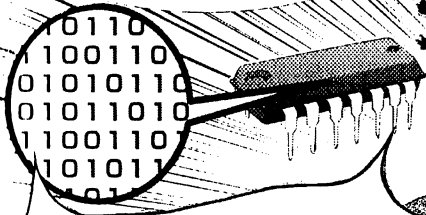
ДА



ДА

ВЫХОДИТ, КОМПЬЮТЕР
САМ ПО СЕБЕ - ЭТО
ЦИФРОВАЯ СХЕМА?

ДАЛЕЕ,
МИКРОСХЕМЫ С ОГРОМНОЙ
СКОРОСТЬЮ ПРОИЗВОДЯТ
НАД ЭТИМИ
НУЛЯМИ И ЕДИНИЦАМИ
ОПЕРАЦИИ (ВЫЧИСЛЕНИЯ)!



ТАКОЕ БЫСТРОДЕЙСТВИЕ
ВОЗМОЖНО ИМЕННО
БЛАГОДАРИ ПРОСТОТЕ
ЦИФРОВОГО МИРА,
СОСТОЯЩЕГО
ИЗ НУЛЕЙ И ЕДИНИЦ!

光の速い (빛의 속도) (Speed of Light)

СРАВНИВАЯ
БЫСТРОДЕЙСТВИЕ
АНАЛОГОВЫХ И
ЦИФРОВЫХ
СХЕМ...

SHURU SHURU SHURU SHURU



TRUCK

...МЫ ПОЛУЧИМ
ВОТ ЧТО!

БЫСТРО!!

Аналоговые схемы

Цифровые схемы

Диапазоны
быстродействия:

Диапазоны
быстродействия:

миллисекунды (10^{-3} с)
(мс)

наносекунды (10^{-9} с)
(нс)

микросекунды (10^{-6} с)
(мкс)

пикосекунды (10^{-12} с)
(пс)

$10^{-3} = 1/10^3 = 0.001$

Другими словами, 1 мс = 0.001 с, 1 нс = 0.000 000 001 с

АНАЛОГОВЫЕ СХЕМЫ
ТОЖЕ БЫСТРЫ,
НО ЦИФРОВЫЕ
ВО МНОГО РАЗ
БЫСТРЕЕ**!!

単純!!!

ПРОСТАТА!!

ПОНЯТНО!
МИР ЦИФРОВЫХ СХЕМ
ОЧЕНЬ ПРОСТ,
ТАМ ТОЛЬКО
НУЛИ И ЕДИНИЦЫ:
(УРОВНИ L И H)!!

ИМЕННО ПОЭТОМУ
ВЫЧИСЛЕНИЯ
ВЫПОЛНЯЮТСЯ
ТАК БЫСТРО!!!

ВЕРНО.
ВОТ ТАКАБА
ТОЖЕ ПРОСТАЧОК,
ПОЭТОМУ Я ВСЕГДА
ПОРУЧАЮ ЕМУ
ПРОСТУЮ РАБОТУ.

И ОН
ВЫПОЛНЯЕТ
ЕЁ ОЧЕНЬ
БЫСТРО.

ОЧЕНЬ
ПОХОЖЕ, ♪
НЕ ТАК ЛУП



* Это чрезмерное упрощение. Вполне себе существуют аналоговые схемы с быстродействием цифровых и даже быстрее. Но их сложно встретить в бытовых устройствах отдельно от цифровых схем.

2. ПОЧЕМУ ЦИФРОВЫЕ СХЕМЫ ПОБЕДИЛИ?

Что такое логические схемы?

В ОБЩИХ ЧЕРТАХ
ЯСНО.

НО ВОТ ВЫЧИСЛЕНИЯ,
КОТОРЫЕ ВЫПОЛНЯЕТ
МИКРОСХЕМА, -
ЭТО, НАВЕРНОЕ,
ТЕ САМЫЕ ЛОГИЧЕСКИЕ
ОПЕРАЦИИ.

ИНТЕРЕСНО, ЧТО
ЭТО ТАКОЕ?
"ЛОГИЧЕСКИЕ" -
ЭТО, НАВЕРНОЕ,
СЛОЖНО.

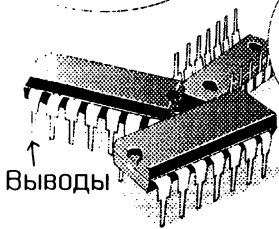
ММ...

НЕТ. ОНИ
НАСТОЛЬКО
ПРОСТЫ, ЧТО
ДАЖЕ ТАКАБА
СМОЖЕТ ОБЪЯСНИТЬ.

ДАВАЙ, ДАВАЙ.

ХВАТЬ

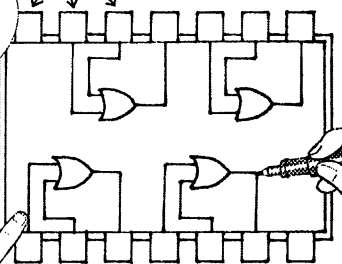
ТАК...
ВНУТРИ ЭТИХ ИС
НАХОДЯТСЯ
ЛОГИЧЕСКИЕ
СХЕМЫ.



НАПРИМЕР, ВНУТРИ
МИКРОСХЕМЫ
СЕРИИ 74 - SN7432,
НАХОДИТСЯ
НЕСКОЛЬКО
ЭЛЕМЕНТОВ ИЛИ (ИЛИ).
ВОТ ЕЁ СХЕМА
С ИСПОЛЬЗОВАНИЕМ
УСЛОВНЫХ ГРАФИЧЕ-
СКИХ ОБОЗНАЧЕНИЙ.

Выводы

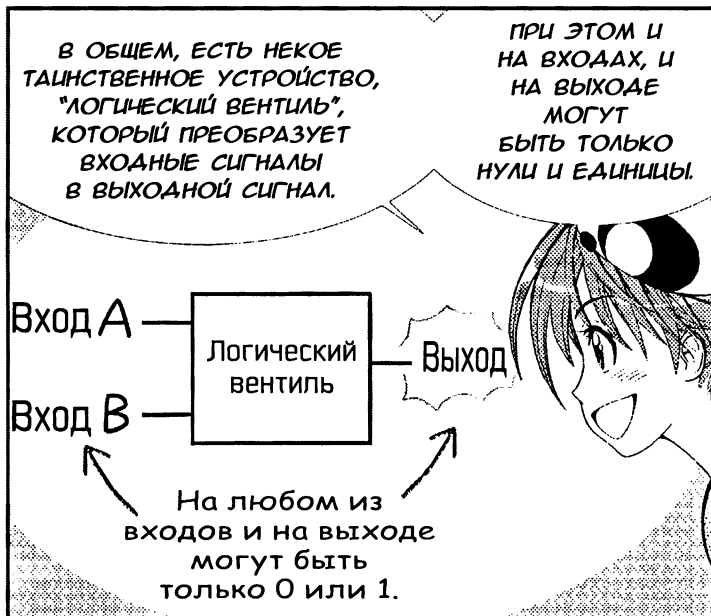
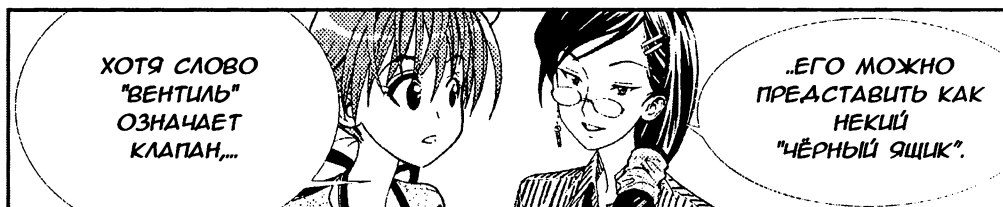
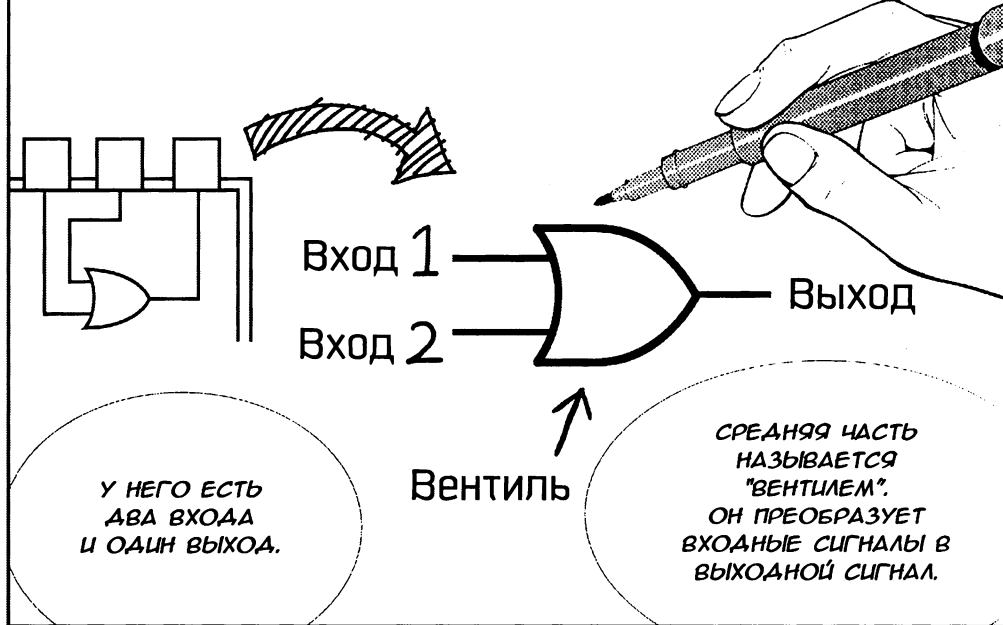
ШУРХ



ШУРХ

ДА, ДА!
ВИЖУ ЧЕТЫРЕ
ОДИНАКОВЫЕ
ФИГУРКИ,
СОЕДИНЁННЫЕ
С ВЫВОДАМИ
МИКРОСХЕМЫ.

ЭТИ ВЫВОДА
ЯВЛЯЮТСЯ
ВХОДАМИ ИЛИ
ВЫХОДАМИ ДЛЯ
НУЛЕЙ И ЕДИНИЦ.
КАЖДАЯ ИЗ
ЭЛЕМЕНТОВ
РАБОТАЕТ
СЛЕДУЮЩИМ
ОБРАЗОМ.

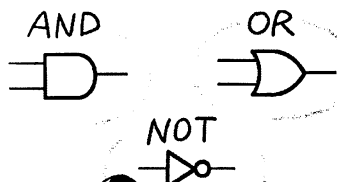


Логические вентили работают просто

ИТАК...

ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ
(ЛОГИЧЕСКИЕ ВЕНТИЛИ)
БЫВАЮТ РАЗНЫХ ВИДОВ.

РАССКАЖУ О
ТИПИЧНЫХ ВЕНТИЛЯХ:
AND, OR И NOT.



НЕ НАДО БОЯТЬСЯ!
ПРАВИЛА ВЕНТИЛЕЙ
ОЧЕНЬ ПРОСТЫ!
ИХ МОЖНО
УПОДОБИТЬ ЛЮБВИ.

ХА-ХА

ПРЕДСТАВЬ, ЧТО ДВА
ВХОДА - ЭТО
ПАРЕНЬ И ДЕВУШКА.
У НИХ МОГУТ БЫТЬ
ДВА СОСТОЯНИЯ: ...

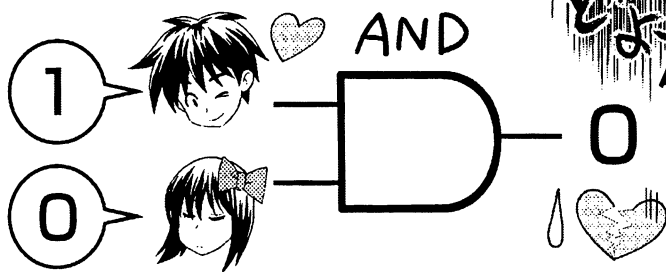
СРАЗУ О ТРЁХ?
МОЖЕТ, ХВАТИТ
ДЛЯ НАЧАЛА ОДНОГО?

ТУДУХ

...ЕДИНИЦА - "ЛЮБИТ",
ИЛИ НУЛЬ -
"НЕ ЛЮБИТ".

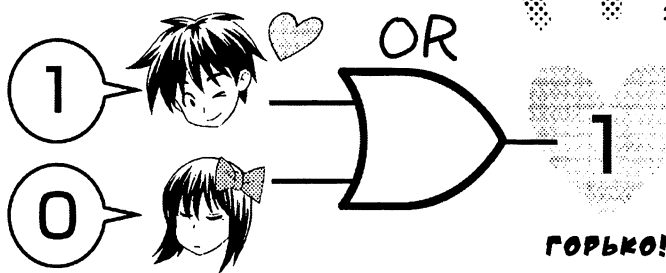
В ЛОГИЧЕСКОМ ВЕНТИЛЕ AND
РЕЗУЛЬТАТ БУДЕТ "СВАДЬБА" (1)
ТОЛЬКО ТОГДА, КОГДА ОБА
ПАРТНЁРА "ЛЮБЯТ".

ТОЛЬКО
КОГДА ОБА!!
КАК СУРОВО!!



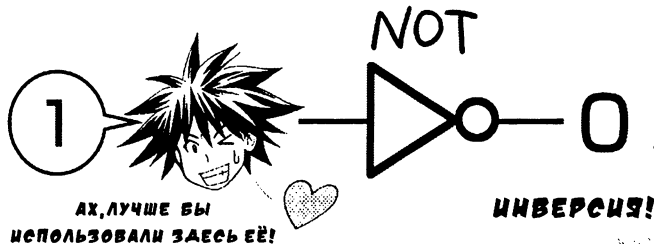
В ЛОГИЧЕСКОМ ВЕНТИЛЕ OR
РЕЗУЛЬТАТОМ БУДЕТ "СВАДЬБА" (1)
В ТОМ СЛУЧАЕ, ЕСЛИ ХОТЯ БЫ ОДИН
ИЗ ПАРТНЁРОВ "ЛЮБИТ" (1).

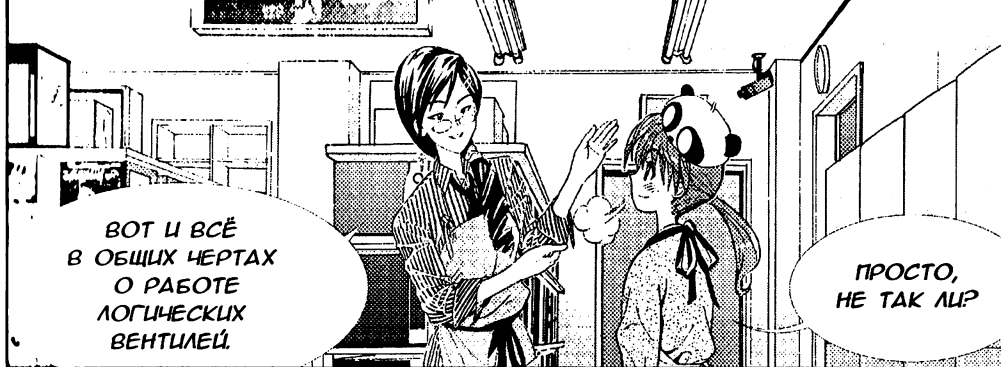
ДАЖЕ ПРИ
НЕРАЗДЕЛЁННОЙ
ЛЮБВИ!
КАКОЕ СЧАСТЬЕ!



В ЛОГИЧЕСКОМ ВЕНТИЛЕ NOT
РЕЗУЛЬТАТОМ БУДЕТ ИНВЕРСИЯ
ВХОДНОГО СИГНАЛА: ЕСЛИ ОН
"ЛЮБИТ" (1), ТО НА ВЫХОДЕ БУДЕТ
"РАЗРЫВ ОТНОШЕНИЙ" (0).

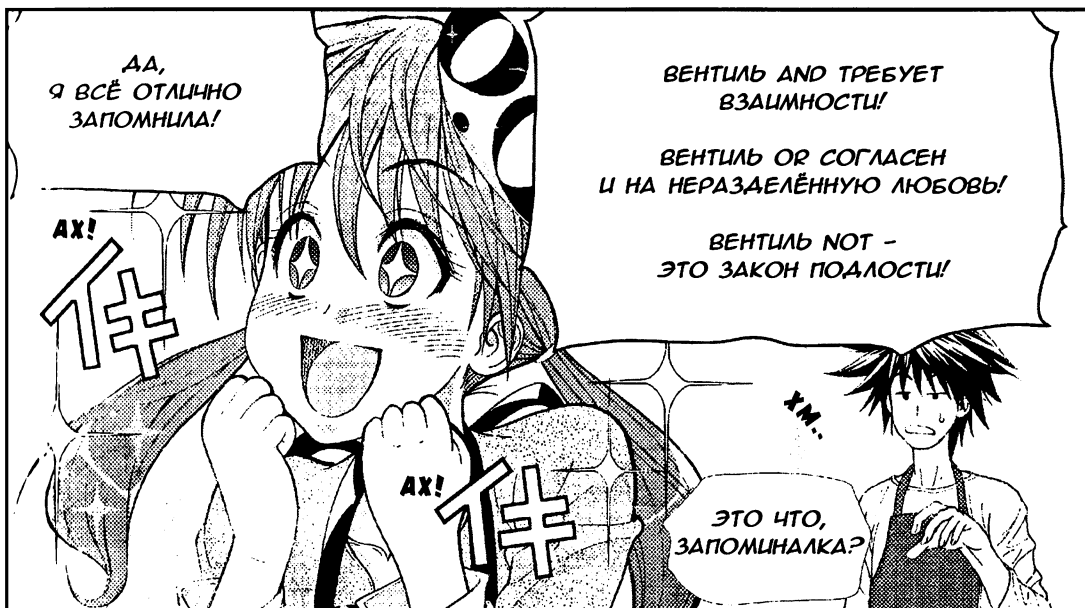
НИЧЕГО СЕБЕ!
ЭТО ЧТО - ЗАКОН
ПОЛОСТИ?!





ВОТ И ВСЁ
В ОБЩИХ ЧЕРТАХ
О РАБОТЕ
ЛОГИЧЕСКИХ
ВЕНТИЛЕЙ.

ПРОСТО,
НЕ ТАК ЛИ?



АА,
Я ВСЁ ОТЛИЧНО
ЗАПОМНИЛА!

ВЕНТИЛЬ AND ТРЕБУЕТ
ВЗАИМНОСТИ!

ВЕНТИЛЬ OR СОГЛАСЕН
И НА НЕРАЗДЕЛЁННУЮ ЛЮБОВЬ!

ВЕНТИЛЬ NOT -
ЭТО ЗАКОН ПОЛОСТИ!

АХ!
≠

АХ!
≠

ЭТО ЧТО,
ЗАПОМИНАЛКА?

※ Подробнее о логических элементах будет рассказано в главе 3.

Цифровые схемы побеждают числом!

ЭЛЕМЕНТАРНЫЕ.....

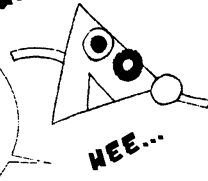
ИТАК, КАЖАБЫЙ ИЗ
ЛОГИЧЕСКИХ ВЕНТИЛЕЙ,
ВХОДЯЩИХ
ЦИФРОВУЮ СХЕМУ,
ПО ОТДЕЛЬНОСТИ МОЖЕТ
ВЫПОЛНЯТЬ ТОЛЬКО ОЧЕНЬ
ПРОСТЫЕ ОПЕРАЦИИ.



ИЛИ-ЛИ...

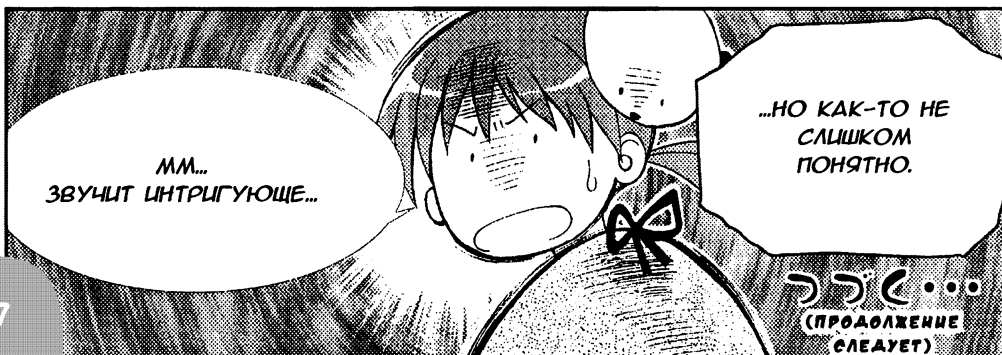


...ПРОСТАКИ!



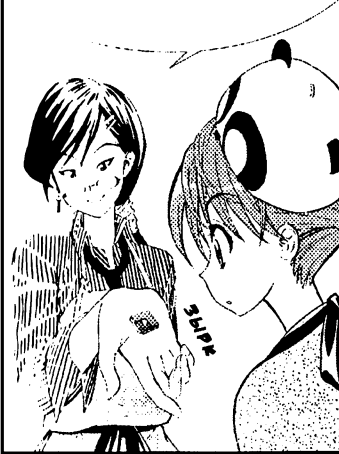
НЭЕ...



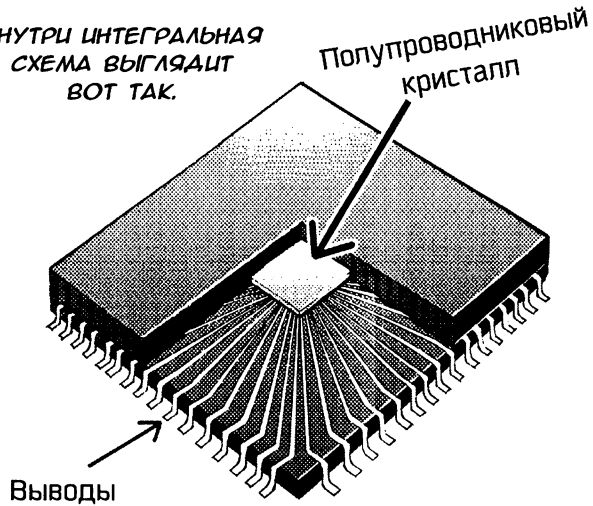


В чём преимущество цифровых схем?

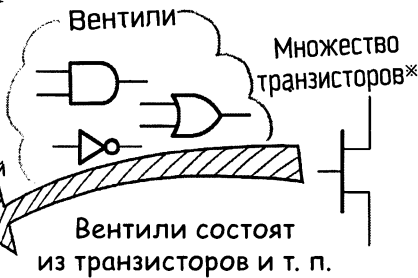
ЧТОБЫ ПОНЯТЬ, В ЧЁМ ПРЕИМУЩЕСТВО ЦИФРОВЫХ СХЕМ, ВЗГЛЯНИ НА ЭТО.



ВНУТРИ ИНТЕГРАЛЬНАЯ СХЕМА ВЫГЛЯДИТ ВОТ ТАК.



ОСНОВОЙ И ДЛЯ АНАЛОГОВЫХ, И ДЛЯ ЦИФРОВЫХ МИКРОСХЕМ СЛУЖИТ ПОЛУПРОВОДНИКОВЫЙ КРИСТАЛЛ.



ДА, ПРИПОМИНАЮ... ВНУТРИ ПОЛУПРОВОДНИКОВОГО КРИСТАЛЛА НАХОДИТСЯ МНОЖЕСТВО ЭЛЕМЕНТОВ, ТАКИХ КАК ТРАНЗИСТОРЫ.

ОНИ ФОРМИРУЮТ ТЕ САМЫЕ ЛОГИЧЕСКИЕ ВЕНТИЛИ, О КОТОРЫХ Я НЕДАВНО РАССКАЗЫВАЛ.



* это условное графическое обозначение транзистора. Подробнее о нём будет рассказано на стр. 86.

ВРЕМЯ ШЛО, ТЕХНОЛОГИИ
РАЗВИВАЛИСЬ.

НАУЧИЛИСЬ РАЗМЕЩАТЬ
ОГРОМНОЕ КОЛИЧЕСТВО
ТРАНЗИСТОРОВ В ОДНОМ
ПОЛУПРОВОДНИКОВОМ
КРИСТАЛЛЕ.

В НАШИ ДНИ В ПОЛУПРОВОДНИ-
КОВОМ КРИСТАЛЛЕ РАЗМЕРОМ
5 ММ МОЖНО РАЗМЕСТИТЬ
ДЕСЯТКИ МИЛЛИОНОВ ВЕНТИЛЕЙ.

НО РАНЬШЕ ТАК
НЕ УМЕЛИ

ДЕСЯТКИ
МИЛЛИОНОВ?!!

5 ММ

ДАЖЕ НЕ СОСЧИТАЕШЬ!
ТЕПЕРЬ Я ПОНИМАЮ,
ПОЧЕМУ ЦИФРОВЫЕ
СХЕМЫ ТАКЕ КРУТЫЕ!

ДАЖЕ КАРЛИКИ
ПОБЕДАТ,
КОГДА ИХ
ДЕСЯТКИ МИЛЛИОНОВ.

ЯСНО.

analog

ДРУГИМИ СЛОВАМИ, ЧЕМ БОЛЬШЕ РАЗВИВАЛИСЬ ПОЛУПРОВОДНИКОВЫЕ ТЕХНОЛОГИИ, ТЕМ ВЫГОДНЕЕ СТАНОВИЛОСЬ ИСПОЛЬЗОВАТЬ ЦИФРОВЫЕ СХЕМЫ ВМЕСТО АНАЛОГОВЫХ.

digital



ПРОГРЕСС ПОМОГ СОЗДАТЬ УМОПОМРАЧИТЕЛЬНОЕ ЧИСЛЕННОЕ ПРОВОСХОДСТВО.

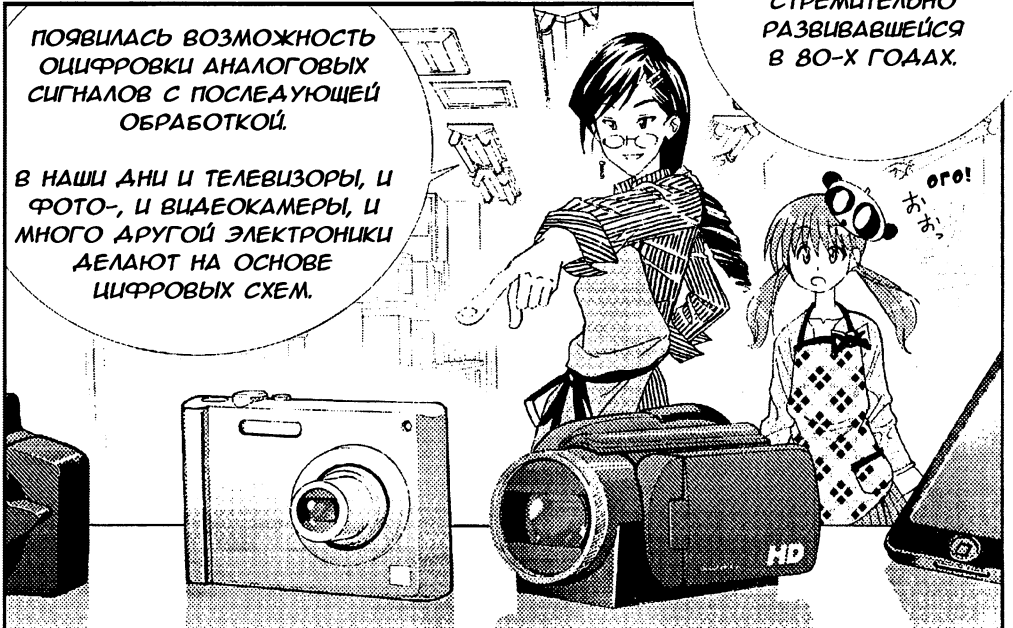


ЦИФРОВЫЕ СХЕМЫ СМОГЛИ ВЗЯТЬ ВЕРХ НАА АНАЛОГОВЫМИ ПРОСТЫМ УВЕЛИЧЕНИЕМ КОЛИЧЕСТВА ВЕНТИЛЕЙ.

ДА УЖ!
ДЕСЯТИМИЛЛИОННАЯ
ТОЛПА!!



ПОЛУЧАЕТСЯ, ЧТО
У АНАЛОГОВЫХ
ПРОСТО НЕ БЫЛО
ШАНСОВ?



* Комплиментарная структура металл-оксид-полупроводник (англ. CMOS).





ЕСЛИ ПОДУМАТЬ...
ТАКИ
УДИВИТЕЛЬНАЯ
ШТУКА,
ТРУДНО
ДАЖЕ
ПОВЕРИТЬ!



ХМ...
АНАЛОГОВЫЕ,
ЦИФРОВЫЕ...

Я И НЕ ЗНАЛА, ЧТО
ОНИ ВНУТРИ УСТРОЕНЫ
СОВЕРШЕННО
ПО-РАЗНОМУ!



ХОТЯ МНЕ
И ВСЁ РАВНО,
КАКОЙ ТЕЛЕВИЗОР -
ЦИФРОВОЙ
ИЛИ АНАЛОГОВЫЙ,
ЛИШЬ БЫ ДАЛИ ♡
ПОРАЗБИРАТЬ.

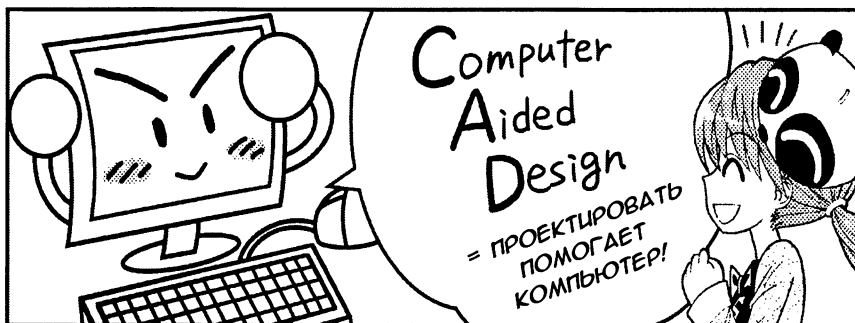


СЛЕДИ,
ЧТОБЫ ОНА
НЕ ПОЛОМАЛА
НАМ ТОВАР!



Я КТО ТУТ,
ОХРАННИК?!
.....

Проектировать цифровые схемы — просто!



Итак, здесь я расскажу, почему проектировать цифровые схемы проще, чем аналоговые.

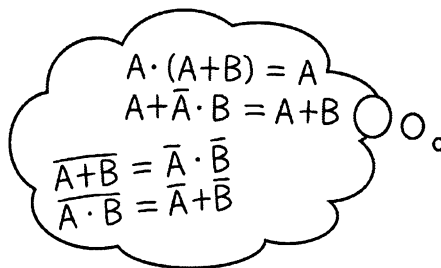
Цифровые схемы пришли на смену аналоговым, поэтому некоторые думают, что цифровые — более сложные, чем аналоговые, но это полнейшее заблуждение.



В самом деле, ведь в цифровых схемах всего два уровня: «низкий» и «высокий» (ноль и единица), и функции у вентилях тоже элементарные, поэтому проектировать их, наверное, проще, чем аналоговые.



Однако в университетах для описания цифровых схем используют алгебру логики. Она кажется сложноватой...



Да уж, всякие формулы...
А нельзя ли как-нибудь без неё обойтись?



Всё в порядке! Алгебра логики — удобный инструмент, но можно проектировать, даже не изучая её досконально.

Достаточно понять очень простые принципы — и можно будет обойтись без сложных формул.



Это радует!!

Значит, я смогу сама спроектировать схему электронного кубика, если постараюсь?

А более сложные электронные устройства тоже просто проектировать?



Хороший вопрос.

Цифровые схемы проектировать просто. Однако, так как они «побеждают числом», при проектировании огромных, сложных схем, например компьютера, действительно могут быть трудности.



Огромных, сложных схем...

Просто взять и нарисовать тут, наверное, не получится...



Да. В этом случае потребуются знания не только цифровых схем, но и, например, компьютерной архитектуры, системного проектирования.



Какая жалость!

В мире не всё так просто, как хотелось бы.



Не отчаивайся!

Ведь благодаря развитию компьютерных технологий у нас есть такой сильный помощник, как САД (Computer Aided Design: проектирование с помощью компьютера)*, который выполнит за нас всю рутинную работу по упрощению цифровой схемы, расчёту её быстроедействия и т. п.

* САПР (система автоматизированного проектирования).



В наши дни проектировщику цифровых схем достаточно описать работу системы на HDL (Hardware Description Language: язык описания аппаратуры) или на одном из компьютерных языков программирования, а САД сам создаст схему. Удобно, не так ли?



Понятно! Компьютер избавляет человека от нудной работы. Надо сказать ему спасибо!



Да. Но человеку тоже приходится использовать голову. Какие возможности должны быть у цифровой системы, чтобы она была привлекательной? Никакой компьютер не сможет решить это за человека, живущего в современную эпоху.



Значит, и в нашу эпоху пригодятся знания об устройстве цифровых схем — хотя бы для того, чтобы в голову приходили хорошие идеи.

Идеи, размышления... Важно развивать эти способности, которые есть только у человека.



Понятно!

Хотя мы не знаем, какие электронные устройства появятся в ближайшем будущем, сами идеи об этом рождаются в голове у человека!

Правда, технологиями для воплощения этих идей на практике в виде цифровых схем мы обязаны компьютерам, которые в наши дни получили такое большое развитие.



СЕГОДНЯ ТЫ МНЕ ТАК
МНОГО ИНТЕРЕСНОГО
РАССКАЗАЛ!
СПАСИБО!



И ДАЖЕ
ПРОВОДИЛ
ДО ДОМА!

НИЧЕГО, НИЧЕГО.
МНЕ ПРОСТО
БЫЛО ПО ПУТИ!




А ГЛАВНОЕ,
НАШ ДИРЕКТОР
СКАЗАЛА, ЧТОБЫ
Я ТЕБЯ ПРОВОДИЛ!!




АА...
НАШЕГО ДИРЕКТОРА
ЛУЧШЕ НЕ ЗЛИТЬ.






ОДНАКО Я ВОСХИЩАЮСЬ
ТВОИМ СТРЕМЛЕНИЕМ
К УЧЁБЕ.
СПЕРВА ПОДУМАЛ, ЧТО ТЫ
СТРАННАЯ КАКАЯ-ТО,
РАЗБИРАТЬ ЛЮБИШЬ.

АА...
ДУМАЮ, МНЕ ПОРА
ОТКАЗАТЬСЯ ОТ
ЭТОЙ ПРИВЫЧКИ -
РАЗБИРАТЬ...




ХМ...
НУ, НАШ ТОВАР А ТАК, ДУМАЮ,
РАЗБИРАТЬ УЖ НИЧЕГО ПЛОХОГО
ТОЧНО НЕ НАДО. В ЭТОМ НЕТ.

КАК?!



ВОТ ЭТО
АА!

ТЫ ПЕРВЫЙ,
КТО ЭТУ МОЮ
ПРИВЫЧКУ
ОДОБРИЛ.



У ВЕЛИКИХ ЛЮДЕЙ
БЫВАЮТ СТРАННОСТИ.
К ТОМУ ЖЕ ЕСТЬ ТАКОЕ
СЛОВО - РЕВЕРС-
ИНЖИНИРИНГ, КОГДА
СНАЧАЛА РАЗБИРАЮТ
ИЗДЕЛИЕ И СМОТРЯТ,
КАК ОНО УСТРОЕНО.



ДОБРЯК?!

ТАКАБА,
ПОХОЖЕ,
ХОРОШИЙ ПАРЕНЬ!

♡♡

Дополнительная информация

Что такое алгебра логики?

Слово «алгебра» происходит от арабского слова «аль-джабр» (восполнение). Используя буквенные обозначения вместо настоящих чисел, алгебра изучает операции над ними, например методы решения уравнений. В школьном курсе алгебры с помощью 4 арифметических операций (сложения, вычитания, умножения и деления) решают уравнения и находят числовые значения неизвестных величин, обозначенных буквами (переменных).

Особенность алгебры логики заключается в том, что здесь используют не арифметические, а логические операции, выполняемые над переменными, которые могут принимать всего два значения: «истина» (1) или «ложь» (0).

К логическим операциям относятся, например, AND (И), OR (ИЛИ), NOT (НЕ), о которых уже говорилось в этой книге. Мы обозначали их графически ми фигурами, но в алгебре логики используют более «математичные» символичные обозначения. На рис. 1 показано соответствие этих обозначений.

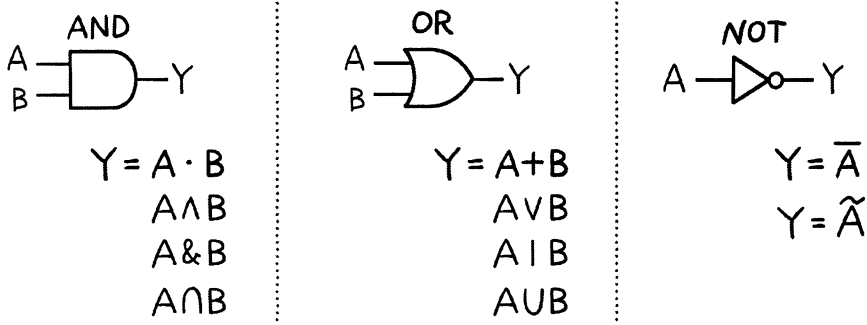
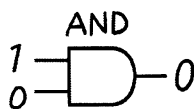


Рис. 1. Нотация MIL* и выражения алгебры логики

Операция AND (И) — логическое умножение, обозначается знаками \cdot , \wedge , $\&$, \cap ; для операции OR (ИЛИ) — логического сложения — используются такие знаки, как $+$, \vee , \mid , \cup .

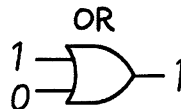
Я буду использовать знаки \cdot и $+$, так как в двоичной системе счисления операции AND соответствует умножение, а операции OR — сложение.

Например...



$$1 \cdot 0 = 0$$

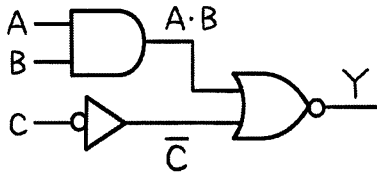
Умножение



$$1 + 0 = 1$$

Сложение

* О нотации MIL пойдет речь в главе 3.



$$Y = \overline{A \cdot B + \bar{C}}$$

Рис. 2. Пример схемы и её логического выражения

На рис. 2 показан пример представления схемы с помощью выражения алгебры логики (логического выражения). Как и в арифметических выражениях, операция AND (логическое умножение) имеет приоритет над операцией OR (логическим сложением). Операция NOT (инверсия) меняет значение выражения под чертой на противоположное: 1 превращается в 0, а 0 — в 1.

Хотя графическая форма записи более наглядно описывает устройство и работу цифровой схемы, иногда логические выражения позволяют сэкономить место на бумаге, а также описать цифровую схему в компьютерной программе, например на языке описания аппаратуры. Таким образом, алгебра логики и цифровые схемы — неразрывно связанные понятия.

Как и в обычной алгебре, в алгебре логики мы занимаемся преобразованием выражений.

Например, законы де Моргана, о которых будет рассказано в главе 2, в виде логических выражений записываются так:

$$\overline{A \cdot B} = \bar{A} + \bar{B} \quad \overline{A + B} = \bar{A} \cdot \bar{B}$$

Законы де Моргана говорят нам о том, что если обратить входы, тип вентиля (All или Exist) и выход, то получится эквивалентный логический элемент.

* All — это вентиль с закруглённым концом, а Exist — с заострённым. О них пойдёт речь на стр.65.

Законы де Моргана

$$\overline{A \cdot B} = \bar{A} + \bar{B}$$

$$\overline{A + B} = \bar{A} \cdot \bar{B}$$



В алгебре логики действуют законы ассоциативности, коммутативности и другие. Основные законы алгебры логики бывают полезны при упрощении схем, о котором я расскажу в главе 4.

Так как алгебра логики — научная основа проектирования цифровых схем, в некоторых вузах заучивают наизусть её основные законы и тренируются преобразовывать формулы, однако большого практического смысла в этом нет, так как проектировать полезные схемы можно и без этого. Правда, умение читать логические выражения — безусловно, важный навык, а некоторым читателям мир алгебры логики может показаться увлекательным. Если у вас есть желание, попробуйте самостоятельно изучить связь алгебры логики с методами проектирования цифровых схем, о которых пойдёт речь в этой книге.

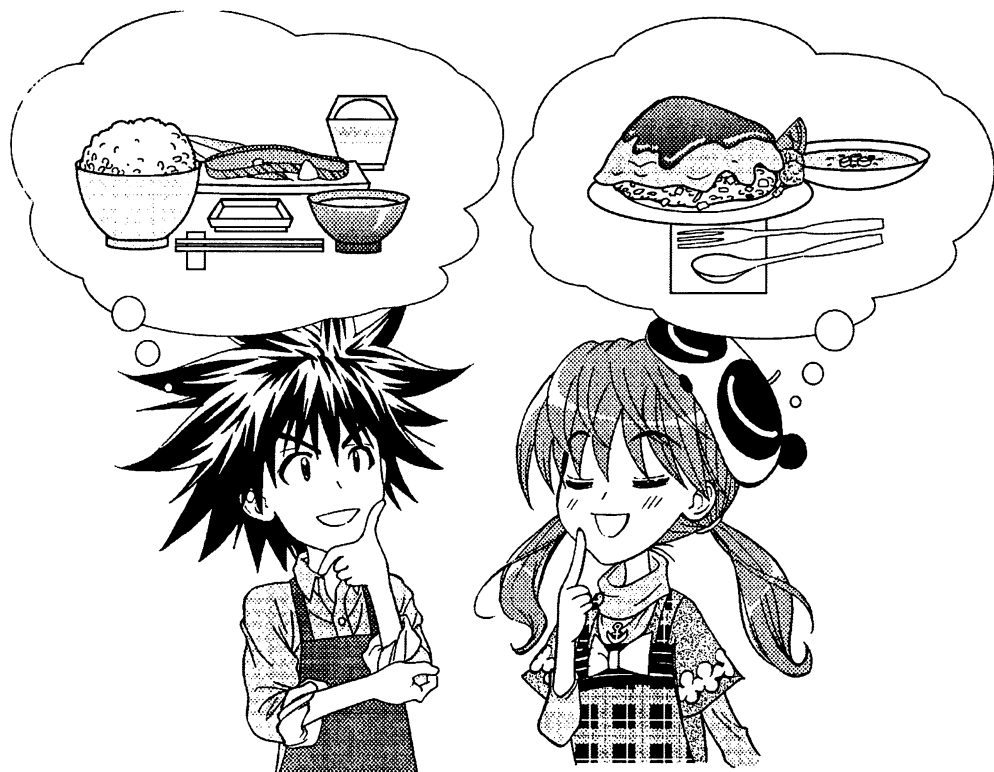
Но даже те читатели, которым не нравится алгебра логики из-за математических формул, без проблем смогут проектировать цифровые схемы. Кроме того, вполне можно обойтись и без знания таких связанных с цифровыми схемами дисциплин, как теория множеств, математическая логика и т. п.

Объяснение терминов главы 2

- **Аналоговые схемы** — электронные схемы для обработки электрических сигналов, изменяющихся в непрерывном диапазоне значений (т. е. аналоговых сигналов), поэтому в таких схемах даже малые изменения сигналов имеют смысл.
- **Цифровые схемы** — электронные схемы для обработки электрических сигналов, принимающих дискретные значения, которых обычно два: «высокий» (H) и «низкий» (L) уровни, соответствующие двоичным числам 1 и 0. Уровень напряжения выше определённого порога считается «высоким» (H), а ниже этого порога — «низким» (L). Другими словами, это — логические схемы, выполняющие операции над множествами значений L и H.
- **Логические операции** — операции над множествами двоичных чисел (0 и 1), такие как логическое умножение (AND, И), логическое сложение (OR, ИЛИ), отрицание (инверсия, NOT, НЕ), изучаемые в алгебре логики. Подробнее см. в «Дополнительной информации».
- **Серия 74** — это стандартные цифровые микросхемы, содержащие базовые логические вентили. Мы рассмотрели чип SN7432, содержащий 4 вентиля OR. Кроме того, в этой серии есть такие чипы, как SN7408, содержащий 4 вентиля AND; SN7404, содержащий 6 вентилях NOT. Расположение выводов этих микросхем стандартизировано и не зависит от производителя. В этой серии выпускались и более сложные микросхемы с триггерами и регистрами (о них будет рассказано в главе 4), мультиплексорами (о них будет рассказано в главе 3). Раньше, чтобы собрать цифровое устройство, мы покупали эти стандартные ИС и соединяли их проводами на печатной плате с отверстиями, но в наши дни эти чипы постепенно исчезают в связи с применением ППВМ (программируемых пользователем вентилях матриц), о которых шла речь в главе 1.
* Отечественные аналоги: SN7432 — К155ЛП1, SN7508 — К155ЛН1, SN7504 — К155ЛН1. — *прим. перев.*
- **Логические схемы** — это цифровые схемы, выполняющие логические операции, но их можно считать синонимом цифровых микросхем, так как обычно цифровые схемы выполняют только логические операции. Но почему же тогда есть два названия? Если мы хотим акцентировать внимание на выполняемых операциях, то говорим «логические схемы», а если на их электрических характеристиках, то используем название «цифровые схемы». Однако этому можно не уделять большого внимания, так как строго установленных правил разграничения этих понятий не существует.
- **КМОП** — см. «Дополнительную информацию» главы 3.

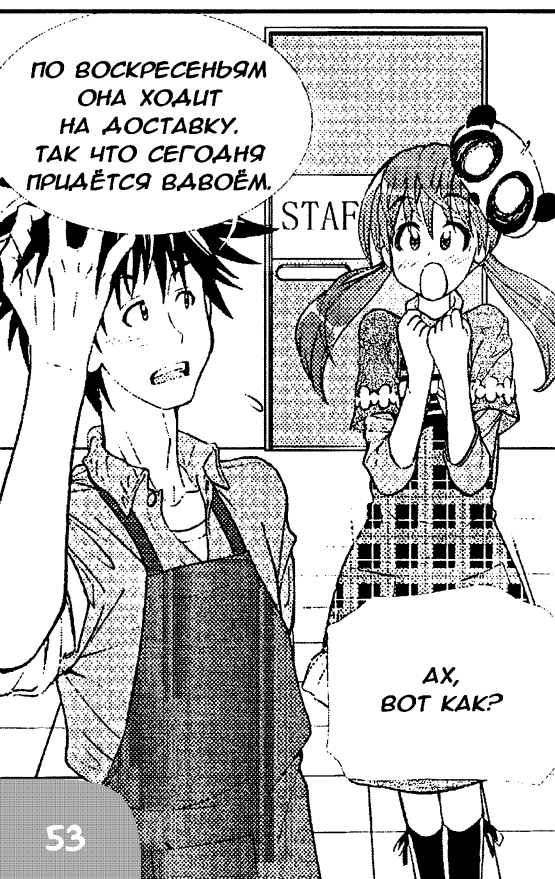
ГЛАВА 3

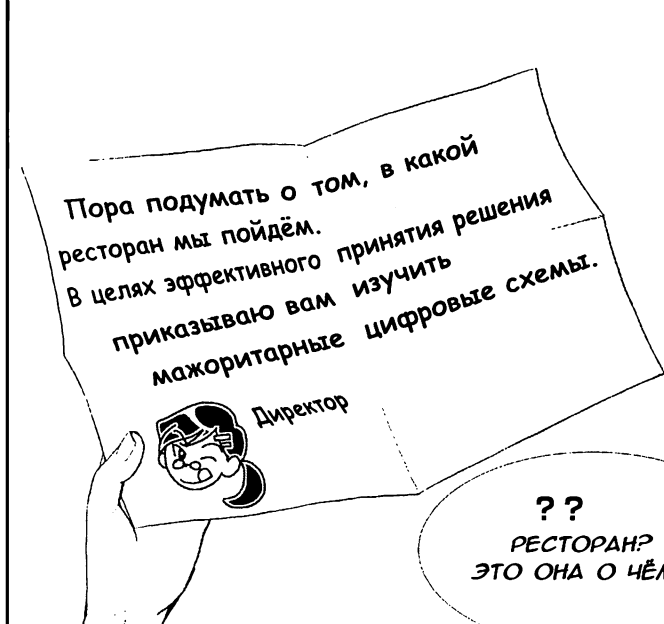
СТРОИМ КОМБИНАЦИОННУЮ СХЕМУ



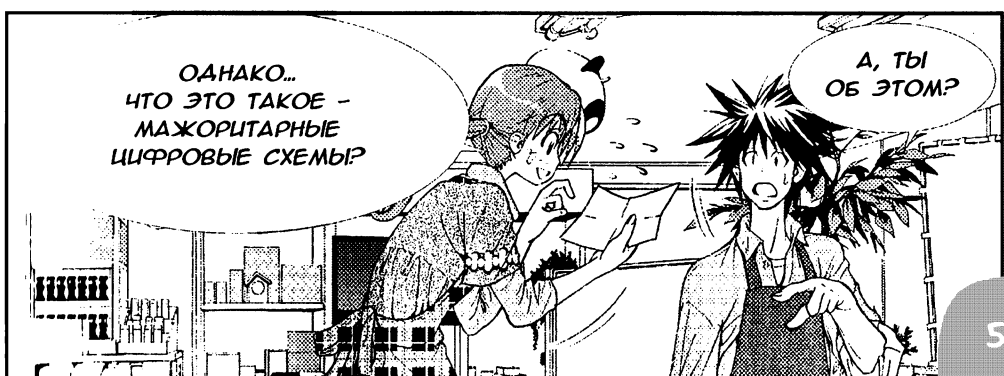
1. ТАБЛИЦА ИСТИННОСТИ, НОТАЦИЯ MIL

Мажоритарные цифровые схемы

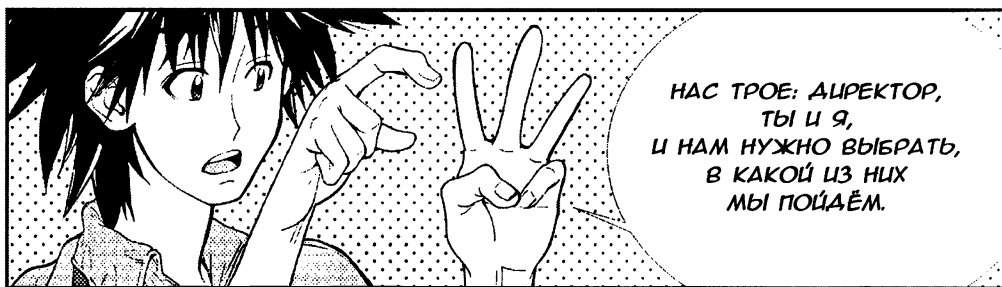
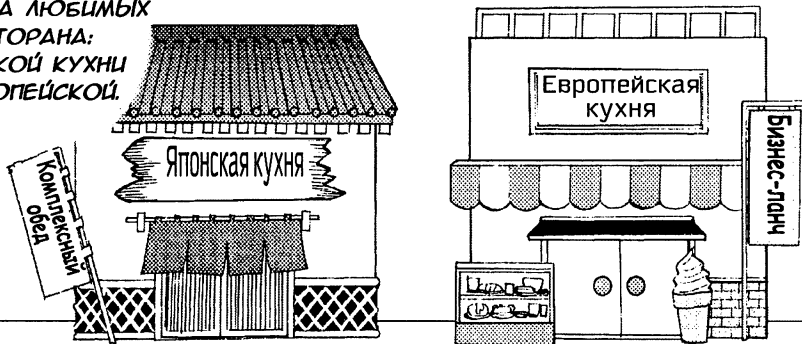




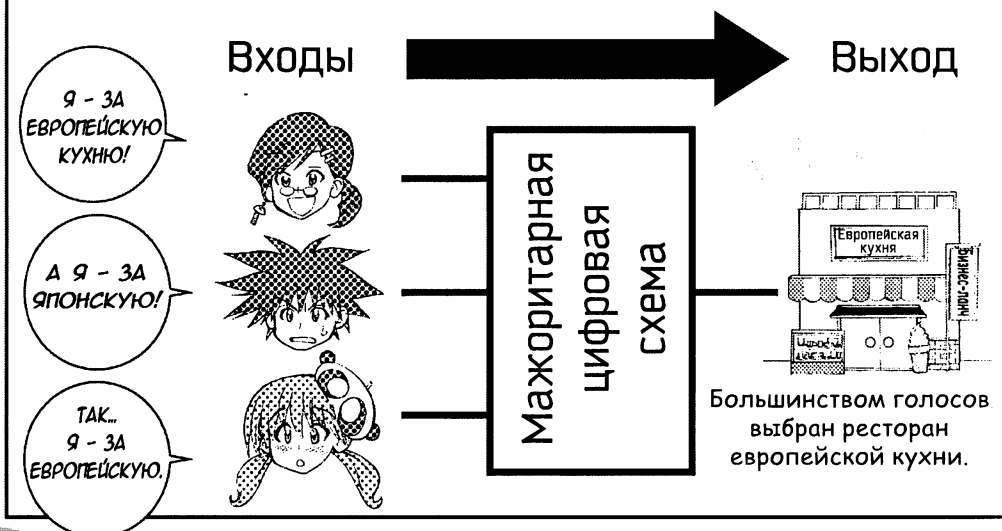
??
РЕСТОРАН?
ЭТО ОНА О ЧЁМ?



У НАШЕГО ДИРЕКТОРА
ЕСТЬ ДВА ЛЮБИМЫХ
РЕСТОРАНА:
ЯПОНСКОЙ КУХНИ
И ЕВРОПЕЙСКОЙ.



ДРУГИМИ СЛОВАМИ, МЫ - ЭТО ТРИ ВХОДА, КОТОРЫЕ
ОПРЕДЕЛЯТ РЕЗУЛЬТАТ НА ВЫХОДЕ.
ЭТО БУДЕТ ВЫГЛЯДЕТЬ, НАПРИМЕР, ВОТ ТАК:



ТАКАВА! 高埜あさひ

PRICTPACTИИ В EAЕ
Y MEHЯ OCOБЫX HEТ -
BCE PABHO,
B KAKOЙ PECTOPAHИ!

何故か
言った通りには
なっていないか
カキニ
聞かされた後
聞かすか?!

КУДА ВЫ
C ДИРЕКТОРОМ
РЕШИТЕ, ТУДА
И ПОЙДУ.

ТЫ НЕ МОЖЕШЬ
НЕ ЧАСТОВАТЬ
В ПРИНЯТИИ
РЕШЕНИЯ!
ВЕДЬ ЭТО
КАСАЕТСЯ ТЕБЯ!

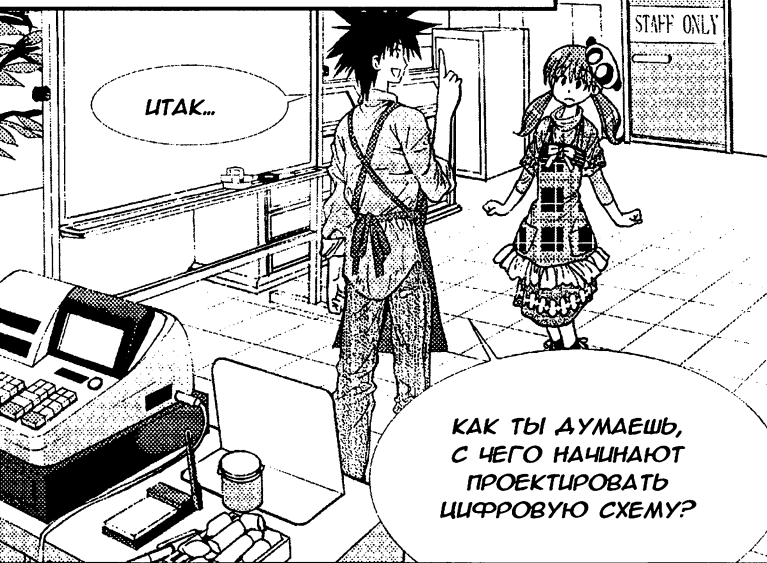
К ТОМУ ЖЕ
УКАЗАНИЯ
ДИРЕКТОРА
ЛУЧШЕ ИСПОЛНЯТЬ!

AAA?

ХУ-ХУ

ХОРОШО!
РАССКАЖИ МНЕ
ПРО ЭТИ
МАЖОРИТАРНЫЕ!

Строим таблицу истинности из L и H



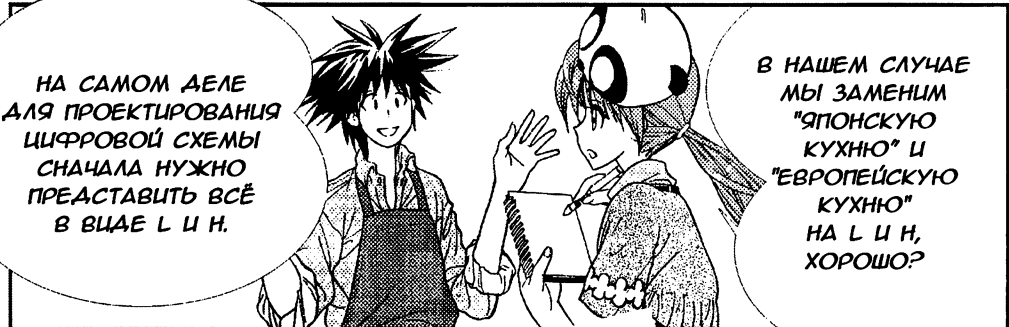
ИТАК...

КАК ТЫ ДУМАЕШЬ,
С ЧЕГО НАЧИНАЮТ
ПРОЕКТИРОВАТЬ
ЦИФРОВОЮ СХЕМУ?



БЕРУТ БУМАГУ
И КАРАНДАШ!

ЭТО УЖ
СЛИШКОМ...



НА САМОМ ДЕЛЕ
ДЛЯ ПРОЕКТИРОВАНИЯ
ЦИФРОВОЙ СХЕМЫ
СНАЧАЛА НУЖНО
ПРЕДСТАВИТЬ ВСЁ
В ВИДЕ L И H.

В НАШЕМ СЛУЧАЕ
МЫ ЗАМЕНИМ
"ЯПОНСКУЮ
КУХНЮ" И
"ЕВРОПЕЙСКУЮ
КУХНЮ"
НА L И H,
ХОРОШО?



ДА!
ЗАЕШЬ МЫ
УСЛОВИМСЯ,
ЧТО...

...ЯПОНСКАЯ КУХНЯ
БУДЕТ L ОТ СЛОВА "РИС",
А ЕВРОПЕЙСКАЯ - H
ОТ СЛОВА "ГАМБУРГЕР".

ХОТЯ НИЧТО
НЕ ЗАПРЕЩАЕТ
СДЕЛАТЬ
И НАОБОРОТ.

ТАКАБА, "РИС"
ПО АНГЛИЙСКИ
ПИШЕТСЯ "RISE",
А НЕ "LISE".

НО ВСЁ
РАВНО
ЗАБАВНО.

ИТАК, КАЖДЫЙ ИЗ ТРЁХ ЧЕЛОВЕК (ВХОДОВ)...

Выбор из двух вариантов

...МОЖЕТ ПРОГОЛОСОВАТЬ ЛИБО ЗА ЯПОНСКУЮ КУХНЮ (L), ЛИБО ЗА ЕВРОПЕЙСКУЮ (H).

ЛИПИ Н

ЛИПИ Н

ЛИПИ Н

$\times \times = 2^3 = 8$ ВОЗМОЖНЫХ КОМБИНАЦИЙ

3 человека

ЗНАЧИТ, КОЛИЧЕСТВО ВОЗМОЖНЫХ КОМБИНАЦИЙ БУДЕТ ДВА В ТРЕТЬЕЙ СТЕПЕНИ, ТО ЕСТЬ $2 \times 2 \times 2 = 8$.

ЯСНО. ЕСЛИ РЕШАЮТ 3 ЧЕЛОВЕКА, ТО КОМБИНАЦИЙ БУДЕТ ВСЕГО ДВА В ТРЕТЬЕЙ СТЕПЕНИ, ЕСЛИ 4 ЧЕЛОВЕКА - ДВА В ЧЕТВЁРТОЙ, А ЕСЛИ N ЧЕЛОВЕК - ДВА В N-ОЙ.

ВЕРНО!

А ТЕПЕРЬ ДАВАЙ ЗАПИШЕМ ВСЕ ЭТИ ВОСЕМЬ ВОЗМОЖНЫХ КОМБИНАЦИЙ!

ОКРИП
キッフ
キッフ
ОКРИП

ВОТ ТАК, В ВИДЕ ТАБЛИЦЫ.

А, В, С - ЭТО МЫ ВТРОЁМ, ТО ЕСТЬ ТРИ ВХОДА.

З - ЭТО ВЫХОД, НА КОТОРЫЙ ВЫВОДИТСЯ РЕЗУЛЬТАТ.

С	В	А	З
L	L	L	L

ЕСЛИ ВСЕ ТРОЕ ВЫБРАЛИ ЯПОНСКУЮ КУХНЮ (L), ТО И РЕЗУЛЬТАТ ПО ПРАВИЛУ БОЛЬШИНСТВА БУДЕТ L.

ЧТО?

А ПОЧЕМУ ТЫ
НАПИСАЛ
ВХОДЫ А, В, С
В ОБРАТНОМ
ПОРЯДКЕ?

ТАК УДОБНЕЕ
ПРЕДСТАВИТЬ НАБОР
ВХОДНЫХ ЗНАЧЕНИЙ
В ВИДЕ АВОИЧНОГО
ЧИСЛА.

С	В	А	З
Л	Л	Л	Л

КСТАТИ, В СЧЁТНИКАХ
СЕРИИ 74 МЛАДШИЙ
РАЗРЯД ТОЖЕ
ОБОЗНАЧАЕТСЯ БУКВОЙ
"А".

ПОНЯТНО.
ТАКУЮ ТАБЛИЦУ
Я И САМА МОГУ
НАРИСОВАТЬ.

С	В	А	З
Л	Л	Л	Л

СКРИП

СКРИП
キユ...

ТАК...
ПОЛУЧИЛОСЬ
В КОМБИНАЦИИ!

ПРОВЕРЬ!!

МОЛОДЕЦ!
ТЫ СОСТАВИЛА
ТАБЛИЦУ
ИСТИННОСТИ!

С	В	А	З
Л	Л	Л	Л
Л	Л	Н	Л
Л	Н	Л	Л
Л	Н	Н	Н
Н	Л	Л	Л
Н	Л	Н	Н
Н	Н	Л	Н
Н	Н	Н	Н

ТАБЛИЦА
ИСТИННОСТИ -
ЭТО...

...ТАБЛИЦА, СОДЕРЖАЩАЯ
ВСЕ ВОЗМОЖНЫЕ
КОМБИНАЦИИ ВХОДНЫХ
ЗНАЧЕНИЙ
И РЕЗУЛЬТАТЫ
ПРЕОБРАЗОВАНИЯ
КАЖДОЙ
ИЗ КОМБИНАЦИЙ
В ЛОГИЧЕСКОЙ СХЕМЕ!!



ОДНАКО...
ТАКУЮ ТАБЛИЦУ
И МЛАДШИЙ ШКОЛЬНИК
СМОЖЕТ НАРИСОВАТЬ.

ПОДРАЗЖАЕТ
ДИРЕКТОРУ?



КАКАЯ ОТ
НЕЁ ПОЛЬЗА?



ПОЛЬЗА
ОГРОМНАЯ!!

ТАБЛИЦА ИСТИННОСТИ -
ЭТО ФУНДАМЕНТ,
НА КОТОРОМ СТРОЯТ
ЦИФРОВУЮ СХЕМУ!!!



!?
ВОТ КАК?

ЭТО
ПОДРАЗЖАЕТ!



АА!
НА САМОМ ДЕЛЕ ВСЕ
ЦИФРОВЫЕ СХЕМЫ
МОЖНО РАЗДЕЛИТЬ НА
ДВА БОЛЬШИХ ТИПА.

СХИМ

СХИМ

Цифровые схемы

Комбинационные

Последовательные*

СЕЙЧАС МЫ ИЗУЧАЕМ
КОМБИНАЦИОННЫЕ СХЕМЫ.
В НИХ ЗНАЧЕНИЕ НА ВЫХОДЕ
ЗАВИСИТ ТОЛЬКО ОТ ТЕКУЩЕЙ
КОМБИНАЦИИ ЗНАЧЕНИЙ
НА ВХОДАХ.

* О последовательных схемах
будет рассказано в главе 5.

АГА...

В САМОМ ДЕЛЕ,
ЕСЛИ КАЖДАЯ ИЗ
ТРОИХ ВЫБЕРЕТ
L ИЛИ H,
ТО РЕСТОРАН
БУДЕТ ВЫБРАН
ОДНОЗНАЧНО.

ЗНАЧИТ,
МАЖОРИТАРНЫЕ СХЕМЫ -
КОМБИНАЦИОННЫЕ?

С	В	А	З
L	L	L	L
L	L	H	L
L	H	L	L
L	H	H	L
H	L	L	L
H	L	H	H
H	H	L	H
H	H	H	H

ВЕРНО!
СТОИТ НАМ НАЧЕРТИТЬ
ТАБЛИЦУ ИСТИННОСТИ...

...И КОМБИНАЦИОННАЯ
СХЕМА ГОТОВА!



ММ...

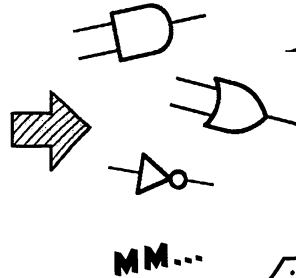
ЧТО?

КАКИЕ-ТО
СОМНЕНИЯ?

ДА,
НЕ ПОНЯТНО, КАК
ЭТО ДЕЛАЕТСЯ
НА ПРАКТИКЕ.

КАК ПРЕДСТАВИТЬ ЭТУ ТАБЛИЦУ
В ВИДЕ ЭЛЕМЕНТОВ AND,
OR, NOT, КОТОРЫЕ МЫ ИЗУЧИЛИ?

С	В	А	Z
L	L	L	L
L	L	H	L
L	H	L	L
L	H	H	H
H	L	L	L
H	L	H	H
H	H	L	H
H	H	H	H



АХ, ИЗВИНИ!
Я ДОЛЖЕН БЫЛ
РАССКАЗАТЬ
ТЕБЕ ЕЩЁ
КОЕ-ЧТО.

ЧТОБЫ
СПРОЕКТИРОВАТЬ
СХЕМУ НА ОСНОВЕ
ТАБЛИЦЫ
ИСТИННОСТИ,...

Нотация
MIL

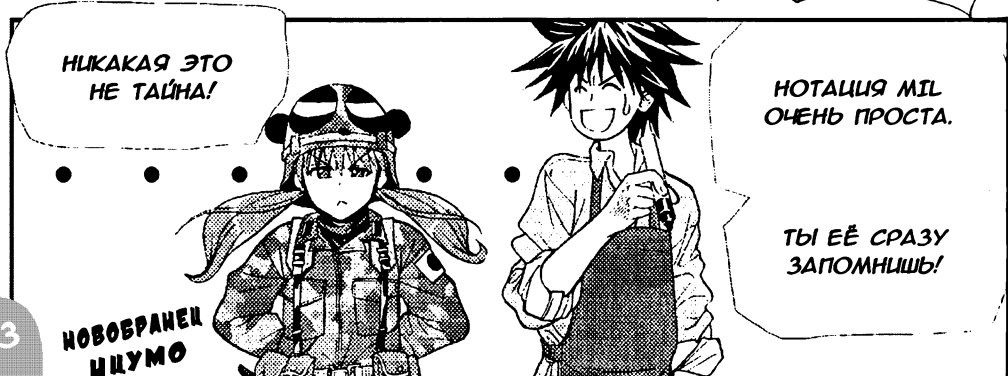
...НАДО ЗНАТЬ
НОТАЦИЮ MIL.

MIL

.....
MIL...?

ОПЯТЬ ЗАГАДОЧНОЕ
СЛОВО...

Что такое нотация MIL?



Входы → Выход | Входы → Выход

СНАЧАЛА ЭТО.

Я УЖЕ РИСОВАЛ ЭТИ ФИГУРЫ, КОГДА РАССКАЗЫВАЛ О БАЗОВЫХ ВЕНТИЛЯХ.

АА!
ЭТО ЭЛЕМЕНТЫ
AND И OR!

МОЛОДЕЦ!
НО НА САМОМ ДЕЛЕ
У ЭТИХ ЗНАЧКОВ
ВЕНТИЛЕЙ ТОЖЕ
ЕСТЬ НАЗВАНИЯ.

ВЕНТИЛИ С ЗАКРУГЛЁННЫМ КОНЦОМ - ЭТО ALL.

"Все входы"
(all)

"Хотя бы один из входов"
(exist)

ВЕНТИЛИ С ЗАОСТРЁННЫМ КОНЦОМ - ЭТО EXIST.

УСЛОВИЕ ВЕНТИЛЯ ТИПА ALL - "ВСЕ ВХОДЫ".

УСЛОВИЕ ВЕНТИЛЯ ТИПА EXIST - "ХОТЯ БЫ ОДИН ИЗ ВХОДОВ".

СКРИП

ПОНЯТНО!
ВЕНТИЛЬ ALL
ОАОБРИТ "СВАДЬБУ"
ТОЛЬКО ПРИ
УСЛОВИИ
ВЗАИМНОСТИ,
А ВЕНТИЛЬ EXIST
"ПОЖЕНИТ"
ДАЖЕ В СЛУЧАЕ
НЕРАЗДЕЛЁННОЙ
ЛЮБВИ
(см. стр. 36)

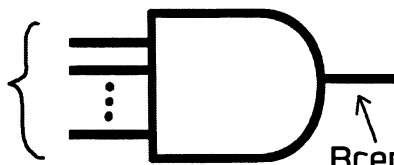
АА,
А ТЕПЕРЬ,
ВНИМАНИЕ!!

CAUTION



Входы → Выход

Иногда
больше
двух!



↑
Всегда
один

РАНЬШЕ Я
ГОВОРИЛ ОБ
ЭЛЕМЕНТАХ С
О ДВУМЯ ВХОДАМИ..

НО НА САМОМ ДЕЛЕ
ИХ МОЖЕТ БЫТЬ
БОЛЬШЕ.

НО ВЫХОДА
ВСЕГДА ОДИН.

НАПРИМЕР,
ЕСЛИ ВХОДОВ ПЯТЬ,
ТО УСЛОВИЕ
ВЕНТИЛЯ ALL -
"ВСЕ ПЯТЬ ВХОДОВ"...

...А УСЛОВИЕ
ВЕНТИЛЯ EXIST -
"ХОТЯ БЫ ОДИН ИЗ
ПЯТИ ВХОДОВ".



Все
(all)



Хотя бы один
(exist)

ВЕДЬ ALL
ПО-АНГЛИЙСКИ
ОЗНАЧАЕТ "ВСЕ".

А EXIST
ПЕРЕВОДИТСЯ
КАК "СУЩЕСТВУЕТ".
В ТОМ СМЫСЛЕ, ЧТО
"СУЩЕСТВУЕТ
ХОТЯ БЫ ОДИН".

ОГО!

АА, ТУТ
ДЕЙСТВИТЕЛЬНО
ТРЕБУЕТСЯ
ВНИМАНИЕ.

УФФ...
ЭТО БЫЛ
НЕ ВРАГ...

КРОМЕ ТОГО,
ЛИНИИ ВХОДА
И ВЫХОДА
НАЗЫВАЮТСЯ
"СИГНАЛЬНЫМИ"...

1100101010
0110010101
1001010111
1011001010

Сигнальные
линии

...ТАК КАК ОНИ
ПРОВОДЯТ ЦИФРОВЫЕ
СИГНАЛЫ 0 И 1
(L И H).

L И H НАЗЫВАЮТСЯ
"ЛОГИЧЕСКИМИ
УРОВНЯМИ".

ТАКИМ ОБРАЗОМ,
ЕСТЬ ВСЕГО АВА ТИПА
ЦИФРОВЫХ СИГНАЛОВ:
"НИЗКИЙ" И "ВЫСОКИЙ",
КОТОРЫЕ МЫ ОБОЗНАЧАЕМ
БУКВАМИ L И H.

Низкий
логический
уровень

Высокий
логический
уровень

ТИПЫ ВЕНТИЛЕЙ
ALL И EXIST.
УРОВНИ, СИГНАЛЬНЫЕ
ЛИНИИ.

ХОП!

ВСЁ ПОНЯЛА!
ТАК ТОЧНО!

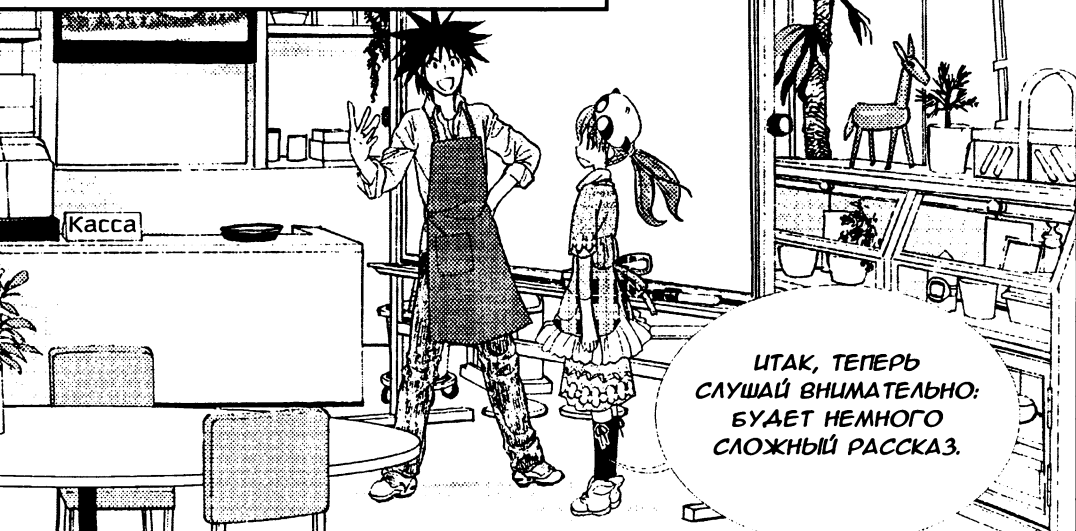
МАТИЛЬДА-
САНИ!

А-А-А

ПОХОЖЕ,
ЧТО ОНА
ЗАЦИКЛИЛАСЬ
НА СЛОВЕ
MILITARY...

※ Матильда Аджан — героиня аниме «Мобильный воин Гандам». — прим. перев.

Активный низкий и активный высокий



ВХОДНЫЕ И ВЫХОДНЫЕ СИГНАЛЫ ЦИФРОВЫХ СХЕМ ПРИНИМАЮТ ЛИШЬ ДВА УРОВНЯ. СМЫСЛ ЦИФРОВОЙ СХЕМЫ - РЕАГИРОВАТЬ НА ИЗМЕНЕНИЯ ВХОДНЫХ СИГНАЛОВ ИЗМЕНЕНИЕМ ВЫХОДНЫХ. ПРИ ЭТОМ МЫ МОЖЕМ ПРОЕКТИРОВАТЬ СХЕМУ ТАК, ЧТОБЫ НУЖНЫЙ НАМ ВЫХОДНОЙ СИГНАЛ ИМЕЛ ЛИБО "НИЗКИЙ" (L), ЛИБО "ВЫСОКИЙ" (H) УРОВЕНЬ, КОТОРЫЙ НАЗЫВАЕТСЯ АКТИВНЫМ УРОВНЕМ.

ММ...
АКТИВНЫЙ УРОВЕНЬ??
КАК-ТО НЕ ОЧЕНЬ
ПОНЯТНО...



ТОГДА
ОБЪЯСНЮ
НА ПРИМЕРЕ.

ПРЕДСТАВЬ
КАРТОЧНУЮ ИГРУ,
В КОТОРОЙ
НЕВОЗМОЖНА НИЧЬЯ -
ТОЛЬКО ВЫИГРЫШ
ИЛИ ПРОИГРЫШ.

И МЫ С ТОБОЙ
СЕЛИ И ВМЕСТО РАБОТЫ
СЫГРАЛИ ДВЕ ПАРТИИ.



ПУСТЬ ИТОГИ ДВУХ ПАРТИЙ ПОДАЮТСЯ НА ВХОДЫ ВЕНТИЛЯ OR.
НО ЕСЛИ МЫ НЕ РЕШИМ, ЧТО СЧИТАТЬ ИТОГОВЫМ ВЫИГРЫШЕМ ИЛИ
ПРОИГРЫШЕМ, - ТО КАК НАМ ОПРЕДЕЛИТЬ ИТОГОВЫЙ РЕЗУЛЬТАТ?

Выиграла!

5

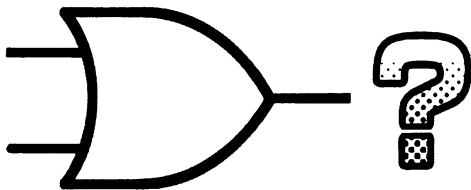
(Первая партия)



(Вторая партия)



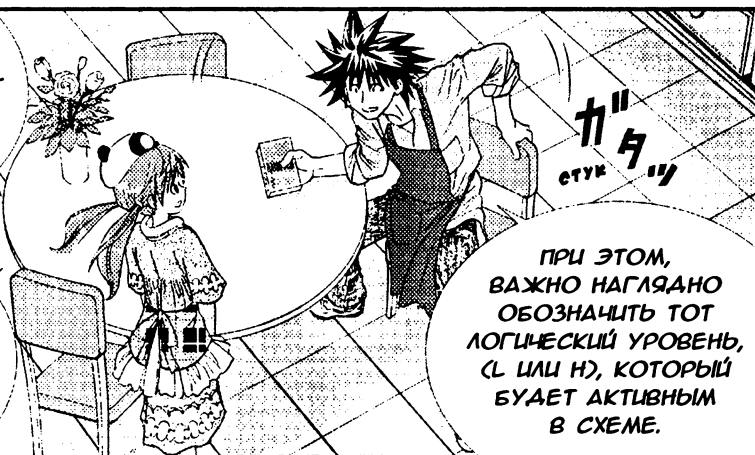
Проиграла...



СЧИТАТЬ ЛИ ЕГО ПОБЕДОЙ, ЕСЛИ БЫЛ ХОТЯ БЫ ОДИН ВЫИГРЫШ?
ИЛИ СЧИТАТЬ ЕГО ПОРАЖЕНИЕМ, ЕСЛИ БЫЛ ХОТЯ БЫ ОДИН ПРОИГРЫШ?

ДА! Я ПОНЯЛА,
КАК ЭТО ВАЖНО -
РЕШИТЬ, НА ЧТО
ОБРАЩАТЬ
ВНИМАНИЕ!

А ЕЩЁ НУЖНО
ПРИСВОИТЬ
УРОВНИ L И H
ВЫИГРЫШУ
И ПРОИГРЫШУ,
ДА?



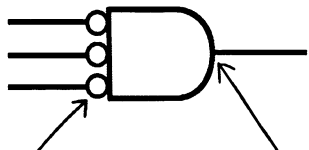
ПРИ ЭТОМ,
ВАЖНО НАГЛЯДНО
ОБОЗНАЧИТЬ ТОТ
ЛОГИЧЕСКИЙ УРОВЕНЬ,
(L ИЛИ H), КОТОРЫЙ
БУДЕТ АКТИВНЫМ
В СХЕМЕ.

В ЭТОМ НАМ
ПОМОЖЕТ
ОБЫКНОВЕННЫЙ
КРУЖОЧЕК.

КРУЖОЧЕК?

ЕСЛИ АКТИВНЫЙ УРОВЕНЬ -
"НИЗКИЙ" (L),
ТО ПОМЕЧАЕМ СИГНАЛЬНУЮ
ЛИНИЮ КРУЖОЧКОМ,
А ЕСЛИ "ВЫСОКИЙ" (H),
ТО НИКАК ЕЁ НЕ ПОМЕЧАЕМ.

ЧТО?!



Входы -
"активные
низкие"

Выход -
"активный
высокий"

[с кружочками]* [без кружочка]

УРОВЕНЬ, НА КОТОРЫЙ
МЫ ОБРАЩАЕМ ВНИМАНИЕ,
НАЗЫВАЕТСЯ "АКТИВНЫМ".

ВХОДЫ ИЛИ ВЫХОД
С КРУЖОЧКАМИ
МЫ БУДЕМ НАЗЫВАТЬ
"АКТИВНЫМИ НИЗКИМИ",
А БЕЗ КРУЖОЧКОВ -
"АКТИВНЫМИ ВЫСОКИМИ".

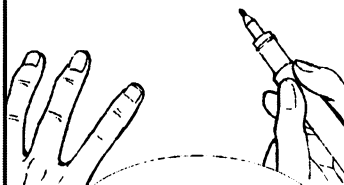
ДРУГИМИ СЛОВАМИ,
ЕСЛИ ЕСТЬ КРУЖОЧЕК,
ТО "АКТИВНЫЙ УРОВЕНЬ"
НА ЭТОЙ ЛИНИИ - "НИЗКИЙ"!

ЯСНО.
ТО ЕСТЬ
НУЖНО
СМОТРЕТЬ
НА ТОЧКИ
СОЕДИНЕНИЯ
ЛИНИЙ
С ВЕНТИЛЕМ?

ТАКОЙ
МАЛЕНЬКИЙ
КРУЖОК,
А ТАКОЙ
ВАЖНЫЙ!



ИТАК, ПЕРЕХОДИМ
К ДЕЛУ.
ПОСМОТРИ
ВНИМАТЕЛЬНО
НА ЭТИ ВХОДЫ,
ВЫХОД
И ВЕНТИЛЬ.



ПОНИМАЕШЬ,
В ЧЁМ ТУТ
СМЫСЛ?

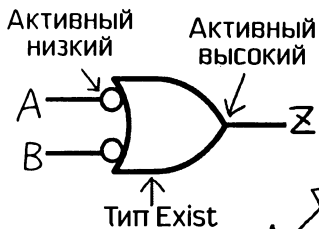
* В России принято называть такие входы инвертирующими. Также и выходы с кружочками называются инверсными.

ТАК... ВХОДЫ
С КРУЖОЧКАМИ -
АКТИВНЫЕ НИЗКИЕ,
А ВЫХОД
БЕЗ КРУЖОЧКА -
АКТИВНЫЙ ВЫСОКИЙ.

ЗАОСТРЁННЫЙ
ВЕНТИЛЬ -
ТИП EXIST, АА?

ВЕРНО! В ОБЩЕМ, ЭТОТ
УСЛОВНЫЙ ЗНАК ПОКАЗЫВАЕТ,
ЧТО ЕСЛИ ХОТЯ БЫ НА ОДНОМ
ИЗ ВХОДОВ ПРИСУТСТВУЕТ 1,
ТО НА ВЫХОДЕ БУДЕТ 1.

ОГО! ЗНАЧИТ,
ЧТОБЫ ПОНЯТЬ
СМЫСЛ, ДОСТАТОЧНО
ВЗГЛЯНУТЬ НА ВХОДЫ,
ВЫХОД И ВЕНТИЛЬ?



B	A	Σ
L	L	H
L	H	H
H	L	H
H	H	L

Эта комбинация нас не интересует, поскольку клапан не переключается в высокое состояние. При этом, на обоих входах - H.

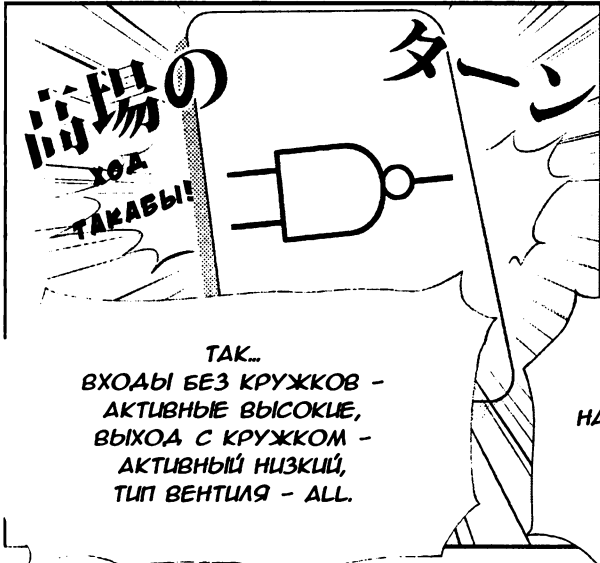


А ЭТУ КАРТУ ОТГАДАЕШЬ?

Если хотя бы на одном из входов - L, то на выходе будет H.

ЭТО - ТАБЛИЦА ИСТИННОСТИ, СОДЕРЖАЩАЯ ВСЕ ВОЗМОЖНЫЕ КОМБИНАЦИИ.

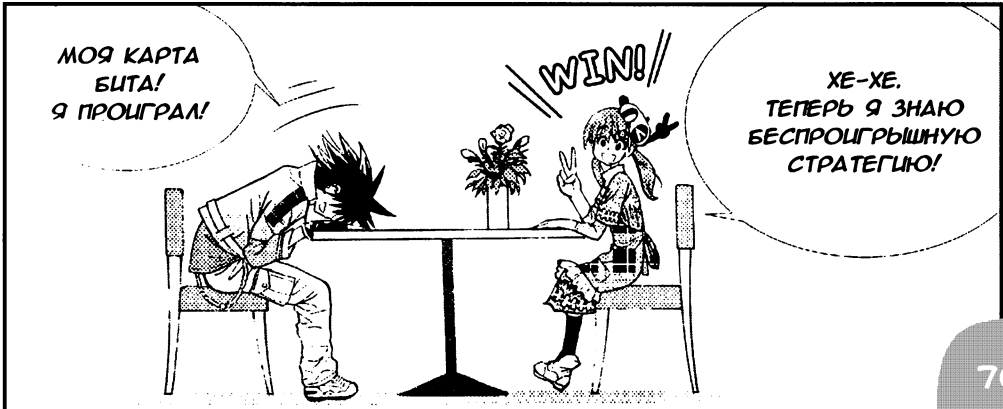
Я ПОНЯЛА! ТАБЛИЦА ГОВОРИТ ТО ЖЕ САМОЕ. ЕСЛИ ХОТЯ БЫ НА ОДНОМ ИЗ ВХОДОВ - L, ТО НА ВЫХОДЕ - H.



ТАК... ВХОДЫ БЕЗ КРУЖКОВ - АКТИВНЫЕ ВЫСОКИЕ, ВЫХОД С КРУЖКОМ - АКТИВНЫЙ НИЗКИЙ, ТИП ВЕНТИЛЯ - ALL.



ЗНАЧИТ, ЕСЛИ НА ВСЕХ ВХОДАХ - H, ТО НА ВЫХОДЕ БУДЕТ L!



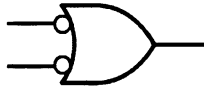
МОЯ КАРТА БЫЛА! Я ПРОИГРАЛ!

ХЕ-ХЕ. ТЕПЕРЬ Я ЗНАЮ БЕСПРОИГРЫШНУЮ СТРАТЕГИЮ!



Законы де Моргана

А СЕЙЧАС
Я РАССКАЖУ ТЕБЕ
КОЕ-ЧТО
ИНТЕРЕСНОЕ.



* Это элемент NAND, о котором
будет рассказано на стр. 76.

ДЕЛО В ТОМ, ЧТО ЭТИ
АВА ЭЛЕМЕНТА НА САМОМ
ДЕЛЕ - ОАНО И ТО ЖЕ.

ОАНО И ТО ЖЕ?
НО ОНИ ВЕДЬ
ТАКЕ РАЗНЫЕ.

МОЖНО ДАЖЕ
СКАЗАТЬ -
ВЗАИМНО
ПРОТИВОПОЛОЖНЫЕ!

Входы

Выход

Вентиль

Активный L	Активный H
Активный H	Активный L
Exist	All



В ТОМ-ТО И ДЕЛО!
ВЕДЬ В ЦИФРОВЫХ
СХЕМАХ НЕТ НИЧЕГО,
КРОМЕ L И H...

...ПОЭТОМУ, ОБРАТИВ
ВСЕ ВХОДЫ, ВЫХОД
И ТИП ВЕНТИЛЯ,
МЫ ПОЛУЧИМ
ТО ЖЕ САМОЕ!



И ЭТО НАЗЫВАЕТСЯ
ЗАКОНАМИ ДЕ МОРГАНА!!

* Иногда их называют
теоремами де Моргана.

ЭТО ЛЕГКО ПОКАЗАТЬ
С ПОМОЩЬЮ
ТАБЛИЦЫ
ИСТИННОСТИ!

В САМОМ ДЕЛЕ,
ТАБЛИЦА - ТА ЖЕ
САМАЯ!

B	A	Z
L	L	H
L	H	H
H	L	H
H	H	L

Если хотя бы на
одном входе — L,
то на выходе — H.

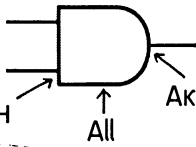
Если на обоих
входах — H,
то на выходе — L.

ОТЛИЧАЮТСЯ ТОЛЬКО СТРОКИ,
НА КОТОРЫЕ МЫ ОБРАЩАЕМ ВНИМАНИЕ!

НО МНЕ ВСЁ
РАВНО КАК-ТО
ТРУДНО В ЭТО
ПОВЕРИТЬ...

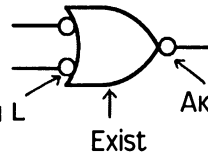
ХОРОШО,
ПРИВЕДУ
ПРОСТОЙ
ПРИМЕР.

Если на обоих входах А и В
уровни Н, то на выходе
тоже будет Н



РАССМОТРИМ ДВА
ВЗАИМНО
ПРОТИВОПОЛОЖНЫХ
ЭЛЕМЕНТА.

Если хотя бы на одном из входов А и В
уровень L, то на выходе
тоже будет L



ВСЁ НАОБОРОТ!

ПРЕДСТАВЬ,
ЧТО ТЫ ХОЧЕШЬ
СТАТЬ ЗВЕЗДОЙ
ЭСТРАДЫ,
КОТОРАЯ ДОЛЖНА
УМЕТЬ ПЕТЬ
И ТАНЦЕВАТЬ.



ЧТО?!!



НА КАСТИНГЕ ТЕБЕ НУЖНО
ПРОЙТИ ДВА ИСПЫТАНИЯ: ...

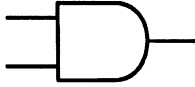


...СПЕТЬ (А)
И СТАНЦЕВАТЬ (В).

НАЗНАЧИВ РЕЗУЛЬТАТАМ -
"ПРОЙДЕНО" ИЛИ
"НЕ ПРОЙДЕНО",
УРОВНИ L И Н -
МЫ МОЖЕМ ПОСТРОИТЬ
ДВА ВАРИАНТА
СХЕМЫ ПРИНЯТИЯ РЕШЕНИЯ.



Если пройдены оба испытания — пение и танец, то результат — принятие.



Если не пройдено хотя бы одно испытание — пение или танец, то результат — отказ.



АГА...
В САМОМ ДЕЛЕ,
ЭТИ ЭЛЕМЕНТЫ,
ПО СУТИ,
ОДИНАКОВЫ!



...МЫ ПОЛУЧИМ
ВОТ ЧТО.

СТАЛО УЖЕ
ПОНЯТНЕЕ.



НЕ ТАК ЛИ?
ЭТО И ЕСТЬ
ЗАКОНЫ ДЕ
МОРГАНА.

НО ТОГАА
НЕ ПОНЯТНО...

ЕСЛИ ЭЛЕМЕНТЫ
СУТЬ ОАНО И ТО ЖЕ,
ЗАЧЕМ НУЖНЫ
РАЗНЫЕ ОБОЗНАЧЕНИЯ?



НЕТ, ТУТ ТЫ
НЕ ПРАВА.
ВОТ МОЖЕШЬ ЛИ ТЫ
СКАЗАТЬ, ЧТО ЭТИ ДВА
ВЫСКАЗЫВАНИЯ
СОВЕРШЕННО
ОДИНАКОВЫ?



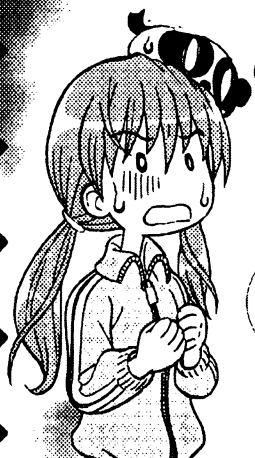
Если пройдены оба испытания — пение и танец, то результат — принятие.
Если не пройдено хотя бы одно испытание — пение или танец, то результат — отказ.

НУ, ХОТЯ СУТЬ
ОАНА И ТА ЖЕ,
ДУШЕВНЫЙ НАСТРОЙ
БУДЕТ ДРУГИМ.
ПЕРВОЕ ЗВУЧИТ КАК-ТО
ОПТИМИСТИЧНЕЕ...

ПРОЙДУ ОБА
ИСПЫТАНИЯ —
И МЕНЯ
ПРИМУТ!



ЗАБАВЮ ОАНО
ИСПЫТАНИЕ —
И МНЕ ОТКАЗУТ...



ВЕРНО!

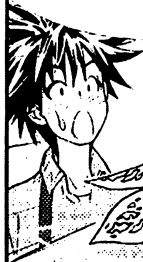
ТАКИМ ОБРАЗОМ,
ДАЖЕ ЕСЛИ СУТЬ
ОАНА, РАССТАВИТЬ
АКЦЕНТЫ МОЖНО
ПО-РАЗНОМУ.



ясно!
они вкладывают
в эти простые
значки свои
чувства.

ПРОЕКТИРОВЩИКИ
СХЕМ ТОЖЕ
ХОТЯТ ТОЧНО
ПЕРЕДАВАТЬ
СВОИ ЦАЕИ.

ПОЭТОМУ ОНИ
И ВЫБИРАЮТ
ИЗ ОДНО
ИЗ ДВУХ
ОБОЗНАЧЕНИЙ,
ПО СУТИ
ТОЖДЕСТВЕННЫХ.



ПЕРЕДАТЬ ЗНАКОМ ХОЧУ
ЧУВСТВО, О КОТОРОМ МОЛЧУ!
★
ЧТО-ТО ВРОДЕ ЭТОГО?

ОГО!
ТЫ ПРЯМО КАК
ЗВЕЗДА!

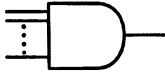



* На самом деле, цель проектирования цифровых схем — построение оптимальной структуры полупроводникового кристалла микросхемы. При этом надо уменьшать количество р-п переходов и для этого упрощать схему, заменяя одни вентили на другие в эквивалентном включении.



Базовые логические вентили в нотации MIL

ПОДВЕДЕМ ИТОГИ
В ОБЩЕМ, В НОТАЦИИ
MIL ИСПОЛЗУЮТСЯ
ВСЕГО ЧЕТЫРЕ
УСЛОВНЫХ
ОБОЗНАЧЕНИЯ.



	
All	Exist
	
Активный L	Передача сигнала дальше по линии. Логического смысла не имеет.

КОМБИНИРУЯ ИХ, СТРОЯТ
РАЗНООБРАЗНЫЕ СХЕМЫ.
РАССКАЖУ О САМЫХ
ТИПИЧНЫХ.

<Базовые вентили в нотации MIL>

Для простоты показаны случаи всего с двумя входами.

	Входы — активные Н	Входы — активные L															
<p>AND</p> <table border="1"> <tr><td>B</td><td>A</td><td>Z</td></tr> <tr><td>L</td><td>L</td><td>L</td></tr> <tr><td>L</td><td>H</td><td>L</td></tr> <tr><td>H</td><td>L</td><td>L</td></tr> <tr><td>H</td><td>H</td><td>H</td></tr> </table>	B	A	Z	L	L	L	L	H	L	H	L	L	H	H	H	<p>Эту таблицу истинности можно трактовать двумя способами.</p> <p>(i) Если оба входа A и B — H, то выход тоже H</p> <p>Входы: активный Н Выход: активный Н Тип вентиля: All</p>	<p>(ii) Если один из входов A или B — L, то выход тоже L</p> <p>Входы: активный L Выход: активный L Тип вентиля: Exist</p>
B	A	Z															
L	L	L															
L	H	L															
H	L	L															
H	H	H															
<p>OR</p> <table border="1"> <tr><td>B</td><td>A</td><td>Z</td></tr> <tr><td>L</td><td>L</td><td>L</td></tr> <tr><td>L</td><td>H</td><td>H</td></tr> <tr><td>H</td><td>L</td><td>H</td></tr> <tr><td>H</td><td>H</td><td>H</td></tr> </table>	B	A	Z	L	L	L	L	H	H	H	L	H	H	H	H	<p>Как и в случае AND, можно трактовать двумя способами.</p> <p>(i) Если один из входов A или B — H, то выход тоже H</p> <p>Входы: активный Н Выход: активный Н Тип вентиля: Exist</p>	<p>(ii) Если оба входа A и B — L, то выход тоже L</p> <p>Входы: активный L Выход: активный L Тип вентиля: All</p>
B	A	Z															
L	L	L															
L	H	H															
H	L	H															
H	H	H															
<p>NOT</p> <table border="1"> <tr><td>A</td><td>Z</td></tr> <tr><td>L</td><td>H</td></tr> <tr><td>H</td><td>L</td></tr> </table>	A	Z	L	H	H	L	<p>(i) Если вход — H, то выход — L</p>	<p>(ii) Если вход — L, то выход — H</p>									
A	Z																
L	H																
H	L																
<p>NAND</p> <table border="1"> <tr><td>B</td><td>A</td><td>Z</td></tr> <tr><td>L</td><td>L</td><td>H</td></tr> <tr><td>L</td><td>H</td><td>H</td></tr> <tr><td>H</td><td>L</td><td>H</td></tr> <tr><td>H</td><td>H</td><td>L</td></tr> </table>	B	A	Z	L	L	H	L	H	H	H	L	H	H	H	L	<p>(i) Если оба входа A и B — H, то выход — L</p> <p>Входы: активный Н Выход: активный L Тип вентиля: All</p>	<p>(ii) Если один из входов A или B — L, то выход — H</p> <p>Входы: активный L Выход: активный H Тип вентиля: Exist</p>
B	A	Z															
L	L	H															
L	H	H															
H	L	H															
H	H	L															
<p>NOR</p> <table border="1"> <tr><td>B</td><td>A</td><td>Z</td></tr> <tr><td>L</td><td>L</td><td>H</td></tr> <tr><td>L</td><td>H</td><td>L</td></tr> <tr><td>H</td><td>L</td><td>L</td></tr> <tr><td>H</td><td>H</td><td>L</td></tr> </table>	B	A	Z	L	L	H	L	H	L	H	L	L	H	H	L	<p>(i) Если один из входов A или B — H, то выход — L</p> <p>Входы: активный Н Выход: активный L Тип вентиля: Exist</p>	<p>(ii) Если оба входа A и B — L, то выход — H</p> <p>Входы: активный L Выход: активный H Тип вентиля: All</p>
B	A	Z															
L	L	H															
L	H	L															
H	L	L															
H	H	L															

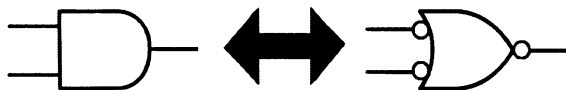
ОГО!

ЕСТЬ И ТАКИЕ ЭЛЕМЕНТЫ,
КАК NAND, NOR.
НО ЕСЛИ ЗНАТЬ НОТАЦИЮ MIP,
ТО ИХ СМЫСЛ СТАНЕТ ЯСЕН.

ЗАМЕТЬ ЕЩЁ,
ЧТО ПРАВАЯ И ЛЕВАЯ
КОЛОНКИ ТАБЛИЦЫ
СВЯЗАНЫ МЕЖАУ СОБОЙ
ЗАКОНАМИ ДЕ МОРГАНА.

ЯСНО!
ПРОСТО ВСЁ ПЕРЕВЕРНУЛИ
НАОБОРОТ, А СУТЬ
ОСТАЛАСЬ ПРЕЖНЯЯ!

И то, и другое — элемент AND



ШАПКА ОСТАНЕТСЯ ШАПКОЙ,
ДАЖЕ ЕСЛИ ПОМЕНЯТЬ МЕСТАМИ
ЧЁРНЫЙ И БЕЛЫЙ.

НИЧЕГО?

УГУ.
МОЖНО СКАЗАТЬ,
ЧТО, И ЛЕВЫЙ, И ПРАВЫЙ
ЭЛЕМЕНТЫ - ЭТО AND.

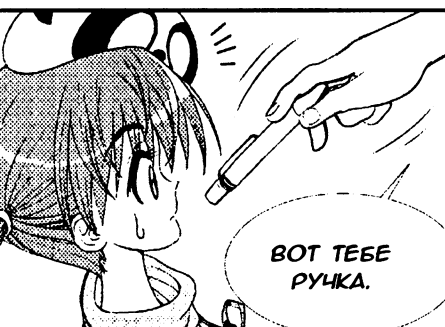
МОЖЕТ БЫТЬ,
СЛИШКОМ МНОГО
ДЛЯ ОДНОГО
ЗАНЯТИЯ?
ВОПРОСЫ ЕСТЬ?

ОДНАКО..
ЗА ЭТО ВРЕМЯ
НЕ ПРИШЛО НИ ОДНОГО
ПОСЕТИТЕЛЯ. С ЭТИМ
МАГАЗИНОМ ВСЁ
В ПОРЯДКЕ?

НУ...
ПРОСТО СЕЗОН
ТАКОЙ,
КОГДА МОЖНО
РАССЛАБИТЬСЯ.

А
ШЛЕП

2. СТРОИМ МАЖОРИТАРНУЮ СХЕМУ



■ Строим схему по таблице истинности (порядок проектирования ДНФ*)

Шаг 1

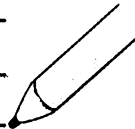
Подчёркиваем строки с Н на выходе и назначаем каждой из них по одному вентилю AND.



Итак, начнём. Мы назначили уровень L японской кухне и уровень H европейской, помнишь?

Сосредоточившись на вопросе «Будет ли выбрана европейская кухня (H)?», мы подчёркиваем строки (комбинации) таблицы истинности с результатом H на выходе.

C	B	A	Z
L	L	L	L
L	L	H	L
L	H	L	L
L	H	H	H ①
H	L	L	L
H	L	H	H ②
H	H	L	H ③
H	H	H	H ④

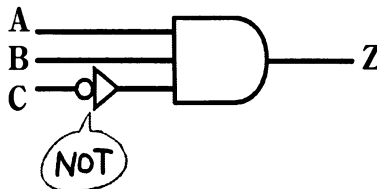


Готово! Уровень H на выходе — в четырёх строках: ①, ②, ③, ④.



Угу. Сначала рассмотрим строку ①: два человека (A и B) проголосовали за европейскую кухню (H), а один (C) — за японскую (L).

С помощью вентиля типа All эту строку можно выразить вот так.

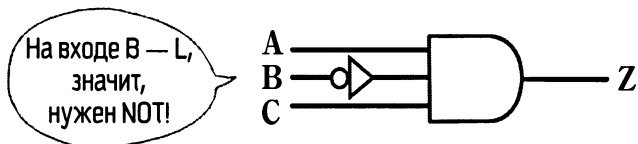


Ого! Мы вставляем элемент NOT только в линию C, передающую входной сигнал уровня L?

* Дизъюнктивная нормальная форма — это формула в алгебре логики, которая представляет собой дизъюнкцию (логическое сложение, элемент ИЛИ) от логических произведений (элемент И), логических переменных и их дизъюнкций. Можно доказать, что любое логическое выражение может быть приведено к виду ДНФ.



Но если так, то для строки ② схема будет вот такой? Это когда А и С проголосовали за европейскую кухню (Н), а В — за японскую (L)...



Правильно! Начертив таким образом элементы для строк ①—④, мы выразим в схематической форме все четыре комбинации, дающие результат «европейская кухня» (Н).



Ясно... Но ведь это — всего лишь четыре отдельных, не связанных друг с другом элемента. Как нам теперь объединить их в одну схему?



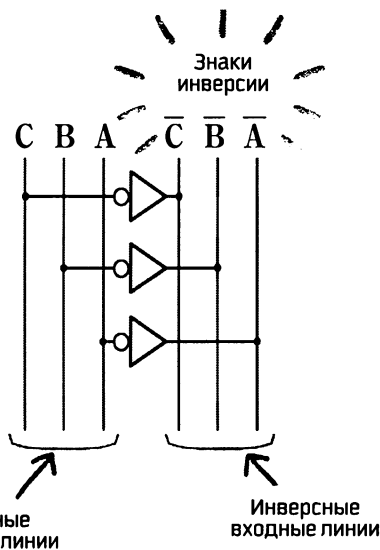
Хороший вопрос. Чтобы объединить все случаи, потребуется одна хитрость: нам надо будет построить сигнальную магистраль — «шину».

Шаг 2

Строим шину, состоящую из всех обычных и инверсных входных линий.



До этого мы вставляли элементы NOT во входные сигнальные линии, на которых в данной комбинации был уровень L. Теперь мы построили общую «шину», которая свяжет три входа схемы с четырьмя вентилями AND, назначенными четырём строкам (комбинациям) таблицы истинности.



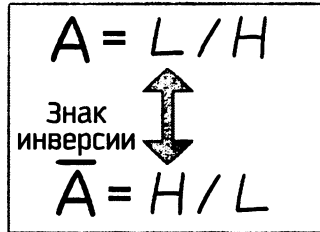


Вот как? Когда есть такая «шина», и всю схему начертить несложно. А что такое «инверсия» и «знак инверсии»?



Инверсия — это замена на противоположность, то есть операция отрицания — NOT. Инверсия A записывается как \bar{A} .

Обычные и инверсные сигнальные линии связаны между собой вот так.



Ясно! Если A проголосовал за японскую кухню (L), то мы используем инверсную входную линию \bar{A} , на которой в этом случае будет уровень H .

Шаг 3

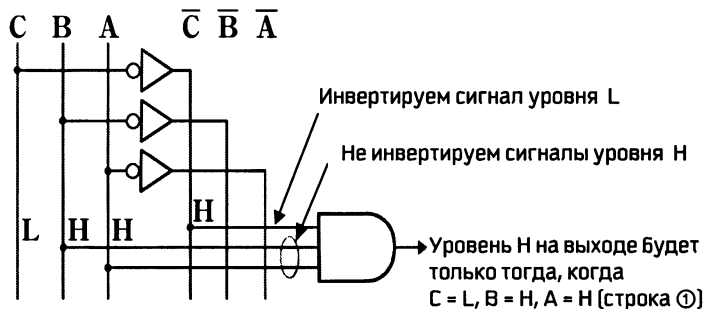
Подключаем те входы схемы, на которых в данной комбинации присутствует уровень H , к входам вентиля AND, назначенного данной комбинации, через обычные линии шины, а те входы схемы, на которых в данной комбинации присутствует уровень L , — через инверсные. Прodelываем вышеописанное для всех подчеркнутых строк таблицы истинности.



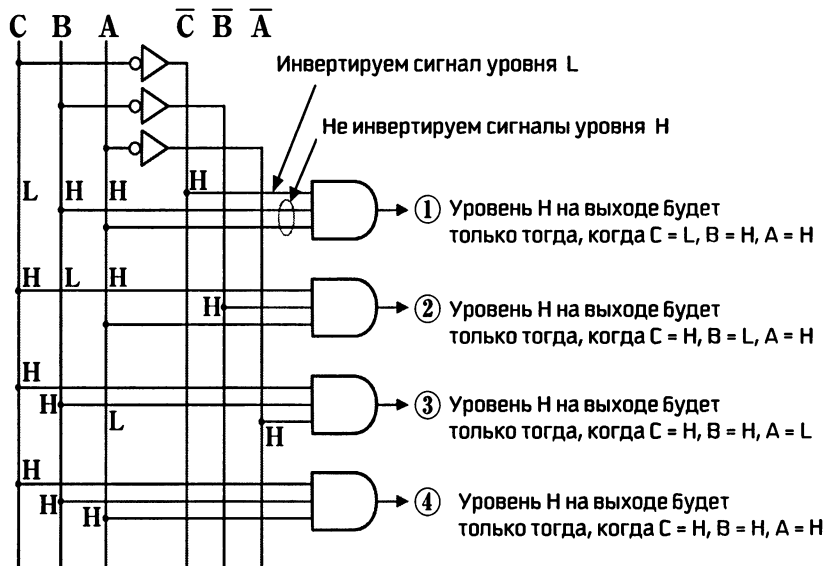
Теперь мы подключаем 3 входы схемы к входам 4 вентилях AND через шину. Входы схемы, на которых в соответствующей комбинации присутствует уровень H , подключаем через обычные входные линии, а те входы схемы, на которых в данной комбинации присутствует уровень L , — через инверсные. Прodelываем это для всех подчеркнутых строк.



Так... Например, в строке ① A и B проголосовали за европейскую кухню (H), а C — за японскую (L)... Значит, мы подключаем к входам вентиля AND входные линии A , B и \bar{C} ?



Если прodelать это для всех строк ①—④, то получится вот так!



Шаг 4

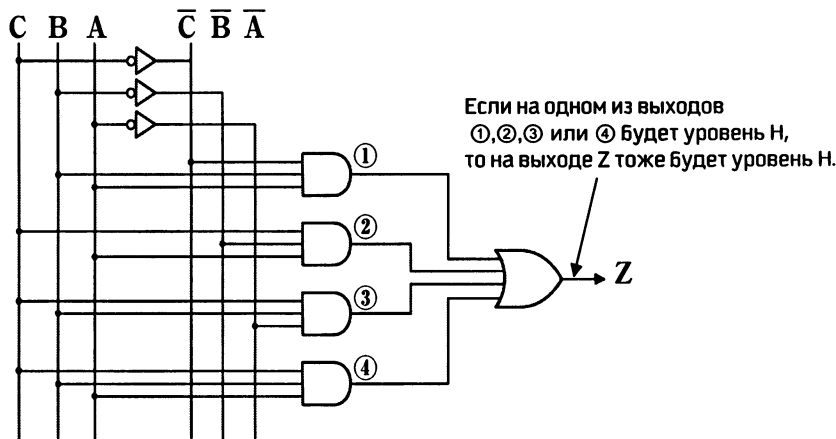
Подключаем выходы всех вентилях AND к входам вентиля OR. Готово!



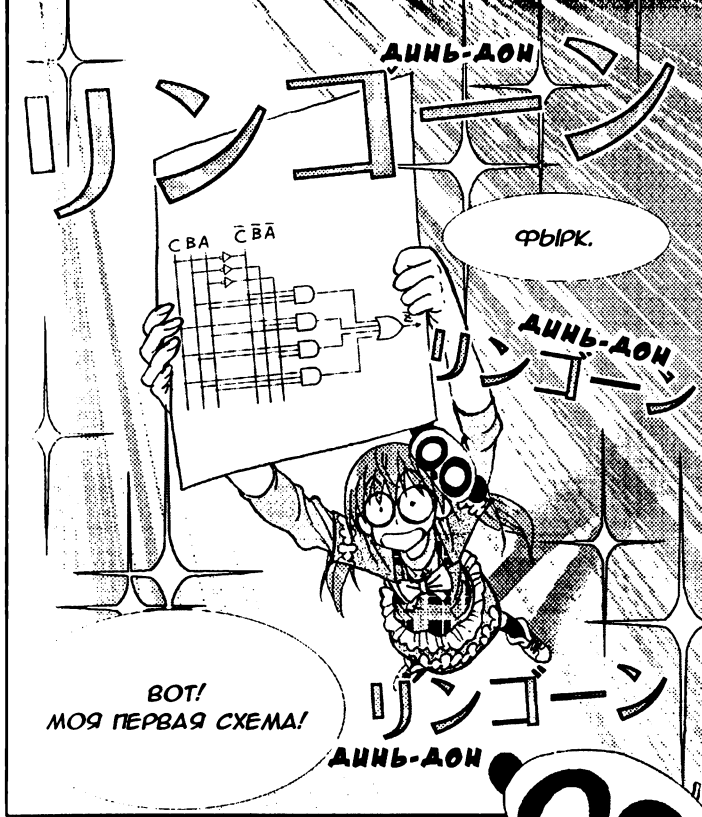
Теперь осталась сущая мелочь! У нас есть четыре вентиля AND, и если на выходе одного из них будет уровень N, то окончательным результатом должна быть «европейская кухня» (N). Для этого достаточно подключить все четыре выхода вентилях AND к входу вентиля OR.



Значит, получится вот так? Всё сделано!



Вот ты и начертила схему! Молодец! Мы использовали метод под названием «проектирование дизъюнктивной нормальной формы (ДНФ)», в котором схема получается напрямую из таблицы истинности. С его помощью ты сможешь построить любую комбинационную схему!



АНЬ-АОН

РЫРК.

АНЬ-АОН

リンゴーン

АНЬ-АОН

ВОТ!
МОЯ ПЕРВАЯ СХЕМА!



БУДУ ЕЁ БЕРЕЧЬ.
ПРИДУ ДОМОЙ,
В РАМКУ ПОВЕШУ.

ИЛИ НЕТ,
НА СИНТОИСТСКИЙ
АЛТАРЬ ПОСТАВЛЮ.

БЛАГОДАРЮ!

НЕТ,
НЕ ОБЯЗАТЕЛЬНО
ТАК ДЕЛАТЬ!

ВЕДЬ ЭТО ЕЩЁ
НЕ ВСЁ.

ДАЛЬШЕ БУДЕТ
ПРОДОЛЖЕНИЕ.

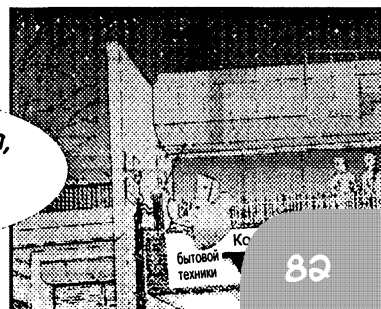


ХЕ-ХЕ

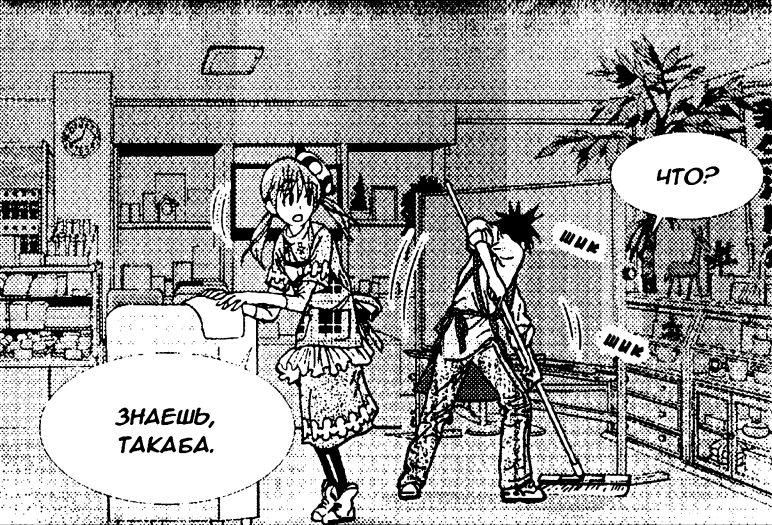
ВОТ...



НА СЕГОДНЯ,
НАВЕРНОЕ,
ХВАТИТ?



Бытовой
техники



ЗНАЕШЬ,
ТАКАБА.

ЧТО?



Я БУДУ ПРИЛАГАТЬ
УСИЛИЯ, ЧТОБЫ СТАТЬ
ВАШЕЙ ШТАТНОЙ
СОТРУДНИЦЕЙ!



ТЫ ТАК СТАРАЕШЬСЯ
ДЛЯ МЕНЯ...
Я ТОЖЕ ХОЧУ
БИТЬ ТЕБЕ ПОЛЕЗНОЙ.

ДА! Я ИСКРЕННЕ
ЖЕЛАЮ ПОМОЧЬ
ВАШЕМУ МАГАЗИНУ.

ХОЧУ РАБОТАТЬ!
НЕ БОЮСЬ ДАЖЕ
УМЕРЕТЬ ОТ
УСТАЛОСТИ!
ОКРУЖЕННАЯ КУЧЕЙ
РАЗОБРАННЫХ
ПРИБОРОВ!



НУ...
УЖЕ ВЕСНА СКОРО.
ДУМАЮ, ДИРЕКТОР
ТЕБЯ ОБЯЗАТЕЛЬНО
ПРИМЕТ.

ВЕДЬ
ВЕСНОЙ...



ЗНАЮ! ВЕСНОЙ НАЧИНАЮТСЯ
ЗАНЯТИЯ В ИНСТИТУТЕ, ФИРМЫ НАБИРАЮТ
НОВЫХ СОТРУДНИКОВ. МНОГИЕ НАЧИНАЮТ
НОВУЮ ЖИЗНЬ, ПОЭТОМУ ДЛЯ
МАГАЗИНОВ Б/У ТЕХНИКИ ЭТО ТОЖЕ
"ЖАРКИЙ СЕЗОН!"

БОЛЬШЕ
ПОКУПАТЕЛЕЙ -
УСПЕШНЕЕ
БИЗНЕС!

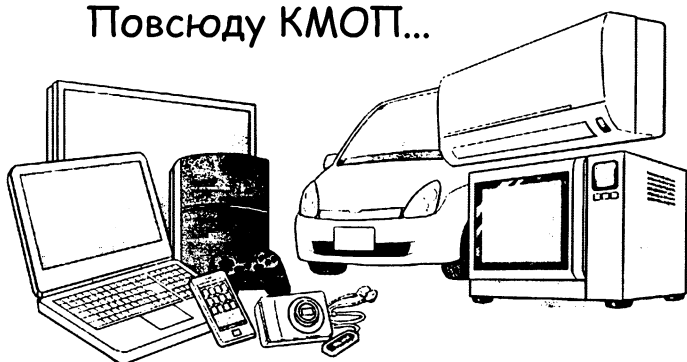
ДА,
КОНЕЧНО.

Что такое КМОП?

Цифровые микросхемы КМОП (комплиментарная структура металл-оксид-полупроводник, англ. CMOS) используются практически во всех окружающих нас электронных устройствах.

На их основе создаются не только разнообразные IT-изделия: компьютеры, игровые приставки, смартфоны, цифровые ТВ-приёмники, цифровые камеры, видеоаппаратура, принтеры, цифровые плееры и др., но и бытовая техника: кондиционеры, микроволновые печи и т. п. Современные автомобили тоже буквально напичканы цифровыми микросхемами КМОП.

Повсюду КМОП...

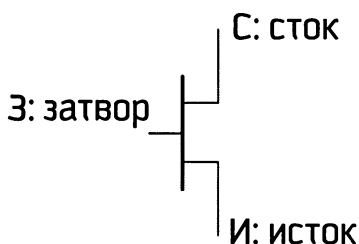


Возьмём, к примеру, плееры. Хотя в их выходных каскадах, создающих звуковой сигнал в наушниках, используются аналоговые схемы, а в схемах памяти (запоминающих элементах) — цифровые ИС, немного отличные от КМОП, но в остальных блоках, например управления воспроизведением, сжатия и сохранения, считывания и преобразования звуковых данных и т. п., всегда используются цифровые микросхемы КМОП. И крупногабаритные IT-изделия — суперкомпьютеры, серверы, производящие облачные вычисления, и т. п. — тоже основаны на КМОП.

В наши дни существуют как быстродействующие микросхемы с высоким потреблением электроэнергии и большим тепловыделением, которые используются, например, в компьютерах; так и менее быстродействующие, но зато более компактные и дешёвые ИС, отличающиеся надёжностью. И те, и другие представляют собой микросхемы КМОП. По технологии КМОП также изготавливают и один из видов фотоматриц — ПЗС-матрицу.

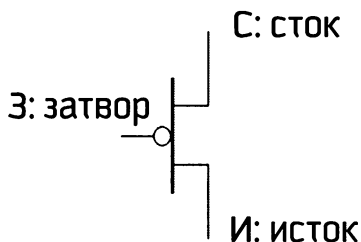


В схеме КМОП используются полевые МОП-транзисторы двух типов: n-канальные (n-МОП) и p-канальные (p-МОП). Полевые транзисторы отличаются от биполярных транзисторов, использующихся в аналоговых схемах, тем, что функционируют подобно переключателям.



n-канальный МОП-транзистор

З = Н: ON — И и С замкнуты
З = L: OFF — И и С разомкнуты



p-канальный МОП-транзистор

З = L: ON — И и С замкнуты
З = Н: OFF — И и С разомкнуты

Рис. 1. Условные обозначения и функционирование n-МОП и p-МОП

На рис. 1 показаны обозначения n-МОП и p-МОП. Они имеют 3 вывода: затвор (З), исток (И) и сток (С) — и управляются напряжением на затворе.

Если считать, что уровень источника питания — Н ("высокий"), а уровень «земли» — L ("низкий"), то n-МОП будет открыт (ON) при З = Н и закрыт (OFF) при З = L. Можно считать, что в состоянии ON исток и сток полностью замкнуты, а в состоянии OFF — полностью разомкнуты.

В случае p-МОП, наоборот, И и С будут замкнуты (ON) при З = L и разомкнуты (OFF) при З = Н.

Схемы, в которых используются оба этих противоположных типа транзисторов, n-МОП и p-МОП, называются комплементарными. Расскажу здесь об устройстве наиболее типичной схемы КМОП. Она состоит из комплементарных пар транзисторов n-МОП и p-МОП, соединённых затворами: когда один из транзисторов пары открыт (ON), другой будет закрыт (OFF). Кроме того, если транзисторы n-МОП соединены друг с другом последовательно, то транзисторы p-МОП — параллельно, и наоборот, если n-МОП соединены друг с другом параллельно, то p-МОП — последовательно.

Взгляните на рис. 2 на следующей странице. Эта схема содержит два n-МОП (Qn1 и Qn2), соединённых друг с другом последовательно, и два p-МОП (Qp1 и Qp2), соединённых друг с другом параллельно. Два входа А и В подключены к затворам пар транзисторов n-МОП и p-МОП соответственно. Сигнальная линия Z посередине схемы является выходом.

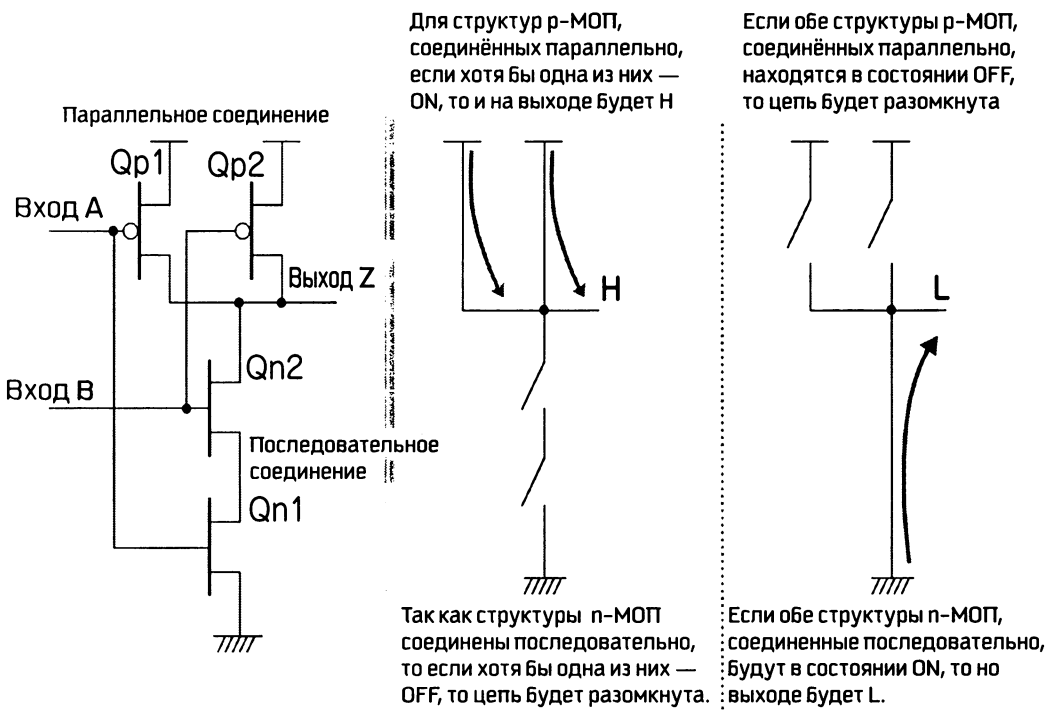


Рис. 2. Вентиль NAND на основе КМОП

Если на одном из входов А, В (или на обоих) — уровень L, то одна из структур р-МОП (или обе) откроются, соединив выход Z с источником питания вверху схемы. Кроме того, одна из последовательно соединённых структур n-МОП (или обе) закроются — цепь разомкнётся. Таким образом, на выходе будет уровень Н.

Если на обоих входах А, В — уровень Н, то оба n-МОП откроются, соединив выход Z с «землей», а оба транзистора р-МОП — закроются. Другими словами, на выходе будет уровень L. Это соответствует поведению вентиль NAND, рассмотренного в данной главе. Подбирая сочетания параллельного и последовательного соединений, можно получать логические схемы с разнообразными функциями.

Благодаря сочетанию параллельного и последовательного соединений цепь питания всегда остаётся разомкнутой, обеспечивая малый потребляемый ток. В связи с необходимостью использования двух типов транзисторов микросхемы КМОП развивались медленнее, чем р-МОП и n-МОП: быстрое развитие и распространение КМОП началось со второй половины 70-х годов благодаря полупроводниковым микротехнологиям.

Раньше степень интеграции ИС увеличивалась каждый год примерно в 1.5 раза (так называемый закон Мура), одновременно росло быстродействие, снижалось энергопотребление, но сейчас темпы развития несколько замедлились. Хотя поговаривают о том, что КМОП достигли своего предела, их развитие всё ещё продолжается.

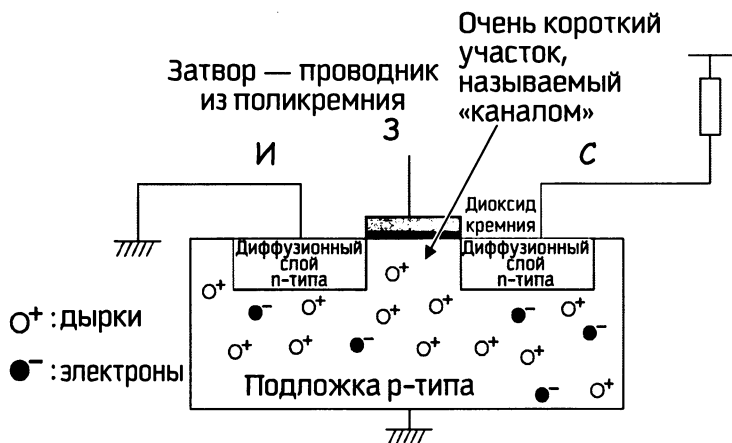
Принцип работы полевого МОП-транзистора

Буква «К» в названии КМОП означает «комплиментарная». Итак, что же такое полевой МОП-транзистор?

На рис. 3 показан n-МОП в разрезе. В нём использованы полупроводники двух типов: n и p.

Полупроводник n-типа содержит свободные электроны, заряженные отрицательно, а полупроводник p-типа — дырки, заряженные положительно. В n-МОП имеется основа (так называемая «подложка») из полупроводника p-типа. Другими словами, в ней много дырок. Её подключают к уровню «земли» (0 В). Внутри подложки формируют две отдельные области n-типа, к которым подключают выводы И (исток) и С (сток).

На миниатюрной площадке между И и С формируют проводник из поликремния (металлического кремния), который называют затвором (З). Под ним находится очень тонкая плёнка диэлектрика (диоксид кремния), которая отделяет затвор от подложки. Исток подключают непосредственно к «земле» (0 В), а сток — через резистор к источнику питания. Однако, когда потенциал затвора равен 0, разделяющая И и С подложка p-типа не позволяет току течь. Это соответствует состоянию OFF — «И и С разомкнуты», показанному на рис. 3.



- Затвор изолирован от подложки p-типа диэлектриком (диоксид кремния).
- И и С изолированы друг от друга подложкой p-типа → состояние OFF .
- В полупроводнике p-типа много дырок, но в этой подложке p-типа есть также немного свободных электронов, не рекомбинировавших с дырками. Именно они играют очень важную роль.

Рис. 3. Устройство транзистора n-МОП (состояние OFF)

Теперь взгляните на рис. 4. Если подать на затвор высокий уровень (почти равный напряжению источника питания), то произойдёт удивительная вещь.

В действительности в подложке р-типа имеется одна хитрость. В ней, как в любых полупроводниках р-типа, содержится много дырок, однако она содержит также атомы примеси, благодаря которым в ней остаётся также немного свободных электронов. Плёнка диэлектрика, отделяющая затвор от подложки, очень тонка, поэтому электроны, заряженные отрицательно, притягиваются электрическим полем и собираются под затвором.

Благодаря этому на очень коротком участке между И и С формируется инверсный слой, обладающий проводимостью n-типа. И и С замыкаются через этот слой, и между ними возникает электрический ток. Транзистор переходит в состояние ON.

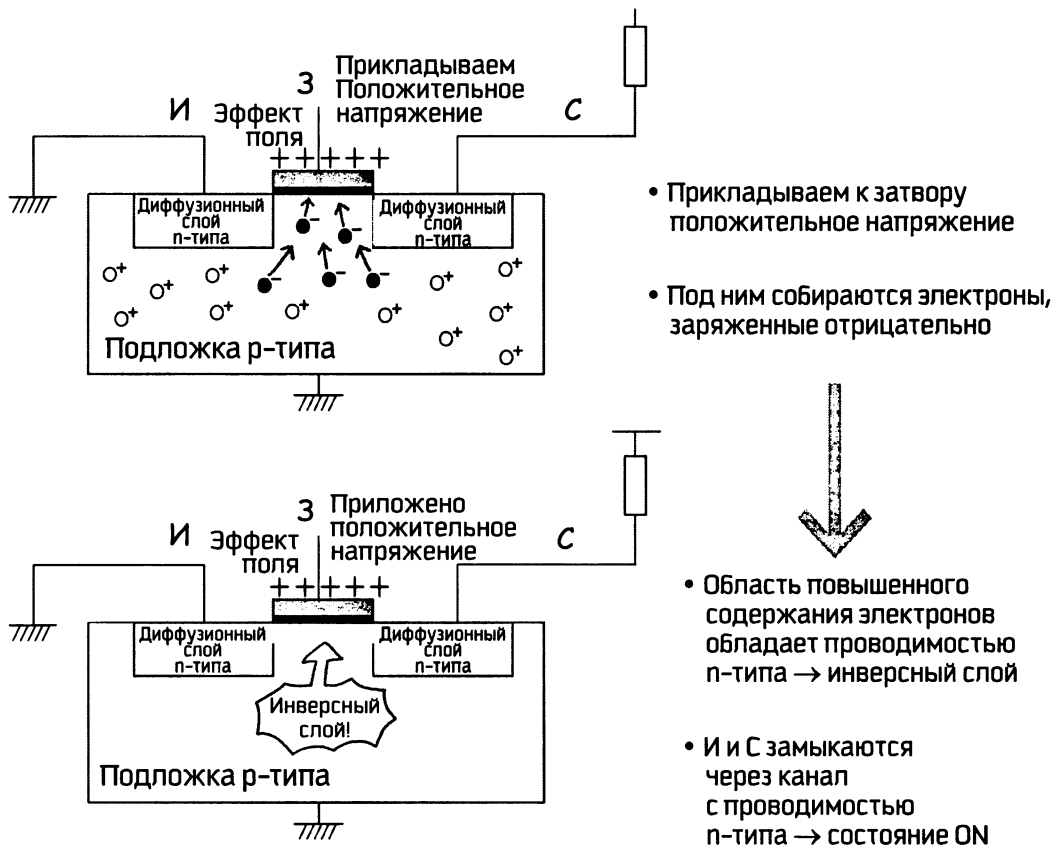


Рис. 4. Устройство транзистора n-MOП (состояние ON)



Именно потому, что И и С замыкаются и размыкаются (ON/OFF) с помощью электрического поля под затвором, этот транзистор и называется «полевым».

Кроме того, так как он состоит из поликремния с металлическими свойствами, плёнки диэлектрика — оксида кремния, а также кремния — полупроводника, его обозначают заглавными буквами МОП.

Принцип действия р-канального МОП-транзистора (р-МОП) совершенно такой же: в нём просто поменяли местами полупроводники n-типа и р-типа, поэтому он работает наоборот: открыт при напряжении на затворе 0 В (состояние ON) и закрывается при подаче на затвор напряжения питания (состояние OFF).

Отличительная особенность полевых МОП-транзисторов — улучшение характеристик при уменьшении размеров.

Во-первых, канал проводимости становится короче, расстояние между И и С уменьшается, благодаря чему повышается быстродействие. Во-вторых, снижается напряжение формирования инверсного слоя — канала проводимости, а значит, и потребляемая мощность, пропорциональная квадрату напряжения. В-третьих, миниатюризация позволяет разместить больше транзисторов в одной ИС. Таким образом, улучшается всё: характеристики, энергопотребление, степень интеграции. Это теория масштабирования, которую предложил американский инженер Роберт Деннард.

В качестве мерила прогресса в области изготовления микросхем используется минимальный размер микрообработки (размер техпроцесса). Чем он меньше, тем более совершенными становятся ИС. На момент написания этой книги в 2013 году самым передовым был техпроцесс 28 нм.

Объяснение терминов главы 3

- **Мажоритарная схема** — схема, выходной сигнал которой определяется состоянием большинства входов. В манге рассказывалось о мажоритарной схеме с тремя входами. Если подключить выходы трёх одинаковых схем к её входам, то даже при отказе одной из них окончательный результат будет верным, если остальные две работают правильно. Это называется принципом тройного модульного резервирования (TMR: Triple Modular Redundancy).
- **Таблица истинности** — таблица, содержащая все возможные комбинации состояний входов и соответствующие состояния выхода. Это основной способ описания работы комбинационных схем. Недостатком является ухудшение читабельности в том случае, если схема имеет слишком много входов.
- **Комбинационные схемы** — цифровые схемы, в которых выходные сигналы определяются только текущей комбинацией входных сигналов. Напротив, в последовательных схемах играет роль также текущее состояние схемы. В этой книге комбинационные схемы рассматриваются в главах 3 и 4, а последовательные — в главе 5.
- **Нотация MIL** — система условных обозначений для чертежей логических схем на основе стандарта MIL-STD-806 ВВС США.
- **Базовые вентили** — основные вентили, часто используемые в логических схемах. В этой книге рассматриваются вентили AND (логическое умножение), OR (логическое сложение), NOT (отрицание или инверсия), NAND и NOR. Иногда к ним добавляют также вентиль XOR (eXclusive OR: исключаящее ИЛИ).
- **Законы (теоремы) де Моргана** — законы, согласно которым при обращении активных уровней всех линий и типа вентиля (All / Exist) логика работы элемента не меняется. В виде логических выражений записываются как $A \cdot B = \overline{\overline{A} + \overline{B}}$ и $\overline{A+B} = \overline{A} \cdot \overline{B}$.
- **Дизъюнктивная нормальная форма (ДНФ)** — логическое выражение, описывающее схему, в которой входные сигналы подаются на вентили AND, выходы которых подключены к общему вентилю OR. ДНФ мажоритарной схемы, описанной в манге, записывается как $\overline{A} \cdot B \cdot C + A \cdot \overline{B} \cdot C + A \cdot B \cdot \overline{C} + A \cdot B \cdot C$.
Напротив, логическое выражение, описывающее схему, в которой входные сигналы подаются на вентили OR, выходы которых подключены к общему вентилю AND, называется конъюнктивной нормальной формой (КНФ). ДНФ и КНФ легко преобразуются одна в другую при помощи законов де Моргана.

ГЛАВА 4 КАК УПРОСТИТЬ СХЕМУ?



1. ИСПОЛЬЗУЕМ КАРТЫ КАРНО

Нерациональные схемы

※1 Ресторан «В пятнадцатом поколении», еда и напитки.
 ※2 Только в марте! Меню «Весенняя сакура».



ЦУМО,
 ОТ ИМЕНИ НАШЕЙ КОМПАНИИ
 ПОЗДРАВЛЯЮ ТЕБЯ
 С ОФИЦИАЛЬНЫМ
 ПРИЁМОМ НА РАБОТУ!

ДАВАЙТЕ ВЫПЬЕМ
 ЗА ПРОЦЕТАЮЩИЙ
 БИЗНЕС!

ОДНАКО...
 ЭТОТ РЕСТОРАН НАШ
 ДИРЕКТОР ВЫБРАЛА
 ЕДИНОЛИЧНО...

А КАК ЖЕ...
 МАЖОРИТАРНАЯ
 СХЕМА?

БОЛЬШОЕ
 СПАСИБО!

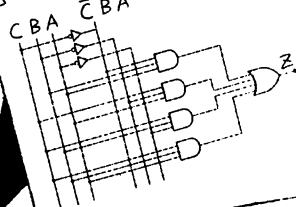
СЕГОДНЯ
 МНЕ ХОЧЕТСЯ
 ЯПОНСКОГО САКЭ!

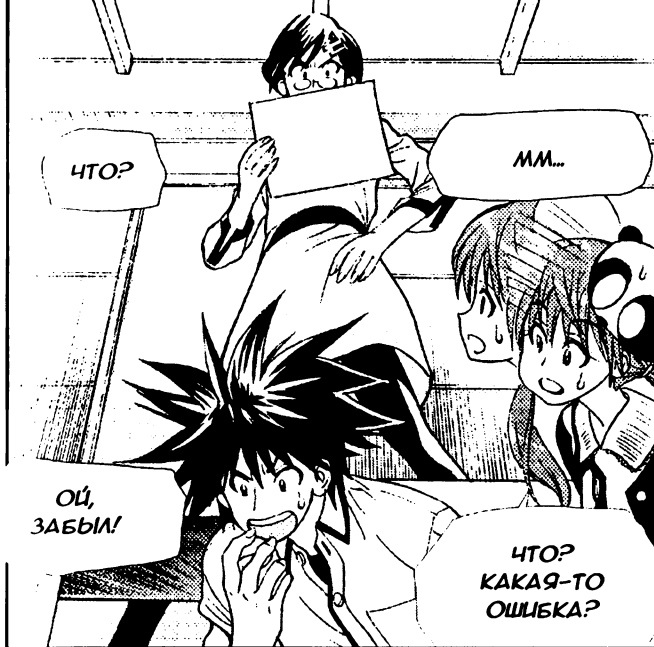
ПОЭТОМУ МЫ
 ПОЙДЕМ В РЕСТОРАН
 ЯПОНСКОЙ КУХНИ!

ДА ЛАДНО ТЕБЕ.
 НУ ЧТО, ИЗУЧИЛИ
 МАЖОРИТАРНЫЕ
 ЦИФРОВЫЕ СХЕМЫ?

ДА! ТАКАБА МНЕ
 ВСЁ ОЧЕНЬ ХОРОШО
 ОБЪЯСНИЛ!

СВА ЦВА (См. стр. 82.)





ЧТО?

ММ...

ОЙ,
ЗАБЫЛ!

ЧТО?
КАКАЯ-ТО
ОШИБКА?



НЕТ, СХЕМА
ПРАВИЛЬНАЯ,
НО В НЕЙ МНОГО
ЛИШНЕГО.

НА САМОМ ДЕЛЕ
ОНА ДОЛЖНА
БЫТЬ ПРОЩЕ.



КАК?
НЕ МОЖЕТ БЫТЬ!

А Я ЕЁ
КАК ТАЛИСМАН
С СОБОЙ НОСИЛА...

АХ, ПРОСТИ МЕНЯ!

НО ТЕМ МЕТОДОМ,
КОТОРОМУ Я ТЕБЯ
НАУЧИЛ, ПРОЩЕ
СДЕЛАТЬ НЕЛЬЗЯ.



ЛАДНО, ДАВАЙТЕ
УСПОКОИМСЯ И
ПОРАЗМЫШЛЯЕМ.
ПРЕДСТАВЬТЕ, ЧТО МЫ
ВТРОЁМ ГОЛОСУЕМ,
КАКУЮ КУХню
ВЫБРАТЬ:
ЯПОНСКУЮ ИЛИ
ЕВРОПЕЙСКУЮ.



ЕСЛИ МЫ ВДВОЁМ
С ЦУМО ПРОГОЛОСУЕМ
ЗА ЕВРОПЕЙСКУЮ,
ТО МНЕНИЕ ТАКАБИ
БУДЕТ УЖЕ НЕ ВАЖНО,
НЕ ТАК ЛИ?

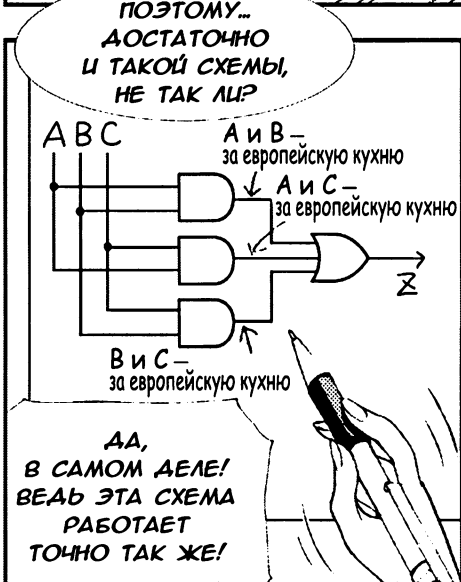
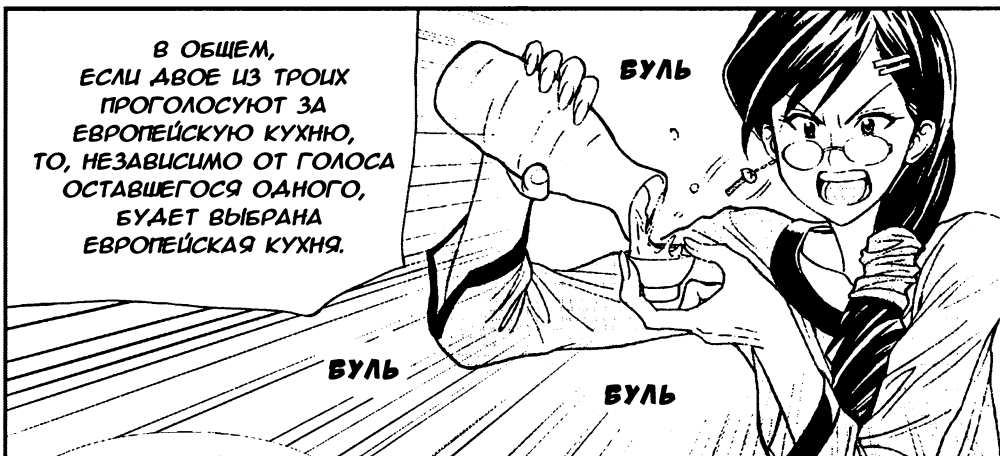


Мажоритарная
цифровая схема



АВУМЪ ГОЛОСАМИ....

...ВЫБРАН РЕСТОРАН
ЕВРОПЕЙСКОЙ
КУХНИ



СУЩЕСТВУЮТ НАДЕЖНЫЕ
МЕТОДЫ УПРОЩЕНИЯ СХЕМ.

ИЗУЧИВ ИХ, КТО УГОДНО
СМОЖЕТ ПРОЕКТИРОВАТЬ
ПРОСТЫЕ СХЕМЫ.

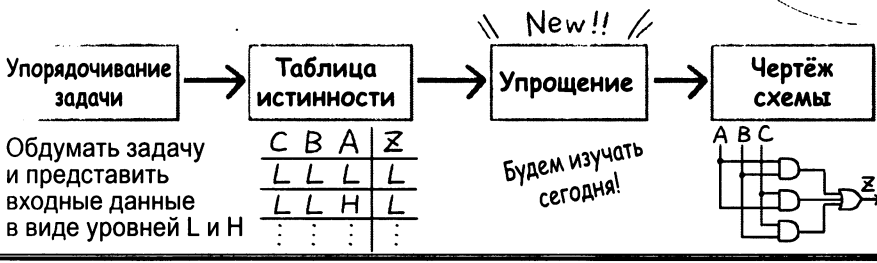


ВОТ, СМОТРИ!
ЭТО - ПРОЦЕСС
ПРОЕКТИРОВАНИЯ
ЦИФРОВЫХ СХЕМ.

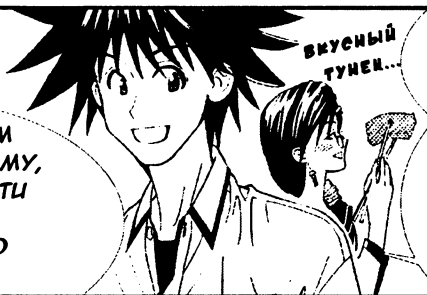


ОГО!
СРАЗУ ПОСЛЕ
"ТАБЛИЦЫ ИСТИННОСТИ"
ЦАЁТ "УПРОЩЕНИЕ!"

«Порядок проектирования цифровых схем»



АА! ПЕРЕД ТЕМ
КАК ЧЕРТИТЬ СХЕМУ,
НУЖНО ПРОВЕСТИ
УПРОЩЕНИЕ.
ПРОСТО Я ЭТО
ПРОПУСТИЛ.



МЕТОДОВ УПРОЩЕНИЯ
СУЩЕСТВУЕТ
МНОЖЕСТВО...

НО ОСОБЕННО
ПОНЯТНЫЙ
И УДОБНЫЙ -
ЭТО "КАРТЫ КАРНО".

ОХ... ДИРЕКТОР, ТЫ ХИТРОВАНКА!
ЭТО САМОЕ ВКУСНОЕ!



ФИЛЕ С
ТРИУФЕЛЯМИ...



КАРТЫ КАРНО?

Исследуем карту Карно

КАРТЫ КАРНО?
А ЧТО ЭТО ТАКОЕ?

ОЧЕНЬ ПРОСТО.
ЭТО ПРЕДСТАВЛЕНИЕ
ТАБЛИЦЫ ИСТИННОСТИ
В ВИДЕ
ДВУМЕРНЫХ ТАБЛИЦ.

ДВУМЕРНЫХ
ТАБЛИЦ?
ЧАС ОТ ЧАСУ
НЕ ЛЕГЧЕ...
ИСПЫТАТЕЛЬНЫЙ
СРОК
ЗАКОНЧИЛСЯ -
И НИКАКИХ
ПОБЛАЖЕК?

СМОТРИ, ВОТ ТАБЛИЦА.
У НЕЁ ЕСТЬ
ГОРИЗОНТАЛЬНАЯ ОСЬ -
ЗАГОЛОВКИ СТОЛБЦОВ,
И ВЕРТИКАЛЬНАЯ ОСЬ -
ЗАГОЛОВКИ СТРОК,
НЕ ТАК ЛИ?

Товары / Месяцы	Телевизоры	Холодильники
Январь	6 шт.	4 шт.
Февраль	8 шт.	5 шт.

ПОЭТОМУ И ГОВОРЯТ,
ЧТО ОНА "ДВУМЕРНАЯ".

ЭТО ПРОДАЖИ НАШЕГО МАГАЗИНА. ♪

ТОГДА ЭТО ПРОСТО!
ТО ЕСТЬ
В КАРТАХ КАРНО
ТОЖЕ ЕСТЬ
ВЕРТИКАЛЬ
И ГОРИЗОНТАЛЬ?

ДА! ДЛЯ ТРЁХ ВХОДОВ
(А, В И С),
БУДЕТ ДВА ВАРИАНТА
КАРТЫ КАРНО:
ШИРОКАЯ И ВЫСОКАЯ.

Карта Карно для трёх входов
(широкая)

		Вход В	Вход А
с \ ВА	00	01	11
0			1
1			

Если А = 1, В = 1,
С = 0, то на выходе
будет 1.

Карта Карно для трёх входов
(высокая)

	с \ В	0	1
0	00		
1	01		1
	11		
	10		

Если А = 1,
В = 1, С = 0,
то на выходе
будет 1.

В КАРТАХ КАРНО ЧАСТО
ДЛЯ НАГЛЯДНОСТИ
ИСПОЛЬЗУЮТ 0 И 1
ВМЕСТО L И H.

ЗАГОЛОВКИ
СТОЛБЦОВ И СТРОК
ТАБЛИЦЫ - ЭТО ВХОДЫ,
А ЕЁ СОДЕРЖАНИЕ -
ЭТО ВЫХОД, АА?

ВЕРНО! НАПРИМЕР,
ЕСЛИ ВЫХОДОВ ЧЕТЫРЕ
(A, B, C И D), ТО
ТАБЛИЦА БУДЕТ
ВОТ ТАКОУ.

Карта Карно для четырёх входов

		Вход В		Вход А	
	ВA	00	01	11	10
DC	00				
	01			1	
	11				
	10				
		Вход D	Вход C		

Если
A = B = C = 1
и D = 0, то
на выходе
будет 1.

ЯСНО. ТЕПЕРЬ Я ЗНАЮ,
КАК ЕЁ СМОТРЕТЬ, И
ДАЖЕ СМОГУ САМА СОСТАВИТЬ.

НУ, ВОТ
ТАК КАК-ТО ...

C	B	A	Z
L	L	L	L
L	L	H	L
L	H	L	L
L	H	H	H
H	L	L	L
H	L	H	H
H	H	L	H
H	H	H	H

(См. стр. 79.)

	ВA	00	01	11	10
C	0	0	0	1	0
	1	0	1	1	1

* Вместо H использовано 1,
а вместо L — 0.

АА! СОВЕРШЕННО
ВЕРНО!

ИЗВИНИТЕ, ЧТО
ЗАСТАВИЛА
ВАШ ЖААТЬ!!

стук

ВАШ ЗАКАЗ!

А, КАК
РАЗ ВОВРЕМЯ!
ВЗГЛЯНИТЕ, ЭТО -
РАЗВЕРНУТАЯ
СКУМЕРИЯ!

КОГДА-ТО
ЛЕВЫЙ И
ПРАВЫЙ БОКА
БЫЛИ СОЕДИНЕННЫ.

ФИНИИ!!

ОТ ТАКИХ ЖЕСТОКИХ
АССОЦИАЦИЙ У МЕНЯ
АППЕТИТ ПРОПАДАЁТ!!!

КРУТЬ

АА, УЖ ЛУЧШЕ ПРЕДСТАВИТЬ
СВЕРНУТЫЙ ЛИСТ БУМАГИ.

Края
соединены!

НЕТ, НА САМОМ ДЕЛЕ
КАРТА КАРНО ПОХОЖА
НА РАЗВЕРНУТУЮ
СКУМЕРИЮ.
ВЕДЬ У НЕЁ ТОЖЕ
СМЫКАЮТСЯ
ЛЕВЫЙ И ПРАВЫЙ,
ВЕРХНИЙ И НИЖНИЙ
КРАЯ.

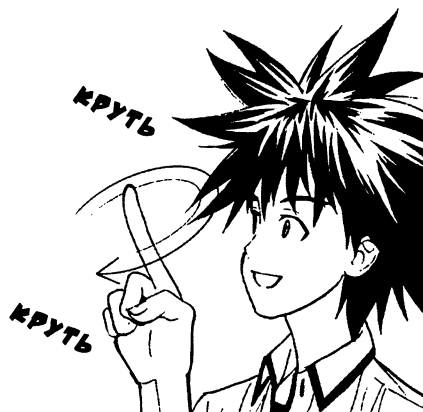
КРУТЬ
КРУТЬ

ПРАВДА?

ЭТО ПОТОМУ, ЧТО В КАРТЕ КАРНО СОСЕДНИЕ ЗАГОЛОВКИ И СТРОК, И СТОЛБЦОВ ВСЕГДА ОТЛИЧАЮТСЯ ДРУГ ОТ ДРУГА ТОЛЬКО ОДИМ РАЗРЯДОМ. ПОСЛЕДНИЙ И ПЕРВЫЙ ЗАГОЛОВКИ ТОЖЕ ОТЛИЧАЮТСЯ ТОЛЬКО ОДИМ РАЗРЯДОМ, ПОЭТОМУ ТАБЛИЦУ МОЖНО "СВЕРНУТЬ В ТРУБКУ", СОМКНУВ ЕЁ ЛЕВЫЙ И ПРАВЫЙ (ИЛИ ВЕРХНИЙ И НИЖНИЙ) КРАЯ.



От ① до ④: инверсия одного разряда повторяется по кругу!



В ЦЕЛОМ ПОНЯТНО...
НО НЕУЖЕЛИ ТАК ВАЖНО,
ЧТОБЫ ТАБЛИЦА
"СВОРАЧИВАЛАСЬ
В ТРУБКУ"?

ДА!
ОЧЕНЬ ВАЖНО!!

ЧТОБЫ ПОНЯТЬ ЭТО,
НУЖНО ИЗУЧИТЬ
ПОРЯДОК УПРОЩЕНИЯ
С ПОМОЩЬЮ
КАРТЫ КАРНО.

НО СНАЧАЛА
НЕМНОГО ПОЕДИМ.

ДАВАЙ!
РАЗВЕРНУТАЯ
СКУМБРИЯ. ♪

♥♥♥...
ВКУСНАЯ
РАЗВЕРНУТАЯ
СКУМБРИЯ...

ВКУСНО

Выделяем группы единиц



Итак, расскажу о порядке упрощения с помощью карт Карно. Самое важное здесь — выделить группы единиц!



Группы единиц?



Например, сосредоточим внимание на единицах в таблице, которую мы только что составили (см. стр. 99).

На самом деле в карте Карно соседние ячейки с единицами можно объединять в «группы». Например, вот так.

$\begin{matrix} \backslash BA \\ C \end{matrix}$	00	01	11	10
0	0	0	1	0
1	0	1	1	1

Below the table, three groups of 1s are circled and labeled with arrows:
- A horizontal group of 1s in the second row, columns 3 and 4.
- A vertical group of 1s in the third column, rows 2 and 3.
- A single 1 in the third row, column 4.

Три группы единиц!



Так... В этом случае мы выделили три группы, да?



Верно! На самом деле удобство заключается в том, что каждой из таких групп достаточно назначить всего по одному вентилю AND!



...Ну и что?



Да, похоже, она не осознаёт, как это удобно!

Аисака-сан, вспомни о методе, о котором я тебе рассказывал.

Ведь в нём требовалось побольше вентилях AND — по одному на каждую единицу, не так ли?!

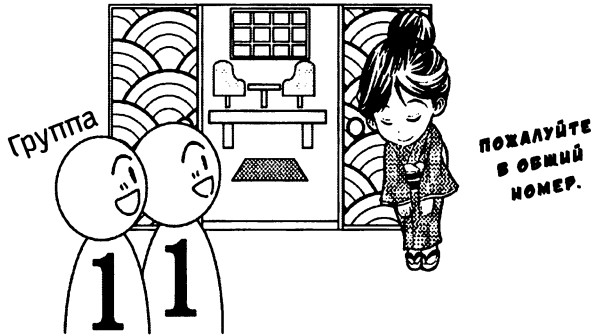


Ах, вот оно что! Общая идея мне ясна.

Объединив единицы в группы, мы уменьшаем количество вентиляей и упрощаем схему, да?

Это как гостиница, в которой группу клиентов-единиц можно разместить в одном общем номере AND! Конечно, так проще и выгоднее, чем подбирать для каждой единицы отдельные номера!

Номер AND



Да, да! Попробую объяснить поподробнее.

Пусть, например, есть вот такая таблица, и мы группируем в ней единицы.

Общий признак: $A = 1$

$\begin{matrix} BA \\ C \end{matrix}$	00	01	11	10
0		1	1	
1				

Другой общий признак: $C=0$

$\bar{C} \cdot A$



В этой группе есть два общих признака: $A = 1$ и $C = 0$, поэтому её можно представить в виде логического выражения $\bar{C} \cdot A$.



Ага!

Ведь ноль (уровень L) выражается с помощью знака инверсии!



Рассмотрим другой пример.
Вот в этой карте Карно общий признак выделенной группы — $B = 0$.

Общий признак: $B = 0$

	BA	↓	↓		
C	00	01	11	10	
0	1	1			
1	1	1			

\bar{B}



Значит, эту группу можно выразить как \bar{B} !



Верно. Наконец, давайте вернёмся к карте Карно для нашей мажоритарной схемы (см. стр. 102).

Как, по-твоему, можно выразить эти три группы?

	BA	00	01	11	10
C	0	0	0	1	0
1	0	0	1	1	1

$C \cdot A$ $B \cdot A$ $C \cdot B$



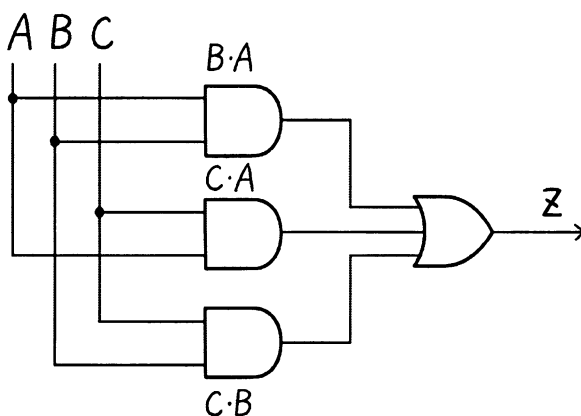
Если немного поразмыслить...
Как $(C \cdot A)$, $(B \cdot A)$ и $(C \cdot B)$!



Верно! Теперь нам остаётся только начертить схему. Порядок следующий.

Порядок черчения схемы после выделения групп

- Прежде всего строим шину, состоящую из входных линий (стр. 80).
- Назначаем каждой группе по одному вентилю AND. Входы схемы, которые в логическом выражении группы имеют знак инверсии, подключаем к входам вентиля AND, назначенного данной группе, через инверсные линии шины, а входы без знака инверсии — через обычные. Прodelьываем вышеуказанное для всех групп.
- В заключение подключаем выходы всех вентилях AND к входам общего вентиля OR.



Упрощённая мажоритарная схема



Получилось! Знаков инверсии на этот раз нет. Точно такая же схема, какую нарисовала директор вначале (см. стр. 96).



Вот именно. Карты Карно позволяют нам упростить схему, даже не задействуя интуицию или вдохновение.

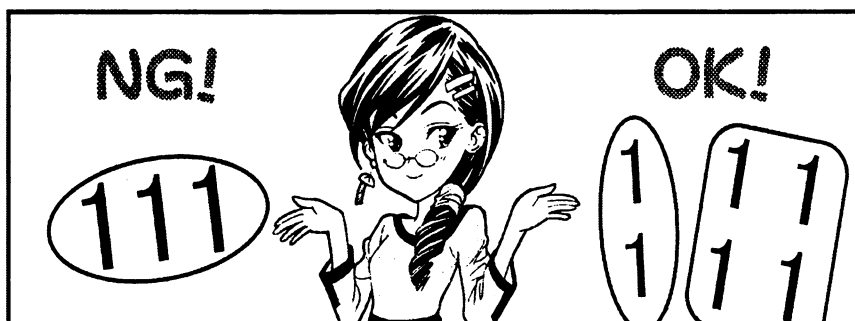


Да, да. Карты Карно — поистине удобная штука. Ах, если бы можно было управлять магазином по такой отработанной методике, не прибегая к интуиции...



Ага, значит, ты принимаешь решения интуитивно?!!

Важные моменты выделения групп



Мм... У меня есть одно сомнение...



В чём?! В моей способности управлять магазином? Или в моих взглядах... на любовь?



Директор, вы немного пьяны! Нет конечно, насчёт этих «групп». Почему в той таблице мы не объединили в одну группу три единицы, идущие подряд? Ведь так, наверное, было бы ещё проще.

BA \ C	00	01	11	10
0	0	0	1	0
1	0	1	1	1

Три единицы
поряд!



Нет, этого делать нельзя ни в коем случае!!
На самом деле существуют правила упрощения с помощью карты Карно.

Правила упрощения с помощью карты Карно

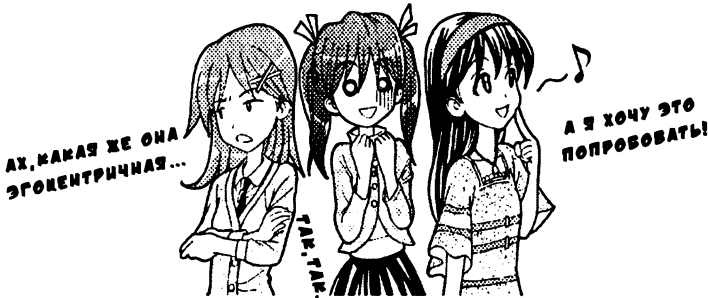
- ★ Группа должна представлять собой прямоугольник или квадрат, длина любой стороны которого равна 1, 2 или 4 ячейкам.
- ★ Группы могут накладываться друг на друга.
- ★ Необходимо стремиться к увеличению размеров групп и уменьшению числа групп



Хе-хе, вот как?! По трое, значит, собраться нельзя?

Ну, это можно понять.

Если три девушки вместе отправятся путешествовать, то, наверное, перессорятся.



Прямо как мы втроем!

Вспомните о развёрнутой скумбрии, которую мы только что съели.



Если бы мы смогли построить для неё карту Карно, то могли бы объединять в группы крайние правые и крайние левые единицы в одних и тех же строках, а также верхние и нижние единицы в одних и тех же столбцах.

Группа из крайней правой и крайней правой ячеек!

BA \ DC	00	01	11	10
00				
01	1			1
11				
10				

$\bar{A} \cdot C \cdot \bar{D}$

Группа из верхней и нижней ячеек!

BA \ DC	00	01	11	10
00	1			
01				
11				
10	1			

$\bar{A} \cdot \bar{B} \cdot \bar{C}$



А, поняла!

Значит, глядя на карту Карно, надо всегда помнить о том, что её можно свернуть в трубку, сомкнув края! Спасибо развёрнутой скумбрии!



Наконец, если выделить группы в карте Карно оказывается невозможным, то это означает, что дальнейшее упрощение схемы невозможно. Это тоже запомни.

Усложняем задачу выбора ресторана



Итак, мы построили мажоритарную схему, реализующую выбор тремя людьми ресторана японской или европейской кухни. Теперь немного усложним задачу, добавив вариант выбора ресторана китайской кухни. Будем решать второём, какую кухню выбрать: японскую, европейскую или китайскую. Как, по-твоему, лучше поступить?



Хе-хе. Гёдза, мабодофу, жареные креветки в соусе чили... Да, и ещё жареная свинина с зелёным перцем.



Ха-ха, не уклоняйся от ответа!

Действительно, цифровую схему выбора из двух вариантов реализовать проще, так как варианты выбора можно выразить двумя уровнями — L и H, на сигнальной линии.

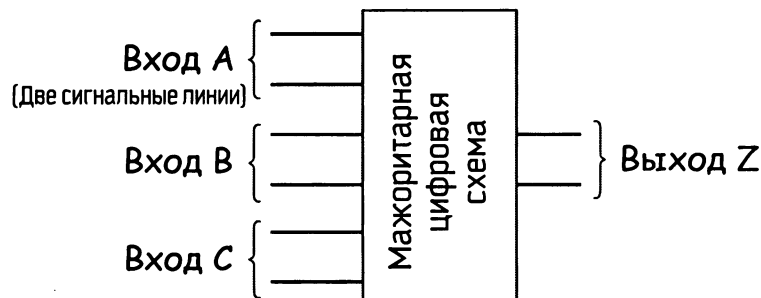
Однако, даже если вариантов три, отчаиваться не стоит, ведь можно использовать и две сигнальные линии.

Тогда можно будет назначить, например, японской кухне — LL, европейской кухне — LH, а китайской кухне — HL.



Да! Тогда общее количество входных сигнальных линий будет равно шести (2 линии/вход \times 3 входа = 6 линий) а выходных — двум.

Выглядеть это будет вот так.





Ага, понятно. В самом деле, так можно всего двумя уровнями — L и H — выразить три варианта выбора: японская, европейская или китайская кухня.



Этот способ называется кодированием. Выразив так варианты выбора, мы сможем построить вот такую таблицу истинности!

C_2C_1	B_2B_1	A_2A_1	Z_2Z_1
L L	L L	L L	L L
L L	L L	L H	L L
L L	L H	L L	L L
L L	L H	L H	L H
L H	L L	L L	L L
L H	L L	L H	L H
L H	L H	L L	L H
L H	L H	L H	L H

L L	L L	H L	L L
L L	H L	L L	L L
L L	H L	H L	H L
H L	L L	L L	L L
H L	L L	H L	H L
H L	H L	L L	H L
H L	H L	H L	H L

L H	L H	H L	L H
L H	H L	L H	L H
L H	H L	H L	H L
H L	L H	L H	L H
H L	L H	H L	H L
H L	H L	L H	H L

L L	L H	H L	H L
L H	L L	H L	H L
H L	L H	L L	L L
L H	H L	L L	L L
H L	L L	L H	L H
L L	H L	L H	L H

← Когда А проголосовал за европейскую кухню, но двое остальных — за японскую

Эта часть таблицы выражает ситуацию, когда мнения разделились.
* Здесь будем считать мнение А решающим.

Таблица истинности с добавлением китайской кухни



Ого! Таблица увеличилась невероятно! Ведь мы добавили вариант выбора, увеличив количество возможных комбинаций, то есть строк.



Да. И ещё появилась одна проблема: что делать, когда мнения троих разделились? Для таких комбинаций нужно придумать какое-то правило, как в этой таблице, где мы приняли мнение А за решающее.



Этот А, разумеется, наш директор: если мнения разделятся, то её голос станет решающим.



Да, разумеется. Кхе...

Хотя эта таблица выглядит очень большой, её можно преобразовать в схему абсолютно тем же методом, который мы изучили сегодня.

Конечно, количество используемых вентилях увеличится, но беспокоиться не стоит, ведь обычно всю рутинную работу делает за нас CAD.



Ого, и здесь CAD! Как он нас выручает!



Да, да. Огромные таблицы и схемы могут на первый взгляд выглядеть страшными, но слишком бояться их не надо, ведь процесс проектирования систематизирован.

Да, ещё об одном. Мы рассмотрели случай, когда голосующих трое, то есть число входов равно трём, но что делать, если их будет ещё больше? Можно ли будет использовать карту Карно?



Так... Таблицу Карно для четырех входов (А, В, С, D) я только что видела, но если их будет ещё больше? Не знаю...



Действительно, если количество входов меньше пяти, то можно обойтись двумерной картой Карно, но если их больше, то потребуется уже трёхмерная. Если же входов больше шести, то использовать её для упрощения будет уже трудно...

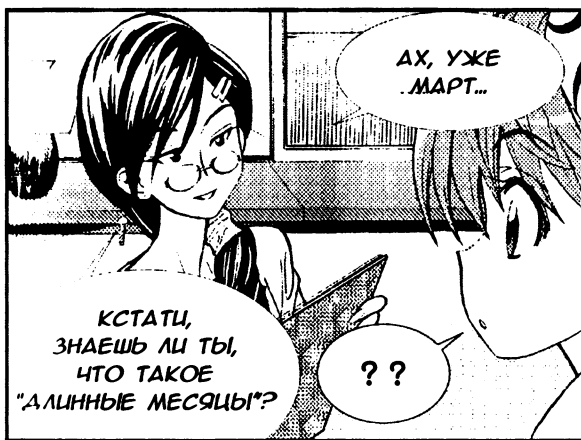
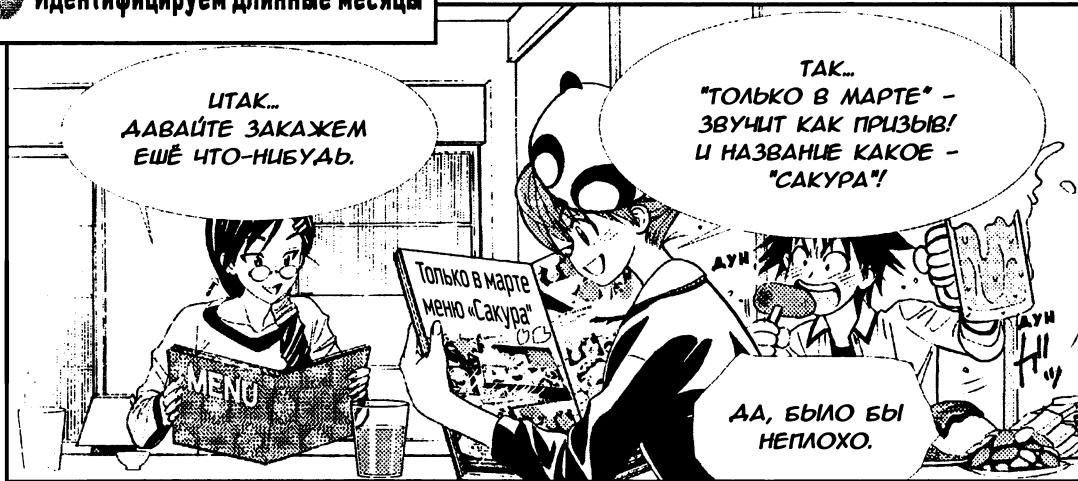


Да, ты прав. Но даже таких случаев бояться не надо, ведь в наши дни в деле упрощения схем можно полностью довериться CAD.

А нам достаточно просто хорошо изучить основы, чтобы понимать суть процесса проектирования.

2. БЕЗРАЗЛИЧНОЕ СОСТОЯНИЕ

Идентифицируем длинные месяцы



ЭТО ПРОСТО!
МЕСЯЦЫ, В КОТОРЫХ 31 ДЕНЬ,
НАЗЫВАЮТСЯ "ДЛИННЫМИ",
А ОСТАЛЬНЫЕ - "КОРОТКИМИ".

Длинные месяцы	Короткие месяцы
Январь, март, май, июль август, сентябрь декабрь	Февраль, апрель июнь, сентябрь ноябрь



НА САМОМ ДЕЛЕ МОЖНО СОЗДАТЬ ЦИФРОВУЮ СХЕМУ, КОТОРАЯ БУДЕТ ИХ ИДЕНТИФИЦИРОВАТЬ. ВВОДИТЬ МЕСЯЦ - И ОНА ОПРЕДЕЛЯЕТ, ДЛИННЫЙ ОН ИЛИ КОРОТКИЙ.

Длинный месяц!!

МАРТ!

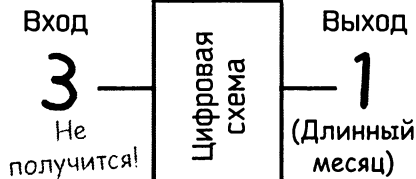
АЗЫНЫ!

НАПРИМЕР, ЕДИНИЦА НА ВЫХОДЕ БУДЕТ ОЗНАЧАТЬ ДЛИННЫЙ МЕСЯЦ, А НУЛЬ - КОРОТКИЙ.

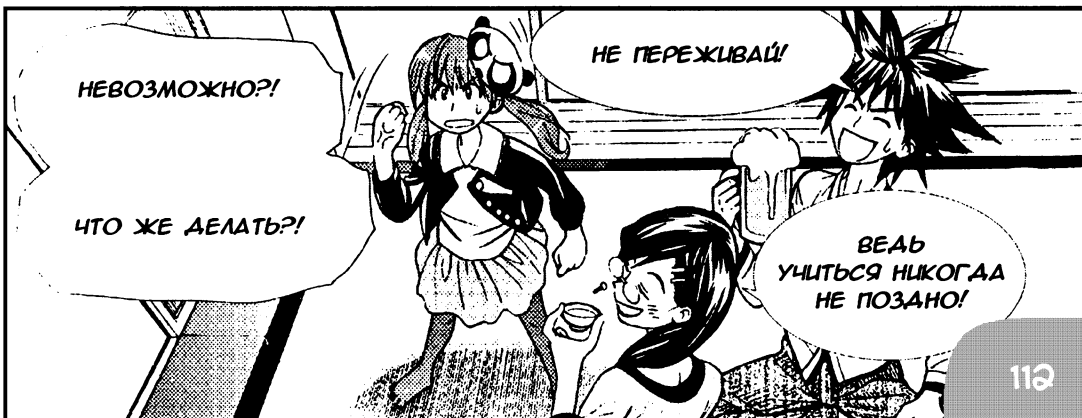
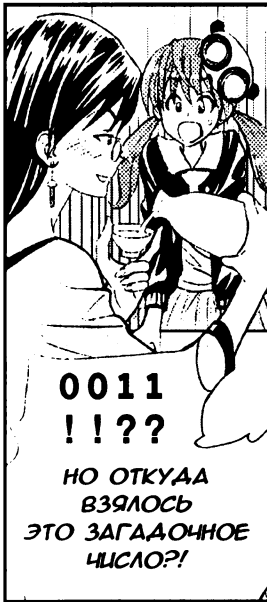
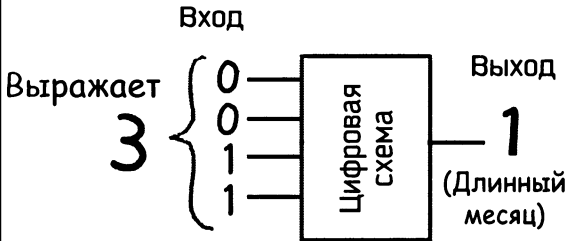
ОГО!
ЭТО ИНТЕРЕСНО!

НО НЕЛЬЗЯ ЗАБЫВАТЬ, ЧТО ЦИФРОВЫЕ СХЕМЫ - ЭТО МИР, В КОТОРОМ НЕТ НИЧЕГО, КРОМЕ 0 И 1 (L И H).

ПОЭТОМУ НОМЕР МЕСЯЦА В ГОДУ (САМЯ МАРТА - 3) НЕПОСРЕДСТВЕННО ВВЕСТИ НЕ ПОЛУЧИТСЯ.



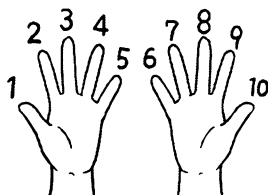
НАПРИМЕР, ЧИСЛО "3" ДЛЯ
МАРТА ПРЯДЕТСЯ ВВОДИТЬ КАК
0011 С ПОМОЩЬЮ
ЧЕТЫРЁХ СИГНАЛЬНЫХ ЛИНИЙ



Десятичные и двоичные числа



Итак, расскажу о двоичных и десятичных числах.
Мы, люди, привыкли считать, используя десять пальцев.
Вот так: один (1), два (2), три (3), четыре(4), пять(5), шесть (6), семь (7),
восемь (8)...



Для записи чисел
используются десять
знаков от 0 до 9,
называемых «цифрами»

**Десятичная
система счисления**

Но цифры заканчиваются на «девяти» (9), поэтому число «десять» придётся записывать уже двумя знаками — 10 (увеличивается количество разрядов, то есть разрядность числа). Эта система счисления называется «десятичной», а выраженные с её помощью числа — «десятичными». Именно их мы используем в повседневной жизни.



Однако, например, в компьютерах, цифровых схемах используется двоичная система счисления, в которой есть всего две цифры — 0 и 1. Особенностью чисел, выраженных с её помощью, то есть двоичных чисел, является очень быстрое увеличение разрядности.

< Сопоставление двоичных и десятичных чисел >

Десятичные	Двоичные
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010

Десятичные	Двоичные
11	1011
12	1100
13	1101
14	1110
15	1111
16	10000
17	10001
...	...



Ого! Количество разрядов так быстро увеличивается!
Глядя на только эти длинные ряды из нулей и единиц, трудно понять, какие числа они обозначают...



Да, действительно, количество разрядов в них больше, поэтому они трудночитаемые. Но вспомни опять о том, что в мире цифровых схем нет ничего, кроме 0 и 1 (уровней L и H). Поэтому нам ничего не остаётся, как использовать в цифровых схемах эту двоичную систему счисления.



Вот как? Поэтому мы должны вводить не «3», а «11»?
Но постой... Почему для «марта», то есть числа «3», мы использовали целых четыре разряда — «0011»? (См. стр. 113.)



Ты хорошо заметила.
В реальных цифровых схемах одно значение вводится по нескольким сигнальным линиям. Например, если сигнальных линий четыре, то можно выразить шестнадцать значений — от 0 до 15.

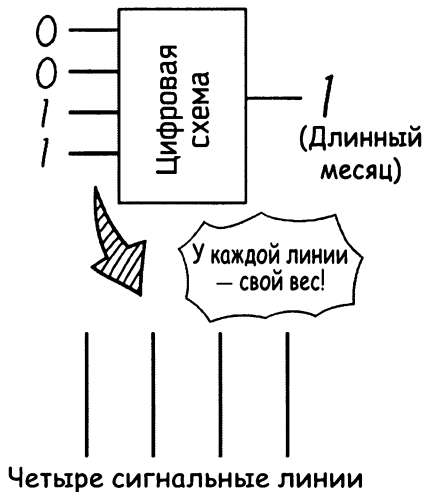
А так как месяцев в году всего 12, то нам потребуется четыре сигнальные линии (другими словами, четыре разряда). Сложновато?



...Да, честно говоря, немного сложно. Мм...
Что всё это значит?



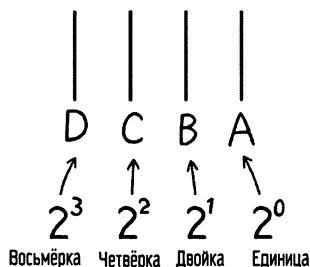
Так... Для ввода используются четыре сигнальные линии.
На самом деле у каждой из этих линий есть свой вес!





Представь три монеты: 1-иеновую, 10-иеновую и 100-иеновую. Хотя количества монет одинаковые, денежный вес (ценность) у них разные, не так ли?

Таким же образом, каждой сигнальной линии назначается вес.



Например, если 4 линии записаны последовательностью DCBA, как на этом рисунке, то крайней справа линии А присваивается вес 1 ($2^0 = 1$), линии В — вес 2 ($2^1 = 2$), линии С — вес 4 ($2^2 = 4$), а линии D — вес 8 ($2^3 = 8$).

Другими словами, А — это единица, В — двойка, С — четвёрка, D — восьмёрка.

Если все разряды $D = C = B = A = 0$, то итог будет 0, а если $D = C = B = A = 1$, то итог будет $8 + 4 + 2 + 1 = 15$.



Так... Проще говоря, А — это 1-иеновая монета, В — 2-иеновая, С — 4-иеновая, D — 8-иеновая. Для каждой из этих монет возможны только два случая: либо она у нас есть (1), причём в единственном экземпляре, либо её у нас нет (0).

	D(2^3)	C(2^2)	B(2^1)	A(2^0)
1 (Есть)				
0 (Нет)				

Если у меня есть все монеты, то у меня всего 15 иен!

Если у меня нет вообще никаких монет, то у меня 0 иен!



Да, каждый из разрядов DCBA может быть равен либо 0, либо 1. Комбинируя их, можно выразить все числа от 0 до 15. Кстати, разряды двоичных чисел называют «битами».



Уфф. Смысл, в общем, уловила. Например, число «3» выражается так: D = 0 (восьмёрки нет), C = 0 (четвёрки нет), B = 1 (двойка есть), A = 1 (единица есть). Поэтому оно записывается как «0011». Всё ясно!



Всё правильно! Можно считать, что с двоичной системой счисления мы разобрались.



Да, но, наверное, нелегко привыкнуть к этим двоичным числам, таким длинным и монотонным. Ведь мы люди, а не компьютеры.



Не переживай, ведь переводить двоичные числа в десятичные или, наоборот, десятичные в двоичные очень просто. Давай потренируемся на числе 18.

Десятичное → Двоичное

Задача:

Перевести десятичное число 18 в двоичную систему счисления.

Шаг 1 Частное / Остаток

2) 18	/		
2) 9		...	0
2) 4		...	1
2) 2		...	0
2) 1		...	0
0		...	1

Делим на 2 до тех пор, пока частное не станет равно 0, выписывая остаток от каждого деления.

Шаг 2

Остатки	0 ①	}	→	⑤ ④ ③ ② ①
	1 ②			10010
	0 ③			
	0 ④			
	1 ⑤			

Выписываем остатки справа налево

Двоичное → Десятичное

Задача:

Перевести двоичное число 10010 в десятичную систему счисления.

Шаг 1

10010	
$2^4 \ 2^3 \ 2^2 \ 2^1 \ 2^0$	← Умножаем значение каждого бита на его вес.

Шаг 2

10010	=	(0 × 2 ⁰) +	}	Суммируем все полученные значения.
		(1 × 2 ¹) +		
		(0 × 2 ²) +		
		(0 × 2 ³) +		
		(1 × 2 ⁴)		

= 0 + 2 + 0 + 0 + 16 = 18



Да, в самом деле, очень просто! Надо только считать внимательно.

■ Проектируем схему идентификации длинных месяцев



Наконец, мы переходим к проектированию схемы.

Для начала попробуй выразить все месяцы в виде двоичных чисел! ...Хотя нет, достаточно только длинных месяцев: января, марта, мая, июля, августа, октября и декабря.



Но почему только длинных?



А ты вспомни, как мы проектировали мажоритарную схему. Ведь и в таблице истинности (см. стр. 79), и в карте Карно (см. стр. 98) нам в конечном итоге понадобились только те комбинации, которые давали на выходе единицу (Н), не так ли?

На этот раз мы проектируем схему идентификации длинных месяцев, поэтому нам достаточно только тех комбинаций, которые должны давать результат «длинный месяц». В реальном проектировании иногда можно пропустить некоторые этапы. Например, опытные проектировщики часто сразу составляют карту Карно, пропуская таблицу истинности.



Действительно, зачем делать лишнюю работу? Энергосбережение! Получится вот что.

Январь (1) → 0001 (бит единицы равен 1)

Март (3) → 0011 (биты двойки и единицы равны 1)

Май (5) → 0101 (биты четвёрки и единицы равны 1)

Июль (7) → 0111 (биты четвёрки, двойки и единицы равны 1)

Август (8) → 1000 (бит восьмёрки равен 1)

Октябрь (10) → 1010 (биты восьмёрки и двойки равны 1)

Декабрь (12) → 1100 (биты восьмёрки и четвёрки равны 1)

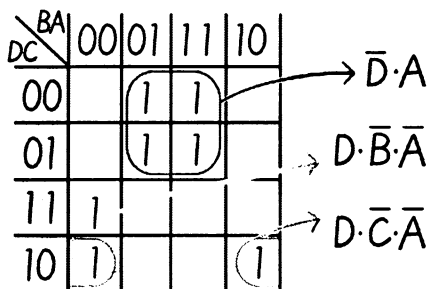


Да. А таблица истинности будет выглядеть вот так.

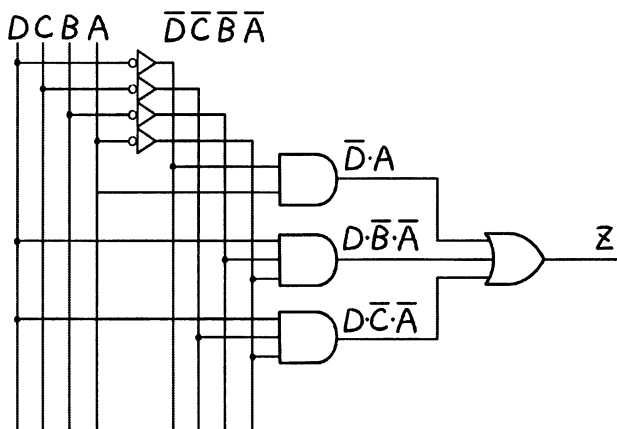
	D	C	B	A	Z
Январь (1) →	0	0	0	1	1 (длинный месяц)
Март (3) →	0	0	1	1	1
Май (5) →	0	1	0	1	1
	⋮	⋮	⋮	⋮	⋮



Вот, карта Карно готова! И группы я тоже выделила. Ицумо, начерти нам схему.



Ого! Быстро, как умелый шеф-повар!
Хорошо, попробую. Ведь нужно просто использовать методы, которые я изучила?
Вот, схема готова!



Молодец, сделано хорошо!
Можно сказать, безупречно... Но есть одна проблема...



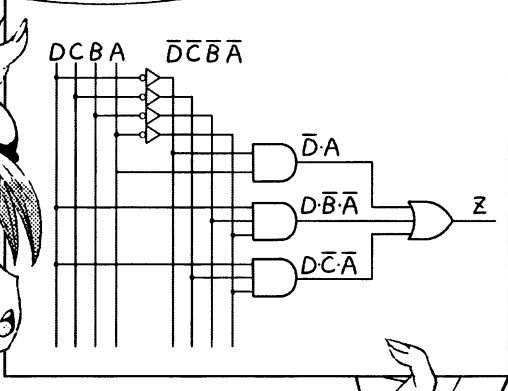
Как? Несмотря на то, что безупречно?!
Неужели опять придётся переделывать? Нехорошее предчувствие!

Безразличное состояние



И ДЛИННЫЕ МЕСЯЦЫ В ВИДЕ
АВОЧНЫХ ЧИСЕЛ ВЫРАЗИЛИ, И
УПРОЩЕНИЕ С ПОМОЩЬЮ КАРТЫ
КАРНО ПРОВЕЛИ...

КАКИЕ ЕЩЁ
МОГУТ БЫТЬ
ПРОБЛЕМЫ?!



НЕТ, ЕСТЬ ПРОБЛЕМА!
ЭТА СХЕМА НЕ ОЧЕНЬ
РАЦИОНАЛЬНА!

НАДО СДЕЛАТЬ
ЕЁ ЭКОНОМНЕЕ!



НО ЗАЧЕМ НУЖНО
НА ВСЁМ
ЭКОНОМИТЬ?
НЕ СЛИШКОМ ЛИ
ТЫ СКУПА?

НУЖНО!
ВЕДЬ РАЦИОНАЛЬНО
СПРОЕКТИРОВАННЫЕ
СХЕМЫ ВЫГЛЯДЯТ
НАМНОГО КРАСИВЕЕ!



ИТАК, ДАВАЙТЕ
ПОРАЗМЫШЛЯЕМ
ВМЕСТЕ.

ВО-ПЕРВЫХ,
В ЭТОЙ ЗАДАЧЕ
МЫ ВВОДИМ ЧИСЛА
ОТ 1 ДО 12, ТАК?

НАМ НЕ НУЖНО
ВВОДИТЬ, НАПРИМЕР,
0, 13, 14 ИЛИ 15.



КОНЕЧНО,
РАЗУМЕЕТСЯ.

НО ПОСТОЙ!

ЧЕТЫРЬМЯ СИГНАЛЬНЫМИ
ЛИНИЯМИ МОЖНО
ВЫРАЗИТЬ
ЧИСЛА ОТ 0 ДО 15!
(См. стр. 116.)

ДАЖЕ ТЕ ЧИСЛА,
КОТОРЫЕ
НАМ НЕ НУЖНЫ...
НАВЕРНОЕ, СЕКРЕТ
НЕРАЦИОНАЛЬНОСТИ
КРОЕТСЯ В ЭТОМ...

ВЕРНО!
ЧИСЛА 0, 13, 14 И 15
НЕ СООТВЕТСТВУЮТ
НИКАКИМ МЕСЯЦАМ,
ПОЭТОМУ ВВОДИТЬСЯ
ОНИ НИКОГДА НЕ БУДУТ.

ДЛЯ ЭТИХ ЗНАЧЕНИЙ,
КОТОРЫЕ НЕ БУДУТ
ВВОДИТЬСЯ, НА ВЫХОДЕ
МОЖЕТ БЫТЬ ХОТЬ 1, ХОТЬ 0.
НАМ ВСЁ РАВНО!

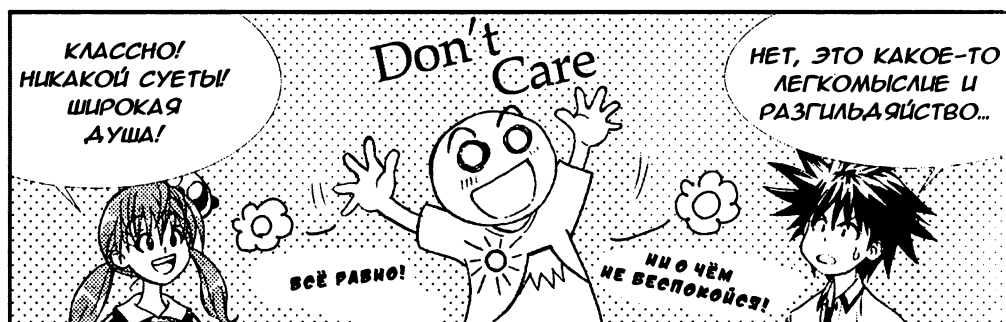
СОВЕРШЕННО
БЕЗРАЗЛИЧНО!!

КАКОЕ-ТО
СЛИШКОМ
НЕБРЕЖНОЕ
ОБЪЯСНЕНИЕ.

АА, ПОХОЖЕ,
ОНА СЛИШКОМ
МНОГО ВЫПИЛА.

АЕРГ

НЕТ...



BA \ DC	00	01	11	10
00	—	1	1	
01		1	1	
11	1	—	—	—
10	1			1

Вот это, например, ячейка 14 (1110).

ТУТ МОЖНО ЗАМЕТИТЬ КОЕ-ЧТО ИНТЕРЕСНОЕ. ДЕЛО В ТОМ, ЧТО НИЧТО НЕ ЗАПРЕЩАЕТ НАМ ПРИНЯТЬ БЕЗРАЗЛИЧНОЕ СОСТОЯНИЕ РАВНЫМ 1, ЧТОБЫ РАСШИРИТЬ ГРУППЫ ЕДИНИЦ.

BA \ DC	00	01	11	10
00	—	1	1	
01		1	1	
11	1	—	—	—
10	1			1

$\bar{D} \cdot A$

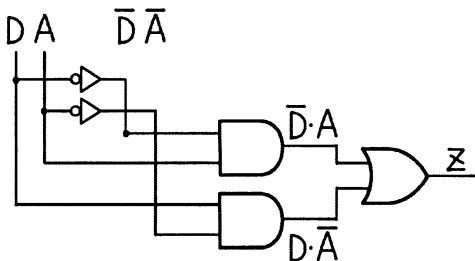
$D \cdot \bar{A}$

Благодаря безразличному состоянию!

ОГО! ЭТО БЕЗРАЗЛИЧНОЕ СОСТОЯНИЕ НАС ВЫРУЧАЕТ!

ОСТАЛОСЬ ВСЕГО ДВЕ ГРУППЫ!

А ЗНАЧИТ, СХЕМУ МОЖНО ЕЩЕ УПРОСТИТЬ.



ОГО! СТАЛО ПРОШЕ, ЧЕМ БЫЛО.

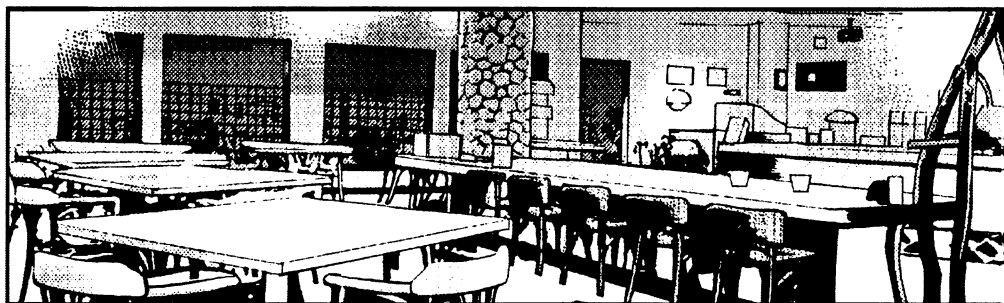
АА, БЕЗ ЭТОГО БЕЗРАЗЛИЧНОГО СОСТОЯНИЯ ПРИ УПРОЩЕНИИ СХЕМ НЕ ОБОЙТИСЬ...

НЕАДРОМ У НЕГО ТАКАЯ ШИРОКАЯ ДУША!

АА УЖ... Я ВОТ ТОЖЕ ХОТЕЛА БЫ ОСТАВАТЬСЯ БЕЗРАЗЛИЧНОЙ К ТОМУ, В ПРИБЫЛИ НАШ МАГАЗИН ИЛИ В УБЫТКЕ...

У НЕЕ СТРЕСС, ЧТО ЛИ?

3. А ЕСЛИ ВЫХОДОВ НЕСКОЛЬКО?

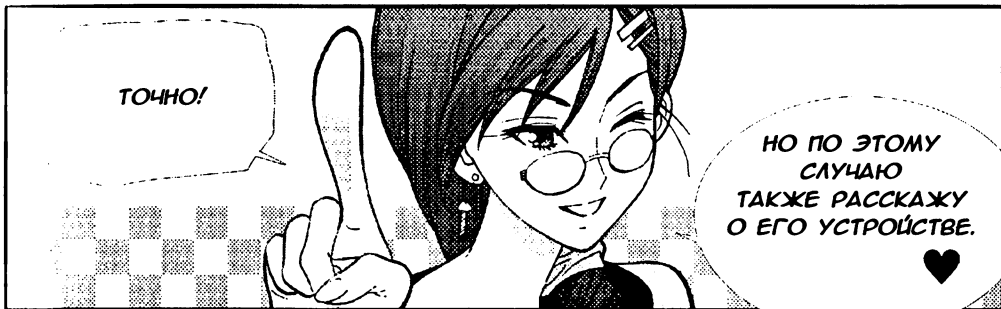




АХ, ЭТА ЗАМЕЧАТЕЛЬНАЯ ИГРУШКА! ТЫ БУДЕШЬ РАССКАЗЫВАТЬ МНЕ ПРО ТО, КАК УСТРОЕН ЭТОТ ЭЛЕКТРОННЫЙ КУБИК?

...НЕТ.
ЭТО ЛЮБИМАЯ ИГРА НАШЕГО ДИРЕКТОРА: У КОГО МЕНЬШЕ ВЫПАДЕТ, ТОТ ПЛАТИТ ЗА ВСЕХ.

ЧТО?!

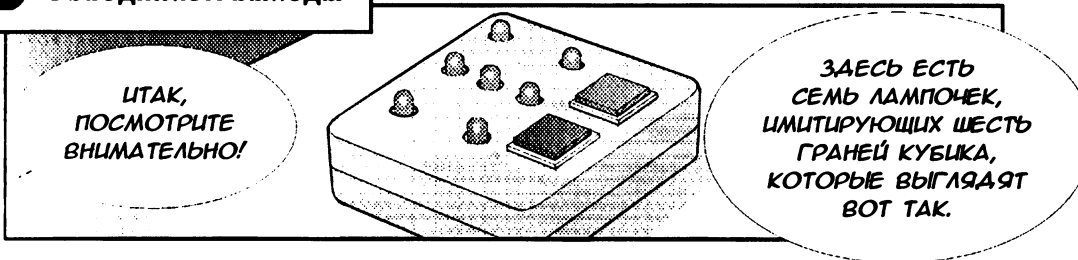


ТОЧНО!

НО ПО ЭТОМУ СЛУЧАЮ ТАКЖЕ РАССКАЖУ О ЕГО УСТРОЙСТВЕ.

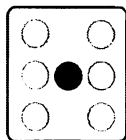
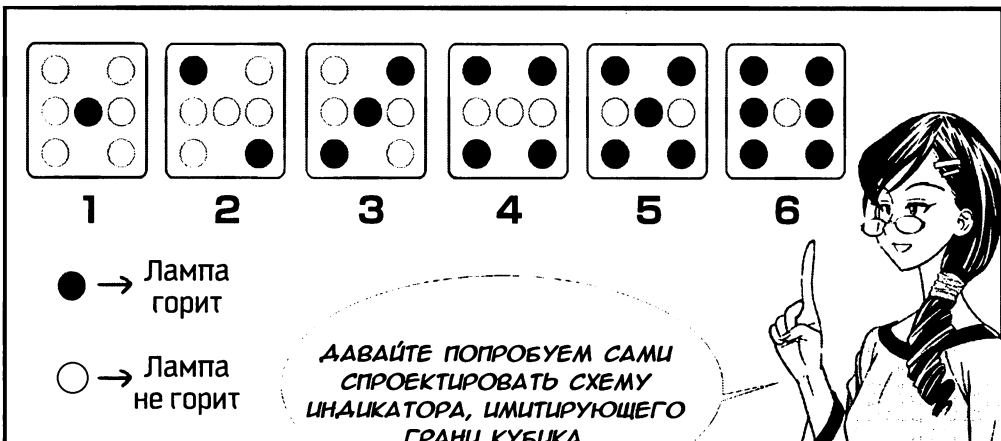


Объединяем выходы

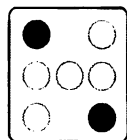


ИТАК, ПОСМОТРИТЕ ВНИМАТЕЛЬНО!

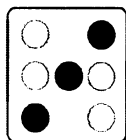
ЗДЕСЬ ЕСТЬ СЕМЬ ЛАМПОЧЕК, ИМИТИРУЮЩИХ ШЕСТЬ ГРАНЕЙ КУБИКА, КОТОРЫЕ ВЫГЛЯДЯТ ВОТ ТАК.



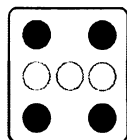
1



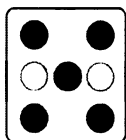
2



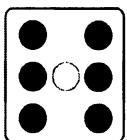
3



4



5



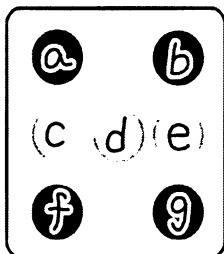
6

● → Лампа горит

○ → Лампа не горит

ДАВАЙТЕ ПОПРОБУЕМ САМИ СПРОЕКТИРОВАТЬ СХЕМУ ИНДИКАТОРА, ИМИТИРУЮЩЕГО ГРАНИ КУБИКА.

АА, ЭТО ПОНЯТНО.
СХЕМА, В КОТОРОЙ
7 ЛАМП ЗАГОРАЮТСЯ
И ГАСНУТ, ИМИТИРУЯ
ОЧКИ НА ГРАНЯХ
КУБИКА.



ОЧЕВИДНО, ЧТО НАДО
ПОДАВАТЬ НА ЛАМПОЧКИ
ВЫХОДНЫЕ СИГНАЛЫ 1
ИЛИ 0 - В ЗАВИСИМОСТИ
ОТ ТОГО, ДОЛЖНЫ ОНИ
ГОРЕТЬ ИЛИ НЕТ.

ВОТ, НАПРИМЕР, ДЛЯ
ГРАНИ 4

При отображении грани 4:

a, b, f, g ... 1 (горят)
c, d, e ... 0 (не горят)

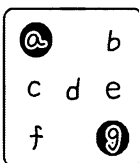


ПРАВИЛЬНО!
НО РАССМАТРИВАТЬ
КАЖДУЮ ИЗ СЕМИ ЛАМПОЧЕК
ПО ОТДЕЛЬНОСТИ -
УТОМИТЕЛЬНОЕ ЗАНЯТИЕ.

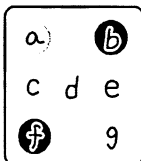


МОЖНО СДЕЛАТЬ ПРОЩЕ,
ЕСЛИ ПОСМОТРЕТЬ
ВНИМАТЕЛЬНО.
ЕСТЬ ЦАЕЦ!

ТАК...
АГА!



Когда a горит,
g тоже горит!
(грани 2, 4, 5, 6)

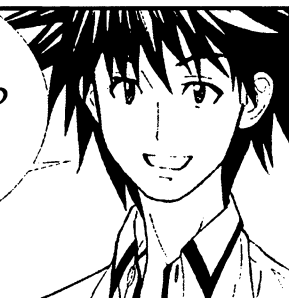


Когда b горит,
f тоже горит!
(грани 3, 4, 5, 6)

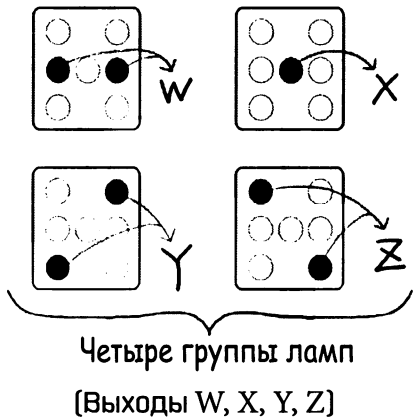


ЛАМПЫ,
РАСПОЛОЖЕННЫЕ В
ПРОТИВОПОЛОЖНЫХ
УГЛАХ, ВЕДУТ СЕБЯ
ОДИНАКОВО.

ЗНАЧИТ, ИХ,
НАВЕРНОЕ, МОЖНО
СГРУППИРОВАТЬ
ПОПАРНО.

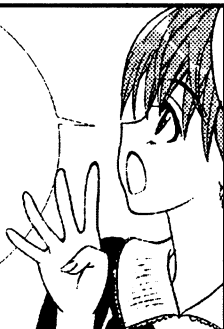


АА! НА САМОМ ДЕЛЕ ПРИ ПРОЕКТИРОВАНИИ ЭТОЙ СХЕМЫ МОЖНО РАССМАТРИВАТЬ НЕ СЕМЬ ЛАМП, А ЧЕТЫРЕ ГРУППЫ ЛАМП.



ТАКИМ ОБРАЗОМ, УМЕЛО ОБЪЕДИНИВ ВЫХОДЫ, МЫ СМОЖЕМ ИСПОЛЬЗОВАТЬ ОДИН ВЫХОДНОЙ СИГНАЛ ДЛЯ НЕСКОЛЬКИХ ЛАМПОЧЕК!

ЯСНО! ВЕДЬ НАМНОГО ПРОЩЕ РАССМАТРИВАТЬ ЧЕТЫРЕ ВЫХОДА, А НЕ СЕМЬ.



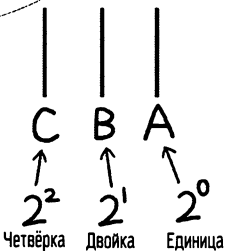
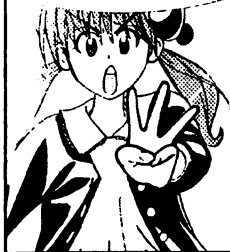
АА, АА. ТЕПЕРЬ МЫ РАССМОТРИМ ВХОДЫ.

ВЫРАЗИ ОЧКИ НА ГРАНЯХ КУБИКА - ЧИСЛА ОТ 1 ДО 6 - В ВИДЕ ДВОИЧНЫХ ЧИСЕЛ, КАК ТЫ СДЕЛАЛА С МЕСЯЦАМИ.



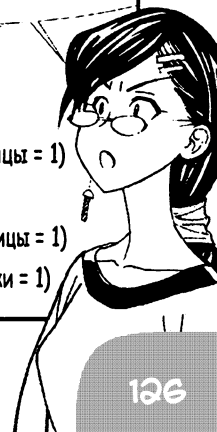
ТАК... ПОТРЕБУЕТСЯ ТРИ СИГНАЛЬНЫЕ ЛИНИИ, ТО ЕСТЬ ТРИ БИТА (РАЗРЯДА)!

(См. стр. 115.)



ПРАВИЛЬНО, НО ЗАДЕСЬ ТРЕБУЕТСЯ ВНИМАНИЕ. ДЕЛО В ТОМ, ЧТО ТРЕМЯ СИГНАЛЬНЫМИ ЛИНИЯМИ (БИТАМИ) МЫ ВЫРАЗИМ ЧИСЛА ОТ 0 ДО 7.

- 1 → 001 (бит единицы = 1)
- 2 → 010 (бит двойки = 1)
- 3 → 011 (биты двойки и единицы = 1)
- 4 → 100 (бит четвёрки = 1)
- 5 → 101 (биты четвёрки и единицы = 1)
- 6 → 110 (биты четвёрки и двойки = 1)





Ой!
НА НАСТОЯЩЕМ КУБИКЕ
НЕ МОЖЕТ
ВЫПАСТЬ НИ НУЛЯ,
НИ СЕМИ ОЧКОВ!

Don't

всё равно!

Care

НИ О ЧЁМ
НЕ ВОСПОМИНИ!

ЯСНО!
ЗНАЧИТ, ЧИСЛАМ 0 И 7
МЫ НАЗНАЧИМ
БЕЗРАЗЛИЧНОЕ СОСТОЯНИЕ
ВЫХОДА!



АА, ВЕРНО.
ИТАК, ТЕПЕРЬ
МЫ ГОТОВЫ
К СОСТАВЛЕНИЮ
ТАБЛИЦЫ
ИСТИННОСТИ
И КАРТЫ КАРНО.

АХ,
ИЗУМИТЕЛЬНО!

レーン
ベラス

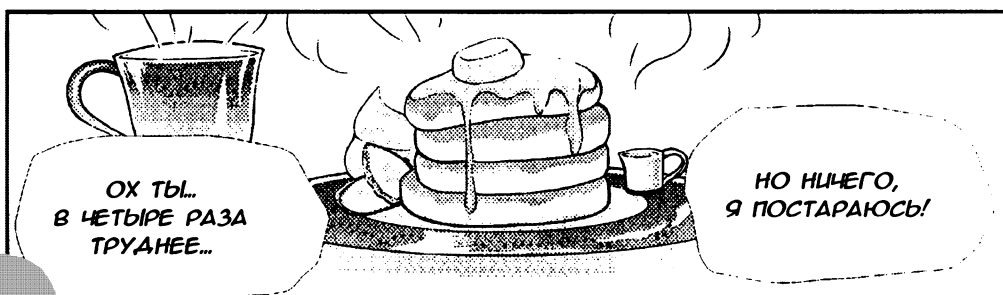


ЗДЕСЬ У НАС
4 ГРУППЫ ЛАМПОЧЕК, КОТОРЫЕ
ДОЛЖНЫ ГОРЕТЬ (1) ИЛИ
НЕ ГОРЕТЬ (0), ПОЭТОМУ
ПРИДЁТСЯ РАССМАТРИВАТЬ
ЦЕЛЫХ 4 ВЫХОДА.

ОГО!
НЕ ОЖИДАЛА....

ТА-ААМ!

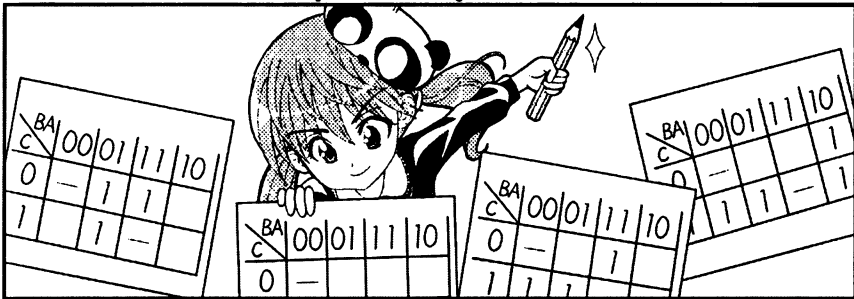
А ЗНАЧИТ,
ТАБЛИЦ ИСТИННОСТИ,
КАК И КАРТ КАРНО,
ТОЖЕ БУДЕТ
ЧЕТЫРЕ!



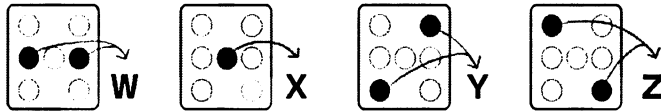
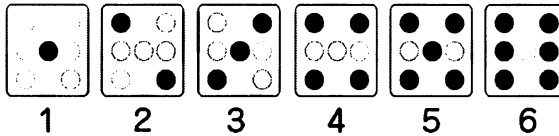
ОХ ТЫ..
В ЧЕТЫРЕ РАЗА
ТРУДНЕЕ...

НО НИЧЕГО,
Я ПОСТАРАЮСЬ!

Проектируем схему индикатора электронного кубика



Итак, давайте не спеша и шаг за шагом обдумаем устройство этой схемы.



Сопоставление, как на рисунке выше, шести граней кубика — от 1 до 6 — и четырёх групп ламп позволяет установить соответствие между гранями и включенными группами ламп.

- ◆ Горит группа лампочек W (1 на выходе W)
Грань 6
- ◆ Горит группа лампочек X (1 на выходе X)
Грани 1, 3, 5
- ◆ Горит группа лампочек Y (1 на выходе Y)
Грани 3, 4, 5, 6
- ◆ Горит группа лампочек Z (1 на выходе Z)
Грани 2, 4, 5, 6



Ясно.
Например, таблица истинности для группы лампочек W будет вот такой, да?

Так как 6 — это 110...

	C	B	A	Z
Грань 6 →	1	1	0	1 (горит)
	⋮	⋮	⋮	⋮
	⋮	⋮	⋮	⋮



Да, когда привыкнешь, будешь сразу составлять карту Карно.
Но не забудь поставить знаки «безразличного состояния» (прочерки) в ячейках 0 (двоичное число 000) и 7 (двоичное число 111).



Попробую составить все четыре...
Карты Карно для групп W, X, Y и Z будут вот такими!
И прочерки для «безразличного состояния» — тоже на месте!

Когда горит группа лампочек W
(1 на выходе W):
грань 6 (110)

BA	00	01	11	10
C	0	—		
	1		—	1

Карта Карно для выхода W

Когда горит группа лампочек X
(1 на выходе X):
границы 1 (001), 3 (011) или 5 (101)

BA	00	01	11	10
C	0	—	1	1
	1	1	—	

Карта Карно для выхода X

Когда горит группа лампочек Y
(1 на выходе Y):
границы 3 (011), 4 (100), 5 (101) или 6 (110)

$\begin{matrix} BA \\ \hline C \end{matrix}$	00	01	11	10
0	—		1	
1	1	1	—	1

Карта Карно для выхода Y

Когда горит группа лампочек Z
(1 на выходе Z):
границы 2 (010), 4 (100), 5 (101) или 6 (110)

$\begin{matrix} BA \\ \hline C \end{matrix}$	00	01	11	10
0	—			1
1	1	1	—	1

Карта Карно для выхода Z



Верно! А после выделения групп единиц у нас получится следующее.
При выделении групп важно правильно использовать безразличное состояние.

$\begin{matrix} BA \\ \hline C \end{matrix}$	00	01	11	10
0	—			
1			\ominus 1	

$\rightarrow C \cdot B$

Карта Карно для выхода W

$\begin{matrix} BA \\ \hline C \end{matrix}$	00	01	11	10
0	—	1	1	
1		1	—	

$\rightarrow A$

Карта Карно для выхода X

$\begin{matrix} BA \\ \hline C \end{matrix}$	00	01	11	10
0	—		1	
1	1	1	—	1

$\rightarrow B \cdot A$
 $\rightarrow C$

Карта Карно для выхода Y

$\begin{matrix} BA \\ \hline C \end{matrix}$	00	01	11	10
0	—			1
1	1	1	—	1

$\rightarrow B \cdot \bar{A}$
 $\rightarrow C$

Карта Карно для выхода Z



Значит, уже можно чертить схему!
Так... Вот, готово! У меня получилось вот что.

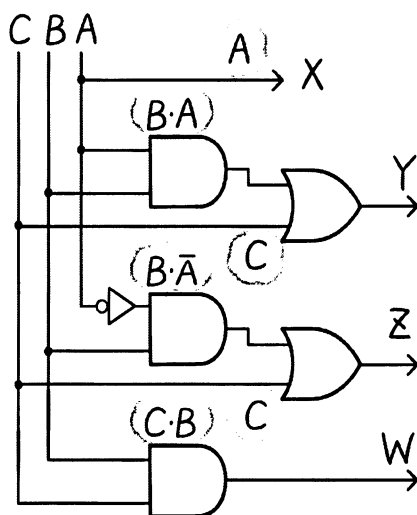


Схема индикатора электронного кубика



Всё о'кэй! Подключив к выходам этой схемы лампочки, мы получим индикатор электронного кубика.



Ах, как это классно!
Я смогла начертить ещё одну схему!



Молодец! Благодаря тому, что мы с самого начала как следует поразмыслили и уменьшили количество выходов с 7 до 4, мы смогли на этот раз сразу получить рациональную схему, не требующую дальнейшего упрощения.



Угу. Это свидетельствует о том, что размышлять о том, как сделать схему более рациональной, избавиться от лишнего в ней, лучше всего с самого начала проектирования, а не на этапе упрощения после составления таблицы истинности.

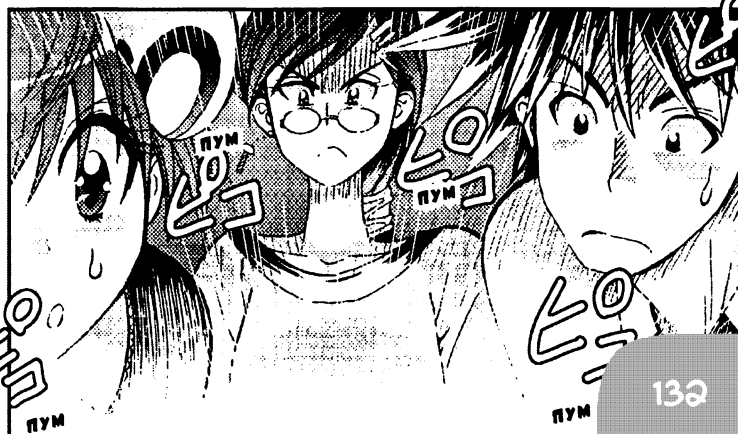
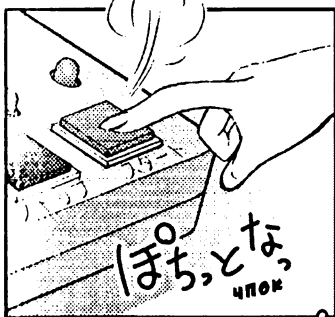
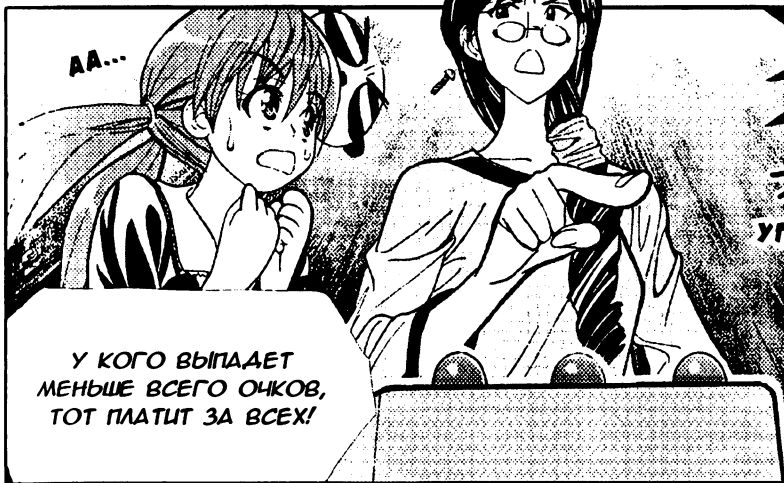
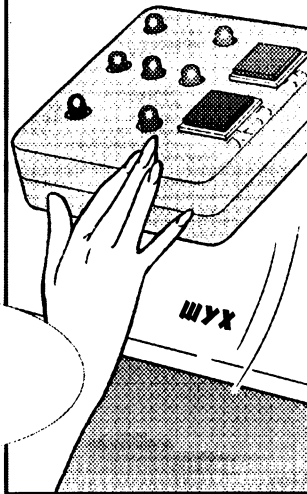
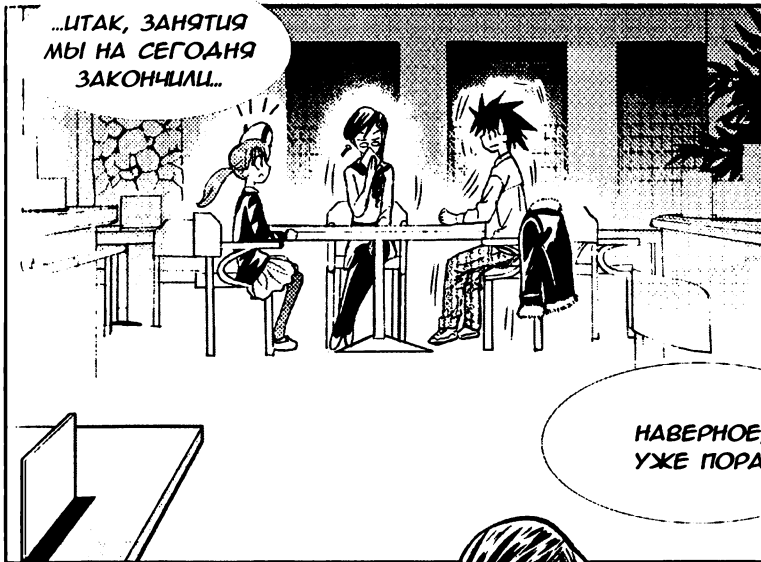


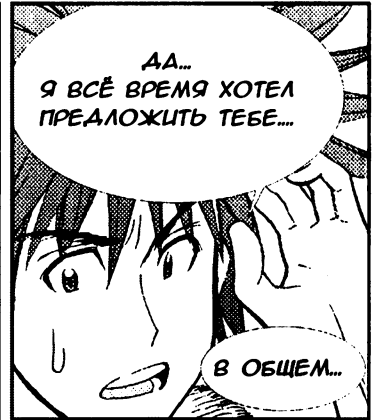
Да. Очень важно постоянно думать о рационализации, не идя ни на какие компромиссы.

Ах, нельзя ли мне тоже уменьшить количество работников, чтобы сократить расходы на персонал? Хи-хи!



Нет, такие рассуждения мне не нравятся!







ЧТО?!!

ТАКАБА,
НЕЛЬЗЯ ЖЕ
СРАЗУ ДОМОЙ
ПРИГЛАШАТЬ!

А МОЖЕТ БЫТЬ,
МОЖЕТ БЫТЬ,
Я ЕМУ ПРАВЛЮСЬ?!



ММ...

АИСАКА-САН,
ТЫ МЕНЯ
СЛЫШИШЬ?

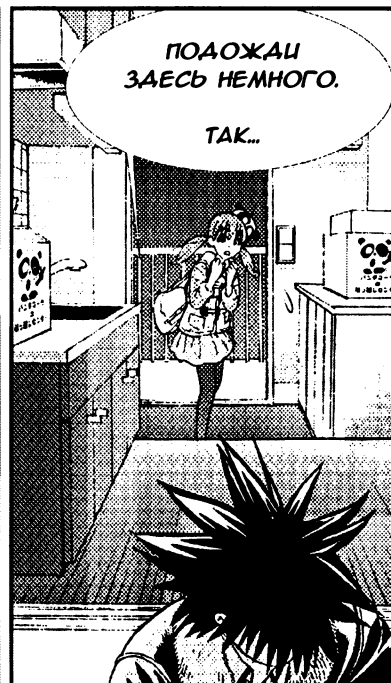


ОЙ!!

АА, Я ПРОСТО
ЗАДУМАЛАСЬ.
КУДА ЭТО МЫ
ПРИШЛИ?!



ЭТО МОЙ ДОМ.



ПОАЖАИ
ЗАЕСЬ НЕМНОГО.

ТАК...



КОМНАТА ТАКАБЫ...
АА, ПРОСТЕНЬКАЯ
ОБСТАНОВКА...

ВОТ!

Я ДАВНО ХОТЕЛ
ПРЕДЛОЖИТЬ ТЕБЕ
ВСЮ ЭТУ ТЕХНИКУ.
МОЖЕШЬ РАЗБИРАТЬ
ЕЁ, КАК ЗАХОЧЕШЬ.

НО ЗАТО
ЧИСТО И
АККУРАТНО!



ЧТО?
ВСЁ, ЧТО
ЗАЕСЬ ЕСТЬ?
ТЕЛЕВИЗОР,
МИКРОВОЛНОВКУ?

АА...
ПРИДЁТСЯ МНЕ
ВЫЛОЖИТЬ ВСЁ
НАЧИСТОТУ.



ЭТОЙ ВЕСНОЙ Я
ЗАКАНЧИВАЮ ИНСТИТУТ
И ВОЗВРАЩАЮСЬ ДОМОЙ.
АА, Я САМ С ХОККАЙДО.

ПОЭТОМУ Я
ХОТЕЛ ИЗБАВИТЬСЯ
ОТ ЭТОЙ ТЕХНИКИ.

ТАААА!



АИСАКА-САН,
ТЫ ЧТО?

А? НЕТ, НИЧЕГО.
Я ОЧЕНЬ РАДА
ТВОЕМУ ПОДАРКУ.
БОЛЬШОЕ СПАСИБО...



ААА!

ГОРИ ВСЁ СИНИМ ПЛАМЕНЕМ!!

ВСЁ ВЕЧЕР
АИСАКА-САН ОТЧАЯННО
РАЗБИРАЛА ПОДАРЕННУЮ
ТЕХНИКУ.

■ Сумматор

В этой главе мы изучили методы проектирования простейших комбинационных схем.

Однако схемы, выполняющие, например, сложение и вычитание двоичных чисел, проектируются на основе совсем других принципов. Дело в том, что количество входов у них слишком велико для использования обычных методов проектирования. Например, для реализации сложения двух 8-битных чисел потребуется 16 входных линий, то есть количество возможных комбинаций входных сигналов составит $2^{16} = 65536$, и проанализировать их вручную становится чрезвычайно трудной задачей.

Поэтому такие комбинационные схемы, имеющие слишком много входных линий, проектируют на основе регулярности, то есть повторяемости некоторых общих закономерностей.

Например, сложение двух 4-битных чисел вручную выполняется как обычно — в столбик, как показано внизу для чисел 4 (0100) и 5 (0101).

$$\begin{array}{r}
 4+5 \\
 100 \text{ ← Переносы} \\
 0100 \text{ ----> 4} \\
 +) 0101 \text{ ----> 5} \\
 \hline
 1001 \text{ ----> 9}
 \end{array}$$

Вход переноса из суммы предыдущих битов
↓
Выход переноса в сумму следующих битов

Рассмотрим элементарный этап сложения. Здесь мы должны, сложив значения трёх битов: входа А, входа В и переноса из суммы предыдущих битов, найти значения двух битов: суммы и переноса в сумму следующих битов.

Совмещённая для обоих выходов таблица истинности для этой операции показана слева. Здесь перенос из суммы предыдущих битов обозначен C_{in} , сумма — S, перенос в сумму следующих битов — C_{out} .

A	B	C_{in}	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



Из этой таблицы истинности можно заметить, что выход C_{out} будет равен 1 тогда, когда не менее двух из трёх входов равны 1, а это — та самая мажоритарная схема, которую уже мы изучили (см. стр. 96). Что же касается той части схемы, которая вычисляет сумму S , то она, к сожалению, не поддаётся упрощению даже методом построения карты Карно.

Схема, реализующая эту таблицу истинности, будет следующей.

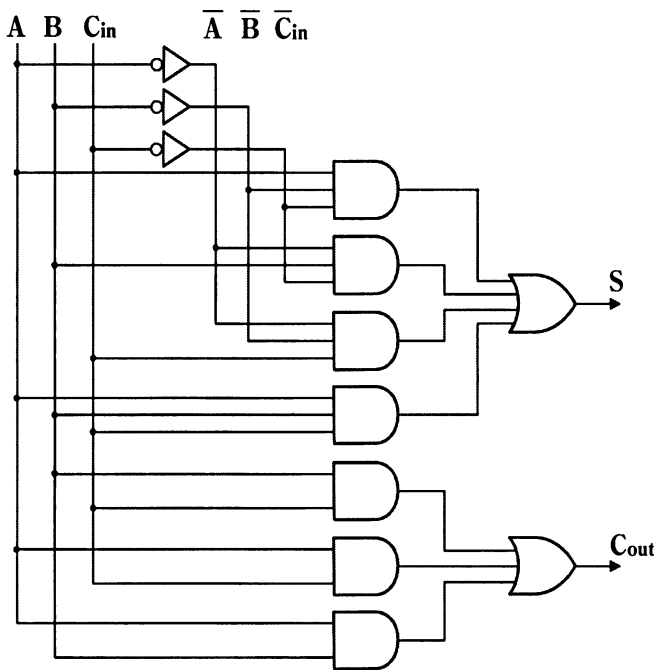


Рис. 1. Полный сумматор

Эту схему называют «полным сумматором». Для сложения двух многобитных чисел несколько таких схем соединяют в цепочку, подключая выход переноса одного полного сумматора к входу переноса следующего.

На рис. 2 на следующей странице приведена структурная схема четырёхбитного сложения (4-битного сумматора), в котором полные одноразрядные сумматоры для упрощения показаны в виде прямоугольников.

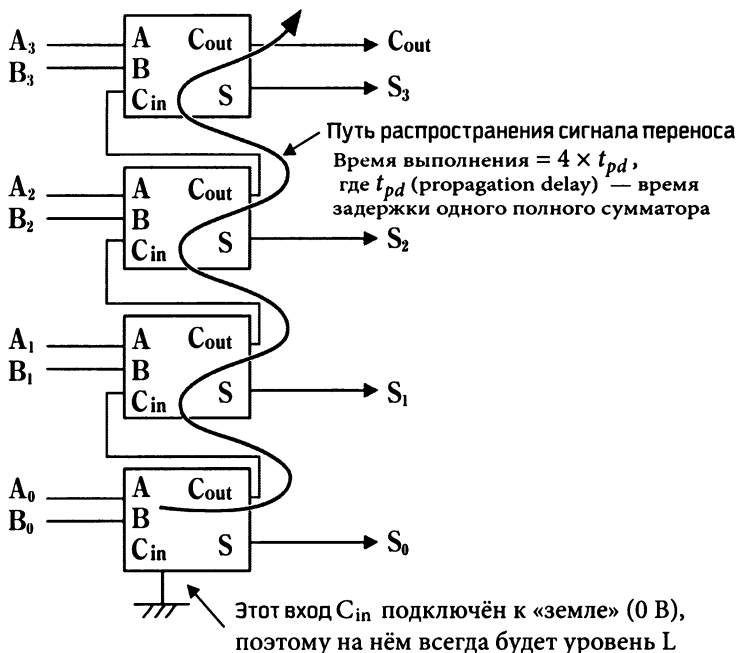


Рис. 2. Сумматор со сквозным переносом (каскадный сумматор)

При взгляде на эту схему можно понять, что она выполняет сложение двух двоичных чисел точно так же, как мы делаем это «в столбик» на бумаге.

По-английски каскадный сумматор называется «ripple carry adder», так как перенос в нём распространяется от бита к биту, подобно «ряби на водной поверхности» (англ. ripple — «рябь»). Таким образом, использование регулярности подразумевает проектирование части схемы, реализующей общие закономерности, и повторение этой части для всех входных линий данной схемы.

Каскадный сумматор обладает одним существенным недостатком: при увеличении разрядности складываемых двоичных чисел увеличивается суммарное время задержки, что приводит к снижению быстродействия. В связи с этим предложено множество схем, ускоряющих распространение переноса, но они, в свою очередь, требуют использования большего количества логических вентилей.

В связи с этим при проектировании цифровых схем иногда приходится искать оптимальный баланс между подобными несовместимыми свойствами на основе компромисса (англ. tradeoff).

В реальном проектировании производится выбор такого варианта реализации, который по быстродействию, количеству вентилей, типам используемых микросхем и электрических цепей и т. п. лучше всего подходит решаемым задачам. В интеллектуальные системы проектирования САД заложен набор различных компромиссных вариантов сумматора, выбор из которых производится автоматически в соответствии с поставленной задачей.

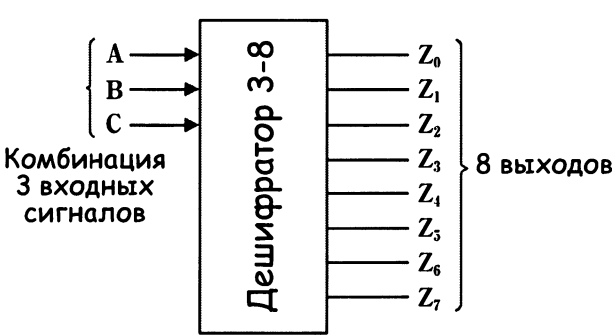
Не только сумматоры, но и другие «регулярные» комбинационные схемы с большим количеством входных линий зачастую не проектируются с нуля, а выбираются из нескольких уже существующих вариантов. Наиболее типичные примеры таких комбинационных схем в общих чертах описаны на следующей странице.

◆ Дешифратор

Схема, отображающая на выходах текущую комбинацию входных сигналов.

Взгляните на нижеприведённую схему, в которой имеется 3 входа, то есть $2^3 = 8$ возможных комбинаций входных сигналов, и количество выходов тоже равно 8. Эту схему называют «дешифратором 3-8». В ней сигнал 1 (или 0) подаётся на выход, номер которого соответствует текущей комбинации сигналов на входах.

Существует множество видов дешифраторов, отличающихся количествами входов, выходов, кодовыми комбинациями.



Входы			Выходы							
С	В	А	Z ₀	Z ₁	Z ₂	Z ₃	Z ₄	Z ₅	Z ₆	Z ₇
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

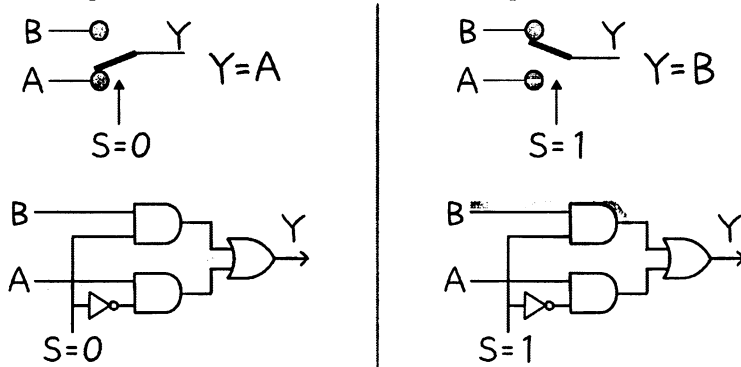
◆ Приоритетный шифратор

Эта схема выполняет функцию, почти обратную дешифратору, генерируя выходной код, соответствующий состоянию входных линий.

Отличие от дешифратора состоит в том, что при одновременной подаче сигналов на несколько входных линий код будет сгенерирован для той из них, у которой самый высокий приоритет.

◆ Мультиплексор

Схема, выбирающая один из нескольких информационных входов и передающая выбранные данные на выход. Мультиплексоры используются для переключения путей передачи данных. Ниже показаны схемы и принцип работы мультиплексора с двумя информационными входами А и В, управляющим входом S и выходом Y. В нём при S = 0 на выход Y передаётся сигнал только с входа А, а при S = 1 — только с входа В.



■ Вычитатель

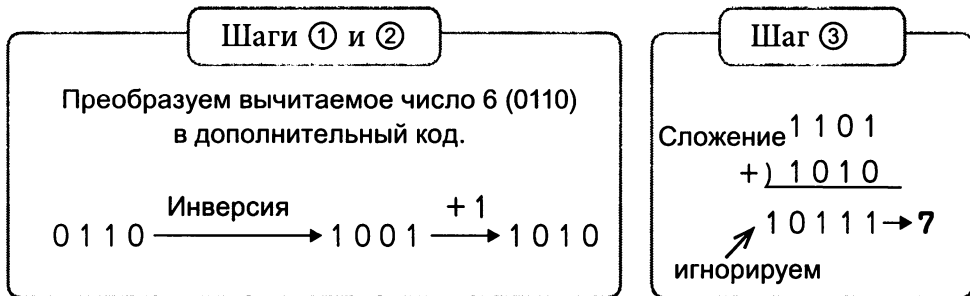
Так же, как и в случае со схемой сумматора, схему вычитателя можно построить в соответствии с алгоритмом вычитания двоичных чисел вручную «в столбик».

Однако в действительности вычитатели проектируют, немного изменяя схемы сумматоров.

Дело в том, что найти разность двоичных чисел ($A - B$) можно следующим методом, состоящим из трёх шагов.

- ① Инвертировать все биты числа B .
- ② Прибавить 1 к результату шага ① (так называемый дополнительный код).
- ③ Прибавить результат шага ② к числу A , игнорируя перенос из старшего бита.

Давайте попробуем найти разность двоичных чисел 13 (1101) и 6 (0110) этим методом, чтобы самим убедиться в том, что мы получим правильный результат, равный 7 (0111).



Только не подумайте, что правильный результат получился случайно!

Секрет здесь заключается в том, что на шагах ① и ② мы нашли для двоичного числа B его дополнительный код (или так называемое «поразрядное дополнение до двойки» — число, при сложении которого с B мы получим двоичное 10000).

Дополнительный код числа B выполняет роль «числа B со знаком минус» ($-B$).

Теперь мы подумаем о том, как реализовать этот метод с помощью сумматора.

Шаг ① очень прост — достаточно использовать вентили NOT. На шаге ② может показаться, что нужен ещё один сумматор, прибавляющий 1 к числу, но мы обойдём эту проблему с помощью одной хитрости. Дело в том, что в самом младшем разряде сумматора вход переноса не используется — на него всегда подаётся 0, но если вместо 0 всегда подавать на него 1, то это будет равносильно прибавлению 1.

Полученная вышеуказанным способом схема вычитателя (иногда называемая разностным сумматором) показана на рис. 3 следующей страницы.

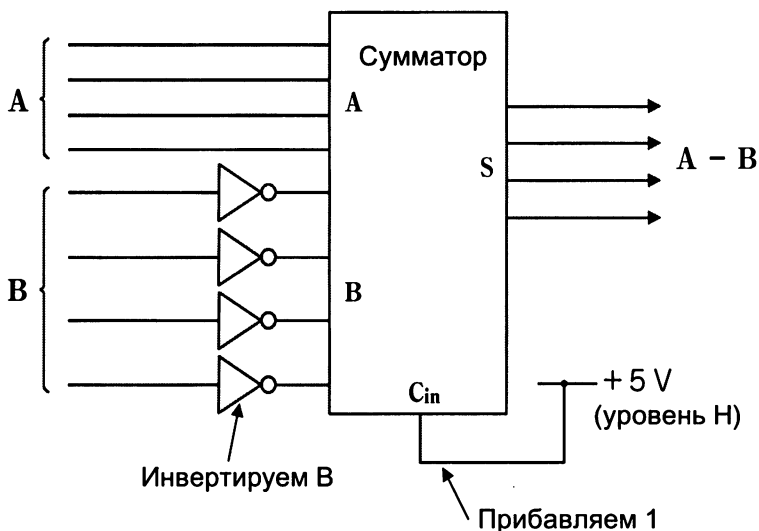


Рис. 3. Вычитатель

Если дополнить эту схему функцией переключения, то можно будет использовать её и как сумматор, и как вычитатель. А если, в дополнение к этому, снабдить эту схему логическими функциями: AND, OR и т. п. ; а также функциями поразрядного сдвига, то мы получим полноценное АЛУ (арифметико-логическое устройство).

АЛУ представляет собой вычислительную схему, позволяющую выбирать выполняемые операции, и является основной частью ЦПУ (центрального процессорного устройства).

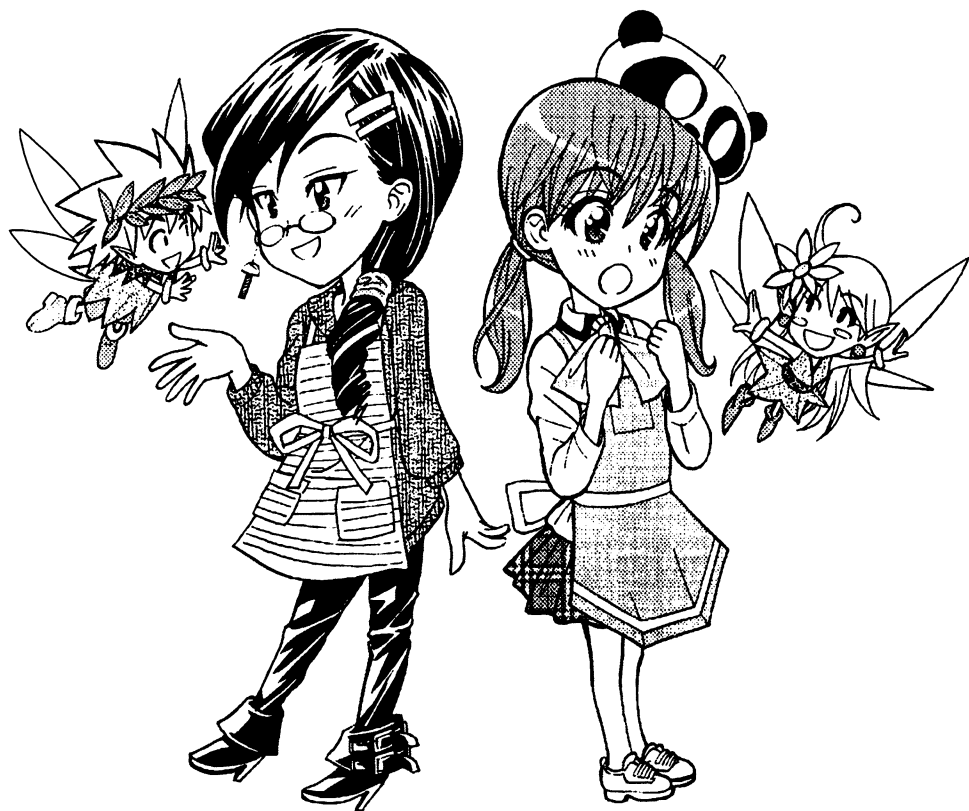
Кстати, умножение и деление — достаточно сложные операции, поэтому во многих случаях они выполняются вне АЛУ в специально предназначенных для этого схемах.

Объяснение терминов главы 4

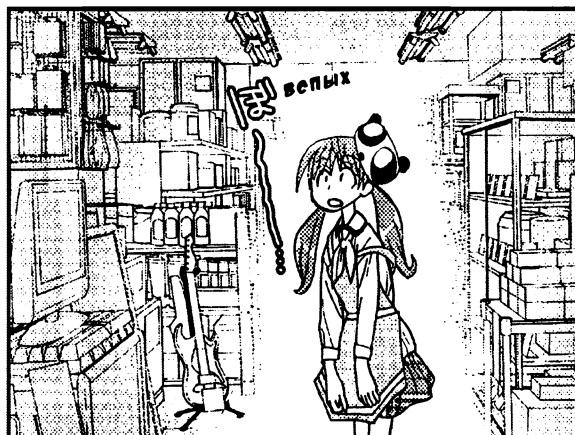
- **Карты Карно** — метод упрощения логических выражений, изобретённый Морисом Карно. В наши дни широко используется для обучения проектированию логики, так как позволяет интуитивно освоить основные принципы упрощения. При количестве входов больше 5 необходимо использовать трёхмерную карту, поэтому предел применимости метода — 6 входов. Существуют и другие методы упрощения: непосредственное преобразование логических выражений, метод Куайна—Мак-Класки и т. д. В реальных условиях проектирование логики осуществляется с помощью языков описания аппаратуры, о которых пойдёт речь в главе 5, после чего схемы генерируются и упрощаются в САД. В связи с этим упрощение схем вручную вышеуказанными методами в настоящее время практически не проводится.
- **Кодирование** — представление любой информации в виде кодов: различных комбинаций нулей и единиц, необходимое потому, что в цифровых схемах не используется никаких сигналов, кроме уровней L и H. Например, все слышали про коды ASCII, используемые для представления букв, цифр и символов в виде 8-разрядных (8-битных) комбинаций единиц и нулей.
- **Двоичные числа** — основополагающий метод представления чисел в виде комбинаций нулей и единиц. Один разряд двоичного числа называют «битом», а восемь бит — «байтом». Для записи больших величин в виде двоичных чисел требуется много разрядов, что ухудшает читабельность, поэтому для наглядного представления в текстовом виде иногда вместо них используются шестнадцатеричные или восьмеричные числа.
- **Безразличное состояние** — состояние, возникающее при вводе запрещённых (невозможных) значений, в котором на выход можно подавать любое значение. В этой книге мы обозначаем его прочерком, но может использоваться также буква X.

ГЛАВА 5

СТРОИМ ПОСЛЕДОВАТЕЛЬНЫЕ СХЕМЫ



1. ЧТО ТАКОЕ ПОСЛЕДОВАТЕЛЬНЫЕ СХЕМЫ





...ВОТ КАК?
И МЕНЯ ТЫ ВЗЯЛА
НА РАБОТУ ТОЖЕ
ПОТОМУ, ЧТО ЗНАЛА
ОБ ЭТОМ?

О ТОМ, ЧТО
ТАКАБА
ЗАКАНЧИВАЕТ
ИНСТИТУТ И
УЕЗЖАЕТ?

...НУ, В ОБЩЕМ,
И ЕЩЁ. ЭТОТ АУРАК
ПРОСИЛ ТЕБЕ
ПЕРЕДАТЬ...!



ЕМУ, ВИАЛТЕ ЛЦ, ОЧЕНЬ
НЕУДОБНО ПЕРЕД ТОБОЮ
ЗА ТО, ЧТО
НЕ СМОГ ДО КОНЦА
РАССКАЗАТЬ ТЕБЕ ПРО
ЦИФРОВЫЕ СХЕМЫ.

А МЕНЯ, ДИРЕКТОРА, ОН
ЗА КОГО ПРИНИМАЕТ?!
КАК БУАТО Я САМА
НЕ МОГУ!

НУ ЧТО, БУДЕМ
ПРОДОЛЖАТЬ?
ИЛИ...



БРОСИТЬ НА ПОЛПУТИ
ИЗУЧЕНИЕ ЦИФРОВЫХ
СХЕМ, О КОТОРЫХ
ТАКАБА МНЕ ТАК
УСЕРДАНО РАССКАЗЫВАЛ?

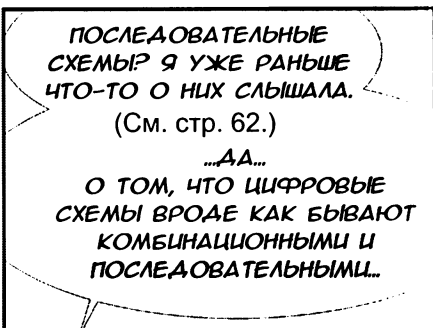
НЕТ!



КОНЕЧНО, Я
ХОЧУ ПРОДОЛЖАТЬ!!

РАЗВЕ МОЖНО
БРОСИТЬ ВСЁ
НА ПОЛПУТИ!!!

У последовательных схем есть память



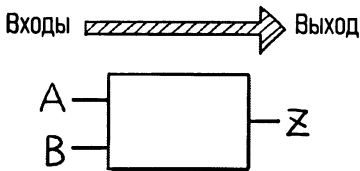
Комбинационные

Последовательные

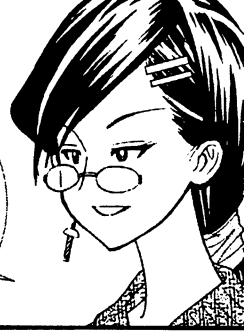


Будем изучать теперь!

Комбинационные схемы



ДА.
В КОМБИНАЦИОННЫХ
СХЕМАХ, КОТОРЫЕ МЫ
ИЗУЧАЛИ ДО ЭТОГО,
УРОВЕНЬ НА ВЫХОДЕ
ОПРЕДЕЛЯЕТСЯ
ТОЛЬКО ТЕКУЩИМИ
УРОВНЯМИ НА ВХОДАХ.

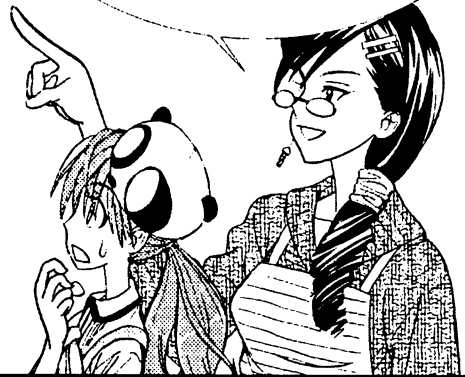


Последовательные схемы



Текущее внутреннее состояние изменяется на следующее под действием входных сигналов.

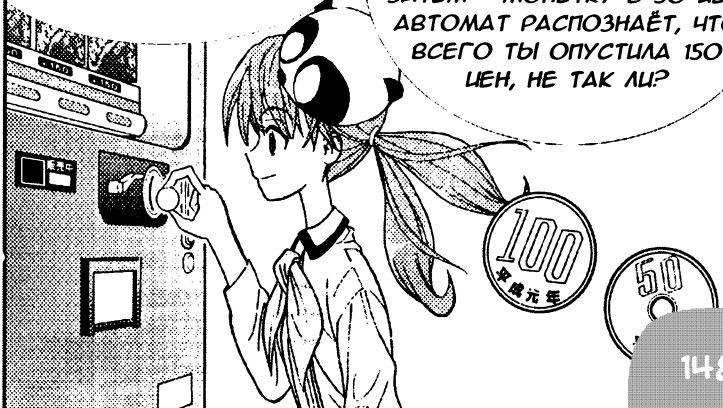
В ПОСЛЕДОВАТЕЛЬНЫХ СХЕМАХ
ТЕКУЩИЕ УРОВНИ НА ВХОДАХ И
ПАМЯТЬ (ВНУТРЕННЕЕ СОСТОЯНИЕ
СХЕМЫ, ЗАВИСЯЩЕЕ ОТ ПРОШЛЫХ
УРОВНЕЙ НА ВХОДАХ) ОПРЕДЕЛЯЮТ
УРОВЕНЬ НА ВЫХОДЕ И СЛЕДУЮЩЕЕ
ВНУТРЕННЕЕ СОСТОЯНИЕ СХЕМЫ!



ММ...
ПАМЯТЬ?
ВНУТРЕННЕЕ
СОСТОЯНИЕ?
ВСЁ, СААЮСЬ...

ХОРОШО, ПРИВЕДУ
ПРИМЕР ИЗ ЖИЗНИ.

ПОКУПАЯ СОК В АВТОМАТЕ,
ТЫ СНАЧАЛА ОПУСКАЕШЬ
ТУДА МОНЕТКУ В 100 ЦЕН, А
ЗАТЕМ - МОНЕТКУ В 50 ЦЕН.
АВТОМАТ РАСПОЗНАЁТ, ЧТО
ВСЕГО ТЫ ОПУСТИЛ 150
ЦЕН, НЕ ТАК ЛИ?



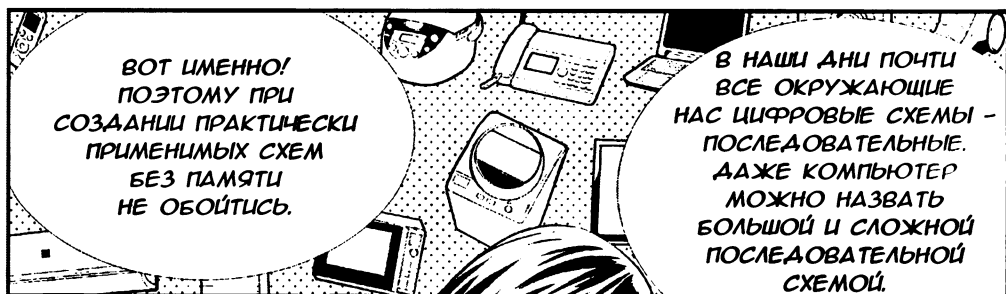


ЭТО ПОТОМУ,
ЧТО АВТОМАТ
ЗАПОМИНАЕТ ПРОШЛОЕ:
ОН ПОМНИТ, ЧТО ТЫ
УЖЕ ОПУСТИЛА
В НЕГО 100 ЦЕН.



АА,
В САМОМ ДЕЛЕ!
ВЕДЬ ЕСЛИ БЫ ОН
НЕ ЗАПОМИНАЛ,
ТО БЫЛА БЫ
НЕПРИЯТНАЯ
СИТУАЦИЯ!

ЧТО ЗА
ШУТОЧКИ?
ВЕРНИТЕ МОИ
ДЕНЬГИ!
НИКАКОГО
ТОЛКУ ОТ
ЭТИХ
АВТОМАТОВ...



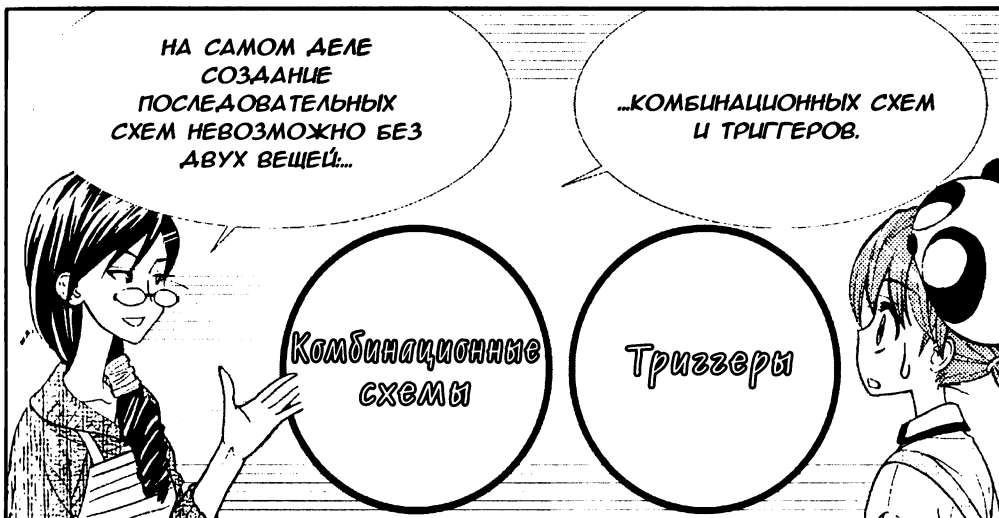
ВОТ ИМЕННО!
ПОЭТОМУ ПРИ
СОЗДАНИИ ПРАКТИЧЕСКИ
ПРИМЕНИМЫХ СХЕМ
БЕЗ ПАМЯТИ
НЕ ОБОЙТСЯ.

В НАШИ ДНИ ПОЧТИ
ВСЕ ОКРУЖАЮЩИЕ
НАС ЦИФРОВЫЕ СХЕМЫ -
ПОСЛЕДОВАТЕЛЬНЫЕ.
ДАЖЕ КОМПЬЮТЕР
МОЖНО НАЗВАТЬ
БОЛЬШОЙ И СЛОЖНОЙ
ПОСЛЕДОВАТЕЛЬНОЙ
СХЕМОЙ.



ТА-ААМ!

ПОЭТОМУ ТОТ,
КТО ОВЛАДЕЕТ
ПОСЛЕДОВАТЕЛЬНЫМИ
СХЕМАМИ,
ОВЛАДЕЕТ
ЦИФРОВЫМИ!

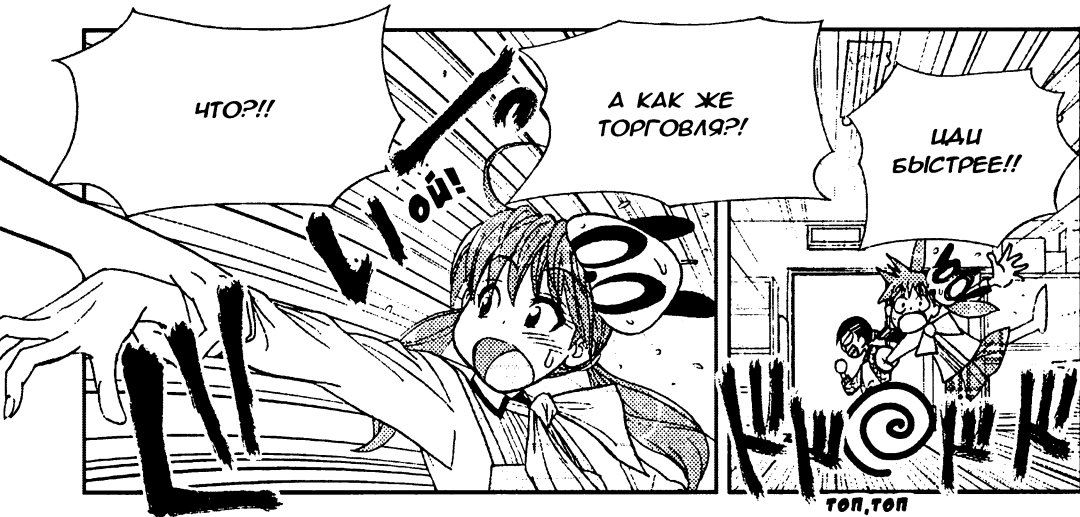




НЕТ...

...ЛУЧШЕ Я ОБЪЯСНЮ
ЭТО НА СВЕЖЕМ ВОЗДУХЕ.
ИДУМО, ПОЙДЁМ!

ХВАТЬ

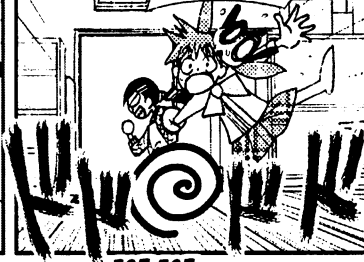


ЧТО?!!

А КАК ЖЕ
ТОРГОВЛЯ?!

ЦАЦ
БЫСТРЕЕ!!

Ой!



ТОП, ТОП

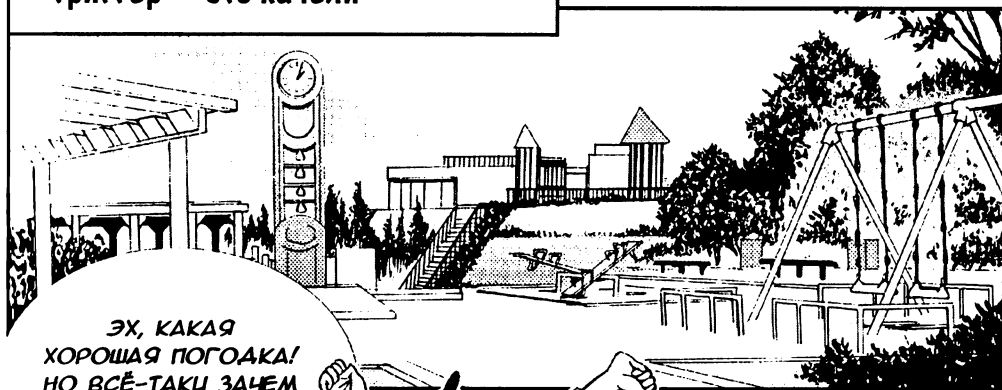


МЫ ПОШЛИ
гулять!
Скоро вернёмся ♡



2. D-ТРИГГЕРЫ

Триггер — это качели



ЭХ, КАКАЯ
ХОРОШАЯ ПОГОДА!
НО ВСЁ-ТАКИ ЗАЧЕМ
МЫ ПРИШЛИ В ПАРК?

НА САМОМ ДЕЛЕ
АНГЛИЙСКОЕ НАЗВАНИЕ
ТРИГГЕРА - FLIP-FLOP,
ПРОИЗОШЛО ОТ
"ХЛОП-ХЛОП" -
РЕЗКОГО СТУКА
КАЧЕЛЕЙ.

О слове «flip-flop» см. также «Объяснение терминов» на стр. 193.

АА?!
КАЧЕЛЕЙ?
ЭТО НАВЕВАЕТ
ВОСПОМИНАНИЯ
О ДЕТСТВЕ.

ТЕПЕРЬ О ГЛАВНОМ!
ТРИГГЕР - ЭТО СХЕМА,
ЗАПОМИНАЮЩАЯ ЦИФРОВОЙ
СИГНАЛ РАЗМЕРОМ В 1 БИТ
(0 ИЛИ 1).

ЭТО ВАЖНО!!

Ох...

ВОСПОМИНАНИЯ
О ДЕТСТВЕ
ЗАКОНЧИЛИСЬ?

X" — ' —
ТУАХ



НЕТ, КАЧЕЛИ —
ХОРОШАЯ АНАЛОГИЯ.
КОГДА НА НИХ
НЕ КАЧАЮТСЯ,
ОНИ НАКЛОНЕНЫ
ЛИБО ВПРАВО,
ЛИБО ВЛЕВО,
НЕ ТАК ЛИ?

И ТРИГГЕР ТОЖЕ
ЗАПОМИНАЕТ
ЛИБО 0, ЛИБО 1.

Последнее приложение
веса — на правое плечо

Запомнили 0



Последнее приложение
веса — на левое плечо

Запомнили 1



АГА, ЯСНО.

ЗАПОМНИТЬ 1 БИТ —
ЗНАЧИТ ЗАПОМНИТЬ
ЛИБО 0, ЛИБО 1.

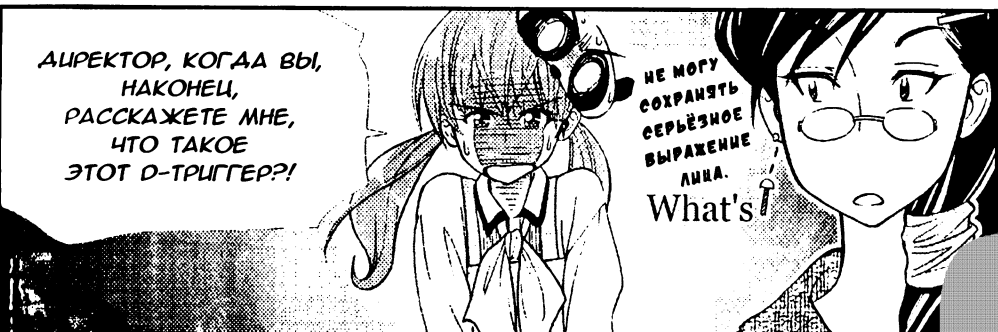
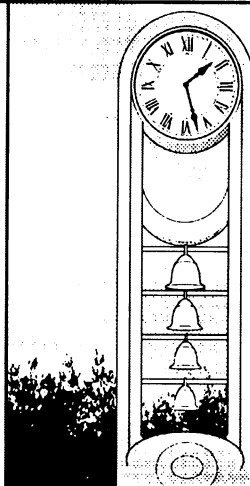
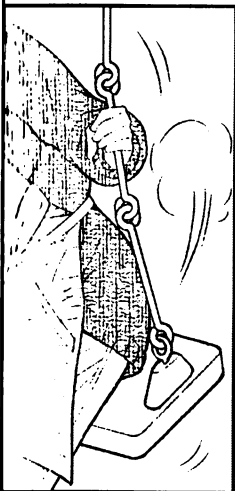
ДЕЙСТВИТЕЛЬНО,
ОЧЕНЬ НАПОМИНАЕТ
КАЧЕЛИ!

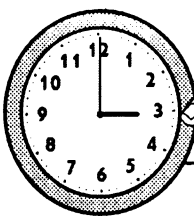
ДА.
НО КРОМЕ ТАКИХ ПРОСТЫХ
УСТРОЙСТВ, ЕСТЬ И БОЛЕЕ
ВАЖНЫЕ ТРИГГЕРЫ,
КОТОРЫЕ ТЕБЕ НУЖНО ОСВОИТЬ.

ЭТО —
0-ТРИГГЕРЫ!!!



D-триггер и тактовый сигнал





ЧАСЫ

D-FF —
самые удобные!

АА, НА САМОМ ДЕЛЕ
ЕСТЬ РАЗНЫЕ ТИПЫ ТРИГГЕРОВ
(СОБЪЕДИНЯЮТСЯ FF
ОТ АНГЛИЙСКОГО FLIP-FLOP),
НО ОСОБЕННО УДОБНЫ
D-ТРИГГЕРЫ.

Другие FF

JK-FF T-FF



В СОВРЕМЕННЫХ ЦИФРОВЫХ
СХЕМАХ ИСПОЛЗУЮТСЯ
ТОЛЬКО ОНИ.

D-ТРИГГЕРЫ УМЕЮТ
ЗАПОМИНАТЬ ДАННЫЕ
ОАНОВРЕМЕННО, ИЛИ,
КАК ГОВОРЯТ, СИНХРОННО
С ПЯВЛЕНИЕМ ОСОБОГО
ЦИФРОВОГО СИГНАЛА,
КОТОРЫЙ НАЗЫВАЕТСЯ
"ТАКОВЫМ СИГНАЛОМ"
(CLK).



CLOCK?
ПО-АНГЛИЙСКИ
ЭТО ОЗНАЧАЕТ
"ЧАСЫ"?



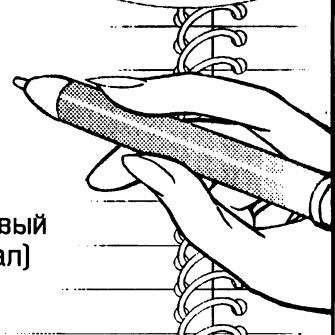
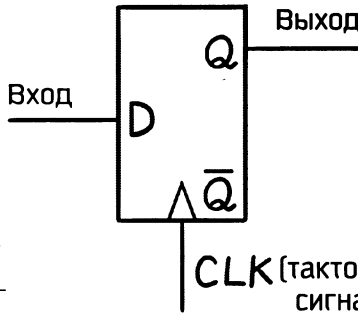
АА,
ПРЕЖДЕ ВСЕГО
ВЗГЛЯНИ ВОТ
НА ЭТО.

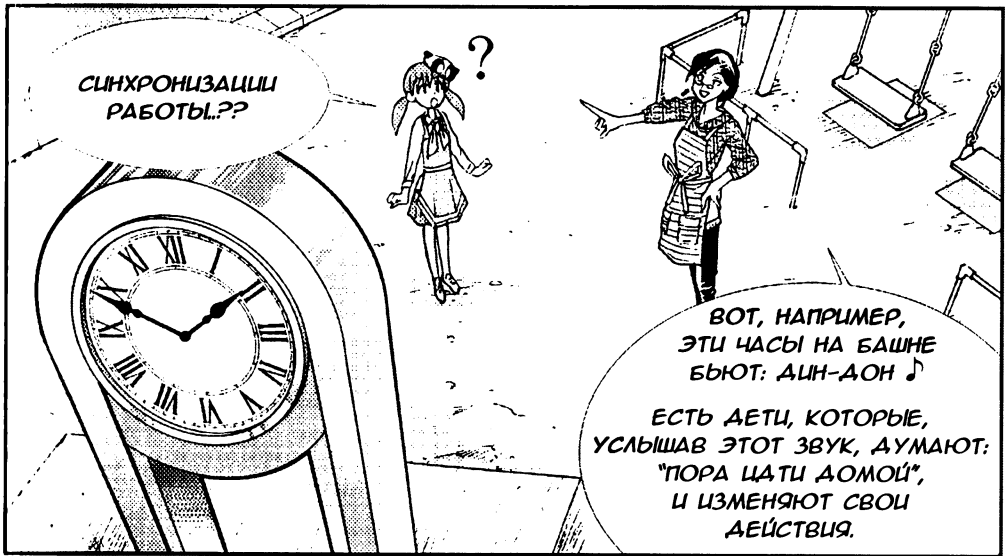


ВОТ ТАК НА ЧЕРТЕЖАХ
ИЗОБРАЖАЕТСЯ D-ТРИГГЕР
(СОКРАЩЁННО - D-FF).

У D-ТРИГГЕРА
ЕСТЬ "ВХОД D"
И "ВЫХОД Q".
ЕСТЬ ЕЩЁ
"УНИВЕРСАЛЬНЫЙ ВЫХОД Q̄",
НО ПОКА НА НЕГО
ОБРАЩАТЬ ВНИМАНИЯ
НЕ НУЖНО.

* Будет использоваться,
например, на стр. 185.







КОЕ-КАК
ПОНЯЛА!

НО ВЕДЬ
ЦИФРОВЫЕ СХЕМЫ -
ЭТО МИР, В КОТОРОМ
НЕТ НИЧЕГО,
КРОМЕ L И H (0 И 1)?
КАК ТАМ МОЖНО ВЫРАЗИТЬ
ЭТОТ "АИЛЬ-ДОН"?

ХОРОШИЙ ВОПРОС!



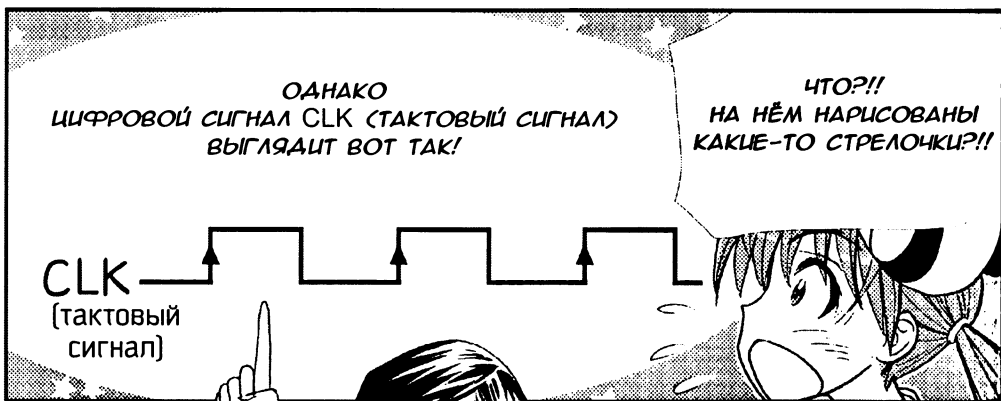
ТЫ ВЕДЬ ПОМНИШЬ,
ЧТО ЦИФРОВОЙ СИГНАЛ МОЖЕТ БЫТЬ
ТОЛЬКО L ИЛИ H (0 ИЛИ 1).
ПОЭТОМУ УРОВНИ И ВХОДНЫХ,
И ВЫХОДНЫХ СИГНАЛОВ -
ЭТО ТОЖЕ ТОЛЬКО L ИЛИ H!

Уровень H →

Уровень L →

ДА, Я ЭТО ПОМНЮ.
ЭТО МЫ УЖЕ
ПРОХОДИЛИ.

(*См. стр. 30 и стр. 67.)



ОДНАКО
ЦИФРОВОЙ СИГНАЛ CLK (ТАКТОВЫЙ СИГНАЛ)
ВЫГЛЯДИТ ВОТ ТАК!

CLK
(тактовый
сигнал)

ЧТО?!!
НА НЁМ НАРИСОВАНЫ
КАКИЕ-ТО СТРЕЛОЧКИ?!!



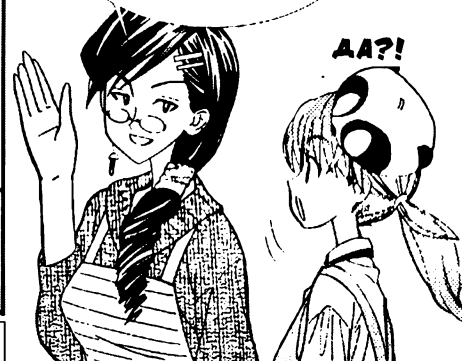
ДА, ИМЕННО
НА НИХ НУЖНО
ОБРАТИТЬ ВНИМАНИЕ!

МОМЕНТ ВРЕМЕНИ,
КОГДА УРОВЕНЬ ТАКТОВОГО
СИГНАЛА ИЗМЕНЯЕТСЯ С L НА H,
НАЗЫВАЮТ
"ПЕРЕДНИМ ФРОНТОМ"
(НАРАСТАЮЩИМ ФРОНТОМ)...

...А КОГДА С H НА L -
"ЗАДНИМ ФРОНТОМ"
(СПАДАЮЩИМ ФРОНТОМ).

Передний фронт	Задний фронт
<p>Уровень CLK изменяется с L на H</p>	<p>Уровень CLK изменяется с H на L*</p>

В НОТАЦИИ MIL
ЭТО ОБОЗНАЧАЕТСЯ
ВОТ ТАК.



АА?!

В России принято обозначать передний фронт треугольником, направленным к триггеру, а задний фронт — треугольником, направленным от триггера.



ОГО!
И НАЗВАНИЯ
ХОРОШИЕ!

НАРАСТАЮЩИЙ!

АА.
АА

СПАДАЮЩИЙ!



ТЕПЕРЬ ВЕРНЁМСЯ
К НАШИМ ЧАСАМ.
МЫ ГОВОРИЛИ,
ЧТО ОНИ БЬЮТ
"АЦНЬ-АОН",
НЕ ТАК ЛИ?

ПЕРЕДНИЙ (ЗАДНИЙ)
ФРОНТ СИГНАЛА CLK -
ЭТО И ЕСТЬ
БОЙ ЧАСОВ!



АГА,
Я ПОНЯЛА!

ПЕРЕДНИЙ ФРОНТ
ТАКТОВОГО СИГНАЛА
ИНИЦИИРУЕТ ИЗМЕНЕНИЯ
В РАБОТЕ СХЕМЫ!

АА, ИМЕННО!
А СЕЙЧАС Я
РАССКАЖУ
О САМОМ
ГЛАВНОМ!

ТУТУМ

О САМОМ
ГЛАВНОМ?
КАК ИНТРИГУЮЩЕ!

Я РАССКАЖУ О ТОМ,
КАК ИЗМЕНЯЕТСЯ РАБОТА СХЕМЫ
В ОТВЕТ НА ПЕРЕДНИЙ ФРОНТ
ТАКТОВОГО СИГНАЛА.

КХЕ

ПРЕЖДЕ ВСЕГО
Я ДОЛЖНА СКАЗАТЬ,
ЧТО ВНУТРИ D-ТРИГГЕРА...

...ЖИВУТ ДВЕ ФЕИ!

* Здесь использована аналогия с двумя феями,
но в действительности эта модель называется
Master-Slave (мастер и помощник).
Подробнее о ней будет рассказано на стр.187.



* Каждая из фей умеет выполнять оба вида работы и меняет вид деятельности при необходимости.





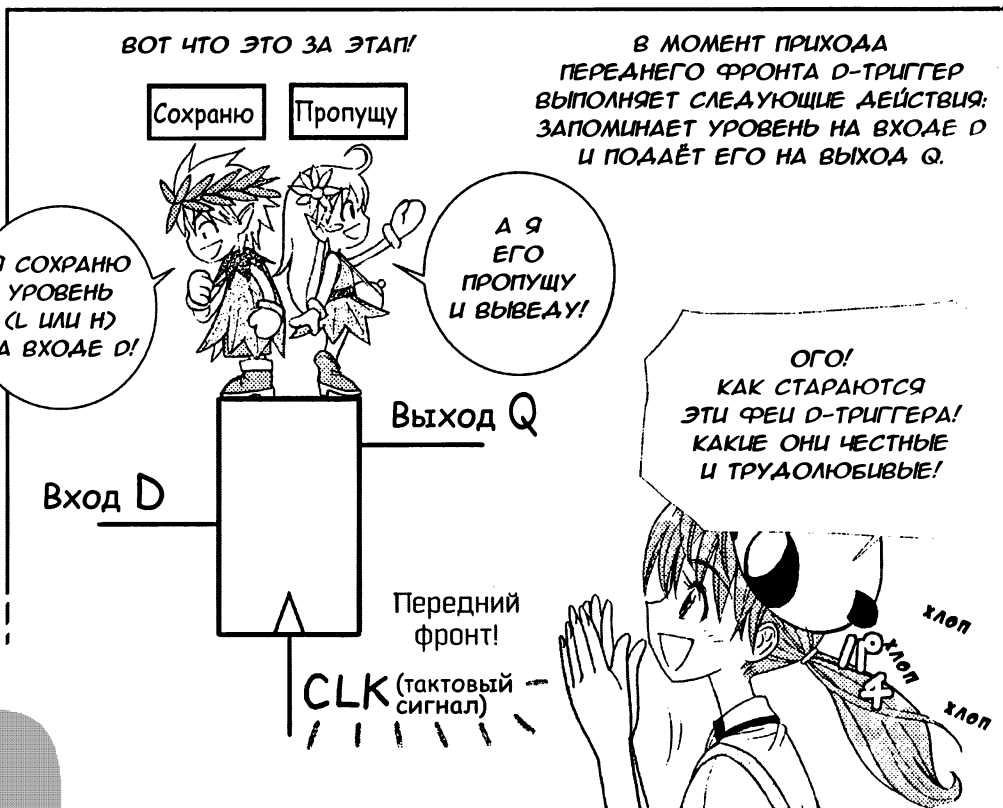
ХОТЯ D-ТРИГГЕР ПОСТОЯННО ФУНКЦИОНИРУЕТ БЛАГОДАРЯ НЕУСТААННОЙ РАБОТЕ ЭТИХ ДВУХ ФЕЙ...

...САМЫЙ ЯРКИЙ ЭТАП В ИХ СОВМЕСТНОЙ РАБОТЕ, БЛАГОДАРЯ КОТОРОМУ D-ТРИГГЕР ОКАЗЫВАЕТСЯ ЧРЕЗВЫЧАЙНО УДОБЕН В ЦИФРОВОЙ СХЕМОТЕХНИКЕ, ВСЕГДА ПРИУРОЧИВАЕТСЯ К МОМЕНТУ ПЕРЕДНЕГО (ИЛИ ЗАДНЕГО) ФРОНТА ТАКОВОГО СИГНАЛА.



ММ...

ЧТО ЖЕ ЭТО ЗА ЭТАП?



ВОТ ЧТО ЭТО ЗА ЭТАП!

Сохраню Пропущу

В МОМЕНТ ПРИХОДА ПЕРЕДНЕГО ФРОНТА D-ТРИГГЕР ВЫПОЛНЯЕТ СЛЕДУЮЩИЕ ДЕЙСТВИЯ: ЗАПОМИНАЕТ УРОВЕНЬ НА ВХОДЕ D И ПОДАЁТ ЕГО НА ВЫХОДЕ Q.

Я СОХРАНЮ УРОВЕНЬ (L ИЛИ H) НА ВХОДЕ D!

А Я ЕГО ПРОПУЩУ И ВЫВЕДУ!

ОГО! КАК СТАРАЮТСЯ ЭТИ ФЕЙ D-ТРИГГЕРА! КАКИЕ ОНИ ЧЕСТНЫЕ И ТРУДОЛЮБИВЫЕ!



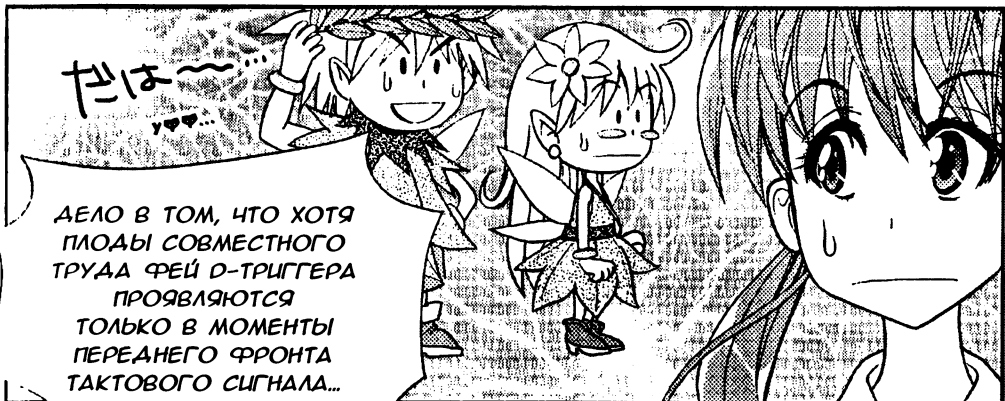


АА, РАБОТАЮТ
ОНИ ЧЕСТНО.

ИМЕННО
БЛАГОДАРЯ ЭТОМУ
В МОМЕНТ ПРИХОДА
ПЕРЕДНЕГО ФРОНТА
НА ВЫХОДЕ Q
ПОЯВЛЯЕТСЯ УРОВНЬ,
КОТОРЫЙ
В ЭТОТ МОМЕНТ
ПРИСУТСТВУЕТ
НА ВХОДЕ D!



ОДНАКО ВНУТРИ
ДЕЛО ОБСТОИТ
СЛОЖНЕЕ,
И СЕЙЧАС Я ТЕБЕ ОБ
ЭТОМ РАССКАЖУ!



だは〜...
うわ...

ДЕЛО В ТОМ, ЧТО ХОТЯ
ПЛОДЫ СОВМЕСТНОГО
ТРУДА ФЕЙ D-ТРИГГЕРА
ПРОЯВЛЯЮТСЯ
ТОЛЬКО В МОМЕНТЫ
ПЕРЕДНЕГО ФРОНТА
ТАКТОВОГО СИГНАЛА...

* О том, как две феи вместе выполняют работу, будет подробнее рассказано на стр.188.



ПОЖАЛУЙТЕ НА
СЛЕДУЮЩУЮ СТРАНИЦУ!

...ТАМ ВНУТРИ ОНИ НЕУСТААННО
РАБОТАЮТ - НЕ ЗА СТРАХ,
А ЗА СОВЕСТЬ.

ТОЛЬКО ВЗГЛЯНИ!

ПОКА НЕТ
ПЕРЕДНЕГО ФРОНТА
ТАКОВОГО СИГНАЛА,
НАШИ ФЕЙ ПОЛНОСТЬЮ
ИГНОРИРУЮТ ЛЮБЫЕ
ИЗМЕНЕНИЯ УРОВНЯ
СИГНАЛА НА ВХОДЕ D!

ОНИ НЕ СОХРАНЯЮТ
НОВЫЕ ДАННЫЕ И,
РАЗУМЕЕТСЯ,
НА ВЫХОДЕ Q ИХ
ТОЖЕ НЕ ПОДАЮТ.



Пропущу

Запомню

ПРОПУЩУ,
ВЕДЬ Я
ГОТОВЛЮСЬ
К ЗАПОМИНАНИЮ
ПО ПЕРЕДНЕМУ
ФРОНТУ.



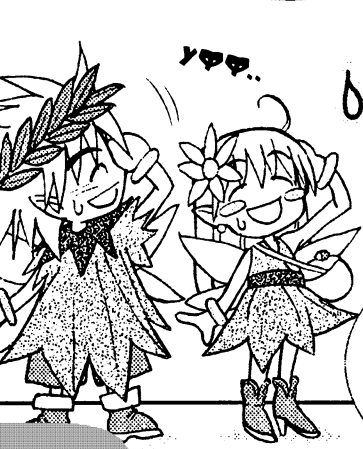
А Я
ЗАНЯТА ТЕМ,
ЧТО БЕРЕЖНО
ХРАНЮ
ДАННЫЕ.



Уровень сигнала
изменился!
Но наши феи
просто не
обращают
на это
внимания!

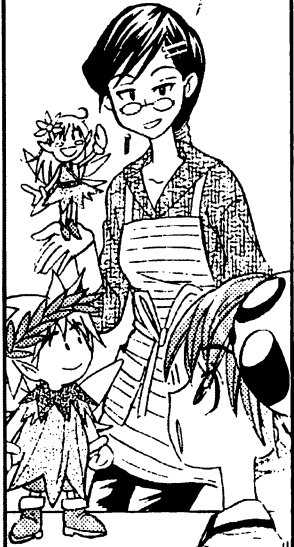
Нет переднего
фронта
CLK (тактовый
сигнал)

ОГО!
Я ПОНЯЛА!
ФЕЙ СОХРАНЯЮТ
НОВЫЕ ДАННЫЕ
ТОЛЬКО ПРИ ПОСТУПЛЕНИИ
ПЕРЕДНЕГО ФРОНТА
ТАКОВОГО ИМПУЛЬСА!



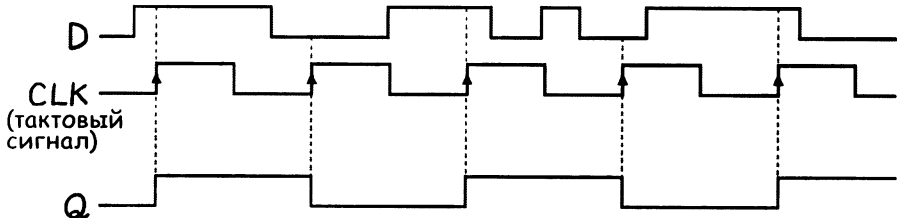
БЛАГОДАря ЭТОМУ
В ОСТАЛЬНОЕ ВРЕМЯ
ДАННЫЕ ХРАНЯТСЯ НАДЕЖНО,
И ТЕКУЩЕЕ СОСТОЯНИЕ
D-ТРИГГЕРА НЕ ИЗМЕНЯЕТСЯ.

АА.
ТАК РАБОТАЕТ
D-ТРИГГЕР.



ДАЛЕЕ,
НА ЭТОМ ГРАФИКЕ
ПРИВЕДЕН ПРИМЕР ИЗМЕНЕНИЙ
ВО ВРЕМЕНИ УРОВНЕЙ
СИГНАЛА НА ВХОДЕ D,
ТАКТОВОГО СИГНАЛА И
СИГНАЛА НА ВЫХОДЕ Q.

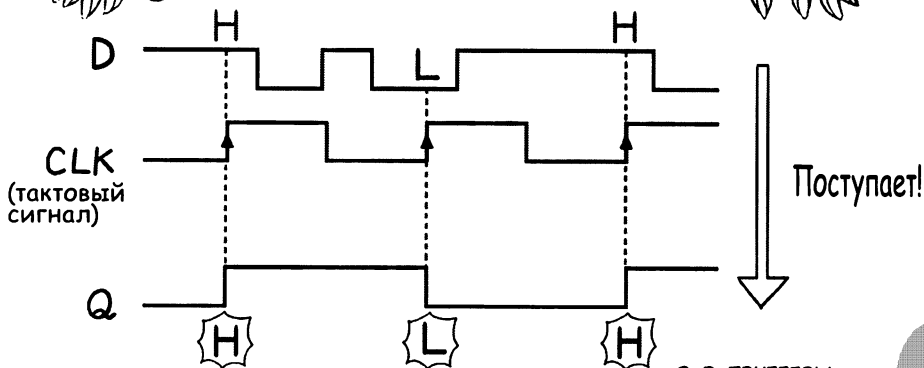
ПОДОБНЫЕ ГРАФИКИ
НАЗЫВАЮТСЯ
"ВРЕМЕННОЙ
ДИАГРАММОЙ".



<Временная диаграмма функционирования D-триггера>

ТАК...
КОГДА ПОСМОТРИШЬ
ВНИМАТЕЛЬНО,
СМЫСЛ СТАНОВИТСЯ
ОЧЕВИДЕН!

УРОВЕНЬ ВХОДА D
ПОСТУПАЕТ НА ВЫХОД Q
ТОЛЬКО ПО ПЕРЕДАНУМУ
ФРОНТУ! В ОСТАЛЬНОЕ
ВРЕМЯ УРОВЕНЬ НА ВЫХОДЕ Q
НЕ ИЗМЕНЯЕТСЯ!





ВЕРНО!
КОГДА ПОИМЁШЬ,
В ЧЁМ СМЫСЛ,
ВСЁ ОКАЗЫВАЕТСЯ
ОЧЕНЬ ПРОСТЫМ.



КСТАТИ,
У МЕНЯ ЕСТЬ
НЕСКОЛЬКО
НАИВНЫЙ
ВОПРОС...

ЧТО ОЗНАЧАЕТ
БУКВА "D"
В НАЗВАНИИ
D-ТРИГГЕРА?

D

А, ТЫ ОБ ЭТОМ?
"D" - ОТ СЛОВА Data ("ДАННЫЕ").
ХОТЯ ЕСТЬ ВЕРСИЯ, ЧТО
ОТ СЛОВА Delay ("ЗАДЕРЖКА").

Date
(Данные)

Delay
(Задержка)

✳ Соединив D-FF в цепочку, можно задержать изменение сигнала на количество тактовых импульсов, соответствующее количеству соединённых в цепочку D-FF. Первоначально D-FF часто использовались именно так.

ого!

ясно!
звучит логично
и легко
запомняется!



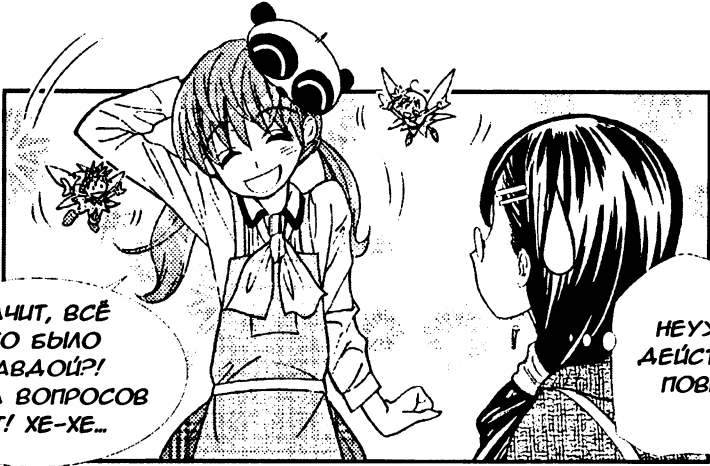
АА, ЕЩЁ ОДИН
ВОПРОС.
А КАК ВСЁ-ТАКИ
УСТРОЕН ЭТОТ
D-ТРИГГЕР
ВНУТРИ?

А ТО МЕНЯ
МУЧАТ
СОМНЕНИЯ...



ЦУМО,
ТЫ ЧТО?

Я ЖЕ ТЕБЕ
УЖЕ СКАЗАЛА!
СХЕМА ПРИВОДИТСЯ
В ДЕЙСТВИЕ
ТРУДОЛЮБИВЫМИ
ФЕЯМИ!




ЗНАЧИТ, ВСЁ
ЭТО БЫЛО
ПРАВОЙ?
ТОГДА ВОПРОСОВ
НЕТ! ХЕ-ХЕ...

НЕУЖТО ОНА
ДЕЙСТВИТЕЛЬНО
ПОВЕРИЛА?!!

* Об устоявие D-триггера будет рассказано в «Дополнительной информации» на стр.187.

Что такое регистр?



ХИ-ХИ ♪

СКРЭБ

СКРЭБ



ДИРЕКТОР, ЗАЧЕМ
ВЫ НАРИСОВАЛИ НА
ЗЕМЛЕ СОРОКОНОЖКУ?



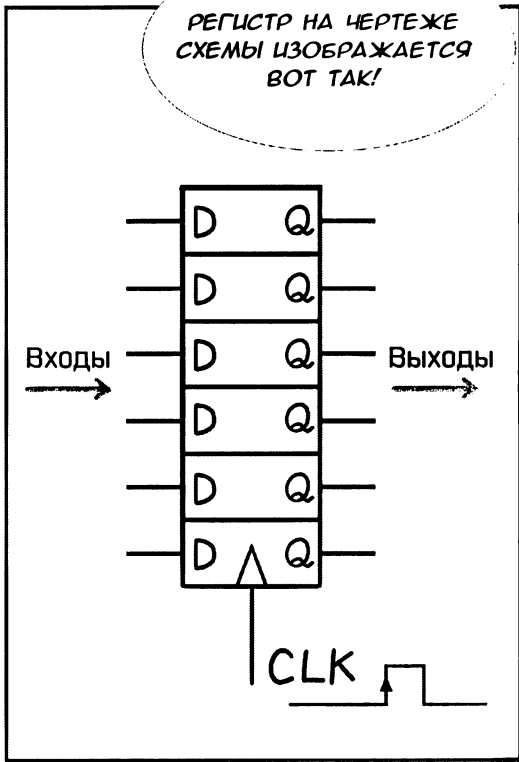
НАВЕРНОЕ,
СТРЕСС
СНИМАЕТЕ?



НЕТ!
ЭТО УСЛОВНОЕ
ОБОЗНАЧЕНИЕ
РЕГИСТРА!!




РЕГИСТР..?




КАК ВЦАНО ИЗ ЭТОГО
УСЛОВНОГО ОБОЗНАЧЕНИЯ,
РЕГИСТР - ЭТО СХЕМА,
СОСТОЯЩАЯ ИЗ НЕСКОЛЬ-
КИХ D-ТРИГГЕРОВ,
ОБЪЕДИНЁННЫХ ОБЩИМ
ТАКОВЫМ СИГНАЛОМ!



АГА...
ДЕЙСТВИТЕЛЬНО,
ОНИ ВЫСТРОЕНЫ
В РЯД.

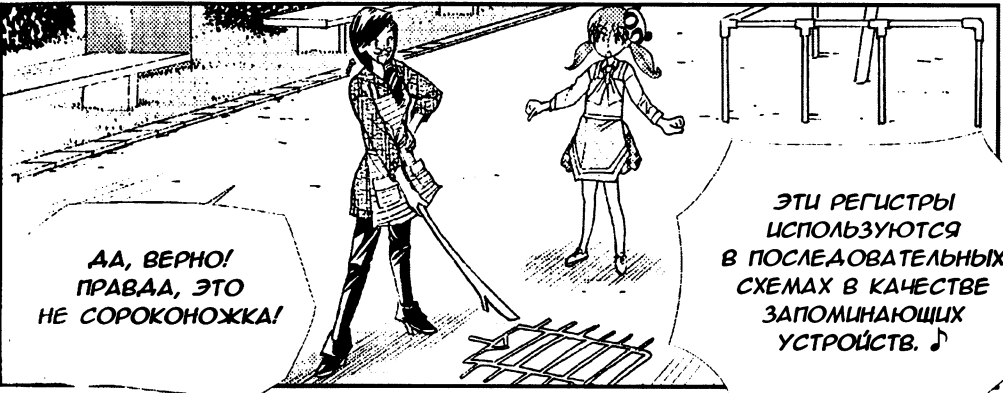


А "СОРОКОНОЖКА"
МОЖЕТ
ЗАПОМНИТЬ СТОЛЬКО
ЦИФРОВЫХ
СИГНАЛОВ (О ЦИЛ 1),
СКОЛЬКО
D-ТРИГГЕРОВ В НЕЙ
СОДЕРЖИТСЯ.



ПРАВДА?!
У СОРОКОНОЖКИ
НА ЭТОМ ЧЕРТЕЖЕ
ШЕСТЬ ПАР ЛАПОК...


ЗНАЧИТ,
ОНА МОЖЕТ
ЗАПОМНИТЬ 6 БИТОВ?




АА, ВЕРНО!
ПРАВДА, ЭТО
НЕ СОРОКОНОЖКА!

ЭТИ РЕГИСТРЫ
ИСПОЛЗУЮТСЯ
В ПОСЛЕДОВАТЕЛЬНЫХ
СХЕМАХ В КАЧЕСТВЕ
ЗАПОМИНАЮЩИХ
УСТРОЙСТВ. ♪

ЯСНО!

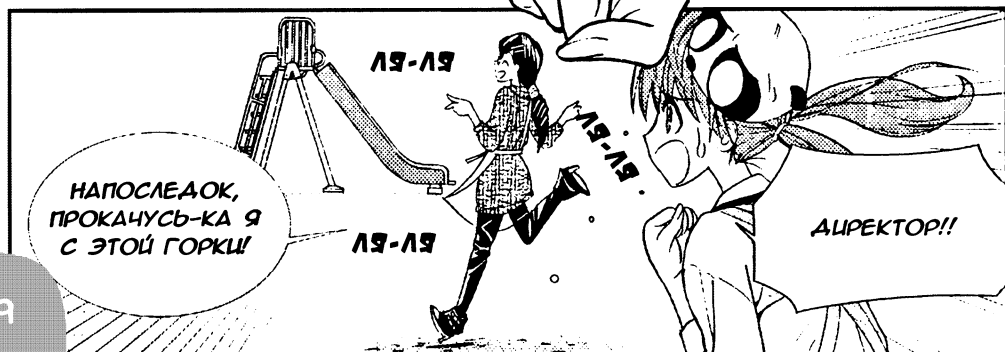
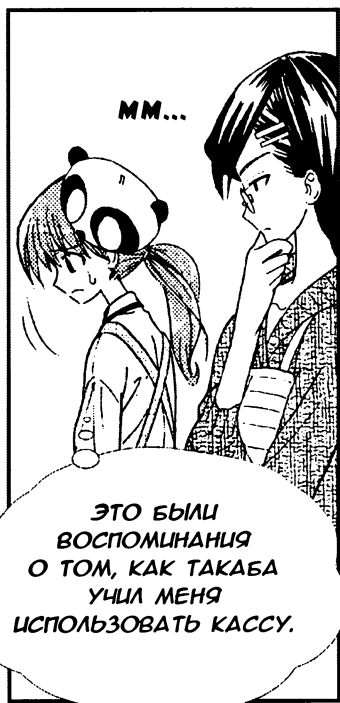


ЗНАЧИТ,
РЕГИСТР - ОЧЕНЬ
ВАЖНОЕ УСТРОЙСТВО?



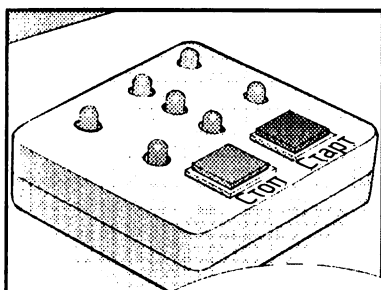
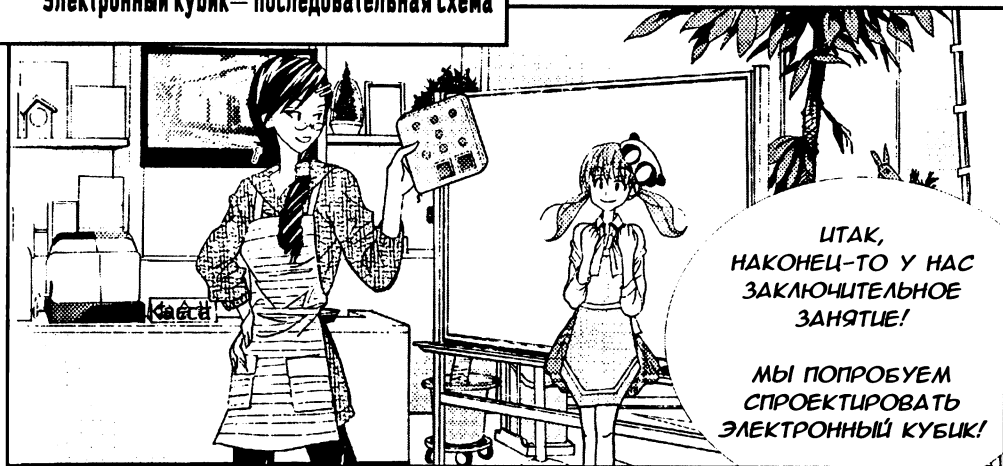
КСТАТИ, СЛОВО
REGISTER ПО-АНГЛИЙСКИ
ОЗНАЧАЕТ
"ЖУРНАЛ ДЛЯ ЗАПИСЕЙ",
А CASH REGISTER -
ЭТО КАССА!

В ОБЩЕМ, СМЫСЛ СЛОВА -
"ЗАПОМИНАТЬ",
"ЗАПИСЫВАТЬ".



3. ПРОЕКТИРУЕМ ЭЛЕКТРОННЫЙ КУБИК

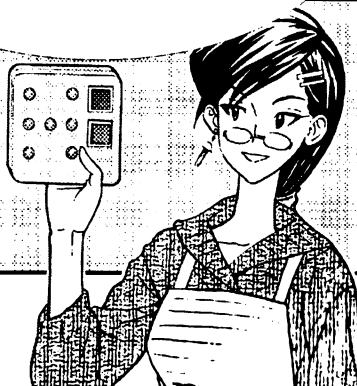
Электронный кубик – последовательная схема



УРА!
МЫ ПОЛЬЗОВАЛИСЬ ИМ
ТАК ЧАСТО, ЧТО Я ОЧЕНЬ
ХОРОШО ПРЕДСТАВЛЯЮ
СЕБЕ, КАК ОН РАБОТАЕТ.

ДО НАЖАТИЯ
НА КНОПКУ "СТОП"
ЛАМПОЧКИ БЫСТРО МИГАЮТ,
ИЗОБРАЖАЯ СМЕНУ
ГРАНЕЙ НАСТОЯЩЕГО КУБИКА.

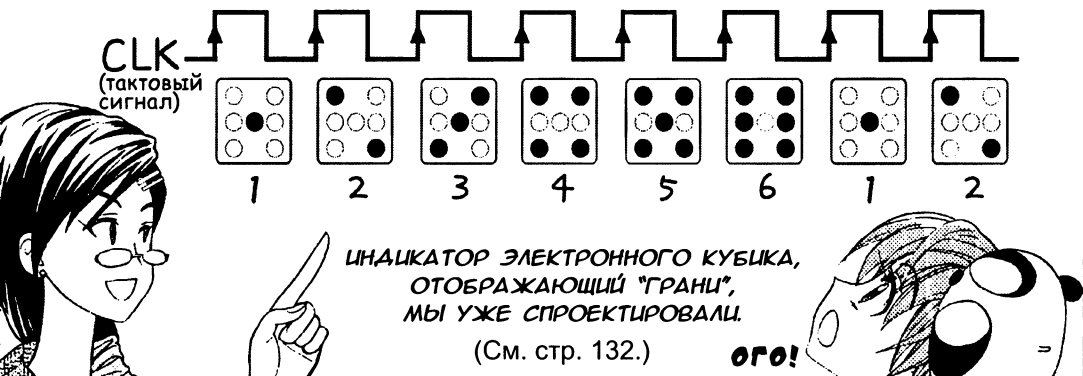
ДА.
ЧТОБЫ "ГРАНИ" ПЕРЕКЛЮЧАЛИСЬ
ДОСТАТОЧНО БЫСТРО,
МОЖНО ИСПОЛЬЗОВАТЬ ТАКОВЫЙ
СИГНАЛ (CLK) С ПЕРИОДОМ
ОКОЛО 0.1 СЕКУНДЫ.



"ГРАНИ" БУДУТ ПЕРЕКЛЮЧАТЬСЯ
ВО ВРЕМЯ ПЕРЕДНЕГО ФРОНТА
ЦИКЛА, КАК ГОВОРЯТ,
"ПО ПЕРЕДАНЕМУ ФРОНТУ",
КАК ПОКАЗАНО НА ЭТОМ РИСУНКЕ.

"ГРАНИ" БУДУТ ПЕРЕКЛЮЧАТЬСЯ В ТАКОМ
ПОРЯДКЕ, ПОВТОРЯЯ ЦИКЛ ОТ 1 ДО 6:

1 → 2 → 3 → 4 → 5 → 6 → 1 → 2 → ...



ЯСНО...
ЕСЛИ ИСПОЛЬЗОВАТЬ
ПЕРИОДИЧЕСКИ
ИЗМЕНЯЮЩИЙСЯ
ТАКТОВЫЙ СИГНАЛ...

...ТО В КАКОЙ БЫ
МОМЕНТ МЫ НИ НАЖАЛИ
НА КНОПКУ "СТОП",
ВЕРОЯТНОСТИ
ВЫПАДЕНИЯ
КАЖДОЙ ИЗ "ГРАНЕЙ"
БУДУТ ОДИНАКОВЫ
И РАВНЫ 1/6.

круть

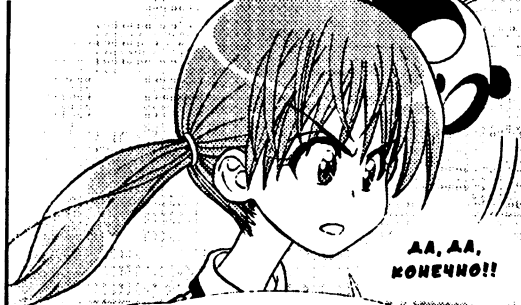
круть

ДА. ВАЖНЫЙ МОМЕНТ
ЗАКЛЮЧАЕТСЯ В ТОМ, ЧТО
"ГРАНИ" ДОЛЖНЫ
ПЕРЕКЛЮЧАТЬСЯ
ПО ПОРЯДКУ:
1 → 2 → 3 → 4 → ...

А НА ЭТО СПОСОБНЫ
ТОЛЬКО
ПОСЛЕДОВАТЕЛЬНЫЕ СХЕМЫ,
ИМЕЮЩИЕ ПАМЯТЬ
(ВНУТРЕННЕЕ СОСТОЯНИЕ).

В САМОМ ДЕЛЕ,
ЭТО ТАК,
ЕСЛИ
ПОДУМАТЬ...

хлоп



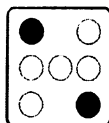
АА, АА,
КОНЕЧНО!!

ТЕКУЩЕЕ ВНУТРЕННЕЕ
СОСТОЯНИЕ - "2",
ЗНАЧИТ, СЛЕДУЮЩЕЕ,
НАВЕРНОЕ, "3".



ВЕАЬ СХЕМА ДОЛЖНА ПОМНИТЬ
ТЕКУЩУЮ "ГРАНЬ", ЧТОБЫ ЗНАТЬ,
КАКАЯ ДОЛЖНА БЫТЬ СЛЕДУЮЩАЯ:
ЕСЛИ ТЕКУЩАЯ - 1, ТО СЛЕДУЮЩАЯ - 2,
ЕСЛИ ТЕКУЩАЯ - 2,
ТО СЛЕДУЮЩАЯ - 3...

Память
(внутреннее
состояние схемы)



НО КАК ЖЕ НАМ
СПРОЕКТИРОВАТЬ
ПОСЛЕДОВАТЕЛЬНУЮ СХЕМУ?

ЗДЕСЬ МЫ БУДЕМ
ИСПОЛЬЗОВАТЬ
ПОРЯДОК ИЗ ТРЁХ ШАГОВ.

СКРИП

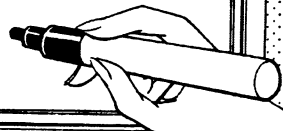


Порядок проектирования последовательной схемы (электронного кубика)

- (1) Чертим диаграмму состояний.
- (2) Назначаем состояниям двоичные числа.
- (3) Проектируем комбинационную схему.

Содержание этих шагов будет объясняться,
начиная со следующей страницы.

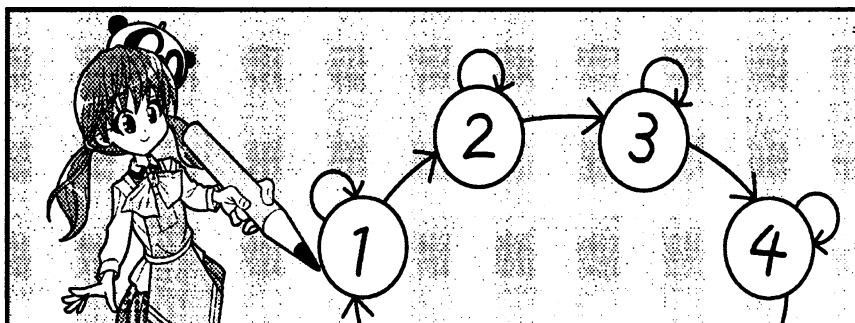
ВОТ!
В ТАКОМ ДУХЕ.



ВСЕГО ТРИ ШАГА
ДО ФИНИШНОЙ ЧЕРТЫ...

ЭТО ТОЧНО ВСЁ?
ТОГАА МНЕ НААА
СОБРАТЬСЯ
СИЛАММ!

1) Чертим диаграмму состояний



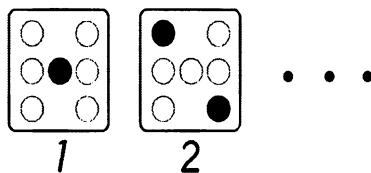
Итак, начинаю объяснять порядок проектирования последовательной схемы (электронного кубика).

Сначала нам нужно задать «внутренние состояния схемы» и начертить диаграмму, показывающую, как они будут переходить одно в другое в зависимости от уровней сигналов на входах.

Другими словами, мы должны начертить «диаграмму состояний»!

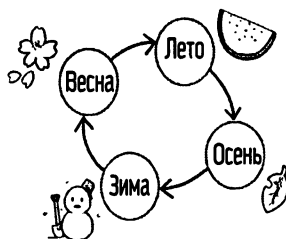


Диаграмму.. состояний...? Раз «состояния» — это внутренние состояния схемы электронного кубика, они должны быть такими: «грань 1», «грань 2» и так далее.



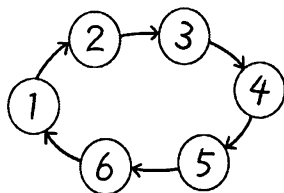
Да! В случае с электронным кубиком всё просто, ведь в качестве внутренних состояний можно использовать сами «грани».

Эти состояния будут переходить одно в другое так же, как, например, сменяют друг друга времена года...



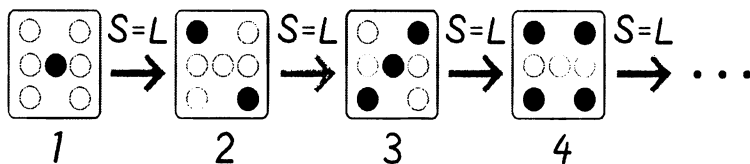


Тогда это очень просто! Нужно сделать, чтобы «границы кубика» сменяли друг друга вот так.



Верно, но нужно подумать также и о кнопке «Стоп»: когда она не нажата, на входе S должен быть уровень L.

При этом текущее внутреннее состояние будет переходить в следующие внутреннее состояние так, как показано на рисунке ниже.

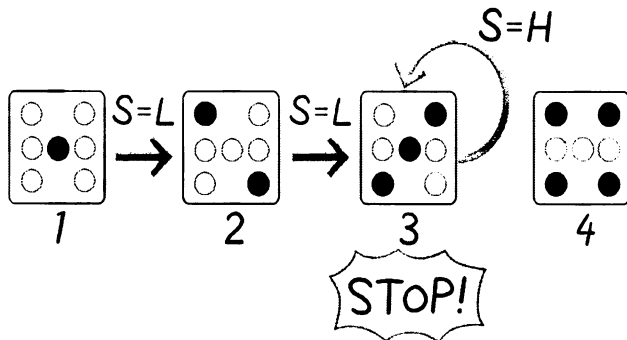


Вспых, вспых, вспых... Как быстро они сменяются!



С другой стороны, после нажатия на кнопку «Стоп» на входе S должен быть уровень H.

При этом текущее внутреннее состояние будет переходить само в себя, как показано на рисунке ниже.



Значит, будет поддерживаться то текущее внутреннее состояние, которое было в момент нажатия на кнопку «Стоп»?

Как будто на настоящем кубике выпало, например, три?



На диаграмме должна присутствовать также информация о возможных входных уровнях: $S = L$ и $S = H$, поэтому она должна выглядеть вот так.

Итак, диаграмма состояний электронного кубика готова!

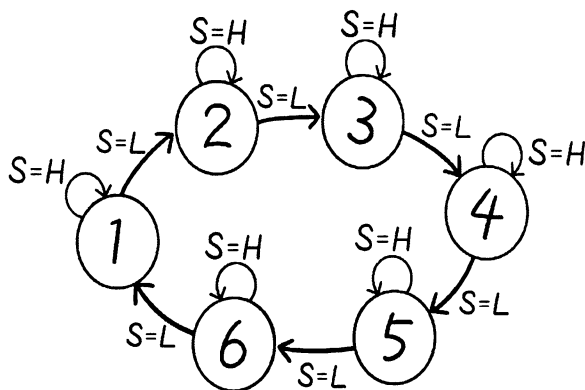


Диаграмма состояний электронного кубика



Правда? Но как она может нам пригодиться?



Хи-хи. Когда Такаба учил тебя проектированию комбинационных схем, он же говорил тебе что-то вроде: «Стоит начертить таблицу истинности — и схема готова»?



Да, говорил (см. стр. 62).
Директор, вы что, ясновидящая?



Точно таким же образом стоит начертить диаграмму состояний — и последовательная схема готова!

Другими словами, последовательные схемы проектируются на основе диаграммы состояний. Когда у нас есть диаграмма состояний, нам остаётся только воспользоваться определёнными методами построения схем.

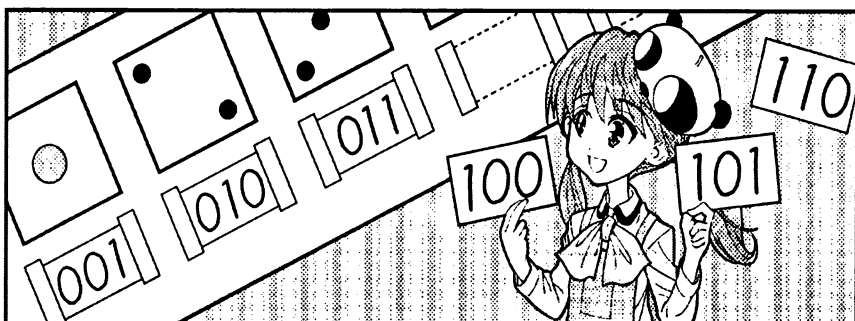


Ого! Правда?
Значит, диаграмма состояний так же важна, как таблица истинности, о которой мне рассказал Такаба.
Всё ясно. Хи-хи-хи...



(Ах, зачем я ей о нём напомнила?!)

(2) Назначаем состояниям двоичные числа



Итак, теперь у нас следующий шаг. А что это значит «назначить состояниям двоичные числа»?



Помнишь, что мы использовали кодирование, когда речь зашла о ресторане китайской кухни (см. стр. 108)? Теперь мы так же закодируем каждое из состояний. Умело назначив каждому состоянию двоичное число (двоичный код), можно существенно упростить схему.

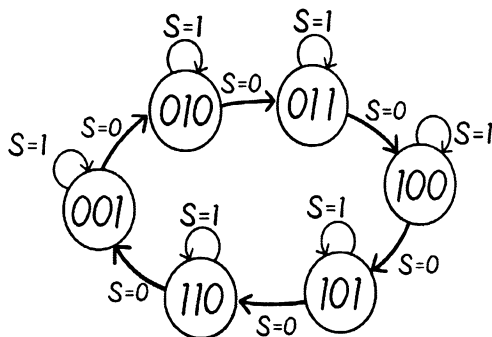
Однако здесь, чтобы особо не напрягаться, мы просто назначим состояниям номера (числа очков) граней кубика. Поступив так, мы сможем подавать коды состояний в неизменном виде на входы уже спроектированного нами индикатора «граней кубика» (см. стр. 132).



Да, так, наверное, будет проще. Значит, шесть состояний у нас будут выражаться вот такими трёхбитными числами: 001 («грань 1»), 010 («грань 2»), 011 («грань 3»), 100 («грань 4»), 101 («грань 5») и 110 («грань 6»).



Верно! И тогда наша диаграмма состояний примет вот такой вид.




Ясно. И уровни на входе S (кнопка «Стоп») тоже выражаем в виде двоичных значений 1 и 0 вместо букв H и L...



Далее, чтобы лучше представить себе задачу проектирования, давай составим ещё «Таблицу переходов состояний». Текущее состояние в ней будет выражаться тремя битами C_2 , C_1 и C_0 , а следующее состояние — тремя битами N_2 , N_1 и N_0 .

Таблица переходов состояний кубика (после назначения кодов)

Вход S	Текущее состояние (выходы)			Следующее состояние					
	C_2	C_1	C_0	N_2	N_1	N_0			
 S = 1 Текущее состояние перейдёт само в себя	1	0	0	1	(1)	0	0	1	(1)
	1	0	1	0	(2)	0	1	0	(2)
	1	0	1	1	(3)	0	1	1	(3)
	1	1	0	0	(4)	1	0	0	(4)
	1	1	0	1	(5)	1	0	1	(5)
	1	1	1	0	(6)	1	1	0	(6)
S = 0 Текущее состояние перейдёт в следующее состояние	0	0	0	1	(1)	0	1	0	(2)
	0	0	1	0	(2)	0	1	1	(3)
	0	0	1	1	(3)	1	0	0	(4)
	0	1	0	0	(4)	1	0	1	(5)
	0	1	0	1	(5)	1	1	0	(6)
	0	1	1	0	(6)	0	0	1	(1)

Номер (число очков)
границ кубика

※ Коды текущего состояния подаются на входы индикатора кубика без изменений, поэтому биты текущего состояния будут также и выходами последовательной схемы.



Ага. Теперь мне абсолютно ясны все связи между битом S (STOP), битами текущего состояния — C_2 , C_1 и C_0 — и битами следующего состояния — N_2 , N_1 и N_0 !



Итак, будем двигаться дальше. Вот мы использовали номера граней кубика в качестве кодов состояний — 001, 010, 011, 100, 101, 110, но как нам теперь запоминать это трёхбитное состояние?



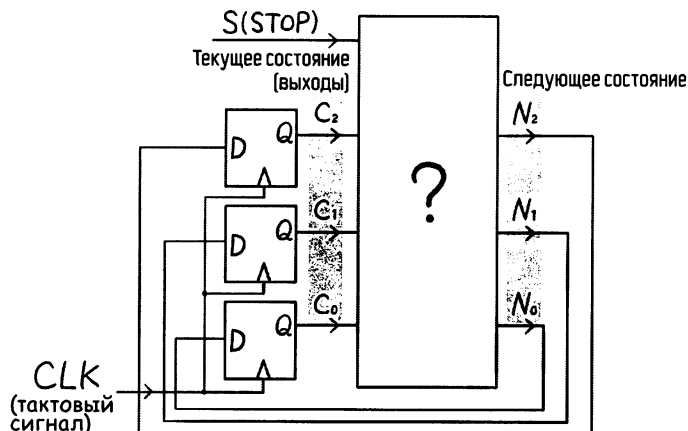
Что?! ...Ага, я поняла... Нужно использовать запоминающие устройства: триггеры или, там, регистр (см. стр. 153 и стр. 168).



Да, правильно! Чтобы запомнить трёхбитное состояние, можно использовать три D-триггера, то есть, по сути, трёхбитный регистр.



Таким образом, теперь у нас есть каркас последовательной схемы электронного кубика, показанный на рисунке ниже.



По кругу — с выходов на входы...

Базовая структура (каркас) последовательной схемы



Ого! Здесь действительно используются три D-триггера, то есть трёхбитный регистр!



Да, текущее состояние будет содержаться в трёхбитном регистре, а следующее состояние будет зависеть от выходов последовательной схемы (C_2 , C_1 и C_0) и от входа S (STOP). Ведь выходы последовательной схемы у нас — это то же самое, что «текущее состояние», так как мы использовали номера граней кубика, выводимые на индикатор, в качестве кодов состояний.

В итоге получается очень просто: текущее состояние (выходы C_2 , C_1 и C_0) и вход S определяют следующее состояние (N_2 , N_1 и N_0).



Ого. А та схема, которая «определит следующее состояние», заключена внутри прямоугольника с вопросительным знаком?



Да, но об этом мы поговорим на следующем шаге. Итак, ещё раз взгляни на эту схему. Ты заметила, что следующее состояние (N_2 , N_1 и N_0) «по кругу» возвращается на входы регистра? Таким образом, оно будет записано в регистр в момент ближайшего переднего фронта сигнала CLK (тактового сигнала).

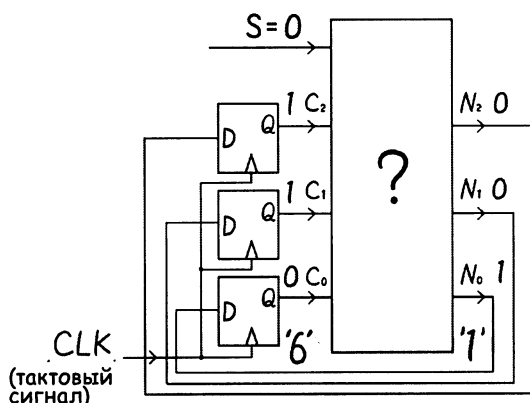


Следующее состояние возвращается «по кругу» и запоминается, превращаясь в текущее состояние... На диаграмме состояний этому соответствует «переход в следующее состояние». Таким образом, эта схема реализует вот такие переходы состояний:

$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 1 \rightarrow 2 \rightarrow \dots$



Верно! На рисунке ниже показан случай текущего состояния 110 ("грань 6") при $S(\text{STOP}) = 0$.



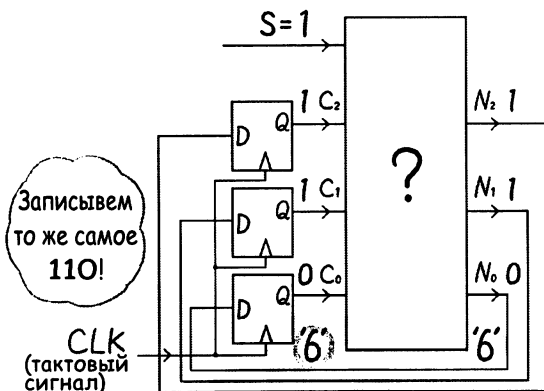
В этом случае следующее состояние, которое должно быть равно 001 ("грань 1"), вернётся на входы регистра и будет записано в него в момент переднего фронта тактового сигнала.



Это соответствует переходу состояния 110 в состояние 001, да?



Да! Кроме того, мы должны спроектировать схему внутри прямоугольника с вопросительным знаком так, чтобы при $S(\text{STOP}) = 1$ биты N_2 , N_1 и N_0 содержали, в нашем случае, значение 110 ("грань 6").

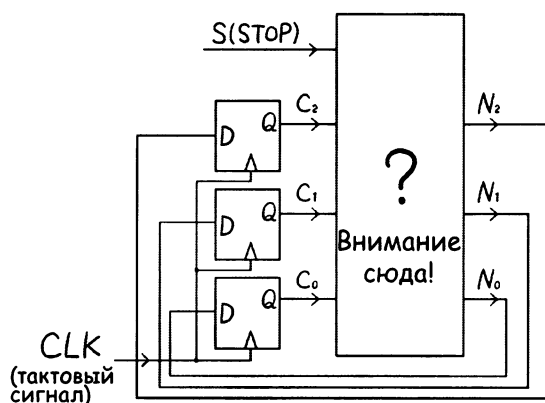
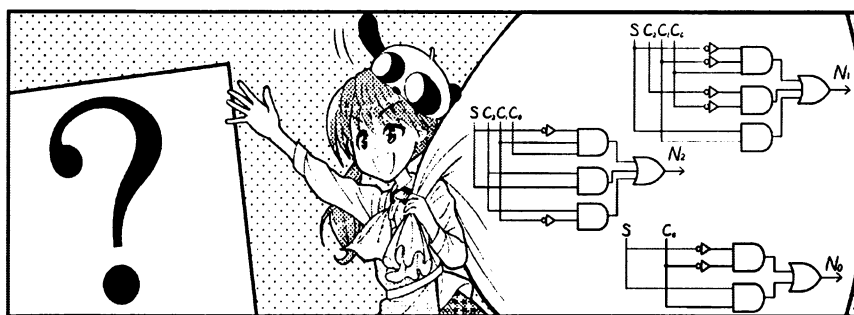


Благодаря этому по переднему фронту тактового сигнала в регистр опять будет записано 110, другими словами, состояние будет переходить само в себя — текущая «грань кубика» меняться не будет.



Ясно! Состояние будет поддерживаться одним и тем же, как будто на настоящем кубике выпало шесть, в нашем случае. Теперь эта базовая структура последовательной схемы стала мне абсолютно ясна!

3) Проектируем комбинационную схему



Итак, наконец-то у нас последний шаг. Мы сосредоточим внимание на той части, которая помечена на чертеже вопросительным знаком.

Как же нам спроектировать схему, генерирующую определённый код «следующего состояния» при вводе в неё кода «текущего состояния»?

На самом деле ты, Ицумо, уже знаешь ответ на этот вопрос. 🎵



Что?! Так... Это должна быть схема, которая на основе текущего состояния (C_2 , C_1 и C_0) будет генерировать следующее состояние (N_2 , N_1 и N_0)? Так... так...



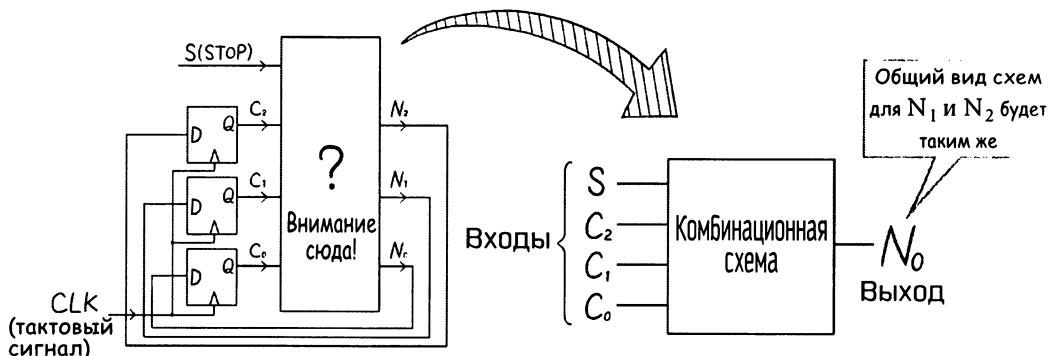
Ладно, дам тебе подсказку... Какая схема при вводе в неё определённого значения сразу выдаст нам требуемое значение? Конечно же, комбинационная!



Как?! Неужели это одна из таких простеньких схем, которые мы изучали до этого?!



Именно. Рассмотрим «прямоугольник с вопросительным знаком». Можно начертить в общем виде часть его схемы для одного из выходов - N_0 , как это сделано ниже. Это позволяет легко заметить, что это и есть «комбинационная схема», так как уровень на её выходе зависит только от текущих уровней на её входах.



Да, в самом деле, это — комбинационная схема! А значит, можно использовать уже изученные мною методы проектирования (см. главу 2)! Так как у нас есть три выхода: N_2 , N_1 и N_0 , нам потребуются три таблицы истинности — по одной для каждого выхода.



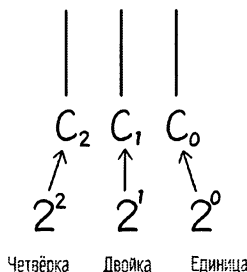
Да, ты права. Но можно использовать «Таблицу переходов состояний» электронного кубика, которую мы недавно составили (см. стр. 178). И ещё один важный момент: при составлении карт Карно по таблицам истинности нельзя забывать о «безразличном состоянии»!

Можешь мне сказать, когда здесь будет безразличное состояние?



Так... Бит S изменяется при нажатии на кнопку «Стоп», поэтому он может быть равен и 0, и 1.

Битами C_2 , C_1 и C_0 , выражающими номер грани кубика, можно представить восемь чисел от 0 до 7 (об идее, лежащей в основе этого подхода, см. на стр. 116).



Если все биты — C_2 , C_1 и C_0 — равны нулю, то это трёхбитное число равно 0.

Если все биты — C_2 , C_1 и C_0 — равны 1, то это трёхбитное число равно 7.

Таким образом, тремя битами можно выразить числа от 0 до 7!



Но на настоящем кубике нет таких граней, как «0» и «7».

Кроме того, для каждой из этих двух комбинаций — 0 и 7 — можно рассмотреть два случая: $S = 0$ и $S = 1$. Таким образом, безразличное состояние можно использовать для четырёх комбинаций!

Четыре случая безразличного состояния на выходах N_2 , N_1 и N_0

$SC_2 \backslash C_0$	00	01	11	10
00	⊖			
01			⊖	
11			⊖	
10	⊖			

Грань 0», $S = 0$

Грань 7», $S = 0$

Грань 7», $S = 1$

Грань 0», $S = 1$



Ого, какой обстоятельный ответ! Таким образом, имеется довольно много случаев безразличного состояния, а значит, возможно упрощение. Приступай!



Ясно! Попробую построить карты Карно и схемы для каждого из трёх выходов — N_2 , N_1 и N_0 ! Начинаю!

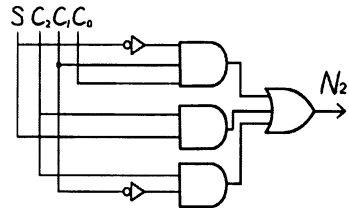
N_2 (самый старший бит)

$SC_2 \backslash C_0$	00	01	11	10
00	—		1	
01	1	1	—	
11	1	1	—	1
10	—			

$\bar{S} \cdot C_1 \cdot C_0$

SC_2

$C_2 \cdot \bar{C}_1$



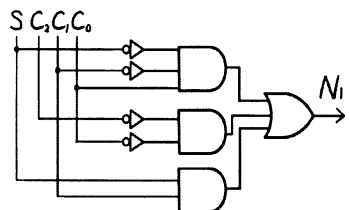
N_1 (средний бит)

$SC_2 \backslash C_0$	00	01	11	10
00	—	1		1
01		1	—	
11			⊖	1
10	—		1	1

$\bar{S} \cdot \bar{C}_1 \cdot C_0$

$\bar{C}_2 \cdot \bar{C}_0$

$S \cdot C_1$

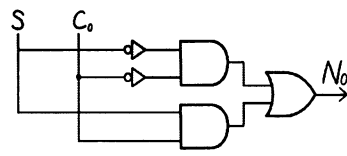


N_0 (самый младший бит)

$\overline{S}C_0$	00	01	11	10
00	-			1
01	1		-	1
11		1	-	
10	-	1	1	

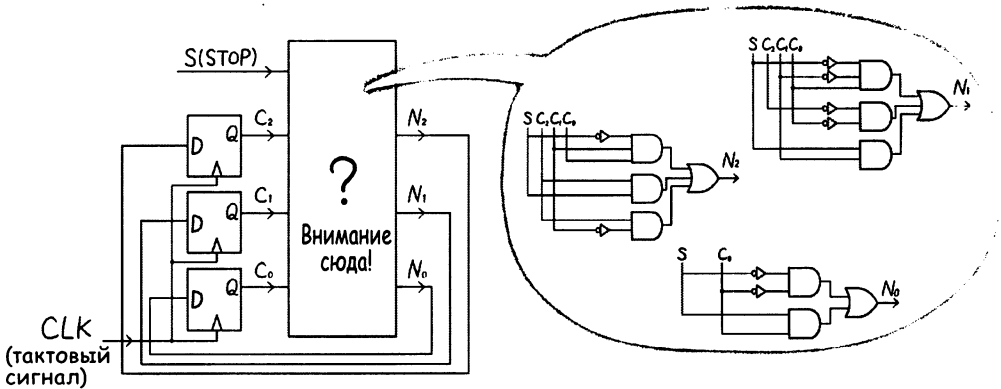
$\overline{S}C_0$

$S \cdot C_0$

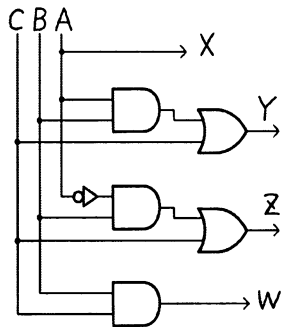


Да! Хорошо сделано!

Итак, теперь вставим эти три части комбинационной схемы вместо прямоугольника с вопросительным знаком.



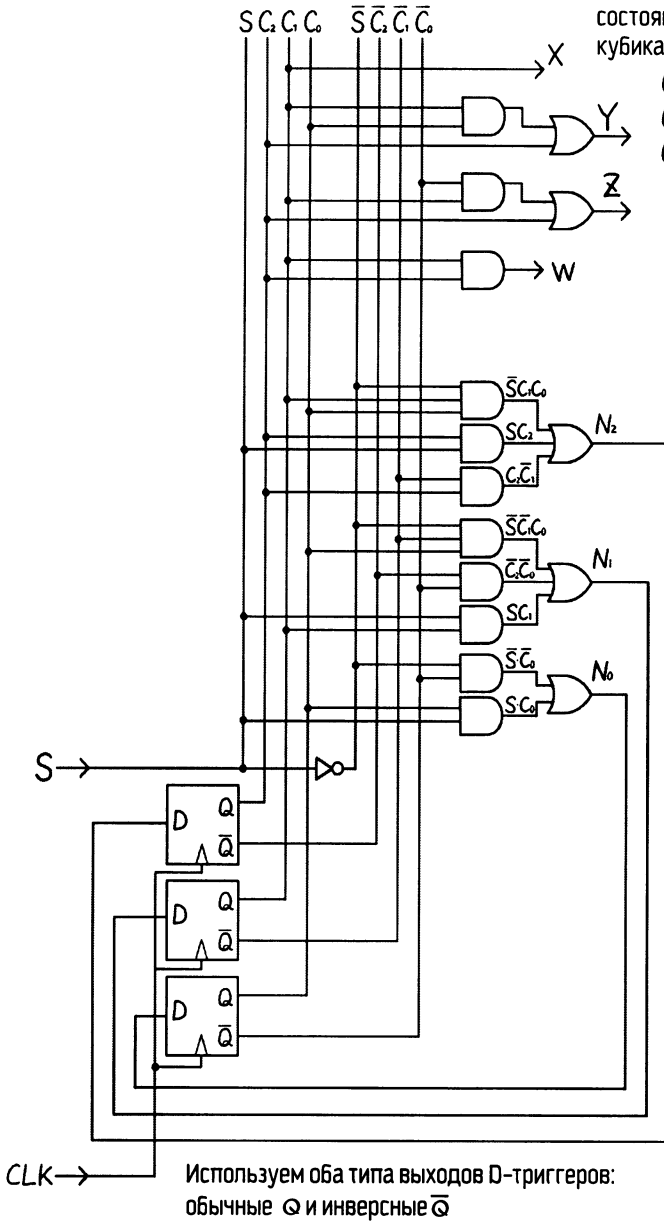
Теперь остаётся только подключить индикатор кубика, который мы спроектировали на днях, — и схема электронного кубика готова!



Индикатор кубика

Так как у нас код текущего состояния равен номеру грани кубика

$C_2 \rightarrow C$
 $C_1 \rightarrow B$
 $C_0 \rightarrow A$



Полная схема электронного кубика



Здорово! Чертёж схемы электронного кубика наконец-то готов!



Молодец, ты хорошо поработала!
Теперь ты прекрасно разбираешься в проектировании последовательных схем. Вот порядок их проектирования, помнишь?

Порядок проектирования последовательной схемы (электронного кубика)

- (1) Чертим диаграмму состояний.
- (2) Назначаем состояниям двоичные числа.
- (3) Проектируем комбинационную схему.



Да! Теперь я прекрасно понимаю смысл каждого из этих шагов!



Стоит заметить, что спроектированная нами последовательная схема относится к «синхронным», так как моменты переходов состояний в ней задаются тактовым сигналом. Существуют также асинхронные последовательные схемы, в которых нет тактового сигнала, однако они почти не используются на практике, поэтому полученные тобой знания применимы практически к любым реально используемым цифровым схемам.



Ого! Как здорово!
Однако создать огромную цифровую схему, такую как компьютер, в качестве одной последовательной схемы, наверное, невозможно?
Нереалистично это как-то...



Да. Огромные цифровые схемы проектируют по частям, разбивая их на несколько последовательных схем.
Однако, как уже здесь говорилось, для этого могут потребоваться знания и умения из других областей, таких как компьютерная архитектура, системное проектирование.



А, вот как?
Как много ещё надо всего изучить...
Но как бы то ни было, как здорово, что мы смогли сами спроектировать электронный кубик! Ура!

Устройство D-триггера

Как запомнить цифровые данные? Можно использовать для этого такую структуру из пары вентилей NOT, в которой выход каждого из вентилей подключён к входу другого.

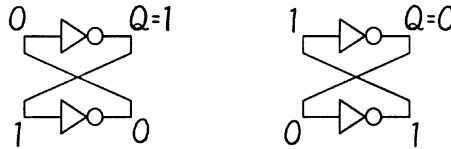


Рис. 1. Самая простая схема памяти

Хотя в этой схеме вообще отсутствуют входы, при $Q = 1$ на входе верхнего вентиля будет 0, в результате чего Q всё время будет равно 1. Аналогично, при $Q = 0$ на входе верхнего вентиля будет 1, в результате чего Q всё время будет равно 0. Другими словами, эта схема может постоянно находиться в одном из двух состояний: $Q = 1$ или $Q = 0$.

Возможно, это поначалу будет немного трудно понять, но на самом деле это и есть простейшая схема памяти.

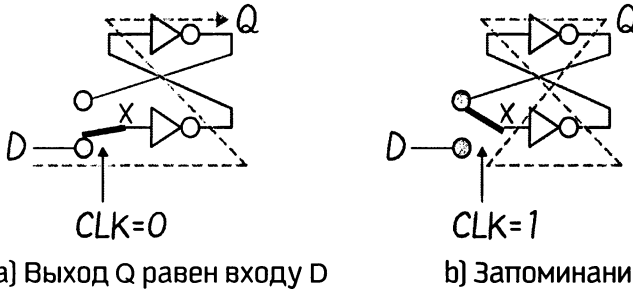


Рис. 2. Схема с переключателем для ввода данных (фlea «D-защёлка»)

Так как без входов эта схема будет мало на что годна, мы оснастим её переключателем, который будет управляться сигналом CLK, как показано на рис. 2. У переключателя есть контакт X, который при $CLK = 0$ подключается к входу D , а при $CLK = 1$ — к выходу верхнего вентиля. Другими словами, при $CLK = 0$ уровень на выходе Q будет равен уровню на входе D . В тот момент, когда CLK станет равен 1, введённые через вход D данные окажутся запертыми» внутри «восьмёрки», присутствуя на выходе Q . Так как при $CLK = 1$ вход D отключен от схемы, запертые данные не изменятся при любых изменениях уровня сигнала на входе D . Это устройство, называемое D-защёлкой, и есть те самая фlea из манги (триггер «мастер»).

Кстати, в качестве переключателя X используется селектор данных (мультиплексор), о котором я рассказывал в «Дополнительной информации» прошлой главы. Однако одна фея А не справится со всей работой D-триггера. Дело в том, что D-защёлка будет хранить введённые данные лишь до момента изменения CLK с 1 на 0, а при CLK = 0 просто будет пропускать данные со входа D на выход Q. Таким образом, запоминание по переднему фронту тактового сигнала D-триггера, нужное для синхронизации переходов между состояниями, не может быть реализовано. В этой ситуации приходится звать на помощь ещё одну фею (триггер «помощник»).

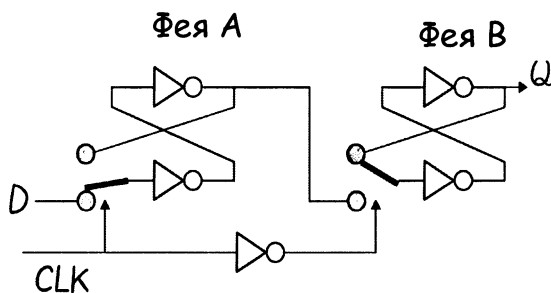


Рис. 3. Устройство D-триггера

Как показано на рис. 3, двух фей соединяют в цепочку, причём на переключатель феи В подаётся инверсный тактовый сигнал. Благодаря этому, пока одна из фей пропускает данные, готовясь их запомнить, другая фея хранит предыдущие данные.

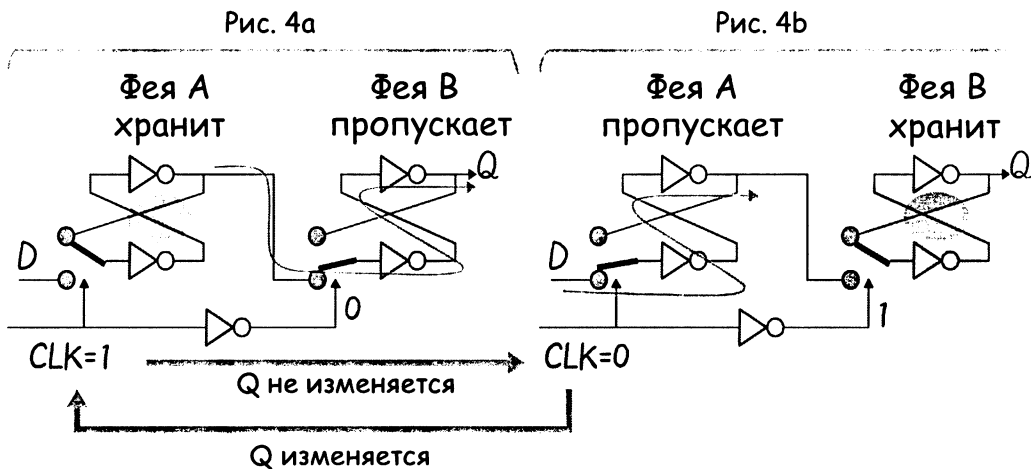


Рис. 4. Функционирование D-триггера



При $CLK = 1$ фея А хранит данные, а фея В пропускает данные, получаемые от феи А, на выход Q (рис. 4а). При изменении CLK с 1 на 0 фея В сохранит данные, которые она до этого получала от феи А и пропускала на выход Q, поэтому уровень на выходе Q не изменится, и снаружи может показаться, что внутри схемы не произошло никаких изменений. Пока фея В хранит данные, фея А пропускает входные данные, готовясь их сохранить (рис. 4б). Наконец, в момент изменения CLK с 0 на 1 (т. е. по переднему фронту тактового сигнала) фея А сохранит новые данные, а фея В начнёт пропускать их на выход Q (возврат к рис. 4а).

Вот так, благодаря слаженной работе двух фей, и работает D-триггер.

За вышеописанной моделью взаимоотношений двух фей закрепилось вводящее в заблуждение название — Master-Slave ("мастер" и «помощник»). В действительности ни одна из фей не является главной, обе они на равных началах выполняют одинаково важную для D-триггера работу. Ни о каком подчинении в их отношениях не может быть и речи. Но раз уж такое название закрепилось исторически, ничего не поделаешь, придётся с этим мириться.



Здесь я рассказал вам только об одном типе триггеров — D-триггере.

В наши дни, в связи с тем, что проектирование осуществляется главным образом с помощью языков описания аппаратуры (о них пойдёт речь в «Послесловии» на стр.208), другие типы триггеров практически вышли из употребления. Исторически, однако, «королём триггеров» считался JK-триггер.

Далее я познакомлю вас с ним, а также с другими типами триггеров.

Дополнительная информация

Различные триггеры

У JK-триггера есть входы J и K, вход для тактового сигнала, выходы Q и \bar{Q} .

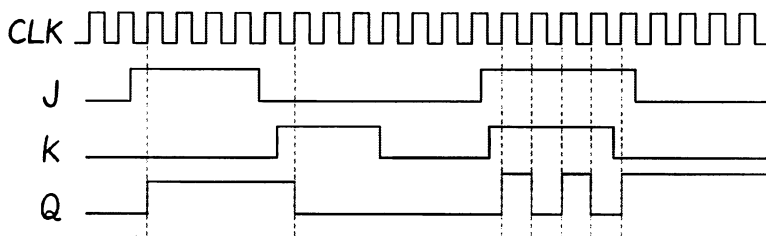
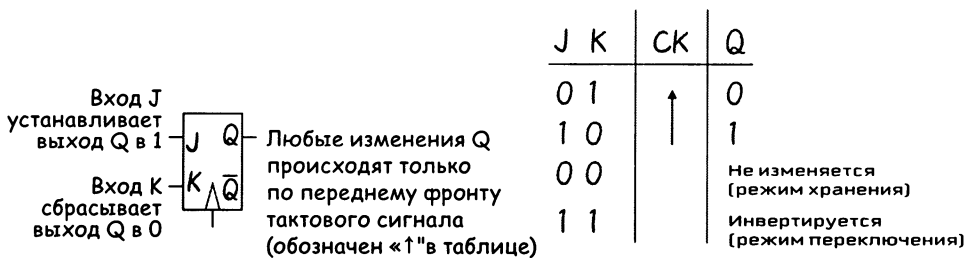
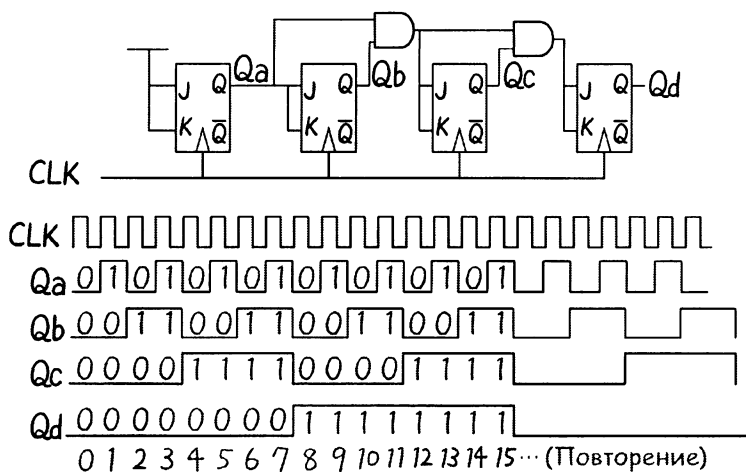


Рис. 5. JK-триггер (JK-FF)

Вход J устанавливает выход Q в 1, а вход K, наоборот, сбрасывает его в 0. Другими словами, если $J = 1$ и $K = 0$, то в момент переднего фронта тактового импульса на выходе Q устанавливается 1, а если $J = 0$ и $K = 1$, то 0. При $J = 0$ и $K = 0$ уровень на выходе не изменяется даже в момент прихода переднего фронта тактового импульса. Таким образом, при $J = 0$ и $K = 0$ JK-триггер находится в режиме хранения текущего состояния.

Интересная особенность JK-триггера проявляется при $J = 1$ и $K = 1$. Дело в том, что в этом случае в момент переднего фронта тактового импульса происходит инверсия уровня Q: он становится равен 0, если был 1, и становится равен 1, если был равен 0. Это называется «режимом переключения» (toggle mode).

Используя режим переключения JK-триггера, легко построить схему, считающую количество тактовых импульсов. Принцип работы синхронного счётчика двоичных чисел прост: по переднему фронту тактового сигнала инвертируются те биты, у которых все предыдущие биты равны 1, а также самый младший бит; остальные биты остаются прежними. Схема синхронного двоичного счётчика, основанного на этом принципе, показана на рис. 6.



Инвертируем биты, для которых все предыдущие биты равны 1, а также самый младший бит.

★ Как видно из временной диаграммы, составленной из выходов Q двоичное число $Q_d Q_c Q_b Q_a$ увеличивается на 1 в моменты переднего фронта сигнала CLK.

Рис. 6. Синхронный счётчик

Так как входы J и K у всех JK-триггеров в этом счётчике обобщены, JK-триггеры в нём способны работать всего в двух режимах: переключения ($J = K = 1$) и хранения ($J = K = 0$). Однако во многих случаях этого оказывается вполне достаточно, поэтому на основе JK-триггера был создан новый тип — T-триггер, в котором входы J и K объединили в один вход T, как показано на рис. 7. Кроме того, если замкнуть вход K на вход J через инвертор (т. е. вентиль NOT) и обозначить общий вход буквой D, то получится уже изученный нами D-триггер.

Именно поэтому JK-триггер, объединяющий в себе функции и T-триггеров, и D-триггеров, считался «королём триггеров». Однако это богатство функций, наоборот, оказывается лишним при проектировании с помощью САД, в связи с чем JK-триггер в настоящее время используется редко.

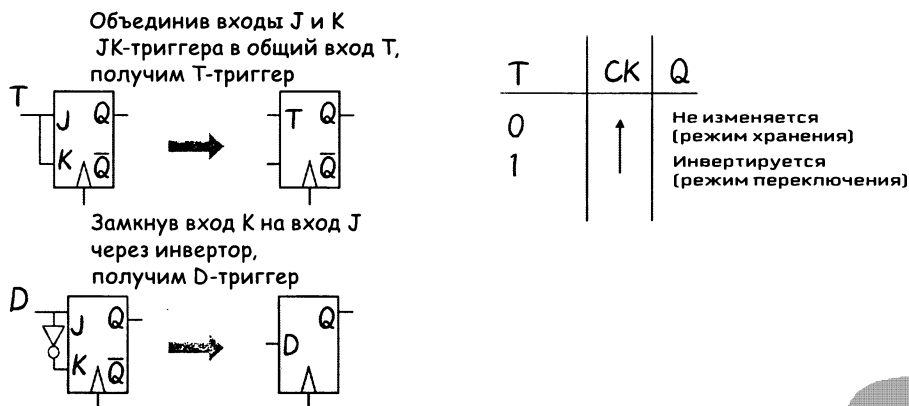


Рис. 7. Связь JK-триггера с T-триггером и D-триггером



Однако D-триггер, который в наши дни считается «самым успешным», не лишён недостатков: он сильно проигрывает JK-триггеру в том, что у D-триггера отсутствует режим хранения текущего состояния, имеющийся у JK-триггера ($J = K = 0$), поэтому D-триггер вынужден запоминать уровень на входе D каждый раз при появлении переднего фронта тактового сигнала. По этой причине D-триггер бесполезен в тех случаях, когда требуется запоминать данные не периодически, а по необходимости.

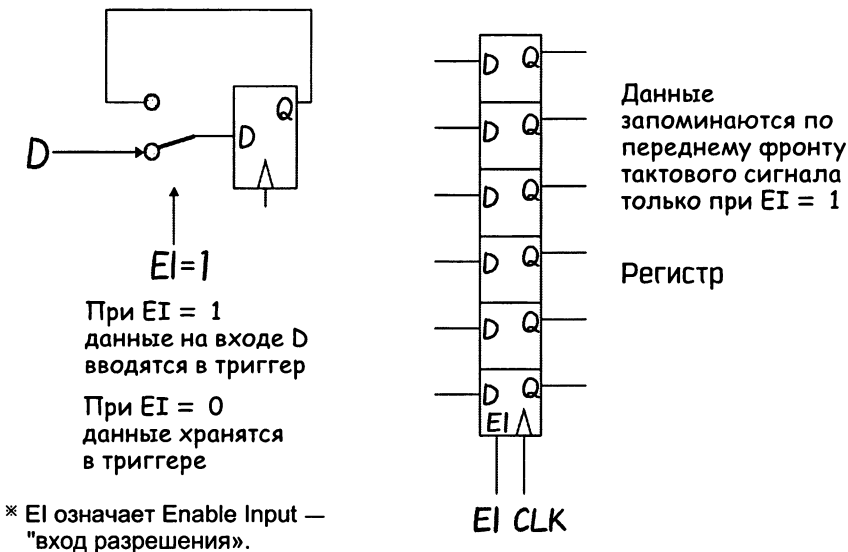


Рис. 8. D-триггер со входом разрешения

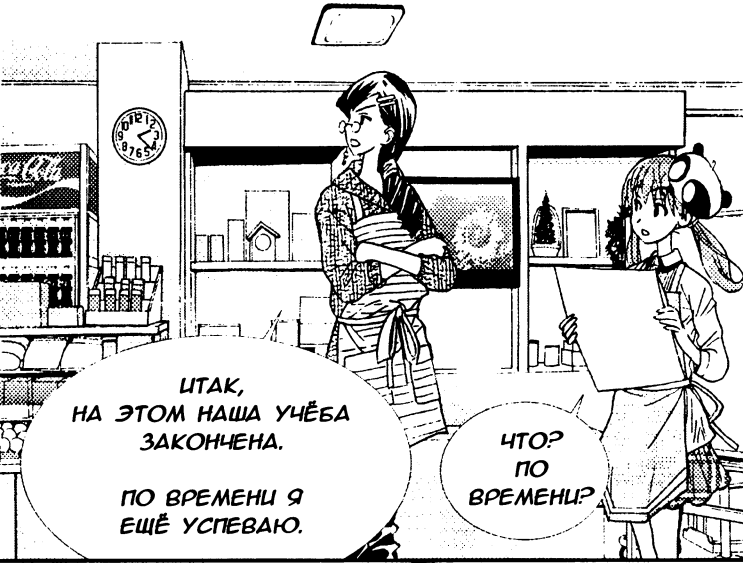
Чтобы решить эту проблему, D-триггер оснащают переключателем, как показано на рис. 8. При $EI = 0$ выход Q подключён к входу D-триггера, поэтому внешние данные могут быть записаны только при $EI = 1$.

Этот модифицированный тип называется «D-триггером со входом разрешения» и в настоящее время является наиболее часто используемым типом триггеров. Показанное на рис. 8 устройство, составленное из требуемого количества D-триггеров со входом разрешения, имеющее общий для всех триггеров вход сигнала разрешения (EI) и общий для всех триггеров вход тактового сигнала (CLK), называется «регистром».

Этот регистр запоминает данные по переднему фронту тактового сигнала только в том случае, если сигнал разрешения (EI) равен 1. В настоящее время при проектировании схем уделяют внимание тому, какие значения и в какое время будут записываться в регистры. Этот метод, о котором я расскажу в «Послесловии», называется «разработкой на уровне передач данных между регистрами» или «разработкой на уровне регистровых передач» (RTL: Register Transfer Level).

Объяснение терминов главы 5

- **Последовательные схемы** — схемы, в которых уровни на выходах определяются текущими уровнями на входах и текущим внутренним состоянием. В отличие от комбинационных схем, изученных в главе 3, способны запоминать прошлое (историю). Почти все реальные современные цифровые схемы — последовательные. Могут быть «синхронными» (другое название — «конечные автоматы»), в которых моменты смены состояний задаются тактовым сигналом, либо «асинхронными». Первые используются чаще, так как проще в проектировании. Конечные автоматы разделяются на автоматы Мили (Mealy machine, в честь английского ученого Георга Мили), в которых выходные сигналы зависят от текущего внутреннего состояния и входных сигналов, и автоматы Мура (Moore machine, также называются автоматами второго рода, в честь Эдварда Мура), в которых выходные сигналы зависят только от текущего внутреннего состояния. Электронный кубик, описанный в манге, является типичным автоматом Мура, так как его внутреннее состояние без изменений подаётся на выходы. Синхронные последовательные схемы состоят из триггеров и комбинационных схем.
- **Триггер** (англ. flip-flop, FF) — название «flip-flop» восходит к хлопанию развешенного белья на ветру, а в манге его уподобили стуку качелей. Используется также в смысле «сальто», но общий смысл этого слова в том, что два состояния резко сменяют друг друга. В цифровых схемах триггеры — это запоминающие устройства с двумя состояниями. В теории электронных схем их называют также «бистабильными мультивибраторами». Как я рассказал в «Дополнительной информации», есть разные типы триггеров, но в настоящее время в основном используется D-триггер, подробно описанный в манге.
- **Регистр** — устройство, служащее для запоминания информации и состоящее из выстроенных в ряд D-триггеров. Для проектирования крупномасштабных цифровых схем, таких как компьютеры, используют упомянутый в «Дополнительной информации» метод разработки на уровне регистровых задач (RTL: Register Transfer Level), уделяя внимание тому, какие значения и в какое время будут записываться в регистры, как эти значения будут перемещаться и обрабатываться. Этот метод в настоящее время является основным методом проектирования. В таких устройствах, как ЦПУ компьютеров, используется множество регистров.
- **Диаграмма состояний** — диаграмма, описывающая работу синхронной последовательной схемы путём построения переходов между её внутренними состояниями. Назначив состояниям коды, можно составить таблицу переходов состояний и спроектировать на её основе синхронную последовательную схему. Хотя в манге мы использовали элементарный способ кодирования, просто назначив состояниям выходные значения последовательной схемы, существуют такие методы, как прямое унитарное кодирование, в котором 1 содержится только в одном бите; код Джонсона, позволяющий уменьшить количество изменений значений битов.



КАК БЫ ТАМ НИ БЫЛО,
БОЛЬШОЕ ВАМ СПАСИБО,
ДИРЕКТОР!

БЛАГОДАРЯ ВАМ
Я САМА СМОГЛА
НАЧЕРТИТЬ СХЕМУ
ЭЛЕКТРОННОГО
КУБИКА!

ИТАК,
НА ЭТОМ НАША УЧЁБА
ЗАКОНЧЕНА.

ПО ВРЕМЕНИ Я
ЕЩЁ УСПЕВАЮ.

ЧТО?
ПО
ВРЕМЕНИ?



НЕТ, ЭТО НЕ ВСЁ!
НА ТВОЁМ ЧЕРТЕЖЕ СХЕМЫ
КОЕ-ЧЕГО НЕ ХВАТАЕТ!!

ТААААМ

И ТЫ, ШУМО,
ДОЛЖНА ИСПРАВИТЬ
ЭТО ПРЯМО СЕЙЧАС!!!



ЧТО?!

НО РАЗВЕ ВЫ
НЕ ГОВОРИЛИ,
ЧТО СХЕМА
ГОТОВА..?

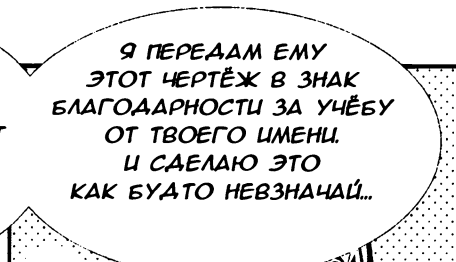


ВОТ ЗДЕСЬ!
ТЫ ДОЛЖНА НАПИСАТЬ
СВОЙ НОМЕР ТЕЛЕФОНА
И АДРЕС ЭЛЕКТРОННОЙ
ПОЧТЫ!



ЧТО?!

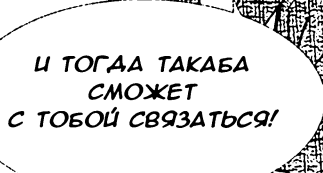
ПРЯМО СЕЙЧАС
Я ПОЕДУ В АЭРОПОРТ
ПРОВОЖАТЬ ТАКАБУ.



Я ПЕРЕДАМ ЕМУ
ЭТОТ ЧЕРТЁЖ В ЗНАК
БЛАГОДАРНОСТИ ЗА УЧЁБУ
ОТ ТВОЕГО ИМЕНИ.
И СДЕЛАЮ ЭТО
КАК БУДАТО НЕВЗНАЧАЙ...

ЛУП

ЛУП



И ТОГДА ТАКАБА
СМОЖЕТ
С ТОБОЙ СВЯЗАТЬСЯ!



ДИРЕКТОР, ВЫ ШУТИТЕ!!

ああ
ou!

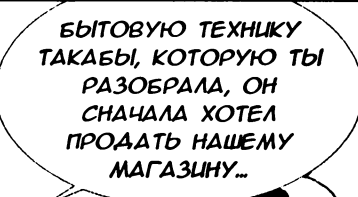
ああ



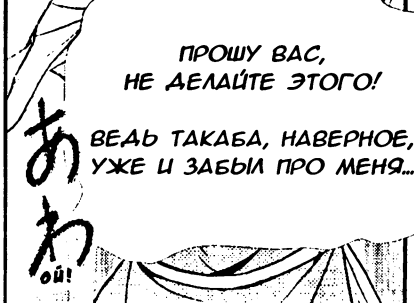
НУ, НЕ ЗНАЮ...

хи-хи-хи

...НО НЕОЖИДААННО
ИЗМЕНИЛ
СВОЁ РЕШЕНИЕ,
СКАЗАВ, ЧТО ХОЧЕТ
ОТААТЬ ВСЁ
"ХОРОШЕНЬКОЎ
НАПАРНИЦЕ".



БЫТОВУЮ ТЕХНИКУ
ТАКАБЫ, КОТОРУЮ ТЫ
РАЗОБРАЛА, ОН
СНАЧАЛА ХОТЕЛ
ПРОДАТЬ НАШЕМУ
МАГАЗИНУ...



ПРОШУ ВАС,
НЕ ДЕЛАЙТЕ ЭТОГО!

ВЕАЬ ТАКАБА, НАВЕРНОЕ,
УЖЕ И ЗАБЫЛ ПРО МЕНЯ...

ああ
ou!



ああ

...!!
ХОРОШЕНЬКОЎ
НАПАРНИЦЕ!

НО ВЕДЬ...

НО ВЕДЬ
ВСЁ РАВНО...
НИЧЕГО ИЗ ЭТОГО
НЕ ПОЛУЧИТСЯ.

ВЕДЬ ОН УЕЗЖАЕТ.
ЭТО ТАК ДАЛЕКО.

НИЧЕГО
НЕ ПОДЕЛАЕШЬ...

С ЭТИМ
ВСЁ КОНЧЕНО.

НИЧЕГО
НЕ ВЫЙДЕТ.

НЕТ,
НЕ ВЫЙДЕТ...

НАДО ПРОСТО
ЗАБЫТЬ...

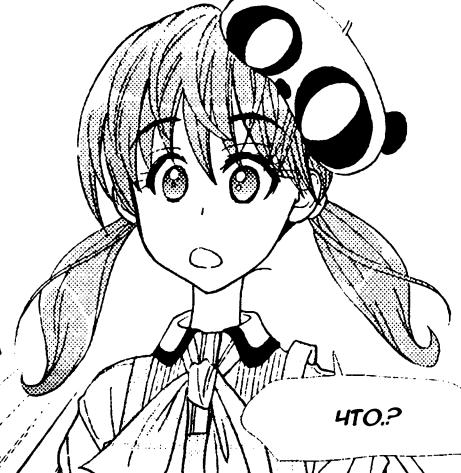
ААА!!
ХВАТИТ!!

БУУУ

ЩИМО, ТЫ ГОВОРИШЬ
ТАК, КАК БУАТО БЫ
ЭТО КОНЕЦ СВЕТА!!


КАКАЯ У НАС
ЭПОХА НА ДВОРЕ?!
КАКОЕ ЭТО ЧУВСТВО,
КОТОРОЕ
НЕ ВЫДЕРЖАЛО
ИСПЫТАНИЯ ЖАЛКОЙ
ТЫСЯЧЕЙ КИЛОМЕТРОВ?!

ЧТО ЖЕ ТЫ
ИЗУЧАЛА ДО
НАСТОЯЩЕГО
МОМЕНТА?!



ТЫ ИЗУЧАЛА
ЦИФРОВЫЕ СХЕМЫ,
НЕ ТАК ЛИ?!

ЧТО?!




ОДНАКО ТЫ ЯВНО
НЕДООЦЕНИВАЕШЬ
ЦИФРОВЫЕ СХЕМЫ И
ОСНОВАННЫЕ НА НИХ
ТЕХНОЛОГИИ!

ВЕДЬ ЭЛЕКТРОНИКА -
ЭТО НЕ ТОЛЬКО
ХОЛОДИЛЬНИКИ И
РИСОВАРКИ.

И СУЩЕСТВУЕТ СКОЛЬКО
УГОДНО ТЕХНОЛОГИЙ,
КОТОРЫЕ СВЯЗЫВАЮТ ЛЮДЕЙ
ЧЕРЕЗ ЛЮБЫЕ РАССТОЯНИЯ!



..... !!



ВЫ ВЕДЬ МОЖЕТЕ
ЗВОНИТЬ ДРУГ ДРУГУ
ПО ТЕЛЕФОНУ,
ОБМЕНИВАТЬСЯ МЭЙЛАМИ
С ПОМОЩЬЮ
КОМПЬЮТЕРА?

А В СВОБОДНОЕ ВРЕМЯ
МОЖНО СЕСТЬ НА ПОЕЗДА
ИЛИ В САМОЛЁТ И
ПОЕХАТЬ В ГОСТИ.

ВЕДЬ ЦИФРОВЫЕ СХЕМЫ
ИСПОЛЬЗУЮТСЯ
И В СВЯЗИ,
И НА ТРАНСПОРТЕ!

ПОЭТОМУ...

ПОРАБОТАЕШЬ ЗАДЕСЬ,
ДЕНЕГ НАКОПИШЬ -
И СМОЖЕШЬ!

ПОЭТОМУ...

...ЕСТЬ МНОГО
ДРУГИХ СПОСОБОВ,
КРОМЕ
"ЗАБЫТЬ О НЁМ"!!

АА... АА...

Я ПОНИМАЮ...

ОДНАКО...

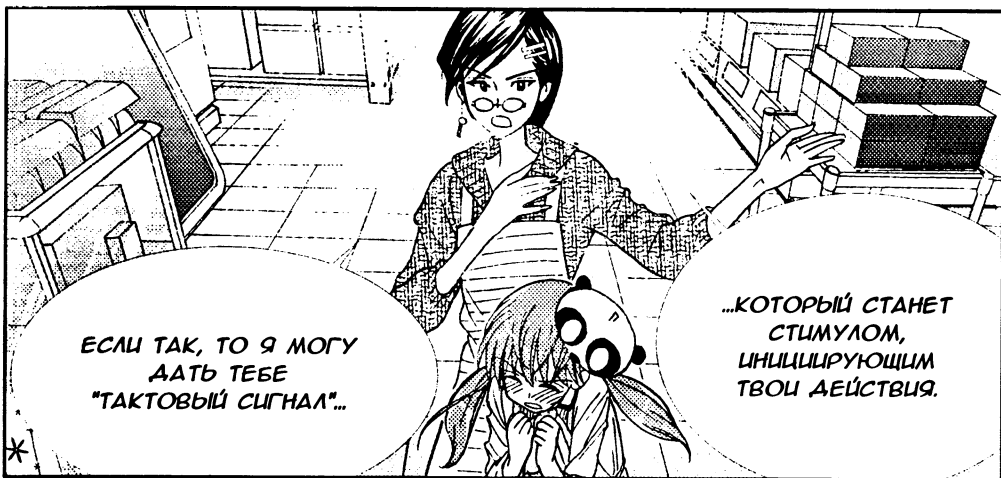
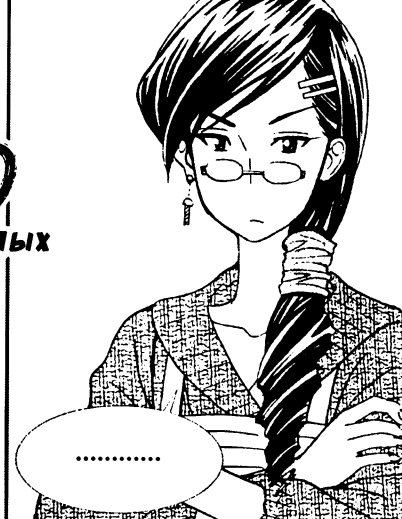
ЩУМО, ТЫ ЧТО,
ПОСЛЕДОВАТЕЛЬНАЯ
СХЕМА?

ДО КАКИХ ПОР
ТЫ СОБИРАЕШЬСЯ
ПОВТОРЯТЬ ЭТО "ТЕКУЩЕЕ
СОСТОЯНИЕ", КОТОРОЕ
ОДНАЖДЫ ЗАПОМНИЛА?
МОЖЕТ БЫТЬ, ПОРА
УЖЕ ПЕРЕХОДИТЬ
К СЛЕДУЮЩЕМУ...
ДЕЙСТВИЮ?



Я И САМА КОРЮ
СЕБЯ ЗА
НЕРЕШИТЕЛЬНОСТЬ.
НО...

ВАМ, НАВЕРНОЕ,
ЭТОГО НЕ ПОНЯТЬ.
А ВДУРУГ ОН
МЕНЯ НА САМОМ
ДЕЛЕ НЕ ЛЮБИТ?
Я ОЧЕНЬ
БЕСПОКОЮСЬ!



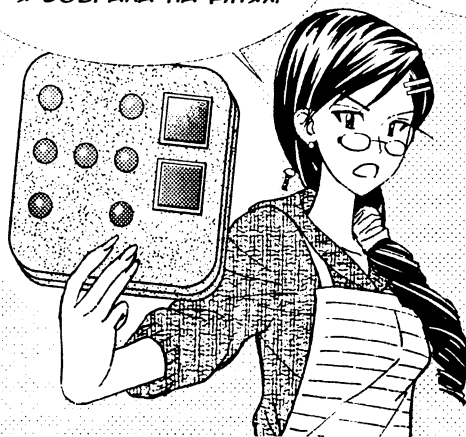
ЕСЛИ ТАК, ТО Я МОГУ
ДАТЬ ТЕБЕ
"ТАКОВЫЙ СИГНАЛ"...

...КОТОРЫЙ СТАНЕТ
СТИМУЛОМ,
ЛИЦИЦИРУЮЩИМ
ТВОИ ДЕЙСТВИЯ.



ВОТ ЭТО!
НОВЫЙ ЭЛЕКТРОННЫЙ
КУБИК, КОТОРЫЙ
Я СОБРАЛА НА ДНЯХ.

ДАВАЙ ПОПРОБУЕМ
БРОСИТЬ "КУБИКИ"
ОДНОВРЕМЕННО.



ЕСЛИ НА ТВОЁМ ВЫПАДЕТ БОЛЬШЕ, ТО ТВОЯ ПОЧАСОВАЯ СТРАВКА УВЕЛИЧИВАЕТСЯ НА 500 ЦЕН.

ЕСЛИ ЖЕ НА МОЁМ ВЫПАДЕТ БОЛЬШЕ, ТО...

!!!?

КАК НАСЧЁТ ЭТОГО? ТЫ ПРЯМО СЕЙЧАС ЕДЕШЬ В АЭРОПОРТ И ПРИЗНАЁШЬСЯ ТАКАБА В ЛЮБВИ?!

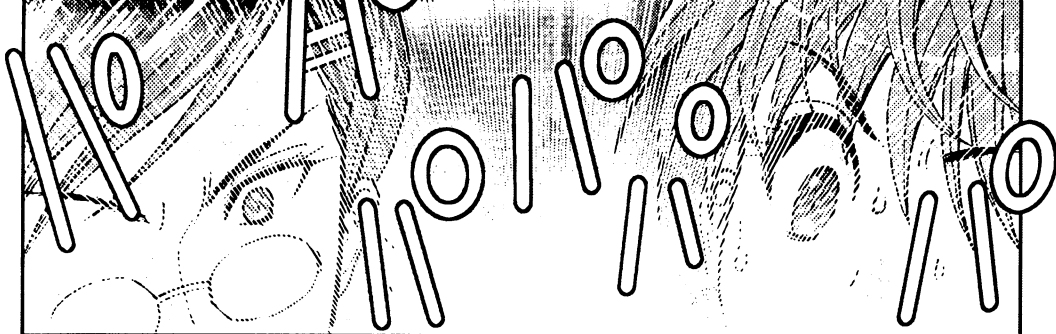
ДИРЕКТОР, ОСТАНОВИТЕСЬ! НЕЛЬЗЯ СМЕШИВАТЬ ТАКИЕ ВЕЩИ, КАК ЗАРПЛАТА ИЛИ ЛЮБОВЬ, С АЗАРТНОЙ ИГРОЙ!!

НЕТ. ВЕДЬ И БИЗНЕС, И ЛЮБОВЬ - ЭТО, ПО СУТИ, АЗАРТНАЯ ИГРА! ЭТО МОЯ ЖИЗНЕННАЯ ФИЛОСОФИЯ!

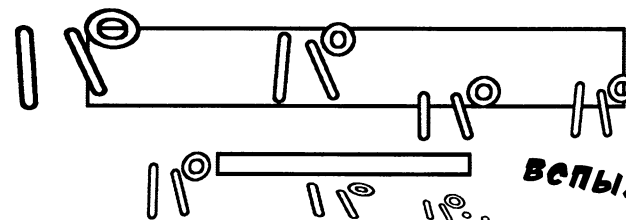
ИТАК, НАЧНЁМ. ЖМИ НА КНОПКУ!

ХВАТЬ

ЭХ!!!



**ВСПЫХ,
ВСПЫХ,**



**ВСПЫХ,
ВСПЫХ,**

ВСПЫХ, ВСПЫХ, ВСПЫХ...

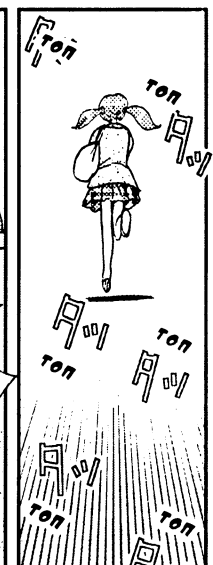
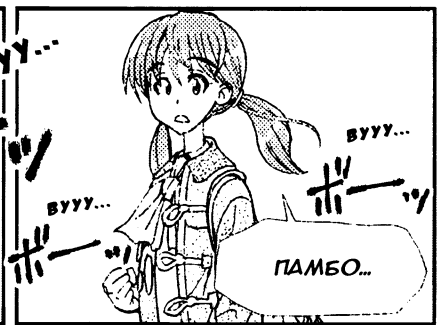
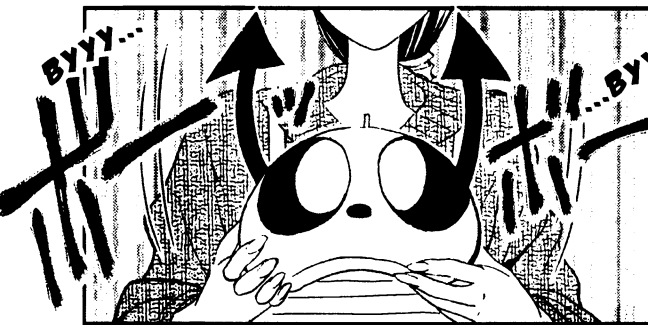
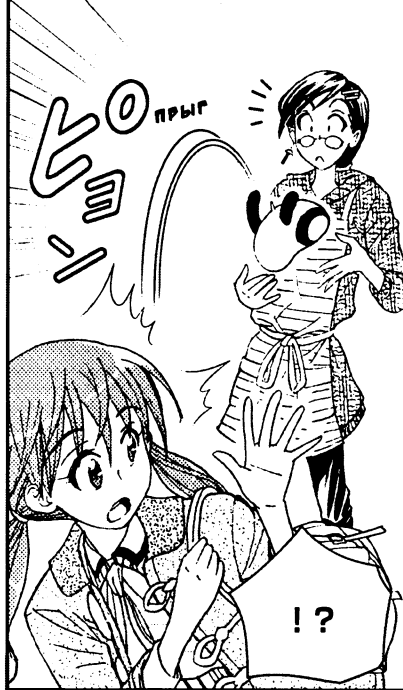


**ДИРЕКТОР,
МНЕ ПОРА!**



ММ...

**БОЛЬШОЕ
ВАМ СПАСИБО!!**



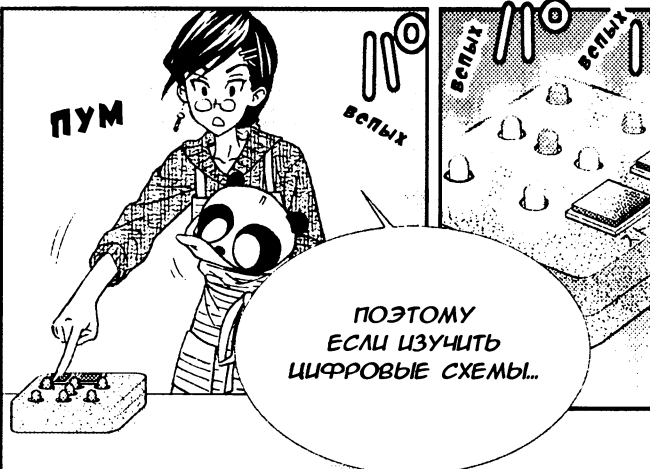
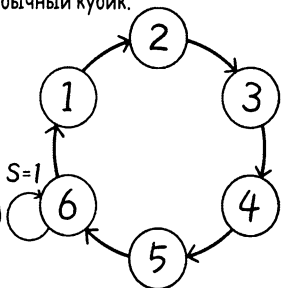


Диаграмма состояний электронного кубика, на котором обязательно выпадет «6»

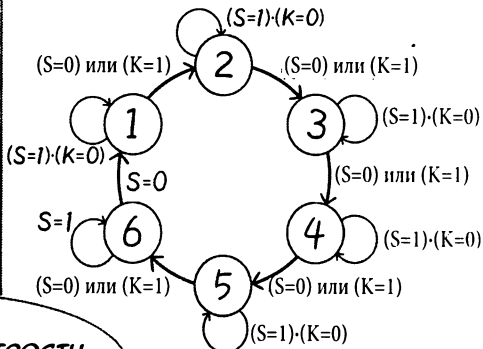
Вариант А

"6" выпадает всегда. В остальном внешне функционирует так же, как обычный кубик.



Вариант В

"6" обязательно выпадет только в том случае, если вход К хитрого переключателя установлен в 1.



ВПРОЧЕМ, ЭТИ ХИТРОСТИ НЕ ПРИГОДИЛИСЬ.

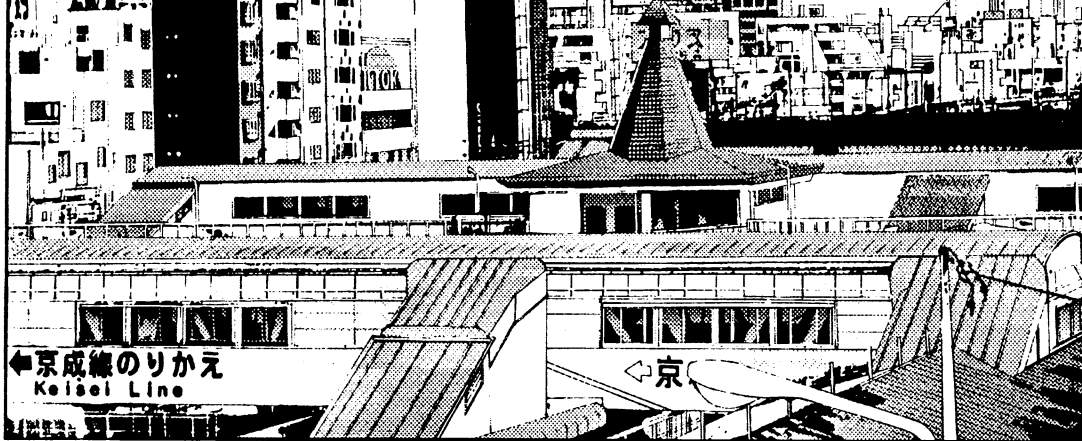
ВЕДЬ НА ОБЫЧНОМ КУБИКЕ ЦУМО...

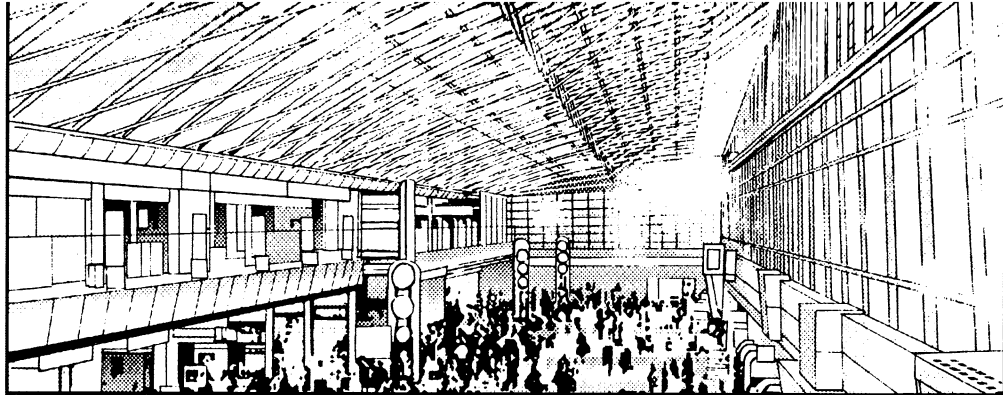
...ВЫПАЛО ВОТ ЭТО!

МОЖЕТ БЫТЬ, ЧЕЛОВЕЧЕСКИЕ ЧУВСТВА МОГУТ ВЛИЯТЬ ДАЖЕ НА ЭЛЕКТРОННЫЕ СХЕМЫ?

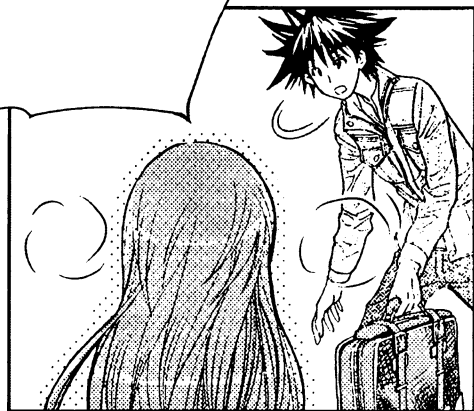
ХОТЯ Я В ТАКОЕ, РАЗУМЕЕТСЯ, НИКОГДА НЕ ПОВЕРЮ...

КОМИССИОННЫ
Бытовой
техники
WOL





ТАКАБА!!!

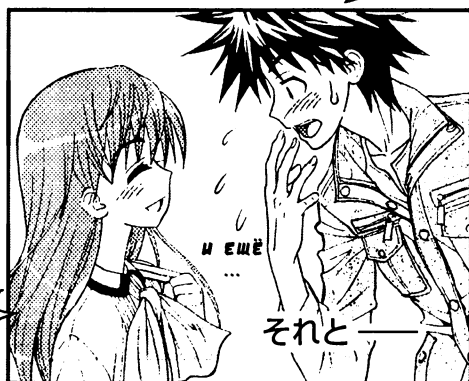




МЕНЯ ЗОВУТ АИСАКА ЦУМО.
МОЙ ОСОБЫЙ НАВЫК -
РАЗБОРКА ЭЛЕКТРОНИКИ.

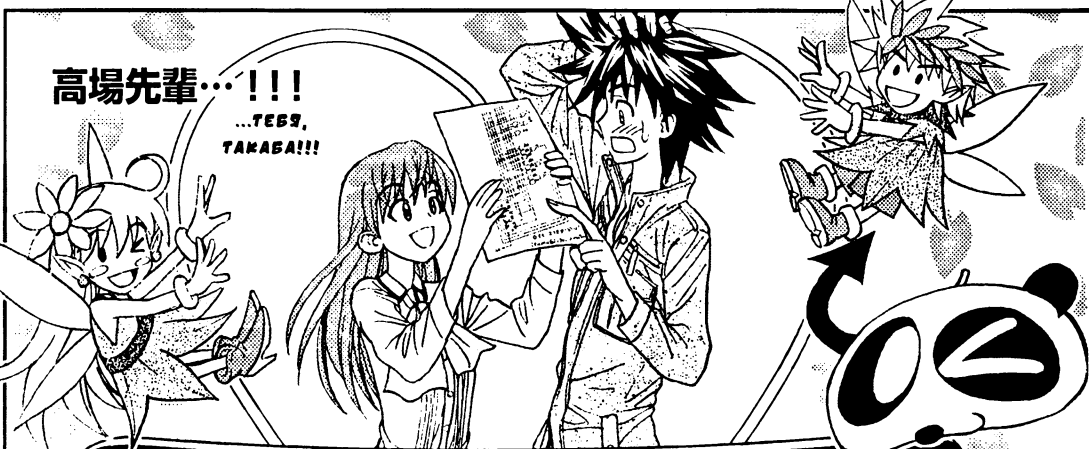


АА, Я ЛЮБЛЮ
БЫТОВУЮ
ЭЛЕКТРОНИКУ...



И ЕЩЁ
...

それと



高場先輩...!!!

...ТЕБЯ,
ТАКАБА!!!

Предчувствие истории о дистанционной любви при поддержке цифровых схем!!

КОНЕЦ.

Проектирование цифровых схем с помощью HDL

В этой книге были изложены основы проектирования цифровых схем. Описанные здесь принципы и методы обязательно пригодятся, несмотря на любой прогресс технологий проектирования.

Но чертят ли схемы в изученной вами нотации MPL, упрощают ли их с помощью карты Карно современные проектировщики цифровых устройств? Конечно же, нет. Современные цифровые схемы проектируются с помощью специально предназначенных для этого языков программирования высокого уровня. По этой причине работа проектировщика в наши дни практически ничем не отличается от работы программиста, пишущего программы для компьютера.

Хотя основным инструментом проектирования являются языки описания аппаратуры (HDL: Hardware Description Language), в последнее время начинают интенсивно применяться также и методы высокоуровневого синтеза (HLS: High Level Synthesis), подразумевающие проектирование в форме, более близкой к компьютерным языкам программирования. В процессе проектирования с использованием языков описания аппаратуры основное внимание уделяется тому, в какие моменты времени и в какие регистры будут записываться данные, каким образом они будут обрабатываться. По этой причине данная методика носит название «разработки на уровне регистровых задач» (RTL: Register Transfer Level).

Среди языков высокого уровня основными являются Verilog HDL (System Verilog) и VHDL, один из которых обычно и используют проектировщики.

К примеру, счётчик электронного кубика, на создание которого герои манги потратили столько усилий, может быть описан с помощью коротенькой программы на Verilog HDL, как показано на следующей странице. Здесь я не буду останавливаться на подробном описании синтаксиса, просто хочу, чтобы вы почувствовали атмосферу.



```

module saikoro(
input clk, rst_n, stop,
output reg [2:0] count);
always @(posedge clk or negedge rst_n) begin
    if(!rst_n) count <= 1;
    else if (!stop) begin
        if(count==6) count <= 1;
        else count <= count + 1;
    end
end
endmodule

```

Этот модуль электронного кубика (saikoro) имеет входы тактового сигнала (clk), сброса (rst_n), остановки (stop) и выход count, выражающий грань кубика.

Так как выход count является регистром, он объявлен с ключевым словом reg; кроме того, указано, что он представляет собой трёхбитное число, у которого старший бит — второй, а младший — нулевой. После ключевого слова always написано что-то вроде магического заклинания, однако на самом деле здесь указано, что схема должна срабатывать по переднему фронту тактового сигнала, а при появлении уровня L на входе сброса должен сразу же осуществляться сброс.

Основная функциональность кубика, однако, описывается в блоке if. Как и в других языках программирования, при выполнении условия, указанного в скобках после ключевого слова if, выполняется код, записанный после этих скобок, а в противном случае выполняется код, записанный после ключевого слова else.

Даже не очень хорошо разбираясь в синтаксисе языка, наверное, можно понять, что при сбросе счётчик становится равным 1; что если вход stop не равен 1, то выполняются манипуляции со значением счётчика: он становится равным 1, если был равен 6, либо увеличивается на 1 в остальных случаях. Возможность подобным образом понимать смысл, даже не обладая знаниями синтаксиса, и является сильной стороной описания с помощью языка.

После того как описание завершено, производится проверка путём логического моделирования. Логическое моделирование является одним из этапов процесса проектирования в CAD (Computer Aided Design — система автоматизированного проектирования, или САПР), во время которого в текстовом и в графическом виде отображаются изменения уровней на выходах и внутреннего состояния абстрактной модели схемы в ответ на определённые комбинации уровней на входах.



Если логическое моделирование прошло успешно, то опять же с помощью САД проводятся логический синтез, логическое сжатие, и затем, после упрощения, выводится результат в виде так называемого схемного набора данных (net list), представляющего собой список межвентильных соединений и обычно не предназначенного для изучения проектировщиком. Работа проектировщика здесь заключается в изучении и оценке информации о результатах синтеза, содержащей частоту тактового сигнала, количество логических вентилях, энергопотребление и т. п. Если вышеуказанные характеристики соответствуют техническому заданию, то задача проектирования считается выполненной, однако в тех случаях, если, например, рабочая частота оказывается ниже требуемой, или если схема содержит слишком много логических вентилях и т. п., приходится повторять этапы логического синтеза, логического сжатия, изменив настройки инструментов проектирования, или даже проектировать схему заново с самого начала.

Распространение методов проектирования с использованием языков описания аппаратуры и САД значительно облегчило работу проектировщика и существенно сократило время проектирования даже больших схем. Однако при проектировании с помощью языков описания аппаратуры разработчик должен описать, куда и в какие моменты времени передаются данные, что по-прежнему представляет собой сложную задачу. В связи с этим в настоящее время появляются такие методы проектирования, в которых достаточно изложить задачу в форме, близкой к компьютерным языкам программирования, а всю кропотливую работу по назначению способов и времени обработки данных выполняет САД.

Подобные методы проектирования носят название «высокоуровневого синтеза» (HLS: High Level Synthesis) и позволяют легко разрабатывать очень сложные схемы, выполняющие сжатие и распаковку видеоданных, зашифрование и расшифровывание, распознавание речи и т. п.

Развитие сред проектирования в наши дни сняло необходимость сосредоточиваться на мелких деталях проекта и позволило уделять больше внимания высокоуровневому проектированию, другими словами, больше размышлять о том, какая система была бы привлекательна для пользователя.

Таким образом, для современного проектировщика важна не столько способность тщательно выполнять кропотливую работу, сколько гибкость, позволяющая осваивать все возможности нового САД, назашоренность мышления и способность генерировать свежие идеи.

СПИСОК ЛИТЕРАТУРЫ



Использованная литература

- ред. Аисо Хидео, авт. Аmano Хидэхару, Такэфузди Ёсиясу,
"Цифровые схемы — это просто!", изд. Ohmsha, 2005 г.
 - авт. Хори Кэйтаро,
"Изучаем цифровые схемы по диаграммам», изд. Ohmsha, 2010 г.
-

Рекомендуемая литература

Для интересующихся проектированием цифровых схем:

- авт. Икаи Кунио, Хори Кэйтаро, Хонда Накадзи,
"Стандартное проектирование цифровых систем», изд. CQ, 1990 г.

Для желающих глубже изучить теорию логических схем:

- авт. Сасао Цутому,
"Теория релейных схем и логическое проектирование», изд Kindai kagaku sha,
2005 г.

Для интересующихся цифровыми элементами: КМОП ИС, триггерами и др.:

- авт. Аmano Хидэхару,
"Логические схемы с точки зрения проектировщика цифровой техники»,
изд. Corona Publishing, 2004 г.

ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ



ЛАТ.

AND — 35, 76

CAD (Computer Aided Desing) — 21, 45

D-защёлка — 187

D-триггер со входом разрешения — 192

D-триггер — 154

DIP (Dual Inline Package) — 21

JK-триггер — 190

n-канальный МОП-транзистор (n-МОП) — 86

NAND — 76

NOR — 76

NOT — 35, 76

OR — 35, 76

p-канальный МОП-транзистор (p-МОП) — 86

SN7432 — 33

T-триггер — 191

Verilog HDL (System HDL) — 208

VHDL — 208

А

Автомат Мура — 193

Автомат Мили — 193

Активный высокий — 70

Активный низкий — 70

Алгебра логики — 49

АЛУ (Арифметико-логическое устройство) — 142

Аналоговые схемы — 29, 52

Б

Безразличное состояние — 122, 143

В

Временная диаграмма — 165

Высокоуровневый синтез (HLS: High Level Synthesis) — 208

Вычитатель — 141

Д

Двоичные числа — 113, 143

Дешифратор — 140

Диаграмма состояний — 174, 193

Дизъюнктивная нормальная форма (ДНФ) — 91

З

Закон Мура — 87

Законы де Моргана — 50, 72, 91

И

ИС (интегральная схема) — 10, 24

К

- Кодирование — 109, 143
- Комбинационные схемы — 62, 91
- Комплиментарная структура металл-оксид-полупроводник (КМОП, CMOS) — 42, 85
- Компромисс — 139

Л

- Логические схемы — 24, 33
- Логический вентиль — 10, 24

М

- Мажоритарная схема — 91
- Модель Master-Slave (мастер и помощник) — 189

Н

- Нотация MIL — 63,91

П

- Полевой МОП-транзистор — 88
- Полный сумматор — 138
- Полупроводниковый кристалл — 39
- Последовательные схемы — 148, 193
- ППВМ (программируемая пользователем вентильная матрица, FPGA) — 21, 22
- Приоритетный шифратор — 140

Р

- Размер техпроцесса — 90
- Разработка на уровне регистровых задач (RTL: Register Transfer Level) — 208

Регистр — 168, 193

С

- Синхронный счётчик — 191
- Сумматор со сквозным переносом (каскадный сумматор) — 139

Т

- Таблица истинности — 60, 91
- Таблица переходов состояний — 178, 193
- Тактовый сигнал — 156
- Теория масштабирования — 90
- Транзисторы — 39
- Триггер (flip-flop, FF) — 153, 193

Ц

- Цифровые схемы — 30,52

Я

- Языки описания аппаратуры (HDL: Hardware Description Language) — 208

■ Об авторе

Аmano Хидехару

Доктор технических наук. В 1948 г. закончил технологический факультет университета Кэйо по специальности «Электротехника». В настоящее время — профессор кафедры информационных технологий научно-технологического факультета университета Кэйо.

<Основные работы>

"Изучение компьютерной архитектуры на практике" (совместное авторство), Baifukan

"Реконфигурируемые системы" (совместное авторство), Ohmsha

"О цифровых схемах понятным языком" (совместное авторство), Ohmsha

"Электронные схемы для цифровых устройств", Corona Publishing

■ Оформление манги: Office sawa

■ Сценарий: Савада Савако

■ Художник: Мэгуро Кодзи

■ DTP: Office sawa

