

НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

СХЕМОТЕХНИКА

№ 5 МАЙ
2002



АВТОЭЛЕКТРОНИКА

ИСТОЧНИКИ ПИТАНИЯ

ЦИФРОВАЯ ТЕХНИКА

ЭЛЕКТРОНИКА В БЫТУ

Схемотехника

№ 5 (19) май 2002

И. О. Главного редактора:

Юлия Герасимова

Редакционная коллегия:

Александр Фрунзе

Виктор Иовчик

Алексей Сигаев

Дизайн и верстка:

Ирина Ермопаева

Ирина Чикина

Отдел распространения:

Марина Трофимова

Юрий Царев

Сергей Лукин

тел.: (902) 112-9838

e-mail: compitech@mtu-net.ru

Отдел рекламы:

Юрий Рубичев

Адрес редакции:

121351 Москва, ул. Обручева,

д. 29, комн. 216

тел./факс: (095) 777-12-14

(902) 188-9192

e-mail: editor@dian.ru

www.dian.ru

Издатель и учредитель

ООО "ИД Скимен"

Отпечатано в ЗАО "Красногорская
типография"

143400 МО, г. Красногорск,

Коммунальный квартал, 2

тел: (095) 562-0433

Тираж 4 500 экз.

Заказ № 1099

Журнал зарегистрирован в
Министерстве РФ по делам печати,
телерадиоаещания и средств
массовых коммуникаций.
Рег. № ПИ77-5262

Редакция не несет ответственности
за информацию, приведенную
в рекламных материалах

Полное или частичное
воспроизведение материалов
допускается только с разрешения
ООО "ИД Скимен"

Информацию о подписке см. на
последней странице журнала

Цена свободная

Содержание

Аудиотехника

Активные фильтры нижних частот для сабвуфера

2

Источники питания

Преобразователь напряжения для питания мультиметра

5

Обратная связь в многоканальных импульсных
обратноходовых преобразователях напряжения

6

Основы схемотехники

Микроконтроллеры? Это же просто!

8

Связь и сетевые технологии

Синхронизация передачи данных: способы кодирования

13

Синхронизация псевдослучайных последовательностей

16

Схемотехника универсальных технологических контроллеров.

Цифро-аналоговые преобразователи

18

Софт

Анализ целостности сигналов на печатных платах

24

PSpice-модели цифровых устройств

29

Цифровая техника

Преобразователи уровней дискретных сигналов

постоянного тока

32

Быстродействующий диспетчер персонального компьютера

RS-232C — RS232C/RS485

34

Генератор прямоугольных импульсов на основе AVR

38

Микросхемы контроллеров ЖКИ фирмы Holtek

и их применение

40

FSK: сигналы и их демодуляция

45

Электроника в быту

Автоматы световых эффектов от «А» до «Я»

50

Автомат для водонагревателя

54

Простая всеволновая антенна

57

Справочный листок

Отечественные ИС стандартной логики. Серии 5554 и 5514

57

МАСТЕР КИТ

Универсальный контроллер электромеханического замка

59

Активные фильтры нижних частот для сабвуфера

В предыдущей публикации [1] были рассмотрены пассивные ФНЧ для автомобильных сабвуферов. Из-за большого затухания в полосе пропускания, несмотря на простоту и другие преимущества, их применение носит ограниченный характер. В основном их используют для усовершенствования уже имеющихся усилителей. Для изготовления новых конструкций желательно, чтобы сам фильтр имел единичный коэффициент усиления, поскольку его обычно включают после предварительного усилителя, где уровень сигнала составляет 50...250 мВ.

Этому требованию в полной мере удовлетворяют активные фильтры нижних частот (ФНЧ). Лучшее всего их сделать на современных операционных усилителях. Малый собственный уровень шума, широкий диапазон питающих напряжений, малое

энергопотребление, высокий коэффициент усиления, низкая цена выгодно отличает ФНЧ на ОУ от аналоговичных устройств на транзисторах.

Для облегчения расчета в настоящей статье приведены уже отработанные схемы ФНЧ (рис. 1-13) вмес-

те с их амплитудно-частотными характеристиками. Фильтры были смоделированы для «стандартной» частоты среза, равной 100 Гц. Все они — второго порядка. Для большего удобства оценки затухания вне полосы пропускания на всех графиках есть контрольная точка «а». Останется только выбрать приглянувшийся фильтр и включить его между выходом предварительного и входом оконечного усилителей. При исправных деталях и правильном монтаже все ФНЧ будут работать без налаживания.

Операционные усилители могут быть любые с внутренними цепями коррекции, например К140УД6, К140УД7. Очень хорошо и стабильно себя ведут К544УД1, а, кроме того, у них и

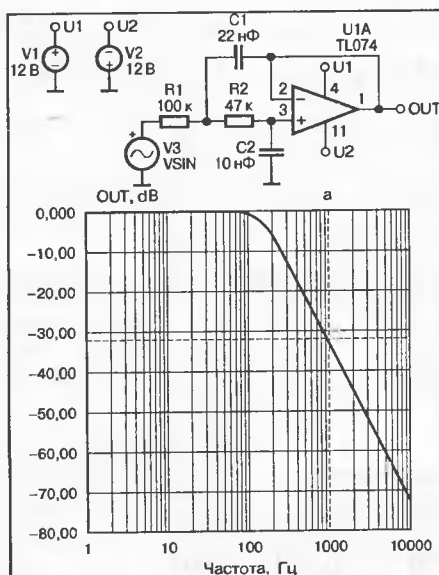


Рис. 1. Sallen-Key (SK) (1) non inverting

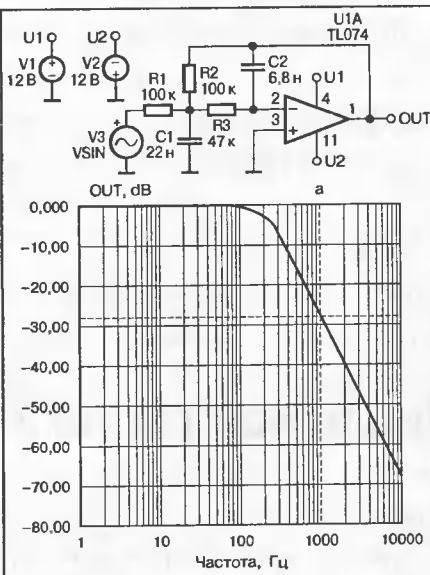


Рис. 3. Multiple feedback (MFB) (1) inverting

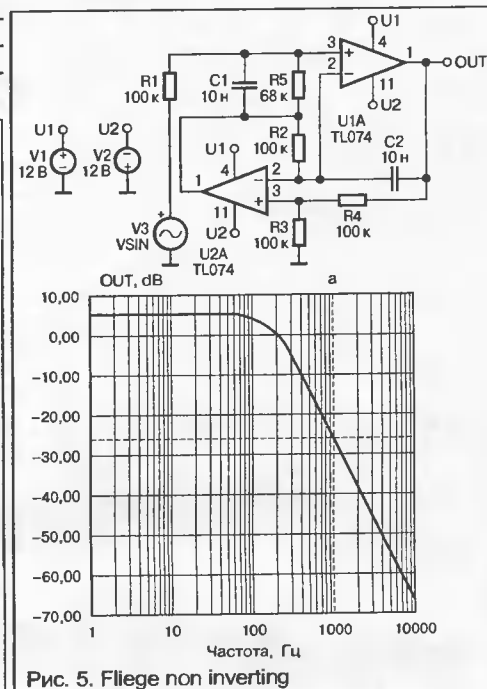


Рис. 5. Fliege non inverting

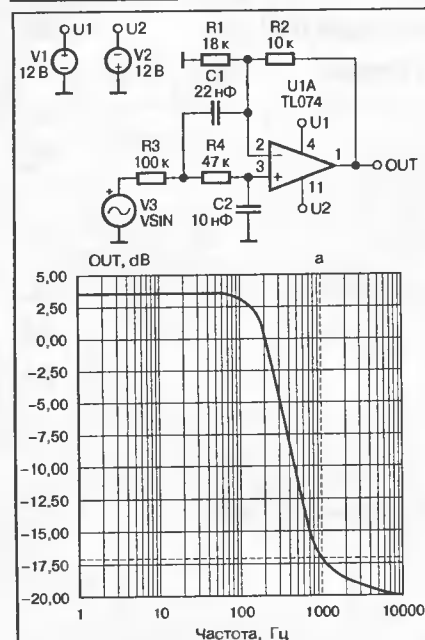


Рис. 2. Sallen-Key (SK) (2) non inverting

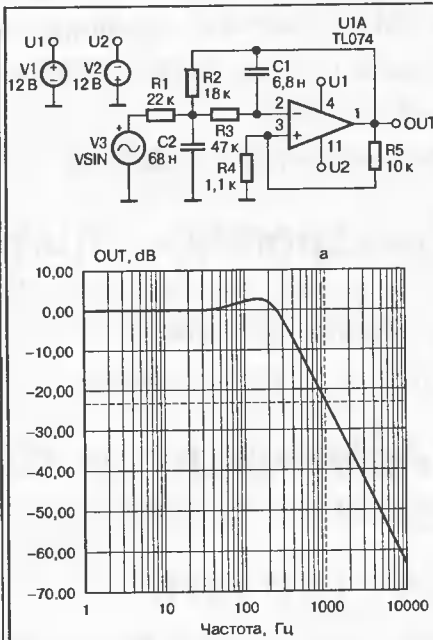


Рис. 4. Multiple feedback (MFB) (2) inverting

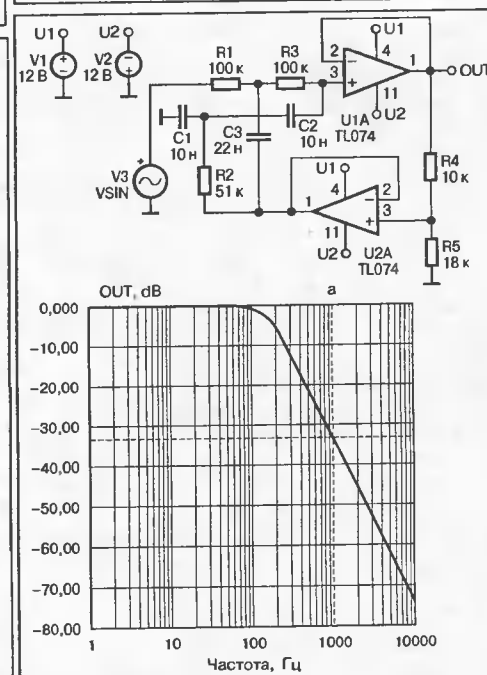


Рис. 6. Twin-T non inverting

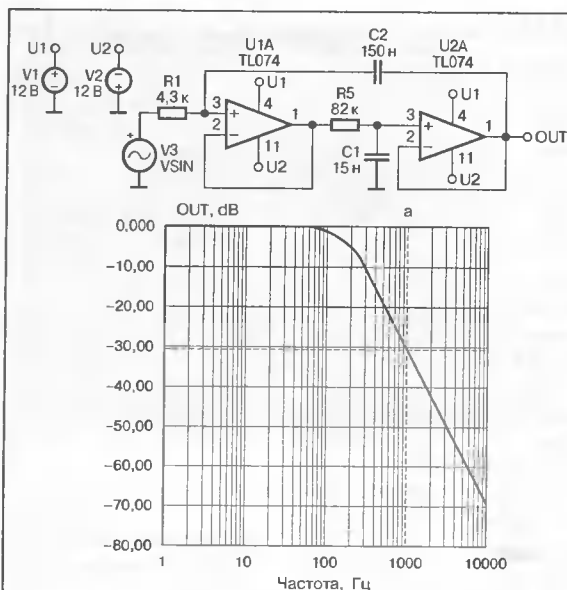


Рис. 7. Bash non inverting

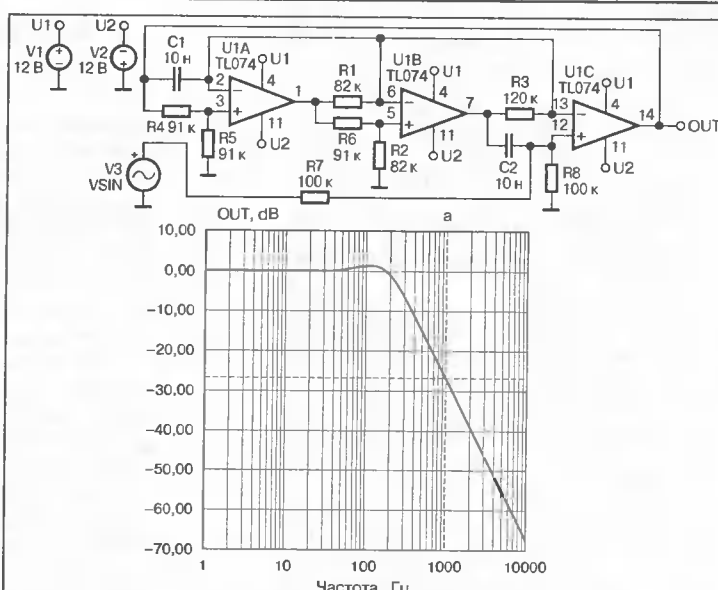


Рис. 10. Mikhael-Bhattacharyya (MB) non inverting

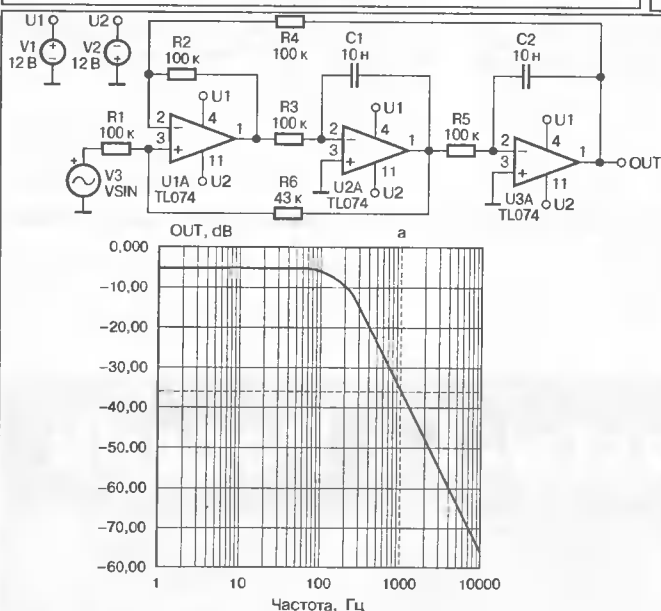


Рис. 8. KHN non inverting

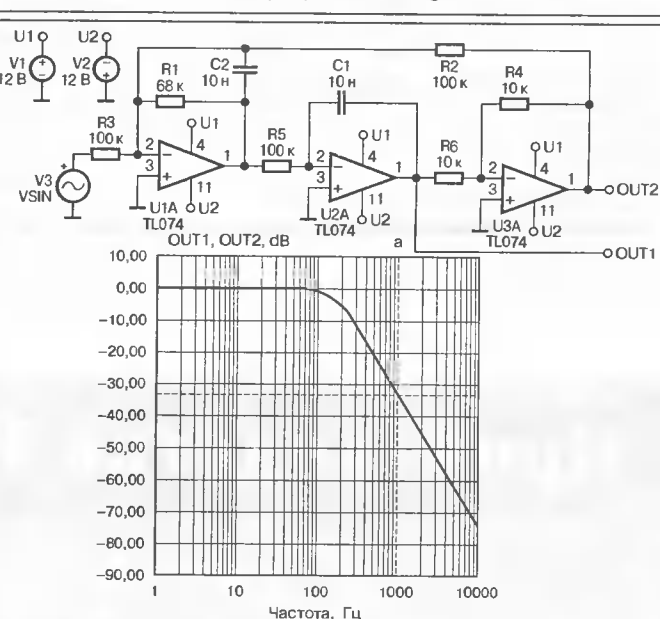


Рис. 11. Tow-Thomas (TT) non inverting

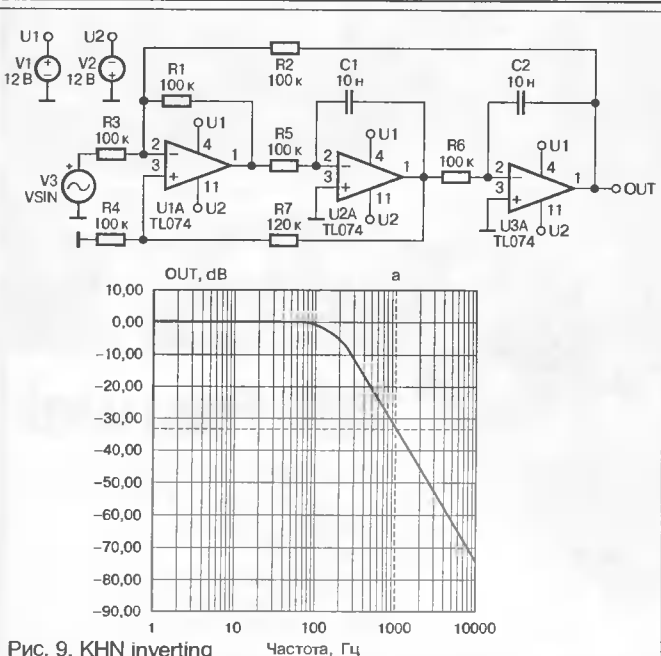


Рис. 9. KHN inverting

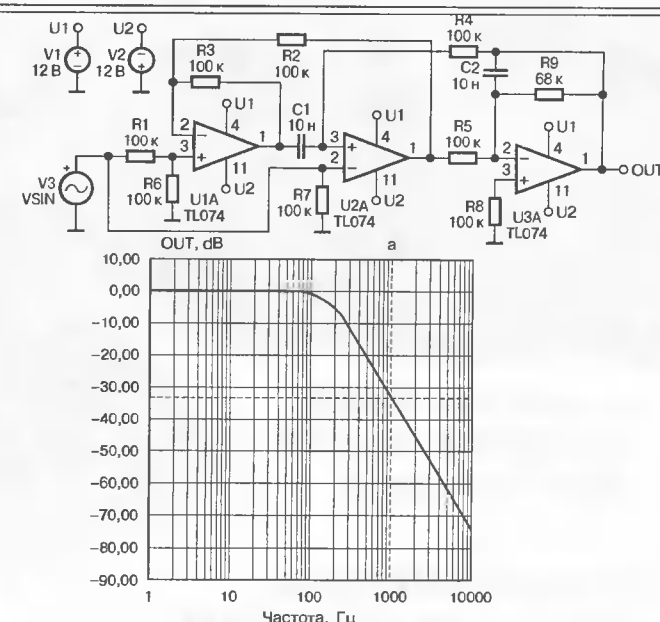
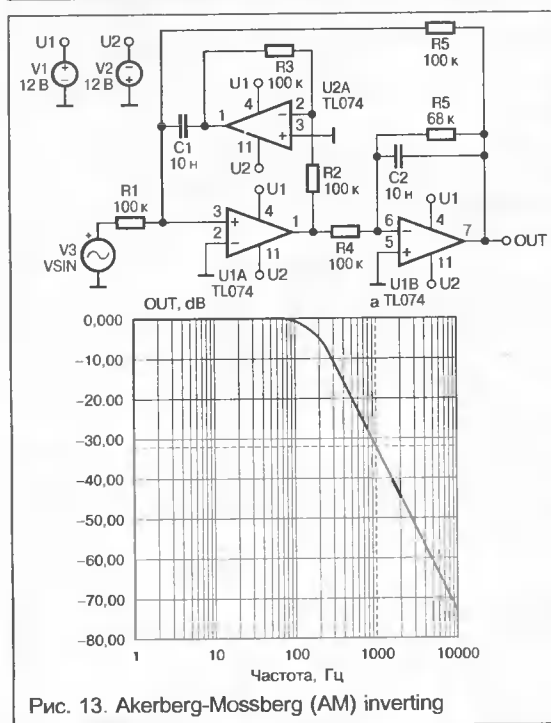


Рис. 12. Berka-Herpy (BH) non inverting



малый собственный шум. Если питание производится от однополярного источника, например в автомобиле, среднюю точку можно получить от делителя из двух одинаковых резисторов 5...10 кОм. Точка соединения этих резисторов и будет средней точкой, обозначенной на схемах как «земля». Резистор, подключенный к минусу питания, необходимо зашунтировать оксидным конденсатором емкостью 100 мкФ.

В заключение хотелось бы отметить, что, если возникнет необходимость пересчитать номиналы деталей на другие частоты или использовать фильтры высших порядков, это легко сделать с помощью программы (рис. 14).



Александр Воробьев,
alex@hit.mldnet.com

Литература:

1. А. Воробьев. Сабвуфер автолюбителю. — Схемотехника, 2002, №3, с. 4, 5.

Производство Fastwel в России



**Контрактная сборка
электронных модулей
любой сложности**

**Заказные разработки
электронного оборудования**

Fastwel

ООО «ФАСТВЕЛ» Москва, 117313, а/я 242

Контрактная сборка
E-mail: product@fastwel.com
Телефон: (095) 234-0639

Заказные разработки
E-mail: sdesign@fastwel.com
Телефон: (095) 234-0639

Питание малогабаритных мультиметров от гальванических батарей при систематическом применении недешево, а от сетевых блоков питания — чревато их порчей. Автор предлагает преобразователь напряжения, гальванической развязка в котором исключает многие неприятности при работе с мультиметром.

Для питания таких приборов по большей части используются малогабаритные батареи напряжением 9 В типоразмера 6F22 («Крона»). Хорошие батареи производства ведущих фирм не отличаются дешёвизной, и у активно работающего электронщика затраты на химические источники ток быстро «догоняют» стоимость «среднего» мультиметра. Чтобы сэкономить на батарейках, некоторые пробуют питать свои приборы от блока питания с понижающим сетевым трансформатором и линейным стабилизатором на 9...10,5 В на выходе. К большому сожалению, такие попытки сэкономить нередко заканчиваются плачевно — мультиметр выходит из строя. Дело в том, что разделительный сетевой трансформатор имеет конечное сопротивление изоляции между первичной и вторичной обмотками и паразитную ёмкость между ними, которая может составлять от нескольких десятков до нескольких тысяч пикофард. Все это в лучшем случае приводит к тому, что показания на дисплее «фонят», в худшем — к повреждению БИС мультиметра.

При увеличении входного напряжения питания или увеличении сопротивления нагрузки напряжение на СЗ стремится возрасти. Это приводит к увеличению тока через светодиод оптрона U1, фототранзистор оптрона открывается больше, ток в его цепи возрастает. Транзисторы VT2, VT4 открываются сильнее, шунтируя базовые цепи ключевых тран-

На холостом ходу при напряжении питания 9 В преобразователь потребляет ток менее 4 мА. Если к выходу устройства

К одному преобразователю следует подключать не более одного прибора вне зависимости от потребляемого им тока.

[illegible]

Обратная связь в многоканальных импульсных обратных преобразователях напряжения

Стабилизация напряжения в импульсных обратных преобразователях может осуществляться за счет отрицательной обратной связи (ООС) с дополнительной обмотки трансформатора или подачей сигнала ООС из выходной цепи. Второй вариант обеспечивает большую стабильность выходного напряжения и меньшее выходное сопротивление. Однако, если выходных напряжений у преобразователя несколько, возникает вопрос, откуда взять сигнал обратной связи? Автор сравнивает несколько возможных способов формирования этого сигнала и пытается ответить на поставленный вопрос.

Построению обратноходовых преобразователей напряжения на микросхеме КР1156ЕУ5 посвящено достаточно много публикаций, например [1, 2]. Для сравнения различных вариантов подачи сигнала обратной связи был изготовлен макет преобразователя с входным напряжением 8 В и двумя выходными напряжениями +12 В и –12 В. Основная часть схемы устройства приведена на рис. 1. Микросхема включена стандартно как повышающей преобразователь, при этом накопительный дроссель заменен на разделительный трансформатор. Цепь VD1VD2 гасит выбросы на индуктивности рассеяния первичной обмотки. Резистор R3 ограничивает коллекторный ток предоконечного транзистора микросхемы и, соответственно, базовый ток выходного. Трансформатор преобразователя намотан на кольцевом сердечнике K20x16x6 из феррита марки M2000HM1-17. Перед намоткой алмазной ножовкой в кольцо пропилен зазор шириной 1,5 мм. Для сохранения прочности кольца в зазор эпоксидной смолой вклеен кусочек стеклотекстолита. В обратноходовых преобразователях очень важна максимальная связь между обмотками, поэтому они выполнены таким образом. Жгутом из семи проводов ПЭШО-0,35 намотано 47 витков, затем секции соединены попарно последовательно. Это образуют обмотки I, II и III по 94 витка с индуктивностью около 400 мкГн и обмотку IV — 47 витков и 100 мкГн.

Для изучения влияния схемы ООС поочередно один из выходов преоб-

разователя нагружался на резистор сопротивлением 100 Ом, при этом проверялось изменение выходного напряжения в каждом его канале. Результаты приведены в табл. 1. Отметим сразу, что при симметричной нагрузке оба выходных напряжения по абсолютной величине всегда изменялись одинаково.

Таблица 1

Канал	Нагруж. канал	Изменение выходного напряжения, В при варианте ООС				
		Рис. 2	Рис. 3, Общ., $-U_{\text{вых}}$	Рис. 3, $+U_{\text{вых}}$, $-U_{\text{вых}}$	Рис. 4	Рис. 5
$+U_{\text{вых}}$	$+U_{\text{вых}}$	-0,35	-0,31	-0,17	-0,24	0,85
	$-U_{\text{вых}}$	-0,01	+0,36	0,16	-0,01	-0,56
$-U_{\text{вых}}$	$+U_{\text{вых}}$	-0,01	0	0,16	0,11	1,32
	$-U_{\text{вых}}$	-0,34	0	-0,17	-0,11	-0,84
Макс. отклонение		-0,35	+0,36	-0,17	-0,24	1,32

Самый простой способ подачи сигнала обратной связи — с выпрямителя, питающегося от обмотки IV. В этом случае элементы, подключенные на рис. 1 к входу обратной связи «Сопр.» микросхемы DA1 (вывод 5) не устанавливались, была подключена цепь по схеме рис. 2. Этот вариант прост, широко используется в одно- и многоканальных источниках питания, обеспечивает хорошую стабильность при колебаниях входного напряжения, но выходное сопротивление преобразователя получается довольно большим. Его вполне можно применять при мало меняющихся токах нагрузки. Результаты изменения абсолютной величины выходного на-

проявления каждого канала при подключении нагрузки в тот же или другой канал приведены в третьей колонке таблицы. Каналы почти независимы — увеличение нагрузки одного из каналов практически не влияет на напряжение в другом.

Рис. 2

Лучшие параметры преобразователя обеспечивает подача сигнала отклонения выходного напряжения от номинального значения через транзисторный оптрон. Типовая схема формирования ООС при помощи так называемого «регулируемого стабилитрона» TL431 (КР142ЕН19А) приведена на рис. 3. Пока напряжение на входе рассматриваемой цепи мало и между управляющим входом микросхемы DA2 (вывод 1) и ее катодом (вывод 2) напряжение менее 2,5 В, ток этого катода (вывод 3) не превышает 1 мА. Весь этот ток проходит через резистор R8 и не вызывает свечения светодиода U1.2 оптрона U1. При

превышении входным напряжением порогового значения анодный ток DA2 резко увеличивается, включается светодиод U1.2 и открывается фототранзистор U1.1 (рис. 1). На вход обратной связи «Comp.» микросхемы DA1 поступает сигнал ООС и напряжение на том выходе преобразователя, к которому подключен вход цепи рис. 3, ста-

билизируется. Пороговое напряжение устанавливают резистором R6, рассчитывая его по формуле

$$R6 = R7(U_{BIX}/2,5 - 1).$$

В одноканальном преобразователе такая цепь стабилизирует выходное напряжение очень жестко. В многоканальном устройстве входы цепи рис. 3 можно подключить к выходу одного из каналов, например, к $-U_{\text{вых}}$ и общему проводу (4-я колонка таблицы). Сопротивление резистора R6 составит 9,1 кОм. В этом случае будет стабилизироваться напряже-

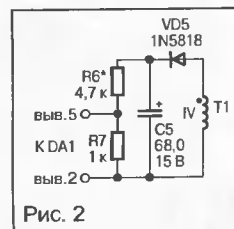


Рис. 2

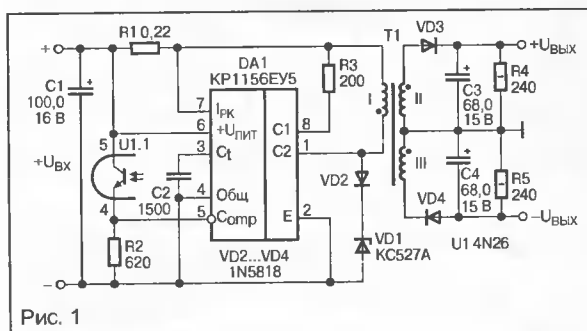


Рис. 1

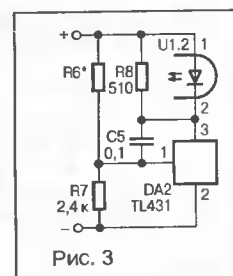


Рис. 3

ние 12 В именно в этом канале независимо от того, какой из них нагружен. Напряжение в канале $+U_{\text{ВЫХ}}$ уменьшается при подключении нагрузки к каналу $+U_{\text{ВЫХ}}$ и увеличивается при подключении к каналу $-U_{\text{ВЫХ}}$, причем весьма заметно.

Этот вариант стабилизации имеет смысл применять в тех случаях, когда для одного из каналов ($+U_{\text{ВЫХ}}$) не важна высокая стабильность напряжения, или если по одному из каналов ($-U_{\text{ВЫХ}}$) потребление постоянно. Нередко его также используют, когда мощность, требуемая от одного из каналов, намного

больше, чем от другого. В этом случае цепь, изображенная на рис. 3 подключают к первому из них, а во второй устанавливают линейный стабилизатор напряжения.

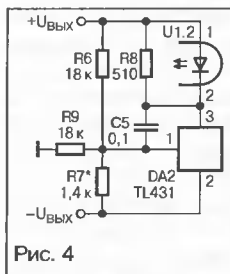


Рис. 4

Если цепь отрицательной обратной связи подключить между выходами $+U_{\text{ВЫХ}}$ и $-U_{\text{ВЫХ}}$, стабильным будет суммарное напряжение каналов (5-я колонка таблицы). Сопротивление резистора R6 в этом случае составит около 20,5 кОм. Изменение выходного напря-

жения каждого канала примерно вдвое меньше, чем у предыдущего варианта, однако ни одно из них не является жестко стабилизированным.



Рис. 5

напряжение в канале $-U_{\text{ВЫХ}}$ и суммарное напряжение двух каналов. Соотношение роли напряжений определяется резисторами R6 и R9, в данном случае равными. Изменяя эти резисторы, можно менять влияние каналов друг на друга, приближаясь к одному из рассмотренных выше вариантов.

В литературе можно встретить еще один способ стабилизации выходных напряжений, показанный на рис. 5. Хотя при симметричной нагрузке все варианты по схемам рис. 3-5 обеспечивают весьма высокую стабильность выходных напряжений, при «перекосе» этот способ резко уступает им. Более того, при увеличении нагрузки в кана-

ле $+U_{\text{ВЫХ}}$ выходное напряжение обоих каналов увеличивается, причем изменение это в несколько раз больше, чем в любом из предыдущих вариантов. Это означает, что выходное сопротивление по каналу $+U_{\text{ВЫХ}}$ имеет отрицательный знак, что может в ряде случаев быть крайне нежелательным.

Каждый из рассмотренных вариантов, кроме последнего, имеет свои преимущества и недостатки, и свои области применения. Автор надеется, что проведенное им сравнение поможет разработчикам многоканальных импульсных обратноходовых преобразователей грамотно подойти к выбору цепей стабилизации выходных напряжений.

Сергей Хвастин,
editor@ dian.ru

Литература:

1. Ю. Семенов. Схемотехника обратноходовых DC/DC преобразователей на основе KP1156EY5. — Схемотехника, 2001, №9, с. 2-5.
2. С. Бирюков. Преобразователи напряжения на микросхеме KP1156EY5. — Радио, 2001, №11, с. 38, 39, 42.

www.platan.ru

ПЛАТАН ЭЛЕКТРОННЫЕ КОМПОНЕНТЫ ВАШЕГО УСПЕХА

BOURNS INC Эффективный выбор для систем защиты и источников питания

Самовосстанавливающиеся предохранители серий MF-R, MF-S, MF-SM, MF-MSM www.bourns.com

- Предназначены для защиты электронных устройств от перегрузки по току или от перегрева.
- Принцип работы основан на свойстве резко увеличивать свое сопротивление под воздействием проходящего тока, превышающего номинальный рабочий ток, или под действием температуры окружающей среды, в несколько раз превосходящей номинальную, и автоматически восстанавливать свои первоначальные свойства после устранения этих причин.

Серия	Диапазон номинальных токов, А	Макс. рабочий ток, А	Сопр. в откл. сост., Ом	Область применения
MF-R	0.10-9.00	60	0.005-2.50	Общего назначения, автомобильная электроника.
MF-S, MF-LS, MF-LR	0.70-4.20	30	0.008-0.085	Защита аккумуляторных батарей от короткого замыкания, перегрева. Небольшые выводы для точечной сварки непосредственно на элемент батареи.
MF-SM	0.30-2.60	60	0.025-0.90	Для поверхностного монтажа. Компьютеры и периферия, автомобильная электроника.
MF-MSM	0.14-1.50	60	0.03-1.50	Для поверхностного монтажа, типоразмер 1812. Применяются в устройствах с высокой плотностью монтажа: жесткие диски, PCMCIA-карты и др.

Фирма «ПЛАТАН» является официальным дистрибутором фирмы «BOURNS» и поставляет полный ассортимент ее продукции со склада и на заказ.

В ПРОГРАММЕ ПОСТАВОК:

- Подстроечные и переменные резисторы
- Миниатюрные кнопки и переключатели
- Цифровые датчики угла поворота (энкодеры)
- Индуктивные компоненты

121351, Москва, ул. Ивана Франко, д. 40, к. 1, стр. 2
Тел./факс: (095) 73-75-999

Почта: 121351, Москва, а/я 100
E-mail: bourns@platan.ru

International TORRECIER

Infineon

MITSUBISHI ELECTRIC

EPCOS

Honeywell

MOTOROLA

intersil

muRata

CRYDOM

DAEJA VIMOP

Kingbright

ARGUSSOFT ЭЛЕКТРОННЫЕ КОМПОНЕНТЫ

Департамент Микроэлектроники

ANALOG DEVICES Новые серии прецизионных операционных усилителей

AMEL

TRACO POWER

BOURNS

CLARE

Sumida

JE-AN

HANTRONIX

Honeywell

SII Solo Instruments Inc.

muRata

Микросхема	Количество усилителей в корпусе	Входное сопротивление, МОм (тип.)	Входной ток самозатяжки (тип.)	КОСЧ, дБ (тип.)	Напряжение питания, В Мин. Макс.
AD8571/2/4	1/2/4	1	10 нА	140	+2.7 +5
AD8551/2/4	1/2/4	1	10 нА	140	+2.7 +5
OP117/12/177	1/2	15	0.5 нА	126	±2.5 ±15
OP177/12/177	1/2/4	20/30/30	5.5 нА	110	+2.7 ±15
AD8610	1	45	2 нА	95	±5 ±13
OP193/293/493	1/2/4	75	15 нА	116	+1.7 ±18
AD8601/2/4	1/2/4	80	0.2 нА	83	+2.7 +6
AD8605/8/8	1/2/4	80	0.2 нА	95	+2.7 +6
OP191/291/491	1/2/4	80	30 нА	80	+2.7 +12

Хотите попробовать? Закажите бесплатные образцы у нас:

Офис в Москве:
Адрес: 129085, Москва, Проспект Мира, 95
Тел.: (095) 217-2437, 217-2519, 217-2505
Факс: (095) 216-95-42
E-mail: components@argusssoft.ru

Офис в Санкт-Петербурге:
Адрес: 191023, Санкт-Петербург, наб. кан. Грибоедова, 36
Тел.: (812) 314-3808, тел./факс: (812) 310-6234
E-mail: spb@argusssoft.ru

Интернет: WWW.ARGUSSOFT.RU

(Продолжение. Начало — №4/2001)

Микроконтроллеры? Это же просто!

Глава 5. Система команд микроконтроллеров X51

Итак, позади уже четыре главы, а мы лишь собираемся начать знакомство с системой команд МК. Увы, если бы я начал систематически излагать ее раньше, я бы здорово перегрузил вас незнакомой информацией, и не уверен, что все продрались бы через этот часток нововведений. Но по ходу изложения предыдущего материала я знакомил вас с командами, входящими в программы, оживляющими то или иное железо. Так что нельзя сказать, что вы понятия не имеете о командах микроконтроллера, и что материал этой главы будет вам в диковинку. На самом деле вам уже в той или иной степени знакомы более 50% команд МК семейства x51. И в настоящей главе я собираюсь лишь сгруппировать все команды в сводные таблицы, ознакомить вас с общепринятыми правилами краткой записи в этих таблицах содержания команд, коротко рассказать о тех командах, которые вам еще не знакомы, да дать о них некоторую дополнительную информацию (о количестве байт в той или иной команде и о времени их выполнения). Я не планирую давать исчерпывающую информацию по командам, как это сделано в книге, написанной А. В. Боборыкиным, Г. П. Липовецким и еще семерыми авторами («Однокристальные микроЭВМ. — М.: МИКАП, 1994»), и рекомендую при необходимости обращаться к ней (в этом аспекте она лучшая из того, что известно мне) или к иной литературе, в том числе и к той, ссылки на которую я привел в самой первой части статьи («Схемотехника», №4, 2001).

Что еще можно сказать о командах МК, прежде чем мы начнем более детальное знакомство с ними? Эти команды составляют, как вы уже знаете из предыдущего, язык ассемблера микроконтроллера. Ассемблер относится к так называемым машинно-зависимым языкам — он свой для каждого семейства МК и отличен от ассемблера любого другого семейства. В этом один из основных его недостатков — переход на новый микроконтроллер, вы вынуждены учить новый язык.

Второй недостаток ассемблера — образно говоря, малая плотность действий МК на страницу текста, написанного на этом языке. Например, в языке Паскаль вывод символа или строки символов на экран компьютера организовывается при помощи одной команды `writeln`. Если вы попытаетесь написать то же самое на ассемблере стоящего в компьютере процессора семейства x86, программа потребует примерно такого же количества строк, как и наши программы для работы с ЖК-индикаторами, рассмотренные в главе 4. А когда дело коснется арифметических вычис-

лений, длина программы на языке высокого уровня, к которым относится и Паскаль, и Си, становится просто неоправданно короче, чем той же программы на языке низкого уровня (т. е. ассемблере). Так что писать на языке высокого уровня оказывается гораздо выгоднее — программы получаются короче, понятнее, да и переносить их с одного процессора или контроллера на другой оказывается очень легко.

Так почему же мы с вами знакомимся с ассемблером, а не с тем или иным языком высокого уровня, версии которых существуют практически для любого контроллера, а уж тем более для такого распространенного, как x51? Вместо ответа приведу такой пример. Когда я приобрел на митинском рынке аналогичный 10-разрядному HT1610 16-разрядный индикатор HT1616, у меня возникли трудности в аппаратном сопряжении его с микроконтроллером. Для отладки продавец индикатора дал мне коды написанной им на Си программы, осуществлявшей вывод на его экран трех идущих с интервалом в пару секунд текстовых строк (эта программа нормально работала на демонстрационном образце). Так вот, объем кода составлял примерно 1,6 кБ. Когда я разобрался с сопряжением, я написал аналог этой программы на ассемблере. Она заняла 82 байта памяти программ плюс число байт, равное количеству выводимых в строках символов (что-то около 40, точно не помню). Другими словами, программа из 15 строк на Си после трансляции потребовала 1600 ячеек памяти программ, а ее ассемблерный аналог — на порядок меньше, байт 120. Впечатляет? Добавлю к этому, что и работают программы, написанные на ассемблере, в несколько раз быстрее аналогов, написанных на языках высокого уровня. Так что по части компактности кода и скорости действия с программой на ассемблере не может сравниться никакой Паскаль или Си.

Думаю, что теперь вам понятно, почему ассемблер, несмотря на все его неудобства, так и не вытеснен языками высокого уровня. И именно по причинам, упомянутым в предыдущем абзаце, программисты еще долго будут им пользоваться. Не будучи большими оригиналами, мы не составим исключения из этого правила, и поэтому приступим к общему знакомству с системой команд ассемблера МК x51.

Общие сведения о системе команд

Система команд x51 содержит 111 базовых команд, которые удобно разделить по функциональному признаку на пять групп: команды передачи данных, арифметических операций, логи-

ческих операций, передачи управления и операций с битами.

Большинство команд (94) имеет формат один или два байта и выполняется за один или два машинных цикла. При тактовой частоте 1 МГц длительность машинного цикла, как вы должны помнить, составляет 1 мкс.

При анализе команд в программировании принято разделять их на две части — код операции (КОП) и операнды. Код операции — это байт, соответствующий команде, а операнд — это тот объект (бит, байт и т. д.), над которым должно совершиться определенное КОП'ом действие. Например, рассмотрим хорошо известную вам команду занесения константы в регистр R1: `MOV R1,#7`. В памяти программ после трансляции этой команде будет соответствовать два байта — 79H и 07H. Первый байт — 79H — это КОП, соответствующий команде занесения константы в регистр R1, а второй байт — это операнд, т. е. то число, которое нужно в этот регистр занести. Как нетрудно догадаться, КОП для другой команды будет отличным от 79H. Например, команде занесения константы в регистр R2 соответствует КОП 7AH, команде сложения аккумулятора с регистром R7 — соответственно 6FH и т. д.

С кодом операции, надеюсь, все ясно. Ниже, по мере знакомства с командами, для всех их я приведу их КОП'ы, и у вас будет возможность увидеть, какой команде соответствует какой код операции. Здесь никакой сложности нет. А вот с операндами будет чуть сложнее, чтобы понять, что это такое, вам придется чуть-чуть поднапрячься.

С какими объектами работают команды МК? Вы уже видели, что с битами, 8-битовыми и 16-битовыми числами, а также с 8-битовыми или 16-битовыми адресами ячеек памяти, где эти числа хранятся. Давайте вспомним те команды, с которыми мы ранее уже знакомы. Мы неоднократно заносили 8-битовые константы в аккумулятор или в регистры, используя команды типа упомянутой `MOV R1,#7`. В главе 3 в регистр DPTR командой `MOV DPTR,#FF25H` мы занесли 16-битовую константу (в данном примере — 0FF25H, что равно 65317 дес.). Мы переносили в аккумулятор числа из ячеек памяти данных (например, командами типа `MOV A,AD00`), а также осуществляли пересылку данных в обратном направлении (в данном случае командой `MOV AD00,A`). Мы пересылали биты из CY в регистры-защелки портов, в разряды некоторых регистров, и наоборот (например, командами типа `MOV P3.1,C` или `MOV C,ACC.5`). И, наконец, мы вызывали подпрограммы командой типа `LCALL DEL4MS` (в последнем случае имя DEL4MS после трансляции превратится в двухбайтовый, т. е. 16-битовый адрес первой команды этой подпрограммы).

Теперь посмотрим, что является операндом в каждом из рассмотренных

выше случаев. В первом (`MOV R1,#7`) операндом является помещаемая в регистр 8-битовая константа — в данном случае 7, но вместо семерки, как вы понимаете, после знака # могло стоять любое число от 0 до 255. Во втором случае (`MOV DPTR,#0FF25H`) операндом являлась 16-битовая или двухбайтовая константа. В третьем случае (`MOV A,AD00`) операнд — это содержимое ячейки, адрес которой определен символическим именем AD00. Почему именно содержимое ячейки, а не записанный через запятую после аккумулятора ее адрес, ведь он также 8-битовый (от 0 до 255; в нашем конкретном случае директивой AD00 .EQU 30H мы присваивали ему численное значение 30H)? Думаю, долго объяснять это «почему» не надо — команда пересылает в аккумулятор именно содержимое ячейки, а не ее адрес. Так что подчеркну еще раз, не адрес в данном случае является операндом, а содержимое ячейки памяти с этим адресом. Как нетрудно догадаться, тот же самый операнд и у команды `MOV AD00,A`. В командах `MOV P3.1,C` или `MOV C,ACC.5` операндом является переносимый бит, хотя адреса этих операндов (разрядов P3.1 и ACC.5) также 8-битовые числа. Как будет показано ниже, адрес бита P3.1, например, — 0B1H, а 5-го разряда аккумулятора (ACC.5) — 0E5H. А вот в команде `LCALL DEL4MS` операндом является 16-битовый адрес первой команды вызываемой подпрограммы, ибо команда `LCALL` работает не с данными, а с адресами.

Да, забыл, мы ведь уже знакомились с командой `SWAP A`. Она обменивает местами старшую и младшую тетрады аккумулятора (тетрада — это 4-битовое число). Эта (и еще одна) команда работает, таким образом, с 4-битовыми операндами.

Итак, я перечислил все возможные для x51 операнды — это единичные биты, 4-битовые, 8-битовые и 16-битовые числа, причем последние в некоторых случаях могут быть как данными, так и адресами. Отмечу еще, что есть команды, оперирующие с двумя операндами, например, `ANL addr,#data`. Эта команда осуществляет логическое И содержимого бита, размещенного в ячейке с адресом `addr`, и числа, записанного после знака #, т. е. оперирует с двумя байтами. Операнды здесь, как вы догадались, содержимое упомянутой ячейки и число `#data`.

После столь просторного объяснения, я полагаю, вы теперь будете легко ориентироваться в терминологии КОП'ов, операндов и их адресов. Для завершения терминологического ликбеза в области системы команд вспомним еще кое-что о методах адресации, без просторного объяснения которых вы не найдете ни одного описания языка ассемблера, неважно, для x51, AVR или PIC-контроллеров.

МК семейства x51 допускают четыре способа адресации к операндам. Ины-

ми словами, сказать микроконтроллеру, где искать тот или иной операнд можно четырьмя различными способами. Они носят названия прямая адресация, непосредственная адресация, косвенная адресация и неявная адресация.

Прямая адресация предполагает, что адрес операнда прямо указан в самой команде. Например, `MOV A,AD00` — адрес операнда (AD00) записан прямо в теле команды. (Если быть точным, то главное, что адрес явно присутствует не столько в приведенной парой строк выше мнемонике команды, сколько в виде второго байта в двух байтах, полученных после ее трансляции).

Еще пример — `SETB Px.y`. Здесь адрес операнда (Px.y) также прямо записан в теле команды (адрес, соответствующий Px.y, будет вторым байтом команды после трансляции). Дальше вы убедитесь, что прямая адресация — самый распространенный способ адресации.

Непосредственная адресация предполагает, что в теле команды записан сам операнд — байт или двухбайтовое число. Примеры — `MOV R1,#7` или `MOV DPTR,#FF25H`. Обратите внимание, в ранее рассмотренном случае (при прямой адресации) в теле команды в явном виде записан адрес операнда, а во втором (при непосредственной адресации) в теле команды в явном виде записан сам операнд. Чувствуете разницу?

Третий вид адресации — косвенная — несколько непривычен для начинающих. Но мы уже сталкивались с ним. Вспомним подпрограмму `DISPLY` в главе 4:

```
DISPLY:
MOV   A,#AD00
ADD   A,R0
MOV   R1,A ;R1=AD00+R0
MOV   A,@R1
и т.д.
```

Обратите внимание на команду `MOV A,@R1`. Она помещает в аккумулятор число из ячейки внутреннего ОЗУ, адрес которой находится в регистре R1. Зачем такие сложности? Бывают случаи, когда мы заранее не знаем, из какой ячейки нужно будет взять операнд для той или иной операции. Как, например, в подпрограмме `DISPLY`: нам для отображения нужно взять данные из ячейки AD00+p, где p=0...7. Точнее, в одном случае данные нужно брать из ячейки с адресом 31H, в другом — 32H, в третьем — 33H, и т. д. Конкретное же значение этого адреса зависит от числа, находящегося в момент вызова подпрограммы в регистре R0, и является суммой этого числа и числа #AD00. Другими словами, значение адреса заранее неизвестно и является результатом вычисления. Именно для подобных случаев и придумана косвенная адресация. Адрес операнда вычисляется по заданному вами алгоритму и помещается в определенный регистр, в данном случае в R1. А далее вы используете команду,

которая предписывает МК найти, какой адрес хранится в этом регистре, после чего перенести в аккумулятор данные из той ячейки внутреннего ОЗУ, адрес которой был найден в нем (т. е. в R1). Подобный метод адресации, когда адрес ячейки памяти, содержащей операнд, находится в каком-либо регистре, носит название косвенной адресации (адрес мы находим косвенно, при помощи R0, R1 или DPTR).

Признаком косвенной адресации является знак @ перед соответствующим регистром (R0, R1 или DPTR). В системе команд x51 адреса для режима косвенной адресации могут быть взяты только из этих трех регистров. Отмечу, что именно использованием в командах с косвенной адресацией регистры R0 и R1 отличаются от остальных регистров общего назначения (R2-R7). В основном возможности всех этих регистров одинаковы.

С косвенной адресацией, надеюсь, все ясно. Приступим теперь к знакомству с неявной адресацией. Вспомним для этого команду `SWAP A`. Она, как вы уже знаете, обменивает местами старшую и младшую тетрады аккумулятора. И хотя в мнемонике команды есть упоминание об аккумуляторе (отдельно стоящая после слова `SWAP` буква A), в коде, полученном после трансляции, этой команде соответствует лишь код операции (0C4H), без каких-либо байтов, прямо или косвенно указывающих на аккумулятор. Другими словами, в коде, соответствующем команде `SWAP A`, нет явных ссылок на аккумулятор. Есть только их (кода и аккумулятора) неявная связь, вытекающая из описания этой команды. Отсюда и название — неявная адресация. Отметим, что этот тип адресации, как вы дальше увидите, характерен для большинства команд, работающих с аккумулятором и с регистрами общего назначения.

Ну вот и все о способах (или методах) адресации МК x51. Отмечу, что у микроконтроллеров некоторых других семейств встречается гораздо большее их количество. Но, разобравшись с вышеописанными основными способами, вы вполне в состоянии постигнуть и все остальные. По крайней мере, если они внятно описаны в книжке, по которой вы будете с ними знакомиться...

Группа команд передачи данных

Команды, входящие в эту группу, сведены в табл. 1.

Рассмотрим их поподробнее. Первой идет уже не раз встречавшаяся вам команда пересылки данных в аккумулятор из регистра общего назначения. Обращаю ваше внимание на то, что за этой одной строкой в таблице стоят на самом деле целых 8 команд: `MOV A,R0`; `MOV A,R1`; `MOV A,R2`; `MOV A,R3`; `MOV A,R4`; `MOV A,R5`; `MOV A,R6`; `MOV A,R7`. Дабы не перегружать таблицу перечислением мнемоник всех 8 команд этой подгруп-

Таблица 1. Команды передачи данных

НАЗВАНИЕ КОМАНДЫ	МНЕМОНИКА	КОП	БАЙТ	ЦИКЛОВ	ОПЕРАЦИЯ
Пересылка в аккумулятор из регистра (n=0...7)	MOV A,Rn	11101rrr	1	1	(A) ← (Rn)
Пересылка в аккумулятор прямоадресуемого байта	MOV A,ad	11100101	2	1	(A) ← (ad)
Пересылка в аккумулятор байта из внут.ОЗУ (i=0, 1)	MOV A,@Ri	1110011i	1	1	(A) ← ((Ri))
Загрузка в аккумулятор константы	MOV A,#data8	01110100	2	1	(A) ← #data8
Пересылка в регистр из аккумулятора	MOV Rn,A	11111rrr	1	1	(Rn) ← (A)
Пересылка в регистр прямоадресуемого байта	MOV Rn,ad	10101rrr	2	2	(Rn) ← (ad)
Загрузка в регистр константы	MOV Rn,#data8	01111rrr	2	1	(Rn) ← #data8
Пересылка по прямому адресу аккумулятора	MOV ad,A	11110101	2	1	(ad) ← (A)
Пересылка по прямому адресу регистра	MOV ad,Rn	10001rrr	2	2	(ad) ← (Rn)
Пересылка прямоадресуемого байта по прямому адресу	MOV add,ads	10000101	3	2	(add) ← (ads)
Пересылка байта из внут.ОЗУ по прямому адресу	MOV ad,@Ri	1000011i	2	2	(ad) ← ((Ri))
Пересылка по прямому адресу константы	MOV ad,#data8	01110101	3	2	(ad) ← #data8
Пересылка во внут.ОЗУ из аккумулятора	MOV @Ri,A	1111011i	1	1	((Ri)) ← (A)
Пересылка во внут.ОЗУ прямоадресуемого байта	MOV @Ri,ad	0110011i	2	2	((Ri)) ← (ad)
Пересылка во внут.ОЗУ константы	MOV @Ri,#data8	0111011i	2	1	((Ri)) ← #data8
Загрузка указателя данных	MOV DPTR,#data16	10010000	3	2	DPTR ← #data16
Пересылка в аккумулятор байта из памяти программ	MOVC A,@A+DPTR	10010011	1	2	(A) ← ((A) + (DPTR))
Пересылка в аккумулятор байта из памяти программ	MOVC A,@A+PC	10000011	1	2	(PC) ← (PC) + 1; (A) ← ((A) + (PC))
Пересылка в аккумулятор байта из внеш.ОЗУ	MOVX A,@Ri	1110001i	1	2	(A) ← ((Ri))
Пересылка в аккумулятор байта из расширенного внеш.ОЗУ	MOVX A,@DPTR	11100000	1	2	(A) ← ((DPTR))
Пересылка во внеш.ОЗУ из аккумулятора	MOVX @Ri,A	1111001i	1	2	((Ri)) ← (A)
Пересылка в расширенное внеш.ОЗУ из аккумулятора	MOVX @DPTR,A	11110000	1	2	((DPTR)) ← (A)
Загрузка в стек	PUSH ad	11000000	2	2	(SP) ← (SP) + 1; ((SP)) ← (ad)
Извлечение из стека	POP ad	11010000	2	2	(ad) ← ((SP)); (SP) ← (SP) - 1
Обмен аккумулятора с регистром	XCH A,Rn	11001rrr	1	1	(A) ↔ (Rn)
Обмен аккумулятора с прямоадресуемым байтом	XCH A,ad	11000101	2	1	(A) ↔ (ad)
Обмен аккумулятора с байтом из внут.ОЗУ	XCH A,@Ri	1100011i	1	1	(A) ↔ ((Ri))
Обмен мл.тетрады аккумулятора с мл.тетрадой байта внут.ОЗУ	XCHD A,@Ri	1101011i	1	1	(A0...3) ↔ ((Ri0...3))

пы (давайте упомянутые 8 команд назовем подгруппой, кто нам мешает?), мы записали в ней одну обобщенную мнемонику MOV A,Rn, уточнив при этом, что n может принимать значения от 0 до 7. Сообщу также, что и при дальнейшем рассмотрении команд мы будем пользоваться подобным способом сведения нескольких (обычно 8 или 2) идентичных команд в единую строку в таблице. Так что, когда в пятой строке рассматриваемой таблицы вы увидите мнемонику MOV Rn,A, для вас уже должно быть очевидно, что за ней стоят команды: MOV R0,A; MOV R1,A; MOV R2,A; MOV R3,A и т. д., вплоть до MOV R7,A.

После сказанного, наверное, будет уже излишним уточнять, что когда в остальных группах команд вы встретите мнемоники типа ADD A,Rn или ANL A,Rn, вы догадаетесь, какие восемь команд описаны каждой из них.

Думаю, что с названием первой рассмотренной команды и с ее обобщенной мнемоникой все ясно. Приступим к анализу ее кода операции. В таблице в рассматриваемой строке стоит следующий КОП: 11101rrr. Как понимать эти три идущие подряд латинские буквы r? Давайте еще раз вспомним, что в двоичном представлении цифре 0 соответствует число 000B, цифре 1 — 001B, цифре 2 — 010B, ..., цифре 7 — 111B. Ну что, догадались? Правильно, представленный в двоичном коде КОП ко-

манды MOV A,R0 — 11101000 (rrr = 000), КОП команды MOV A,R1 — 11101001 (rrr = 001), КОП команды MOV A,R2 — 11101010 (rrr = 010), ..., КОП команды MOV A,R7 — 11101111 (rrr = 111). Так что запись 11101rrr также является обобщенной, вместо трех букв r мы должны вставить в эту мнемонику трехбитовый двоичный номер участвующего в этой операции регистра общего назначения.

Кстати, а почему rrr, а не, к примеру, jjj? Думаю, и это очевидно — и в русском, и в английском слово регистр начинается с r (r), так что ее использование выглядит гораздо более естественным, чем любой другой буквы.

И последнее в первой строчке: символическое описание выполняемой операции. Заключенная в скобки буква A означает *содержимое аккумулятора*, а Rn в скобках — *содержимое соответствующего регистра общего назначения*. Стрелка показывает направление пересылки. (A) ← (Rn) — это значит переслать в аккумулятор содержимое n-го регистра. Сравните это с названием команды, фигурирующим в начале рассматриваемой строки. Сходится?

Кажется, с первой командой все. А впереди еще 110! Ладно, справимся, не боги горшки обжигают. Обожаем что-нибудь и мы...

Вторая команда называется «Пересылка в аккумулятор прямоадресуемо-

го байта». То есть, в аккумулятор нужно переслать байт, адрес которого указан в мнемонике: MOV A,ad. Идущие в ней после запятой символы ad и означают адрес той самой ячейки-источника, записанный в явном виде (номер от 0 до 7FH), либо в виде символического имени (AD00 или что-то в этом роде).

С кодом операции, как мне кажется, все очевидно — 11100101B. Обратите внимание, что в отличие от предыдущей команды, рассматриваемая является двухбайтовой. При ее трансляции первым байтом будет упомянутый код операции, а вторым — как вы думаете, какой? Правильно, адрес этого самого прямоадресуемого байта.

Ну, и в завершение рассмотрения этой команды, краткое описание совершаемого ею действия: (A) ← (ad), что значит переслать в аккумулятор содержимое ячейки с адресом ad. На всякий случай еще раз повторю, что заключенная в скобки буква A означает *содержимое аккумулятора*, а ad в скобках — *содержимое ячейки памяти с этим адресом*. Стрелка показывает направление пересылки.

Третья команда — «Пересылка в аккумулятор байта из внутреннего ОЗУ». Другими словами, в аккумулятор нужно переслать из ОЗУ байт, адрес которого хранится в регистре R0 или R1. Обратите внимание, пересылаем байт все из того же внутреннего ОЗУ, из которого

пересылались данные и предыдущей командой. Но, заметьте, что хотя и источник пересылки, и приемник в обеих командах одинаковы, способы адресации в них различны. В предыдущей команде адрес прямо указывался в теле команды, а в текущей — извлекается из соответствующего регистра, т. е. указывается не прямо, а косвенно. Я намеренно уже в третий раз подробно объясняю, что такое способ косвенной адресации, т. к. это одно из самых сложных, на мой взгляд, понятий в системе команд.

Мнемоника команды — `MOV A,@Ri`, где *i* может принимать значение 0 или 1, а ее КОП — 1110011i. Думаю, нет смысла объяснять, что команде `MOV A,@R0` соответствует КОП 11100111B, а команде `MOV A,@R1` — КОП 11100111B. Команда однобайтовая, выполняется, как и предыдущие две, за один машинный цикл. Краткое описание совершаемого ей действия: $(A) \leftarrow ((Ri))$. Обратите внимание, *Ri* заключено в двойные скобки. Это не опечатка. Двойные скобки означают, что речь идет о содержимом ячейки, адрес которой и сам в свою очередь является содержимым того или иного регистра. Неясно? Попробую чуть подробнее.

В самом деле, как мы договорились ранее, (Ri) — это содержимое регистра *Ri*; но ведь это содержимое — не что иное, как адрес ячейки, принимающей участие в пересылке! Далее вспомним, что содержимое той или иной ячейки обозначается скобками, в которые заключен ее адрес. В нашем случае адрес — это (Ri) , т. е. запись, уже имеющая один комплект скобок. Заключив ее во вторые скобки, как раз и получим пересылаемый в аккумулятор операнд — содержимое ячейки, адрес которой находится в регистре *Ri* (т. е. является содержимым этого регистра). Думаю, что теперь уже причина появления в описании команды двойных скобок стала понятной всем.

И последняя из команд пересылки данных, в которых приемником байта является аккумулятор — это приведенная в четвертой строке команда загрузки в аккумулятор константы (`MOV A,#data8`). В теле команды включена константа, которая и переносится в аккумулятор. В мнемонике команды эта константа размещается после символа, называемого программистами “решетка” (*#*), а в кодах программы, полученных после трансляции вы обнаружите ее после кода операции (01110100B). Думаю, ясно, что `#data8` — это сокращенное обозначение 8-битового числа (0...255), которое в рассматриваемой команде и является этой самой константой. Обратите внимание, насколько эта команда схожа с командой пересылки в аккумулятор прямоадресуемого байта — обе они двухбайтовые, обе осуществляют пересылку данных в аккумулятор, обе в теле содержат 8-битовую константу. Но в команде `MOV A,ad` эта константа трактуется как адрес ячейки-источника, а в `MOV A,#data8` — именно как помещаемая в аккумулятор константа. Характер трактовки определяется,

повторюсь еще раз, наличием или отсутствием знака “решетки”: если его нет, то константа — это 8-битовый адрес, если есть — байт данных.

Также, думаю, очевидно и краткое описание совершаемого ей действия: $(A) \leftarrow \#data8$. Обратите внимание, `#data8` не заключено ни в какие скобки, т. к. это сами данные, а не адрес, где они находятся.

Мы очень подробно рассмотрели все команды пересылки данных, в которых приемником байта является аккумулятор. Их, как видите, всего четыре, и в табл. 1 они расположены в первых четырех строках. Следующие три строки — это команды пересылки данных, в которых приемником байта является регистр общего назначения (*Rn*, *n*=1-7). Думаю, всем уже очевидно, что каждая из этих строк вобрала в себя по 8 родственных команд, различающихся лишь номером регистра. Также, думаю, нет необходимости объяснять, что означают три латинские *g* в обобщенной мнемонике каждой из этих обобщенных команд, поскольку все совершенно идентично тому, что говорилось при анализе команды пересылки в аккумулятор из регистра.

Как видите, в регистр общего назначения можно переслать данные из аккумулятора (`MOV Rn,A`), из прямоадресуемого байта (`MOV Rn,ad`) или непосредственно загрузить в него константу (`MOV Rn,#data8`). Первая команда — однобайтовая, две другие — двухбайтовые, причем вторая, в отличие от первой и третьей (а также и четырех, рассмотренных ранее), выполняется не за один, а за два машинных цикла. Собственно, говорить об этих командах больше нечего — все, необходимое для их понимания, рассказано при анализе команд пересылки данных в аккумулятор. Так что двинемся дальше.

Следующие пять строк — это команды пересылки данных, в которых приемником байта является прямоадресуемая ячейка внутреннего ОЗУ с адресом *ad* (*ad* — это адрес той самой ячейки-приемника, заданный или в явном виде (номер от 0 до 7FH) или в виде символического имени — что-то типа AD00). Первые две из этих команд должны уже быть вам очевидны — пересылка в ячейку ОЗУ байта из аккумулятора (`MOV ad,A`) или из одного из РОН (`MOV ad,Rn`, *n*=1-7). Не столь очевидна третья команда, мудрено именуемая “Пересылка прямоадресуемого байта по прямому адресу”. Правда она не сложнее ранее рассмотренных. При ее выполнении микроконтроллер извлекает байт данных из прямоадресуемой ячейки памяти (т. е. из ячейки с указанным в теле команды адресом *ads*) и пересылает его в другую прямоадресуемую ячейку. Адрес второй ячейки — *add* — также прямо указан в теле команды, причем, что важно, указан в ней первым (вспомните, в одной из предыдущих глав я уже говорил, что в ассемблере x51 первым всегда записывается адрес приемника информации, а вторым — источника). Команда `MOV add,ads` — трехбайтовая, выполняется за два машинных цикла. Пожалуй, больше говорить о ней

нечего. Также, как и о двух оставшихся командах пересылки данных, в которых приемником байта является прямоадресуемая ячейка внутреннего ОЗУ — `MOV ad,@Ri` и `MOV ad,#data8`. Первая предписывает микроконтроллеру переслать в ячейку с адресом *ad* байт из ячейки, адрес которой хранится в регистре *R0* или *R1*, а вторая — загрузить в ячейку с адресом *ad* восьмибитовую константу.

Следующие три строки — это команды пересылки данных, в которых приемником байта является косвенно адресуемая ячейка внутреннего ОЗУ (т. е. адрес ячейки хранится в *R0* или в *R1*). Как видите, в косвенно адресуемую ячейку внутреннего ОЗУ можно переслать данные из аккумулятора (`MOV @Ri,A`), из прямоадресуемого байта (`MOV @Ri,ad`) или непосредственно загрузить в нее константу (`MOV @Ri,#data8`). Все, что касается этих команд, вы вполне в состоянии самостоятельно понять из информации, приведенной в табл. 1. Кстати, также как и то, что касается команды `MOV DPTR,#data16` — мы подробно рассмотрели ее в главе, посвященной регистрам микроконтроллера.

Поздравим себя, мы осилили 16 команд. На первый взгляд не так уж это и много, но лиха беда начало! Чем дальше мы будем продвигаться в их изучении, тем скорее это продвижение будет происходить, ибо все больше и больше материала будет вам уже знакомо. Надеюсь, я вас обнадежил. Но даже если и нет, все равно двинемся дальше.

Следующие две команды позволяют загрузить в аккумулятор байт данных из памяти программ. Первая из них — пересылка с использованием указателя данных (`MOVC A,@A+DPTR`) — нам уже знакома: помните, в главе 4 мы с ее помощью отыскивали байты, выводимые в 28-разрядный регистр сдвига матричного светодиодного индикатора HCMS-2xxx? Она выполняет следующее действие: складывает содержимое аккумулятора и *DPTR*, находит ячейку памяти программ с адресом, равным полученной сумме, извлекает из нее информацию и помещает извлеченное в аккумулятор. С названием, мнемоникой, количеством байт и временем выполнения команды вам все должно быть очевидно. Обратите внимание на краткое описание совершаемого ей действия: $(A) \leftarrow ((A) + (DPTR))$. Оно также должно быть вполне доступно для вашего понимания. $(A) + (DPTR)$ означает взять содержимое аккумулятора и регистра *DPTR* и сложить их. А далее нужно взять содержимое ячейки с адресом, равным полученной сумме (это обозначается заключением суммы $(A) + (DPTR)$ в скобки) и переслать его в аккумулятор, о чем свидетельствует направленная к (A) стрелка. Как видите, разобраться с правилами записи кратких описаний выполняемых командами действий весьма полезно. По ним можно безошибочно определить, что эти команды делают.

Вторая команда — пересылка с использованием программного счетчика *PC* (`MOVC A,@A+PC`). Пока она нам ни

о чем не говорит. Но посмотрите краткое ее описание, и сразу все станет ясно: встретив ее, МК увеличивает на 1 значение PC, затем увеличивает это значение с содержимым аккумулятора, находит ячейку памяти программ с адресом, равным полученной сумме, извлекает из нее информацию и помещает извлеченное в аккумулятор. Правда, хоть убейте, плохо представляю, зачем эта команда может быть полезна. Но не исключаю, что кто-нибудь из читателей придумает интересный вариант ее применения.

Еще хочу обратить ваше внимание на букву C в конце мнемоники самой операции (MOVX). Эта буква показывает, что пересылка осуществляется из памяти программ (C — первая буква слова Code, обычно ассоциируемого с кодами программы, т. е. с памятью программ). Нечто подобное мы уже видели, когда столкнулись с командой, извлекающей данные из внешней памяти данных (MOVX): там на конце мнемоники самой операции стояла буква X, что связано было со словом eXtended — расширенная, а в текущем контексте — внешняя. Кстати, именно к рассмотрению таких команд мы сейчас и перейдем.

Первая из них — пересылка в аккумулятор байта из внешнего ОЗУ. Ее мнемоника — MOVX A,@Ri. Сравните ее с рассмотренной выше MOV A,@Ri. Они почти одинаковы, разница лишь в том, что первая работает с внешним ОЗУ, а вторая — с внутренним. В обычной практике команда MOVX A,@Ri почти не используется, она полезна лишь при работе МК с микросхемой типа 8155, 8156, 1821PU10, применяемыми сейчас крайне редко. Обратные действия совершаются командой MOVX @Ri,A.

С командами MOVX A,@DPTR и MOVX @DPTR,A мы уже познакомились в третьей главе, и особо добавлять к сказанному там нечего. Поэтому перейдем к командам работы со стеком.

Первая (PUSH ad) называется загрузкой в стек. Мы уже говорили о том, как МК использует стек для хранения адреса возврата из подпрограмм. Но, помимо этого, в стеке можно также хранить и данные. Упомянутая команда переносит содержимое ячейки с адресом ad в стек, после чего эту ячейку можно использовать в любых других целях. По завершении операции, потребовавшей освобождения этой ячейки от хранившихся в ней данных, их можно вернуть из стека в ячейку обратно командой POP ad.

Для чего это нужно? Очень часто оказывается, что при выполнении тех или иных действий нам не хватает регистров общего назначения. Тогда можно попытаться сохранить в стеке содержимое одного или нескольких регистров, которое в текущей операции не используется, и воспользоваться ими, восстановив их содержимое после завершения операции. Но здесь есть одна проблема. В командах PUSH ad и POP ad нужно прямо указывать адрес ячейки. Если у аккумулятора и у регистра B эти адреса неизменны, то у регистров общего назначения они зависят от номера используемого банка PОН. Поэтому сохра-

нение в стеке содержимого регистров общего назначения может оказаться проблематичным. В свете сказанного у МК x51 в стеке обычно сохраняют при необходимости лишь содержимое аккумулятора и регистра B.

Обращаю ваше внимание на то, что если вы используете стек для хранения содержимого ячейки памяти, перенесите его туда командой PUSH ad, восстановить содержимое нужно до того, как вы вызовете какую-либо подпрограмму, т. к. вызов изменяет содержимое указателя стека, и при восстановлении содержимого в регистр попадут не те данные. Точно также, если сохранение совершается внутри какой-либо подпрограммы, то восстановить содержимое регистра (регистров) нужно до того, как наступит черед выполнения команды RET.

И еще по поводу работы со стеком. Легче всего представить себе принцип его работы следующим образом. Вспомните магазин автомата Калашникова или ему подобного оружия. Когда вы набиваете его патронами, каждый вставляемый в него патрон после вставки оказывается верхним, а тот, который был верхним до этого, опускается вниз. Предположим, вы набили его несколькими патронами — тремя, пятью, десятью, число в данном случае неважно. Вопрос: каким по счету в процессе стрельбы выйдет из магазина последний вставленный в него патрон? Ответ очевиден — первым. А предпоследний? Ясное дело, вторым. А самый первый? Конечно же, последним.

Точно также происходит процесс загрузки данных в стек и извлечения их оттуда. Последнее занесенное в него число должно извлекаться из него первым по счету, предпоследнее — вторым, предпредпоследнее — третьим и т. д. Поэтому, если вы вначале загрузите в стек, например, содержимое аккумулятора, а затем — регистра B, то извлекать их оттуда нужно в обратной последовательности, т. е. вначале восстановить содержимое B, а затем — аккумулятора. Если не соблюсти это правило, то после сохранения данных в стеке содержимое регистров перепутается со всеми вытекающими отсюда последствиями.

Ну вот и все про стек. Перейдем к последним четырем командам группы

команд передачи данных. Это команды обмена содержимого аккумулятора соответственно с содержимым регистра общего назначения (XCH A,Rn) прямоадресуемого байта (XCH A,ad) с косвенно адресуемым через R0 или R1 байтом из внутреннего ОЗУ (XCH A,@Ri). Объяснять здесь особо нечего, все, что нужно для понимания размещенной в табл. 1 информации, касающейся этих команд, выше уже подробно объяснено. А последняя команда рассматриваемой группы — XCHD A,@Ri. Она похожа на команду XCH A,@Ri, но, в отличие от последней, в XCHD A,@Ri обмениваются лишь младшие 4 бита аккумулятора с младшими четырьмя битами соответствующей ячейки памяти, в то время как старшие 4 бита и того и другого остаются неизменными.

Итак, мы закончили рассмотрение группы команд передачи данных. Кое-кому из вас после знакомства с первой из них наверняка показалось, что знакомство это будет очень долгим и трудным. Но как видите, чем дальше мы продвигались вглубь системы команд, тем легче это происходило. Так что наберитесь терпения, дальше это знакомство будет идти все легче и легче, и вы заметите не успеете, как я оповещу вас, что оно завершено, и что вы знакомы со всеми командами МК семейства x51.

Александр Фрунзе
alex.fru@mtu-net.ru

ARGUSSOFT ЭЛЕКТРОННЫЕ КОМПОНЕНТЫ

Департамент Микроэлектроники ОФИЦИАЛЬНЫЙ дистрибутор фирм:



Программируемые логические микросхемы

ATF15xxAS/ASL/AE — высокопроизводительные логические матрицы. Режим внутрисистемного программирования (ISP) позволяет изменять внутреннюю конфигурацию микросхемы непосредственно в печатной плате через порт JTAG. Микросхемы полностью совместимы с CPLD фирмы ALTERA серии EPM7xxK/EP3xxK, однако, имеют расширенную разводочную матрицу. Микросхемы программируются через кабель ATDH1150 фирмы ATMEL, также возможна загрузка через ByteBlaster фирмы ALTERA. Микросхемы с индексом AE имеют 3.3/5 В толерантные входы.

Название	Кол-во микросхем	Кол-во выв. выв.	Задержка распространения, нс	Кол-во ячеек	Аналог фирм ALTERA
ATF1502ASL/AE	32	36	7.5 - 25	44	EPM7032S
ATF1504ASL/AE	64	36/52/68	7.5 - 25	44/68/84/100	EPM7064S
ATF1506ASL/AE	128	68/84/100	7.5 - 25	64/100/160	EPM7128S
ATF1516ASL/AE	256	от 100	10 - 25	160/192/208	EPM7256S
ATF1532ASL/AE	512	от 100	5 - 25	160/192/208	EPM7512S

Основные технические характеристики:

- общее и индивидуальное управление тристабильным состоянием
- программно управляемая крутизна фронта выходного сигнала
- открытый управляемый режим «открытый коллектор»
- количество циклов перезаписи — не менее 10000
- режим защиты от несанкционированного чтения конфигурации

Для проектирования систем на базе программируемых микросхем фирмы ATMEL рекомендуется использовать бесплатный пакет ATMEL-WinCUP или ATMEL-ProChip Designer.

Более подробную информацию можно получить на сайте: <http://atmel.argusssoft.ru>

Офис в Москве:
Адрес: 128085 Москва, Проспект Мира, 95
Тел.: (095) 217-2487, 217-2518, 217-2505
Факс: (095) 216-66-42
E-mail: info@argusssoft.ru

Офис в Санкт-Петербурге:
Адрес: 191023, Санкт-Петербург, наб. р. Гривордовская, 36
Тел.: (812) 314-3808, тел./факс: (812) 310-6234
E-mail: info@argusssoft.ru

Интернет: WWW.ARGUSSOFT.RU



(Продолжение. Начало — №12/2001)

Синхронизация передачи данных: способы кодирования

Распознавание ячеек ATM в битовом и байтовом потоках данных

Структура ячейки

Технология ATM (Asynchronous Transfer Mode) — асинхронный режим передачи — предусматривает размещение данных внутри 53-байтовых ячеек (рис. 27). Как показано на рисунке, ячейки следуют по линии непрерывным потоком, хотя возможны и иные варианты их транспортирования (об этом позже).

Ячейка состоит из заголовка и поля данных (соответственно 5 и 48 байтов). Заголовок содержит четыре байта для размещения управляющей информации (28 битов — номер логического канала,

ки. Если соответствующий байт (самый «новый» в бегущей строке) представляет собой контрольную сумму (CRC) полученных ранее четырех байтов, то делается предположение о том, что заголовок найден. Если это предположение оправдано, то в дальнейшем моменты обнаружения заголовков будут следовать с шагом, равным длине ячейки. Если совпадение было случайным (его вероятность равна $1/256$), то оно не найдет подтверждения при последующей проверке событий на периодичность.



Рис. 27. Непосредственная передача ячеек ATM по линии связи

плюс четыре флаговых бита) и один контрольный байт CRC (Cyclical Redundancy Check — циклический избыточный код).

Байт CRC вычисляется передатчиком и представляет собой контрольную сумму четырех предыдущих байтов. Приемник после получения ячейки проводит аналогичные вычисления и сопоставляет «свой» результат с имеющимся в ячейке кодом CRC. При отсутствии ошибок коды совпадают.

Но возникает вопрос: как приемник определяет границы ячейки, ведь она не содержит флагов начала или конца?

Использование кода CRC в процессе распознавания границ ячеек

В общем виде методика поиска границ состоит в следующем.

На начальном этапе поиска приемник просматривает поступающий поток ячеек (рис. 27) сквозь «непрозрачную маску» с вырезанной в ней щелью, сквозь которую видны 40 битов данных. Другими словами, перед приемником проходит «бегущая строка» длиной 5 байтов. В каждом такте строка смещается на один бит.

Приемник в каждом такте проверяет содержимое бегущей строки на предмет обнаружения в ней заголовка ячейки.

Как следует из рис. 27, истинный заголовок в наилучшем случае попадет в «поле зрения» приемника через $53 \times 8 - 1 = 423$ такта после начала поиска. Это соответствует ситуации, при которой поиск начался как раз в тот момент, когда предыдущий заголовок только что сместился на один бит и уже не полностью попадает в область бегущей строки.

После выделения цепи правильных совпадений на фоне случайных, приемник точно знает положение границ ячеек. Синхронизация с передатчиком достигнута. Но описанный процесс анализа потока битов продолжится, и каждый новый заголовок проверяется на правильность. Если в заголовке обнаружена корректируемая ошибка, то она исправляется (код CRC позволяет вычислить положение искаженного бита). Если ошибка некорректируемая, то ячейка уничтожается. Последовательность заголовков с некорректируемыми ошибками означает потерю синхронизации, поэтому приемник возвращается к начальному этапу поиска границ ячеек.

Далее будут рассмотрены некоторые подробности аппаратного воплощения данной методики поиска.

Схема «деления» двоичных чисел

Код CRC можно назвать «контрольной суммой», характеризующей некоторый массив данных. Но если говорить более точно, то это контрольный остаток от «деления» некоторого двоичного чис-

ла, построенного на основе массива, на двоичное число, выбранное в качестве делителя. Слово «деление» не напрасно взято в кавычки, так как соответствующая операция, как будет показано, выполняется довольно своеобразно — «лесенкой», но вместо вычитания используется поразрядное логическое суммирование по модулю два. Результат этой операции в общем случае не совпадает с результатом обычного деления.

Применительно к ячейкам ATM, делитель представлен двоичным числом 100000111. Это число соответствует так называемому образующему полиному $X^8 + X^2 + X + 1$ (четыре его члена можно сопоставить с четырьмя единичными битами делителя). Полином, в свою очередь, имеет прямое отношение к структуре обратных связей в схеме «деления», приведенной на рис. 28.

Схема содержит кольцевой сдвиговый регистр из восьми D-триггеров T и три логических элемента Иключающее ИЛИ. На вход синхронизации поступает последовательность тактовых импульсов. В каждом такте положительный фронт импульса подтверждает истинность очередного бита данных D на входе регистра.

Состояние $F7(n+1) \dots F0(n+1)$ регистра в такте $(n+1)$ определяется его состоянием $F7(n) \dots F0(n)$ в предыдущем такте n, а также новым значением входного бита данных $D(n+1)$ и может быть выражено соотношениями (1):

$$F7(n+1) = F6(n) \quad (1)$$

$$F6(n+1) = F5(n)$$

$$F5(n+1) = F4(n)$$

$$F4(n+1) = F3(n)$$

$$F3(n+1) = F2(n)$$

$$F2(n+1) = F1(n) \oplus F7(n)$$

$$F1(n+1) = F0(n) \oplus F7(n)$$

$$F0(n+1) = D(n+1) \oplus F7(n)$$

Чтобы установить взаимосвязь процесса «деления» с последовательностью состояний схемы (рис. 28), рассмотрим примеры формирования кода CRC и его использования для проверки правильности принятых данных.

Формирование заголовка ячейки передатчиком

На начальном этапе формирования кода CRC передатчик дополняет исходные четыре байта заголовка ячейки пятым байтом 000...0, как показано на рис. 29. Это эквивалентно умножению исходного 32-разрядного числа на 28. Полученное 40-разрядное число используется в качестве делимого. Делитель равен 100000111.

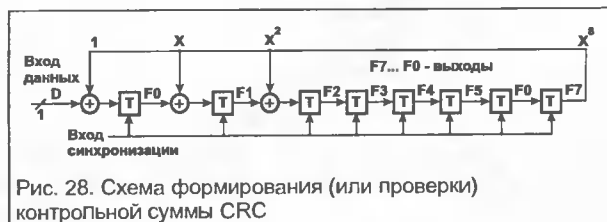


Рис. 28. Схема формирования (или проверки) контрольной суммы CRC

Процесс преобразования 40-разрядного числа, как уже отмечалось, напоминает обычное деление и так же может выполняться «лесенкой». Нули в старших разрядах делимого игнорируются. Копия делителя «пододвигается» под группу разрядов делимого, содержащую в старшем разряде лог. 1, независимо от значений последующих разрядов. Полученная пара кодов поразрядно суммируется по модулю два ($0 + 0 = 0$, $0 + 1 = 1$, $1 + 0 = 1$, $1 + 1 = 0$).

Результат дополняется справа новыми битами делимого, которые для этого «сносятся» вниз таким образом, чтобы общее число битов нового фрагмента, подлежащего обработке, соответствовало разрядности делителя. Появление нескольких нулей в старших разрядах текущего результата, как и при обычном делении, приводит к ускоренному сдвигу очередной ступени «лесенки» вправо.

Целая часть частного не используется, остаток представляет собой байт CRC. Он «прикладывается» передатчиком к четырем исходным байтам (взамен ранее «приложенного» байта 000...0) и пересылается в линию. Первым передается старший разряд байта 1, последним — младший разряд байта CRC. В таком же порядке эта

группа из пяти байтов (заголовок ячейки) поступает в приемник.

Чтобы сопоставить только что описанный процесс получения кода CRC с процессом «прокрутки» кода через кольцевой сдвиговый регистр (рис. 28), рассмотрим ряд состояний этого регистра (рис. 30, левая часть).

В исходном состоянии в регистре записан нулевой код. В первом такте ($n = 1$) по положительному фронту синхроимпульса в левый разряд регистра принимается старший бит 40-разрядного числа (делимого). Но поскольку в нашем примере этот бит равен нулю, состояние регистра не изменяется. Во втором, третьем и четвертом тактах при вводе последующих нулевых битов делимого повторяется то же самое. В пятом такте старший значащий разряд (лог. 1) делимого заносится в регистр, и в последующих тактах наблюдается продвижение группы единиц в направлении выхода F7.

В двенадцатом такте лидирующая лог. 1 достигает выхода F7. Поэтому логические элементы Исключающее ИЛИ во входных каскадах регистра настраиваются на режим инвертирования соответствующих сигналов, распространяющихся слева направо. Инвертирование «проявляется» в тринадцатом такте, в результате в регистре фиксируется код 01001111. В этом коде

В сороковом такте ($n = 40$) в регистре фиксируется код CRC.

Проверка правильности заголовка ячейки приемником

Рассмотрим теперь ситуацию, при которой в приемник поступают пять байтов, из которых пятый — код CRC, построенный на основе обработки первых четырех. Приемник решает следующую задачу: действительно ли пятый байт представляет собой правильный код CRC, т. е. соответствует ли он контрольной сумме предшествующих четырех байтов?

Вернемся к рис. 29 и рассмотрим его нижнюю часть. Делимое содержит в младшем байте код 00000111, который соответствует коду CRC, вычисленному передатчиком. Деление «лесенкой» на десяти ступенях дает те же результаты, что и ранее. Но на одиннадцатой ступени получаем нулевой остаток. Это свидетельствует о том, что контрольная сумма правильная (что и требовалось проверить).

В правой части рис. 30 показан ряд состояний кольцевого сдвигового регистра приемника при проверке правильности кода CRC. Отличия от группы состояний аналогичного регистра передатчика наблюдаются в тактах 38-40. В частности, в контрольном, сороковом такте имеем нулевое содержимое регистра, что подтверждает правильность приема заголовка.

Отметим, что нулевые коды получены также в тактах 1-4 и 31. Но в нашем примере заранее известно, что эти такты промежуточные, поэтому моменты получения промежуточных нулевых кодов не являются для нас значимыми событиями. Иное дело — в реальной жизни, когда такие события не должны оставаться незамеченными.

Поиск заголовка в непрерывном битовом потоке данных

Рассмотренный ранее способ проверки правильности кода CRC с помощью схемы, приведенной на рис. 28, применим лишь в идеальной «статической» ситуации, когда заранее известно, что начальное состояние сдвигового регистра — нулевое, а в первом такте ($n = 1$) в регистр заносится старший бит заголовка.

Реальная ситуация динамична в том смысле, что в поле зрения приемника находится бегущая строка из 40 битов. В каждом такте содержимое строки смещается на один бит. Поэтому в каждом такте нужно успеть проверить правильность пятого байта — предположительно кода CRC. Если применить рассмотренную ранее методику, то нужно в начале каждого такта установить кольцевой сдвиговый регистр в нуль, а оставшийся тактовый интервал разбить на 40 микротактов для выполнения процедуры, показанной в правой части рис. 30. Но такое решение слишком прямолинейно, чтобы быть эффективным.

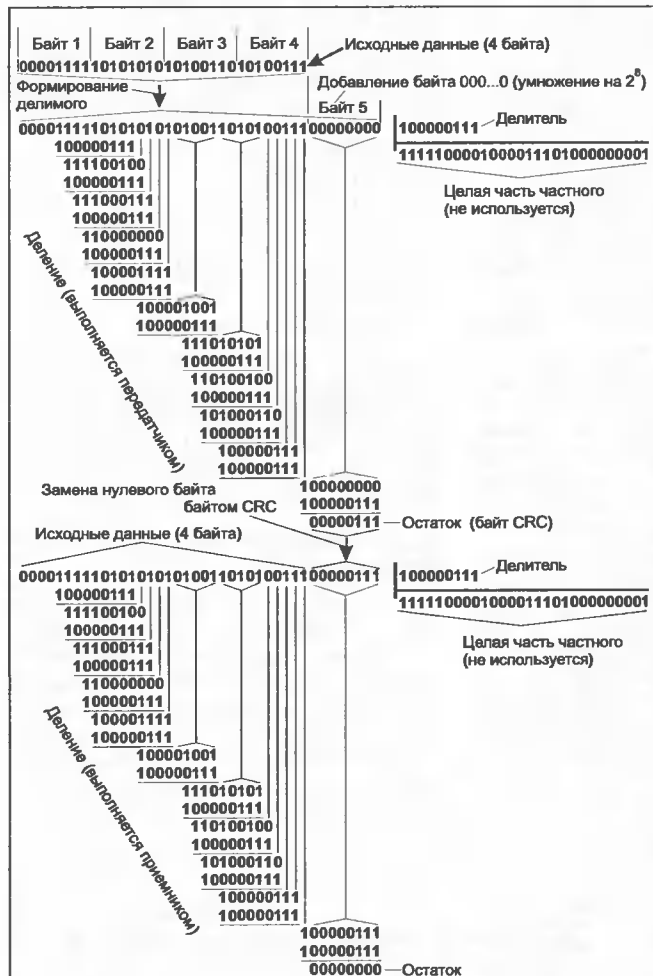


Рис. 29. Формирование кода CRC передатчиком (верхняя часть рисунка) и проверка полученных данных приемником (нижняя часть)

Рассмотрим решение, предложенное в [23], рис. 31. Здесь проверка очередного 40-разрядного кода выполняется за один такт в конвейерном режиме. Новый бит, вводимый в бегущую строку, обычным образом учитывается в новой контрольной сумме. Но одновременно с этим новая контрольная сумма корректируется из-за того, что один бит «вытолкнут» за пределы бегущей строки, и схема должна уничтожить его былой вклад в контрольную сумму.

Сравнивая схемы, показанные на рис. 28 и 31, можно заметить, что последняя представляет собой расширение первой. Кольцевой сдвиговый регистр содержит дополнительные цепи коррекции результата. Введен 40-разрядный сдвиговый регистр для хранения бегущей строки, в которой ожидается появление заголовка ячейки (точнее, кода с правильной контрольной суммой, претендующего на то, чтобы именоваться заголовком). Элемент ИЛИ регистрирует

появление нулевого кода в кольцевом сдвиговом регистре. Все D-триггеры T-синхронизируются общим тактовым сигналом.

Состояние $F7(n+1) \dots F0(n+1)$ кольцевого сдвигового регистра в такте $(n+1)$ определяется состоянием $F7(n) \dots F0(n)$ этого регистра в предыдущем такте n , состоянием разряда $H39(n)$ в такте n , а также новым значением входного бита данных $D(n+1)$, и может быть выражено следующими соотношениями:

$$\begin{aligned} F7(n+1) &= F6(n) \\ F6(n+1) &= F5(n) \oplus H39(n) \\ F5(n+1) &= F4(n) \oplus H39(n) \\ F4(n+1) &= F3(n) \\ F3(n+1) &= F2(n) \\ F2(n+1) &= F1(n) \oplus F7(n) \\ F1(n+1) &= F0(n) \oplus F7(n) \oplus H39(n) \\ F0(n+1) &= D(n+1) \oplus F7(n) \end{aligned} \quad (2)$$

Предположим, что в некоторый момент t бегущая строка (регистр $H39 \dots H0$) содержит код $L(t)$. Примем пока на веру, что

в этот же момент в кольцевом сдвиговом регистре присутствует результат 40-тактной проверки кода $L(t)$ по приведенной ранее методике. Этот результат (код 10110010 на рис. 32) отличается от нулевого, что находится в согласии с условием, что пятый байт кода $L(t)$ не является контрольной суммой (CRC) четырех предыдущих байтов.

В момент $(t+1)$ в бегущей строке фиксируется код $L(t+1)$. Этот код мы уже рассматривали в предыдущем примере. Он содержит правильную контрольную сумму (CRC), равную 00000111. Из рисунка следует, что простой ввод в кольцевой сдвиговый регистр очередного бита, т. е. простой (без поправки) переход от кода $L(t)$ к коду $L(t+1)$ приводит к результату, равному 01100010. Этот результат нельзя считать правильным, так

как ранее было показано, что при правильной контрольной сумме результат должен быть нулевым. Таким образом, если не учитывать поправку, то операцию можно выполнить быстро, за один такт, но, к сожалению, неправильно.

В чем же тут дело? Проблема в том, что результат, равный 01100010 получен при обработке не 40-, а 41-разрядного числа, в котором старший разряд принадлежит коду $L(t)$, а остальные — коду $L(t+1)$. Иными словами, в полученном результате неправильно учитывается вклад в контрольную сумму «исчезнувшего» бита, отброшенного за пределы бегущей строки. Нужно уничтожить этот вклад. Но каков он?

Чтобы ответить на этот вопрос, отметим, что отброшенный разряд отображает число $2^{40} = 1000 \dots 0$ (сорок нулей после единицы). Если «разделить» (например, «лесенкой») это число на наш делитель (100000111), то получим в остатке искомый вклад исчезнувшего бита. Он равен 01100010 (соответствующий образующий полином $X^6 + X^5 +$

Формирование кода CRC				Проверка заголовка ячейки			
n	D	F0 ... F7		n	D	F0 ... F7	
1	0	00000000		1	0	00000000	
2	0	00000000		2	0	00000000	
3	0	00000000		3	0	00000000	
4	0	00000000		4	0	00000000	
5	1	10000000	Байт 1	5	1	10000000	
6	1	11000000		6	1	11000000	
7	1	11100000		7	1	11100000	
8	1	11110000		8	1	11110000	
9	1	11111000		9	1	11111000	
10	0	01111100		10	0	01111100	
11	1	10111110		11	1	10111110	
12	0	01011111	Байт 2	12	0	01011111	
13	1	01001111		13	1	01001111	
14	0	11000111		14	0	11000111	
15	1	00000111		15	1	00000111	
16	0	11100001		16	0	11100001	
17	1	00010000		17	1	00010000	
18	0	00001000		18	0	00001000	
19	1	10000100		19	1	10000100	
20	0	01000010	Байт 3	20	0	01000010	
21	0	00100001		21	0	00100001	
22	1	01110000		22	1	01110000	
23	1	10111000		23	1	10111000	
24	0	01011100		24	0	01011100	
25	1	10101110		25	1	10101110	
26	0	01010111		26	0	01010111	
27	1	01001011		27	1	01001011	
28	0	11000101	Байт 4	28	0	11000101	
29	0	10000010		29	0	10000010	
30	1	11000001		30	1	11000001	
31	1	00000000		31	1	00000000	
32	1	10000000		32	1	10000000	
33	0	01000000		33	0	01000000	
34	0	00100000		34	0	00100000	
35	0	00010000		35	0	00010000	
36	0	00001000		36	0	00001000	
37	0	00000100		37	0	00000100	
38	0	00000010		38	1	10000010	
39	0	00000001		39	1	11000001	
40	0	11100000	CRC	40	1	00000000	

Рис. 30. Процессы формирования кода CRC передатчиком (левая часть рисунка) и проверки правильности заголовка ячейки приемником (права часть)

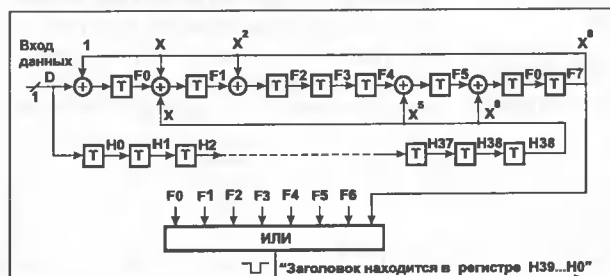


Рис. 31. Схема опознания ячейки в непрерывном потоке битов. Цепи синхронизации триггеров не показаны

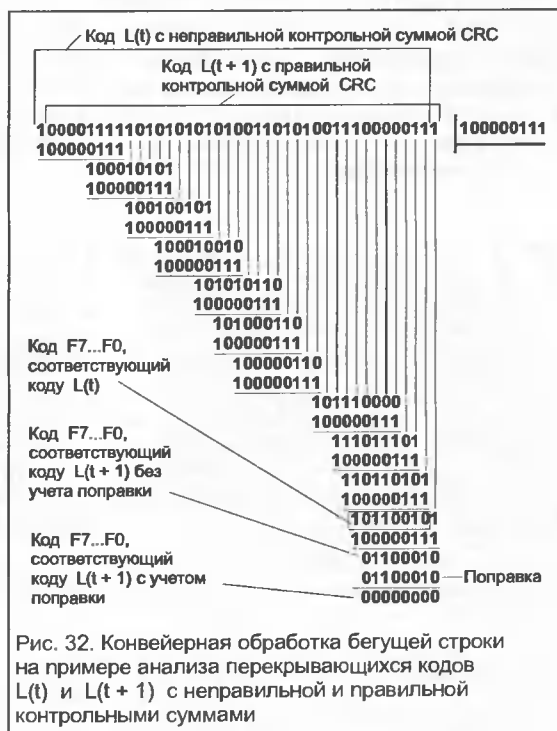


Рис. 32. Конвейерная обработка бегущей строки на примере анализа перекрывающихся кодов $L(t)$ и $L(t+1)$ с неправильной и правильной контрольными суммами

Х). Остается вычесть этот мешающий вклад (поправку) из результата обработки 41-разрядного числа. Но, как мы уже обсуждали, «деление» выполняется довольно своеобразно, и операция вычитания заменяется поразрядным суммированием чисел по модулю два.

Такое суммирование выполняется на заключительном этапе процедуры, показанной на рис. 32. Конечный результат, как и ожидалось, нулевой. Как следует из схемы, приведенной на рис. 31, введение поправки совмещено во времени с вводом очередного бита данных D , поэтому проверка кода в бегущей строке выполняется за один такт. Если бит, исчезнувший из бегущей

строки, — нулевой, то поправка не нужна.

Таким образом, если в такте j проверка содержимого бегущей строки выполнена правильно, то и в следующем такте $j + 1$ гарантируется правильность результата (ошибками передачи пренебрегаем). Из этого можно заключить, что схема (рис. 31) обеспечивает непрерывное слежение за содержимым бегущей строки, причем каждый 40-разрядный код анализируется за один такт.

Конечно, для начального заполнения «конвейера» нужно потратить 40 тактов после предварительной установки всех триггеров T в нулевое состояние.

После установления надежной синхронизации приемника с передатчиком (на уровне установления границ ячеек) приемник использует код CRC для коррекции одиночных ошибок в заголовке. Это возможно, в частности, благодаря тому, что к моменту обнаружения ошибки заголовок хранится в регистре $H39$ —

$H0$, поэтому можно проинвертировать ошибочный бит, т. е. исправить его (цепи коррекции одиночных ошибок в заголовке ячейки на рисунке не показаны).

Борис Шевкопляс,
Игорь Сухман,
Анатолий Бернов,
boriss@zemail.ru

Продолжение следует

Синхронизация псевдослучайных последовательностей

Псевдослучайные последовательности, а именно M -последовательности (псевдослучайные последовательности максимального периода) нашли широкое применение в радиоэлектронной технике: радиолокации, связи и др. Известные способы синхронизации, например, способ синхронизации «по зачетному отрезку» или «многократная передача с мажоритарной обработкой на приеме» имеют ряд недостатков, обусловленных потерей информации при квантовании сигнала. Предлагаемый в настоящей статье способ синхронизации псевдослучайных последовательностей основан на аналогово-цифровой обработке сигнала. При этом уменьшается время вхождения в синхронизм за счет увеличения вероятности правильного приема элемента сигнала.

Пусть псевдослучайная последовательность (ПСП) формируется линейным рекуррентным регистром (ЛРР). Под ЛРР мы будем понимать электронную цепь, позволяющую формировать последовательность, определяемую линейным рекуррентным уравнением. Структурная схема ЛРР состоит из двух основных элементов — сумматоров по модулю два и ячеек памяти (рис. 1).

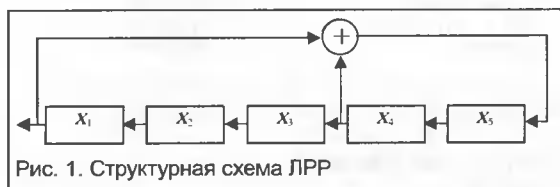


Рис. 1. Структурная схема ЛРР

Известно, что если ЛРР построен на основе неприводимого примитивного многочлена, то он генерирует ПСП максимального периода (M -последовательность) с периодом, равным

$$L = 2^n - 1,$$

где L — период последовательности; n — длина регистра. В нашем случае $L=31$.

Отличительными особенностями описываемого способа синхронизации [2] и одновременно его достоинствами являются:

- отказ от предварительного квантования сигнала, приводящего к потере некоторой части информации;

- аналогово-цифровая обработка сигнала;
- учет рекуррентных свойств ПСП по предсказанию очередного значения элемента;
- вычисление оценочных значений элементов ПСП как внутри тактовых интервалов, так и на их границах.

Указанные свойства позволяют уменьшить время вхождения в синхронизм за счет повышения вероятности правильного приема элемента ПСП.

Сущность предлагаемого способа синхронизации заключается в следующем. Время синхронизации определяется в соответствии с выражением

$$T_{\text{синхр.}} \sim 1/p^n,$$

где $T_{\text{синхр.}}$ — время синхронизации; p — вероятность правильного приема элемента ПСП; n — длина регистра.

Заметим, что при использовании длинных последовательностей даже незначительное увеличение вероятности правильного приема ведет к резкому уменьшению времени синхронизации.

Отказ от предварительного квантования сигнала и аналогово-цифровая обработка требуют использования некоторых преобразований на передаче и приеме. В частно-

сти, в канале связи сигнал должен иметь биимпульсную форму. Для формирования таких сигналов необходимо применять ЛРР, использующий в своей структуре ячейки памяти с аналоговыми значениями, а в цепи обратной связи — перемножители. Структурная схема аналогового ЛРР представлена на рис. 2.

Такое преобразование соответствует операции суммирования по модулю два, если произвести перекодировку по правилу:

$$\begin{aligned} 0 &\rightarrow +1; \\ 1 &\rightarrow -1. \end{aligned}$$

Таблицы истинности для дискретных и аналоговых значений в этом случае имеют вид, представленный ниже, где X_1, X_4 — дискретные значения, Y_1, Y_4 — аналоговые значения.

X_1 / Y_1	X_4 / Y_4	$X_1 \oplus X_4 / Y_1 \times Y_4$
1/-1	1/-1	0/+1
0/+1	1/-1	1/-1
1/-1	0/+1	1/-1
0/+1	0/+1	0/+1

При установлении синхронизации неискаженные элементы ПСП во время передачи могут иметь вид, показанный на рис. 3, а. Под воздействием шумов и помех в канале связи элементы сигнала могут быть искажены таким образом, что пользуясь известными способами невозможно определить истинное значение элемента ПСП. На приемной стороне смесь сигнала, шума и помех имеет вид, приведенный на рис. 3, б. Стробирующие импульсы с частотой, равной тактовой частоте ПСП, показаны на рис. 3, в, где T — длительность элемента сигнала.

С целью избежания внесения дополнительных искажений принимаемый сигнал на входе приемника не квантуют на два уровня, а дискретизируют с частотой, в k раз превышающей тактовую, и получают аналоговые дискретизирован-

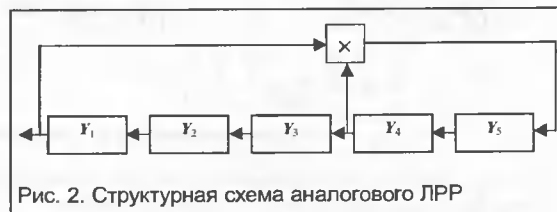


Рис. 2. Структурная схема аналогового ЛРР

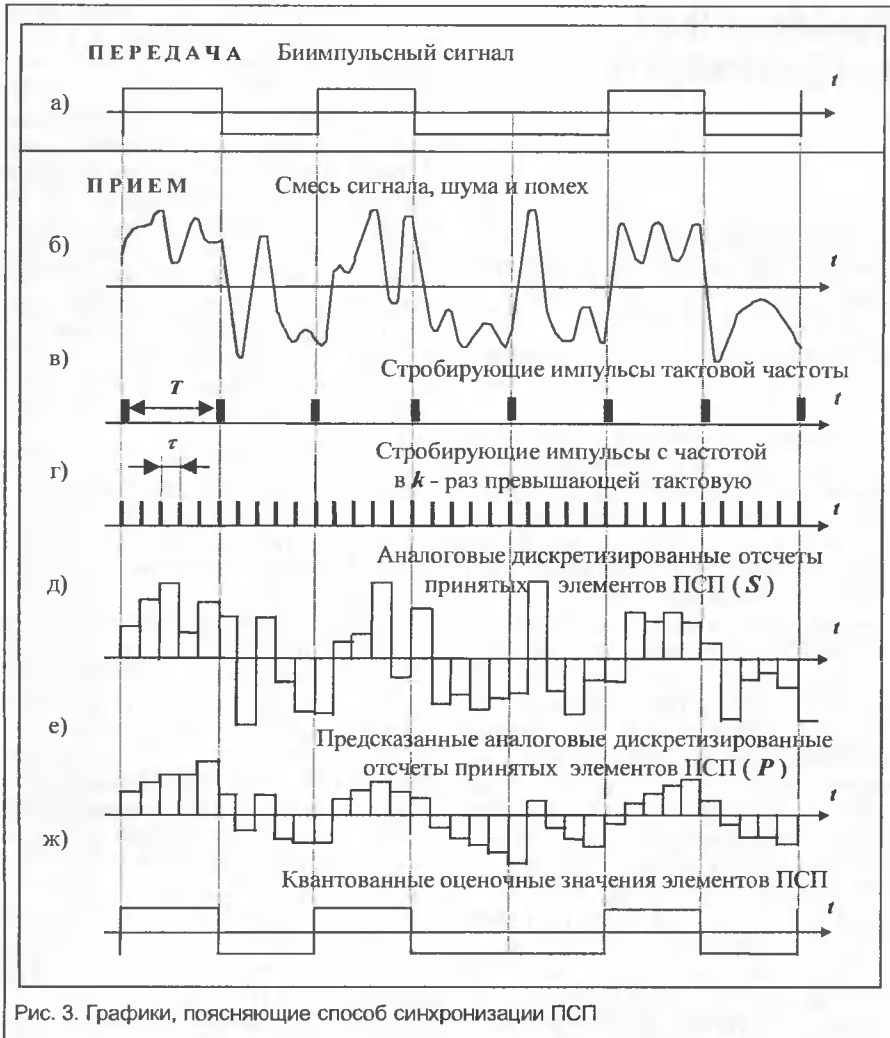


Рис. 3. Графики, поясняющие способ синхронизации ПСП

ные отсчеты (ДО) элемента сигнала с длительностью t . Эксперименты показали, что значение k целесообразно выбирать в пределах от 2 до 10, т. к. при $k=1$ получают вырожденный случай и эффекта улучшения не наблюдается, при $k>10$ значительно увеличивается время оценивания сигнала, а точность практически не увеличивается. Стробящие импульсы с частотой, в k раз превышающей тактовую частоту, показаны на рис. 3, г, аналоговые ДО принятых элементов ПСП — на рис. 3, д.

На рис. 3, е представлены предсказанные значения принятых элементов сигнала, полученные на основе рекуррентного правила преобразования n ранее принятых элементов ПСП.

Вычисление оценочных значений внутри тактовых интервалов и на их

границах производится с целью наиболее точной оценки элемента ПСП.

Оценочное значение можно записать в следующем виде:

$$S_1^* = C_1 P + C_2 (S_1 - C_1 P),$$

где C_1, C_2 — весовые коэффициенты; P — предсказанное аналоговое значение ДО принятого элемента сигнала; S_1 — принятое аналоговое значение ДО элемента ПСП.

С целью определения точности сделанных оценок вычисляют коэффициент корреляции между полученным оценочным значением элемента ПСП и принятым значением ДО. Полученные значения коэффициентов суммируют до тех пор, пока не будет превышен заданный порог. Значение порога суммирования задают в зависимости от уровня шумов и помех.

Итоговые оценочные значения квантуют на два уровня и записывают в ЛРР, сформированный на основе того же характеристического полинома, что и на передающей стороне. Квантованные

оценочные значения элементов ПСП представлены на рис. 3, ж. В случае, когда суммарное значение коэффициента корреляции превысит заданный порог, подается управляющее воздействие на начало генерации в ЛРР ПСП, синхронной с принимаемой последовательностью.

Структурная схема устройства синхронизации представлена на рис. 4. Устройство содержит следующие основные элементы:

- выделитель тактовой частоты (ВТЧ) — для выделения тактовой частоты из принимаемой смеси сигнала, шума и помех;
- дискретизатор (Д) — для дискретизации принимаемой смеси с частотой, в k раз превышающей тактовую частоту;
- линия задержки (ЛЗ) — для задержки принятой смеси на время ее обработки в блоке формирования оценочного сигнала;
- блок формирования оценочного сигнала (БФОС) — для формирования оценочного сигнала в соответствии с приведенным выше правилом;
- блок коммутации (БК) — для обеспечения заданных режимов работы;
- квантователь (Кв) — для квантования аналоговых оценочных значений;
- коррелятор (К) — для вычисления коэффициента корреляции между полученными оценочными значениями и принятыми ДО элементов ПСП;
- устройство управления (УУ) — для формирования управляющего воздействия на начало генерации ПСП;
- линейный рекуррентный регистр (ЛРР) — для генерации ПСП, синхронной с принимаемой.

Результаты статистического моделирования свидетельствуют о наличии существенного уменьшения времени синхронизации в сравнении с традиционными способами. Предлагаемый способ и устройство синхронизации может найти применение для повышения помехоустойчивости и уменьшения времени синхронизации последовательностей с большим периодом, а также при синхронизации в условиях сложной помеховой обстановки. Устройство реализуется на современной элементной базе, например, микроконтроллерах фирмы Atmel.

Александр Журченко,
Дмитрий Онышко
oda78@pisem.net

Литература

1. В. И. Журавлев. Поиск и синхронизация в широкополосных системах. — М.: Радио и связь, 1986.
2. А. Б. Журченко и др. Патент на изобретение № 2127954 «Способ и устройство синхронизации M-последовательности».
3. R. B. Ward. Acquisition of pseudo-noise signals by sequential estimation. IEEE Trans., 1965, Dec., COM-13, 14, pp. 475-483.

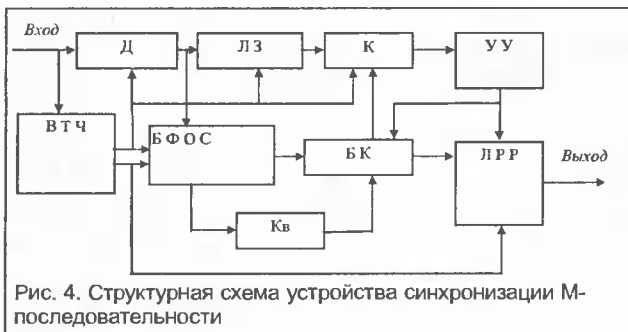


Рис. 4. Структурная схема устройства синхронизации M-последовательности

Схемотехника универсальных технологических контроллеров. Цифро-аналоговые преобразователи

Настоящая статья продолжает цикл публикаций, посвященных схемотехнике универсальных технологических контроллеров. В статье детализированы критерии выбора схем цифро-аналоговых узлов, приведен обзор микросхем, выпускаемых наиболее известными производителями мира и описаны микросхемы, наиболее удовлетворяющие выработанным критериям.

Критерии выбора цифро-аналоговых преобразователей для универсальных технологических контроллеров

Цифро-аналоговым преобразователем (Digital-to-Analog Converters) называется схема, предназначенная для преобразования входного цифрового кода в выходной аналоговый ток или напряжение. Микросхемы цифро-аналоговых преобразователей (далее просто DAC) выпускаются рядом фирм: Analog Devices [1], Burr-Brown [2], Exar [3], Fairchild Semiconductor [4], Hitachi [5], Intersil [6], Linear Technology [7], Maxim [8], National Semiconductor [9], Sipex [10], Signal Processing Technologies (Toko) [11], Texas Instruments [12], и др. Данные по DAC, выпускаемым этими фирмами, можно найти на [13].

Наиболее важными параметрами DAC являются разрядность, тип интерфейса, быстродействие, тип корпуса, характеристики питания, тип выхода (по току или по напряжению).

В универсальных технологических контроллерах аналого-цифровые преобразователи (ADC), как правило, используются с последовательным интерфейсом, поскольку они зачастую выполняют задачи контроля или измерения медленно изменяющихся величин (см. предыдущие статьи цикла). С цифро-аналоговыми преобразователями ситуация несколько сложнее, т. к. они обычно управляют какими-либо исполнительными устройствами и, в конечном итоге, процессами. Это обуславливает необходимость применения как медленнодействующих, так и относительно быстродействующих DAC.

Разрядность DAC определяет точность преобразования. В настоящее время

выпускаются микросхемы с разрядностью от 8 до 22 и выше. Обычно в универсальных технологических контроллерах используют DAC с разрядностью 8 и 10, что обеспечивает шаг квантования выходной управляемой величины 1/256 и 1/1024 соответственно.

Выпускаемые в настоящее время микросхемы DAC могут иметь параллельный или последовательный интерфейс. Параллельный интерфейс чаще всего используется для более быстродействующих DAC, а последовательный — для медленнодействующих. При этом, в качестве последовательного интерфейса предпочтительно использовать интерфейс SPI (Serial Peripheral Interface), т. к. большинство современных микроконтроллеров оснащаются аппаратной реализацией именно этого интерфейса.

Быстродействие цифро-аналогового преобразователя также важно. Современные DAC по быстродействию можно условно разделить на три группы: низкого быстродействия — до 100 ksp/s, среднего быстродействия — от 100 до 500 ksp/s, высокого быстродействия — более 500 ksp/s. Для универсальных технологических контроллеров наиболее часто используются DAC низкого или среднего быстродействия исходя из соотношения стоимость/быстродействие.

Современные микросхемы DAC производятся с токовым выходом (без встроенного масштабирующего операционного усилителя) и с выходом по напряжению (со встроенным выходным операционным усилителем). Кроме того, в случае, если микросхема в одном корпусе содержит несколько DAC, она может иметь комбинированный выход (часть с токовым выходом, часть с выходом по напряжению) или иметь выхо-

ды по напряжению с различной нагрузочной способностью. Конечно, с точки зрения минимизации количества корпусов цифро-аналогового узла, более предпочтителен выход по напряжению. Однако, ввиду того, что обычно разработчики универсальных контроллеров стремятся уместить выход, т. е. ставят выходной операционный усилитель, который кроме всего прочего еще и формирует выходное напряжение необходимого уровня, тип выхода микросхем DAC не имеет существенного значения.

Следует также отметить, что современные микросхемы DAC могут использовать встроенный или внешний узел опорного напряжения.

Микросхемы DAC могут иметь одно или несколько различных напряжений питания. Очевидно, что для технологических микросистем желательно, чтобы микросхема имела одно питание, равное 5 В.

Обычно в универсальных технологических контроллерах используется достаточно большое количество DAC двух типов. Первая группа включает медленнодействующие преобразователи с разрядностью 10 и последовательным интерфейсом. Обычно может использоваться 4, 6 или 8 независимых преобразователей. Вторая группа — относительно быстродействующие преобразователи с разрядностью 8. В этой группе могут быть микросхемы как с последовательным, так и с параллельным интерфейсом (обычно от 2 до 6 преобразователей).

Большое значение при выборе микросхемы имеет тип корпуса. Как уже отмечалось в первой статье цикла, оптимальным для технологических контроллеров является корпус типа DIP, который позволяет устанавливать микросхему на панельку, что значительно упрощает ремонт и наладку контроллера. Кроме того, желательно, чтобы корпус имел малое количество выводов. Обычно используются микросхемы с количеством выводов от 8 до 16.

Предварительный отбор микросхем DAC

В табл. 1 приведены данные на наиболее распространенным DAC с параллельным интерфейсом (за исключением телевизионных и сверхбыстродействующих), отсортированные по нарастанию разрядности, а затем по нарастанию количества каналов в корпусе.

Тип	Разрядность	Количество DAC	Напряжение питания, В	Выход	Быстродействие, мкс	Интерфейс параллельный	Опорное напряжение	Корпус
HI5660	8		5 ÷ 3	V	125/60 MSPS		Int.	
MX7523	8	1	5 ÷ 16	I	0,15 мкс		Ext.	DIP16, WSO16
MAX5018	8	1	-5,2	I	0,0045 мкс		Ext.	NDIP24
DAC0800	8	1	±5 ÷ ±15	V	100 нс			SOIC, MDIP
DAC0808	8	1	±5 ÷ ±15	V	150 нс			SOIC, MDIP
DAC0830	8	1	5 ÷ 15	V	1000 нс			MDIP
DAC0832	8	1	5 ÷ 15	V	1000 нс			SOIC, MDIP, PLCC

Таблица 1

Продолжение таблицы 1

Тип	Разрядность	Количество DAC	Напряжение питания, В	Выход	Быстродействие, мкс	Интерфейс параллельный	Опорное напряжение	Корпус
HI3338, CA3338/A	8	1	5	V	50 MSPS		Ext.	
HI1171	8	1	5	V	40 MSPS		Ext.	
HA17008RP	8	1		V	150 нс			DIP16
HA17008RFP	8	1		V	150 нс			SOP16
HA17408P	8	1		V	250 нс			DIP16
HD49307	8	1		V	30 нс			SOP56
TLC5602	8	1	5	V	30 нс			
THS5641A	8	1	3 ÷ 5		35 нс			
THS5641	8	1	3 ÷ 5		35 нс			
MP7523	8	1	4,5 ÷ 16	I				PDIP, CDIP, SOIC
MP7524	8	1	4,5 ÷ 16	I				PDIP, CDIP, SOIC, PLCC
MP7524A	8	1	4,5 ÷ 16	I				PDIP, SOIC
AD7524	8	1	5; 15				Ext.	
DAC08	8	1	+15; -15				Ext.	
AD9708	8	1	2,7 ÷ 5,5; 3; 3,3; 5		35 мкс			
AD9701	8	1	-5,2		8 мкс			
AD5330	8	1					Ext.	
AD7801	8	1					Int./Ext.	
TLC7524	8	1	5 ÷ 15	I	0,1 мкс		Ext.	
DAC0802	8	1	±5 ÷ ±15	V	100 мкс			SOIC, MDIP
MAX5140	8	1	-5,2	I	0,003	uP/8	Ext.	NDIP24
MAX5480	8	1	5	I	0,4	uP/8	Ext.	QSOP16
MAX7624	8	1	12 ÷ 15	I	0,25 (max)	uP/8	Ext.	DIP16, SO16
MX7224	8	1	15; 11,4 ÷ 16,5 & -5	V	5 (max)	uP/8	Ext.	PLCC20, DIP18, WSO18
MX7524	8	1	5 ÷ 15	I	0,4	uP/8	Ext.	DIP16, SO16
HI5628	8	2	5 ÷ 3	V	125,60 MSPS		Int.	
HI1177	8	2	5	V	40 MSPS		Ext.	
MP7528	8	2	4,5 ÷ 16	I				PDIP, CDIP, SOIC, PLCC
MP7529A	8	2	4,5 ÷ 16	I				PDIP, SOIC, PLCC
MP7529B	8	2	4,5 ÷ 5,5	I				PDIP, SOIC, PLCC
AD7528	8	2	5; 15				Ext.	
AD7628	8	2	12; 15				Ext.	
AD5332	8	2					Ext.	
AD7302	8	2					Int./Ext.	
TLC7628	8	2	11 ÷ 15	I	0,1 нс		Ext.	
TLC7528	8	2	5 ÷ 15	I	0,1 нс		Ext.	
MX7528	8	2	5 ÷ 15	I	0,18 (max)	uP/8	Ext.	DIP20, WSO20
MX7628	8	2	5 ÷ 15	I	0,35 (max)	uP/8	Ext.	DIP20, WSO20
HI1178	8	3	5	V	40 MSPS		Ext.	
FMS3810	8	3	5	V	100 МГц			
FMS3815	8	3	5	V	150 МГц			
TMC3503x30	8	3	5	V	30 МГц			
TMC3503x50	8	3	5	V	50 МГц			
TMC3503x80	8	3	5	V	80 МГц			
TMC3533x30	8	3	3,3	V	30 МГц			
TMC3533x50	8	3	3,3	V	50 МГц			
TMC3533x80	8	3	3,3	V	80 МГц			
TL5632	8	3	5		10 нс			
THS8134B	8	3	3 ÷ 5		5 нс			
THS8134A	8	3	3 ÷ 5		5 нс			
THS8134	8	3	3 ÷ 5		5 нс			
MP7643	8	4	5, 5, 10	V				PDIP, SOIC
MP7226	8	4	5/15	V				PDIP, SOIC, PLCC
MP7628	8	4	4,5 ÷ 5,5	I				PDIP, CDIP, SOIC, PLCC
DAC8408	8	4	5				Ext.	
AD5334	8	4					Ext.	
AD7305	8	4					Ext.	
TLC7226	8	4	15	V	5 нс		Ext.	
TLC7225	8	4	5 ÷ 15	V	5 нс		Ext.	
MAX505	8	4	5; ±5	V	6	uP/8	Ext.	SSOP24, NDIP24, WSO24
MAX506	8	4	5; ±5	V	6	uP/8	Ext.	DIP20, WSO20
MX7225	8	4	15; 12 ÷ 15 & -5	V	4 (max)	uP/8	Ext.	PLCC28, NDIP24, WSO24
MX7226	8	4	15; 12 ÷ 15 & -5	V	4 (max)	uP/8	Ext.	PLCC20, DIP20, SB20, WSO20
MP7228	8	8	5/15	V				PDIP, CDIP, PLCC, SOIC
MX7228	8	8	15; 12 ÷ 15 & -5; +5	V	5 (max)	uP/8	Ext.	PLCC28, NDIP24, WSO24

Продолжение таблицы 1

Тип	Разрядность	Количество DAC	Напряжение питания, В	Выход	Быстродействие, мкс	Интерфейс параллельный	Опорное напряжение	Корпус
UCC5950	10	1	5	V	2,5 нс			
HI20201	10	1	-5,2	V	160 MSPS		Ext.	
HI5760	10	1	5 ÷ 3	V	125/60 MSPS		Int.	
HI5721	10	1	+5/- 5,2	V	125 MSPS		Int.	
HI3197	10	1	5; +5/-5	V	125 MSPS		Int.	
HI5780	10	1	5	V	80 MSPS		Int.	
HI2315	10	1	5	V	80 MSPS		Int.	
THS5651A	10	1	3 ÷ 5		35 нс			
MP7533	10	1	4,5 ÷ 16	I				PDIP, CDIP, SOIC, PLCC
MP7633	10	1	4,5 ч 16	I				PDIP, CDIP, SOIC, PLCC
AD9760-50	10	1	2,7 ÷ 5,5; 3; 3,3; 5		35			
ADV7128	10	1	5					
AD9750	10	1	5		35			
AD9760	10	1	2,7 ÷ 5,5; 3; 3,3; 5		35			
AD9731	10	1	+5; -5,2		3,8			
AD9732	10	1	5		4,75			
AD9751	10	1	3		35			
AD5331	10	1					Ext.	
AD7393	10	1					Ext.	
ADV7127	10	1	3,3; 5					
AD7533	10	1	15			Unbuffered	Ext.	
MX7520	10	1	5 ÷ 15	I	0,5	uP/10	Ext.	SB16, DIP16, WSO16
MX7530	10	1	5 ÷ 15	I	0,5	uP/10	Ext.	SB16, DIP16, WSO16
MX7533	10	1	5 ÷ 15	I	0,6	uP/10	Ext.	SB16, DIP16, WSO16
MAX503	10	1	+5; ±5	V	25	uP/4, uP/8	Ext./ Int.	SSOP24, NDIP24, WSO24
HI5728	10	2	5 ÷ 3	V	125/60 MSPS		Int.	
AD9761	10	2	2,7 ÷ 5,5; 3; 3,3; 5		35			
AD9763	10	2	3 ÷ 5,5; 3; +3,3; 5		35			
AD5333	10	2					Ext.	
AD7397	10	2					Ext.	
HI3050	10	3	5	V	50 MSPS		Ext.	
FMS3110	10	3	5	V	100 МГц			
FMS3115	10	3	5	V	150 МГц			
TMC3003x30	10	3	5	V	30 МГц			
TMC3003x50	10	3	5	V	50 МГц			
TMC3003x80	10	3	5	V	80 МГц			
TMC3033x30	10	3	3,3	V	30 МГц			
TMC3033x50	10	3	3,3	V	50 МГц			
TMC3033x80	10	3	3,3	V	80 МГц			
THS8133A	10	3	3 ÷ 5		5 нс			
THS8133	10	3	3 ÷ 5		5 нс			
AD5335	10	4					Ext.	
AD5336	10	4					Ext.	
HI5731	12		+5/-5,2	V	100 MSPS		Int.	
HS7541A	12	1	15 В @ 2 мА	I	0,6 нс			
SP7545	12	1	15 В @ 2 мА	I	2 нс			
HI5860	12	1	5 ÷ 3	V	125 MSPS		Int.	
HI5735	12	1	+5/-5,2	V	80 MSPS		Int.	
HA17012P	12	1		V	250 нс			DIP20
THS5661A	12	1	3 ÷ 5		35 нс			
LTC1450L	12	1		V				
LTC7541A	12	1		I				
LTC7545	12	1		I				
MP1230A/31A/32A	12	1	4,5 ÷ 16 В	I				PDIP, CDIP, SOIC
MP7541B	12	1	4,5 ÷ 16 В	I				PDIP, CDIP, SOIC
MP7542	12	1	4,5 ÷ 5, 5 В	I				PDIP, CDIP, SOIC
MP7545B	12	1	4,5 ÷ 16 В	I				PDIP, SOIC, PLCC
MP7645B	12	1	4,5 ÷ 16 В	I				PDIP, CDIP
DAC7613	12	1		V	10			SSOP24
DAC7621	12	1		V	10		Int.	SSOP20
DAC7541	12	1		V	1			DIP18, SO18
DAC7545	12	1		V	2			DIP18, SO18
DAC80	12	1		V	4		Int.	DIP24, Hermetic24
DAC811	12	1		V	4		Int.	DIP28, SO28
DAC813	12	1		V	4		Int.	DIP28, SO28

Продолжение таблицы 1

Тип	Разрядность	Количество DAC	Напряжение питания, В	Выход	Быстродействие, мкс	Интерфейс параллельный	Опорное напряжение	Корпус
AD7545A	12	1	5; 12; 15				Ext.	
AD9752	12	1	5		35			
AD9762	12	1	2,7 ч 5,5; 3; 3,3; 5		35			
AD9753	12	1	3		35			
AD5340	12	1					Ext.	
AD5341	12	1					Ext.	
AD7392	12	1					Ext.	
DAC8562	12	1					Int.	
MAX5013	12	1	+5; -5,2	I	0,013		Int.	DIP28, PLCC28
AD7542	12	1	5			Nibble Load (3 x 4Bits)	Ext.	
AD7541A	12	1	5; 12; 15			Unbuffered	Ext.	
MAX502	12	1	± 12 ч ± 15	V	5	uP/12	Ext.	NDIP24, WSO24
MAX507	12	1	± 12 ч ± 15	V	5	uP/12	Int.	NDIP24, WSO24
MAX7645	12	1	15	I	1 (max)	uP/12	Ext.	PLCC20, DIP20, WSO20
MX565A	12	1	± 15	I	0,25 (max)	uP/12	Int.	NDIP24, WSO24
MX566A	12	1	-15	I	0,35 (max)	uP/12	Ext.	NDIP24, WSO24
MX7245	12	1	12 ч 15; ± 15	V	5 (max)	uP/12	Int.	NDIP24, PLCC28
MX7521	12	1	5 ч 15	I	0,5	uP/12	Ext.	SB18, DIP18, WSO18
MX7531	12	1	5 ч 15	I	0,5	uP/12	Ext.	SB18, DIP18, WSO18
MX7541	12	1	5 ч 16	I	1 (max)	uP/12	Ext.	DIP18, WSO18
MX7541A	12	1	5 ч 16	I	0,6	uP/12	Ext.	18/PLCC, DIP18, SB18, WSO18
MX7545	12	1	15; 5	I	2 (max)	uP/12	Ext.	DIP20, WSO20
MX7545A	12	1	5 ч 15	I	1 (max)	uP/12	Ext.	PLCC20, DIP20, WSO20
MX7845	12	1	± 15	V	5 (max)	uP/12	Ext.	PLCC28, NDIP24, WSO24
MX7542	12	1	5	I	2 (max)	uP/4	Ext.	DIP16, WSO16
MAX530	12	1	5; ± 5	V	25	uP/4, uP/8	Int.	SSOP24, NDIP24, WSO24
MAX501	12	1	± 12 ч ± 15	V	5	uP/8	Ext.	NDIP24, WSO24
MAX508	12	1	± 12 ч ± 15	V	5	uP/8	Int.	DIP20, WSO20
MX7248	12	1	12 ч 15; ± 15	V	5 (max)	uP/8	Int.	DIP20, PLCC20
MX7548	12	1	12 ч 15; 5	I	1 (max)	uP/8	Ext.	PLCC20, DIP20, WSO20
MAX5012	12	1	-5,2	I	0,013	ECL	Int.	DIP28, PLCC28
MAX555	12	1	-5,2	V	0,004	ECL	Ext.	Thermally Enhanced PLCC68
AD7548	12	1	5; 12; 15			Byte	Ext.	
TLV5639	12	1	2,7 ч 5,5	V	1 нс	P12	Int.	
TLV5619	12	1	2,7 ч 5,5	V	1 нс	P12	Ext.	
TLV5633	12	1	2,7 ч 5,5	V	1 нс	P8	Int.	
TLV5613	12	1	2,7 ч 5,5	V	1 нс	P8	Ext.	
SP9502	12	2	+5 В @ 4 мА; -5 В @ 4 мА	V BiPolar	4 uS			
SP9602	12	2	+5 В @ 1,6 мА; -5 В @ 1,6 мА	V BiPolar	30 uS			
HI5828	12	2	5 ч 3	V	125 MSPS		Int.	
HSP50415	12	2	3,3	V	100 MSPS		Int.	
DAC7801	12	2		V	0,8			DIP24
DAC7802	12	2		V	0,8			DIP24
AD7547	12	2	12; 15				Ext.	
DAC8222	12	2	5; 15				Ext.	
AD9765	12	2	3 ч 5,5; 3; +3,3; 5		35			
AD5342	12	2					Ext.	
AD5343	12	2					Ext.	
AD7396	12	2					Ext.	
AD8582	12	2					Int.	
AD7549	12	2	15			Nibble (3 x 4Bits)	Ext.	
MX7547	12	2	12 ч 15	I	1,5 (max)	uP/12	Ext.	PLCC28, NDIP24, WSO24
MX7847	12	2	± 15	V	4	uP/12	Ext.	NDIP24, WSO24
MX7549	12	2	5 ч 15	I	1,5 (max)	uP/4, uP/8	Ext.	PLCC20, DIP20, WSO20
MX7537	12	2	12 ч 15	I	1,5 (max)	uP/8	Ext.	PLCC28, NDIP24, WSO24
MX7837	12	2	± 12 ч ± 15	V	4	uP/8	Ext.	NDIP24, WSO24
AD7537	12	2	12; 15			Byte	Ext.	
DAC8248	12	2	5; 15			Byte	Ext.	
SP9504	12	4	+5 В @ 8 мА; -5 В @ 8 мА	$\pm V$	4 uS			
SP9604	12	4	5 В @ 3 мА; -5 В @ 3 мА	$\pm V$	30 uS			

Продолжение таблицы

Тип	Разряд- ность	Коли- чество DAC	Напряжение питания, В	Выход	Быстродействие, мкс	Интерфейс параллельный	Опорное напряжение	Корпус
MP7680	12	4	4,5 ÷ 5,5	I				PDIP, CDIP, PQFP
DAC7624	12	4		V	10			DIP28, SO28
DAC7625	12	4		V	10			DIP28, SO28
DAC7724	12	4		V	10			SOIC24, DIP24
DAC7724	12	4		V	10			SO28, PLCC28
DAC7725	12	4		V	10			SOIC28, PDIP28
DAC7725	12	4		V	10			SO28, PLCC28
AD5344	12	4					Ext.	
MAX526	12	4	+12 ÷ +15 & -5	V	3	uP/8	Ext.	SB24, NDIP24, WSO24
MAX527	12	4	±5	V	3	uP/8	Ext.	SB24, NDIP24, WSO24
MP7613	12	8	+12/-12	V				PQFP, PGA, PLCC
MAX547	13	8	±5	V		uP/13	Ext.	FP44, PLCC44
HS3140	14	1	+15 В @ 2 мА	I	2 uS			
SP7514	14	1	+15В@2 мА	I	2 uS			
DAC14135	14	1	+5	V				TSSOP
HI5960	14	1	+5 ÷ +3	V	125 MSPS		Int.	
ISL5314	14	1	+5 ÷ +3	V	125 MSPS		Int.	
HI5741	14	1	+5/-5,2	V	100 MSPS		Int.	
THS5671A	14	1	3 ÷ 5		35 нс			
LTC1591	14	1		I				PDIP, CDIP, SOIC
MP7614	14	1	4,5 ÷ 16	I				
AD7538	14	1	+12; +15	I			Ext.	
AD9754	14	1	+5	I			Int./Ext.	
AD9755	14	1	+3	I			Int./Ext.	
AD9764	14	1	+2,7 ÷ +5,5; +3; +3,3; +5	I			Int./Ext.	
AD9772	14	1	+2,7 ÷ +3,6; +3; +3,3	I			Int./Ext.	
AD9774	14	1	+2,7 ÷ +5,5; +3; +3,3; +5	I			Int./Ext.	
AD7538	14	1	+12; +15				Ext.	
AD9754	14	1	+5		35			
AD9764	14	1	+2,7 ÷ +5,5; +3; +3,3; +5		35			
AD9774	14	1	+2,7 ÷ +5,5; +3; +3,3; +5		35			
AD9772	14	1	+2,7 ÷ +3,6; +3; +3,3		11			
AD9755	14	1	+3					
MX7535	14	1	+12 ÷ +15	I	1,5 (max)	uP/14	Ext.	PLCC28, DIP28, SB28, WSO28
MX7538	14	1	+12 ÷ +15	I	1,5 (max)	uP/14	Ext.	PLCC28, NDIP24, SB24, WSO24
MX7536	14	1	+12 ÷ +15	I	1,5 (max)	uP/14, uP/8	Ext.	PLCC28, DIP28, WSO28
MX7534	14	1	+12 ÷ +15	I	1,5 (max)	uP/8	Ext.	PLCC20, DIP20, SB20, WSO20
AD7534	14	1	+12; +15	I		Byte	Ext.	
AD7534	14	1	+12; +15			Byte	Ext.	
AD7536	14	1	+12; +15; +12 ÷ +15	I		Byte	Ext.	
AD7536	14	1	+12; +15; +12 ÷ +15			Byte	Ext.	
AD7840	14	1	+15; -15	V		+ Serial	Int./Ext.	
AD9767	14	2	+3 ÷ +5,5; +3; +3,3; +5	I			Int./Ext.	
AD9767	14	2	+3 ÷ +5,5; +3; +3,3; +5		35			
AD7836	14	4	+15; -15; +5	V			Ext.	
AD7835	14	4	+15; -15; +5	V		Byte	Ext.	
MP7611	14	8	+12/-12	V				PQFP, PGA, PLCC
AD7841	14	8	+12; -12; +5; +15; -15; +5	V			Ext.	
SP9316	16	1	+15 В @ 0,5 мА	I	2 uS			
LTC1597	16	1		I				
LTC1599	16	1		I				
LTC1657	16	1		V				
MP7616	16	1	4,5 ÷ 16	I				PDIP, CDIP, SOIC
MP7626	16	1	4,5 ÷ 16,5	I				PDIP, CDIP, PLCC

Окончание таблицы

Тип	Разрядность	Количество DAC	Напряжение питания, В	Выход	Быстродействие, мкс	Интерфейс параллельный	Опорное напряжение	Корпус
DAC7641	16	1		V	10			TQFP-32
DAC7741	16	1		V	5		Int.	TQFP48
DAC712	16	1		V	10		Int.	DIP28, SO28
DAC715	16	1		V	10		Int.	DIP28, SO28
DAC702	16	1		V	8		Int.	Hermetic24
DAC703	16	1		V	8		Int.	DIP24, SO24
DAC707	16	1		V	8		Int.	DIP28
DAC708	16	1		V	8		Int.	Hermetic24
DAC709	16	1		V	8		Int.	Hermetic24
AD768	16	1	+5; -5	I			Int./Ext.	
AD669	16	1	+15; -15; +5	V			Int./Ext.	
AD7846	16	1	+12; -12; +5; +15; -15; +5	V			Ext.	
AD768	16	1	+5; -5		25			
DAC16	16	1	+5; -15			Unbuffered	Ext.	
DAC16	16	1	+5; -15	I		Unbuffered	Ext.	
AD569	16	1	+12; -12	V		Byte	Ext.	
AD660	16	1	+15; -15; +5	V		Byte, +Serial	Int./Ext.	
DAC725	16	2		V	8		Int.	DIP28
DAC7644	16	4		V	10			SSOP48
DAC7744	16	4		V	10			SSOP48
AD760	18	1	+15; -15; +5	V		Byte, +Serial	Int./Ext.	

Литература:

1. <http://www.analog.com>
2. <http://www.burr-brown.com>
3. <http://www.exar.com>
4. <http://www.fairchildsemi.com>
5. <http://www.hitachi.com>
6. <http://www.intersil.com>
7. <http://www.linear-tech.com>
8. <http://www.maxim-ic.com>
9. <http://www.national.com>
10. <http://www.sipex.com>
11. <http://www.tokoam.com>
12. <http://www.ti.com>
13. <http://www.itis.spb.ru>

Олег Николаичук
onic@ch.moldpac.md

(Продолжение следует)




MAXIM DALLAS SEMICONDUCTOR

Поставки от официального дистрибьютора



125315, г. Москва, ул. Усиевича, 24/2, ООО «АТОС»
Телефоны и Факсы: (095) 787-48-05, 155-40-35
Интернет: www.atos.ru, info@atos.ru



www.platan.ru ЭЛЕКТРОННЫЕ КОМПОНЕНТЫ ВАШЕГО УСПЕХА

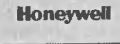
www.platan.ru

- ВИРТУАЛЬНЫЙ МАГАЗИН РЕАЛЬНЫХ ЭЛЕКТРОННЫХ КОМПОНЕНТОВ**
 Наш сайт предоставляет широкие возможности оформления заявок на поставку электронных компонентов:
 - ✓ Вы можете разместить заявку на требуемые компоненты и отслеживать ее прохождение на всех этапах. Очень удобный интерфейс и абсолютная безопасность!
 - ✓ Воспользуйтесь нашей удобной системой приема заявок на компоненты, которые находятся на складе или на которые можно разместить заказ.
 - ✓ С нашего сайта Вы можете скачать нашу базу в формате EXCEL, проставить нужное Вам количество компонентов и отправить этот файл по нашему адресу.

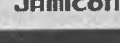
ВСЕ ЗАЯВКИ, ПОЛУЧЕННЫЕ В ЭЛЕКТРОННОМ ВИДЕ, ОБРАБАТЫВАЮТСЯ В ПЕРВУЮ ОЧЕРЕДЬ.
- НОВОСТИ, ТЕХНИЧЕСКАЯ ИНФОРМАЦИЯ, СТАТЬИ, ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ**
 Новые поступления на склад ПЛАТАН, новости от мировых производителей. Подробная техническая документация на поставляемые компоненты, статьи и книги по электронной тематике. Бесплатное программное обеспечение.













121351, Москва, ул. Ивана Франко, д. 40, к. 1, стр. 2 Почта: 121351, Москва, а/я 100
тел./факс: (095) 73-75-999 E-mail: ir@platan.ru

Анализ целостности сигналов на печатных платах

В последнее время все актуальнее становится проблема автоматизации расчета целостности сигналов при проектировании печатных плат. Сейчас приняты два метода проектирования быстродействующих устройств.

Первый метод заключается в использовании физических ограничений (Physical Constraint). В этом случае используются геометрические соображения, такие как длина и ширина проводника, которые заносятся в правила проектирования САПР (design rules). Однако данный подход обычно не позволяет учесть эффекты из-за тактовых сигналов и фронтов, особенностей диэлектрических свойств материалов платы и других негеометрических факторов. В результате это приводит к тому, что конструкторы проектируют излишне консервативную топологию платы, но в то же время не застрахованы от проблем, связанных с недостаточным учетом вышеперечисленных факторов. Альтернативный подход — синтез соединений (Interconnect Synthesis, IS) — появился несколько лет назад. При этом подходе интегрально учитываются как геометрические факторы, так и электрические свойства материалов платы и компонентов. Электрическая верификация проекта в этом случае выполняется постоянно, начиная от стадии размещения компонентов и завершая последними этапами разводки (сглаживание и т. п.). В этом случае, с одной стороны, удается повысить плотность трассировки, с другой — уменьшить взаимные помехи.

Явление взаимных наводок (Crosstalk) заключается в том, что сигнал, передаваемый по линии связи, передается на соседнюю дорожку из-за наличия взаимных емкостей и индуктивностей. Очевидно, что и в аналоговых, и в цифровых цепях наводки могут вызвать ошибочные переключения, сбросы и т. п. Проблема особенно обостряется для цепей тактовой синхронизации при сокращении длительности фронтов. Плотные платы, в которых расстояния между дорожками малы, еще более усугубляют проблему наводок.

Необходимость предварительного анализа наводок с использованием со-

ответствующих средств САПР связана с тем, что в результате резкого уменьшения сроков разработки изделия попросту не остается времени на окончательную отработку.

Следует помнить, что параллельность проводников еще не означает обязательные наводки.

На рис. 1 представлены результаты моделирования процессов в параллельных проводниках с использованием программного обеспечения компании Mentor Graphics.

На осциллограмме слева время нарастания фронта импульса несколько нс, на правой — 0,5 нс, топология разводки одинакова, частота следования импульсов — 20 МГц. Как видим, из-за уменьшения длительности фронта наводки возросли с 0,5 до 3 В. Таким образом, очевидной становится необходимость проведения предварительного моделирования платы с рабочими параметрами критических цепей.

В качестве примера соответствующего программного обеспечения рассмотрим утилиту анализа целостности сигналов **ACCEL Signal Integrity**.

Окно утилиты **ACCEL Signal Integrity** представлено на рис. 2. Для загрузки информации из PCB файла существуют два пути.

Во-первых, можно одновременно с **ACCEL Signal Integrity** открыть графический редактор **ACCEL PCB** и использовать команду **Get Nets**.

Во-вторых, можно воспользоваться пиктограмму **Open SULTAN File**. Файл **SULTAN** представляет собой файл, содержащий необходимое описание топологии платы и ее электрических характеристик.

Затем с помощью команды **Edit/Designator** (рис. 3) определяются свойства компонентов, имеющих определенный префикс.

Эта операция позволяет определить основные электрические свойства компонентов. Здесь нелишне напомнить, что при создании проекта в графических редакторах следует

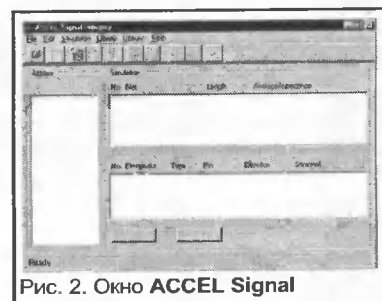


Рис. 2. Окно **ACCEL Signal**

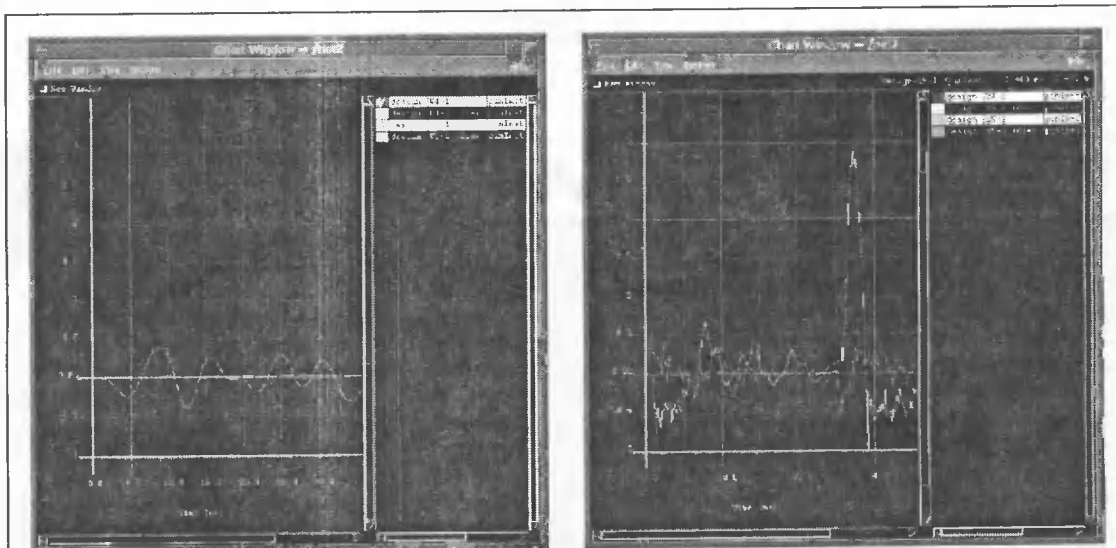


Рис. 1

придерживаться стандартных обозначений компонентов.

Следующим шагом является определение цепей питания с использованием команды **Edit Supply Nets** (рис. 4).

В этом окне каждой цепи питания задается соответствующий уровень напряжения.

Для проведения анализа необходимо определить структуру слоев печатной платы с помощью команды **Edit Layer Stack** (рис. 5).

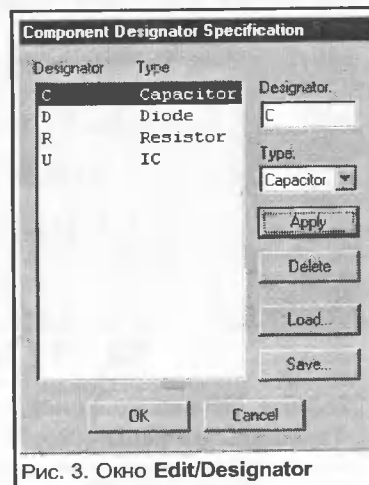


Рис. 3. Окно **Edit/Designator**

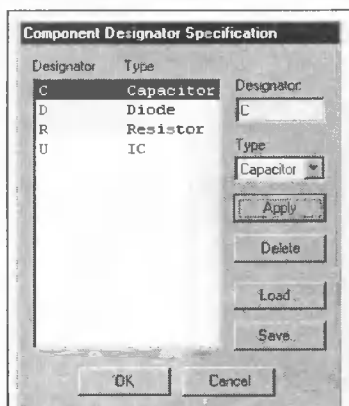


Рис. 4. Окно команды **Edit Supply Nets**

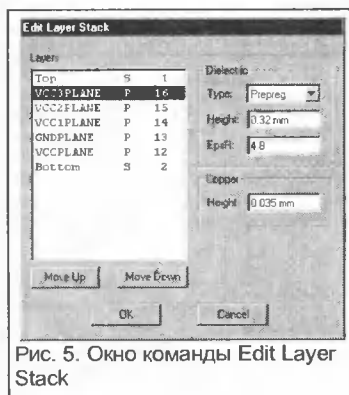


Рис. 5. Окно команды **Edit Layer Stack**

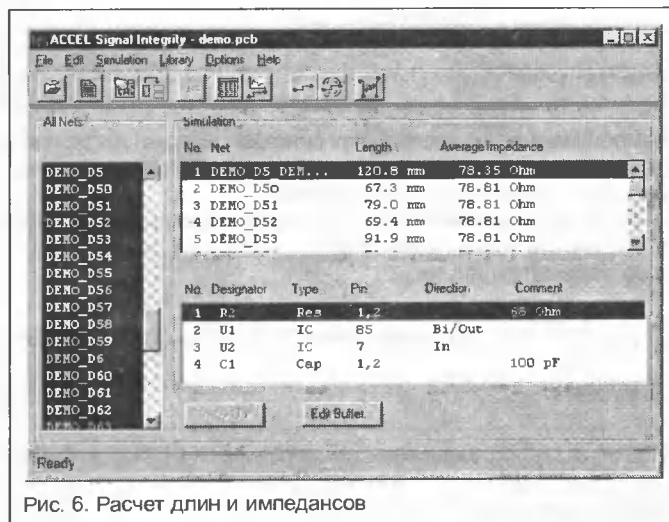


Рис. 6. Расчет длин и импедансов

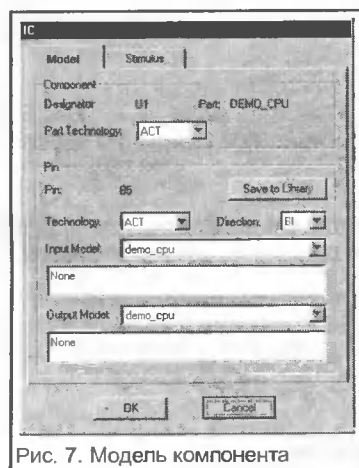


Рис. 7. Модель компонента

В данном окне задается последовательность слоев, толщина металлизации, расстояние между слоями металлизации и диэлектрическая постоянная материала платы. Кроме того, вводится информация о типе слоя (сигнальный, или плоскость металлизации).

Затем в окне в левой части экрана (рис. 2) выбираются цепи для проведения анализа.

С помощью команды **Edit/Take Over** цепи загружаются в окно моделирования (рис. 6).

В окне **Simulation** отображается список цепей, а также их длины и импедансы.

В нижней части экрана в списке выбираются компоненты и с помощью кнопки **Edit Buffer** производится редактирование информации о свойствах модели компонента (рис. 7) и его электрических характеристиках тестовых сигналов (рис. 8). Модель ком-

пONENTА может быть сохранена в библиотеке, а тестовые сигналы — в файле STM, который может использоваться при дальнейших процессах анализа.

Для резисторов и конденсаторов задаются значения их номинальных сопротивлений и емкостей (рис. 9, 10). Команды меню **Simulate** позволяют выполнить различные виды анализа.

Нажатием кнопки **Impedance View** (команда **Simulation/Screening**) вызывается список цепей с указанием их длин и рассчитанными импедансами (рис. 11).

Для инициализации анализа искажений сигналов используется команда **Simulation/Reflection** или соответствующая пиктограмма. На рис. 12 представлены результаты анализа искажений тестового сигнала.

Используя пиктограммы измерений (рис. 13), можно получить числовые значения интересующих параметров импульса (фронта, среза, выбросов и т. д.).

Способы нагрузки проводников могут быть выбраны с использованием окна **Termination Advisor** (рис. 14).

Анализ перекрестных наводок (**Crosstalk simulation**) выполняется с помощью команды **Simulation/Crosstalk**. Выбор пиктограммы **Set Victim Net** позволяет определить цепь-«жертву», наводки на которую будут анализироваться.

Запуск анализа выполняется с помощью пиктограммы **Crosstalk Simulation**.

Результаты анализа перекрестных наводок представлены на рис. 15.

Рассмотрим алгоритм, лежащий в основе работы анализатора импедансов.

Как известно, импеданс каждой дорожки печатной платы зависит от толщины слоя металлизации и ее ширины, диэлектрической постоянной материала платы, а также рас-

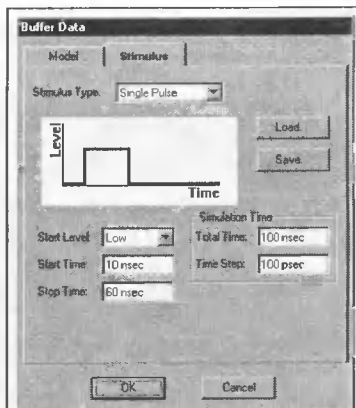


Рис. 8. Информация о тестовых сигналах

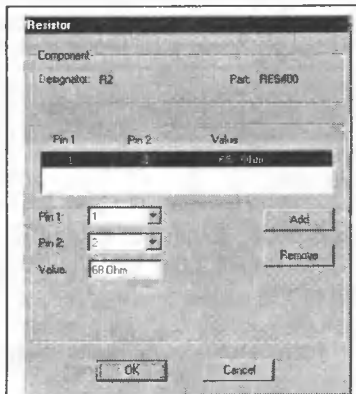


Рис. 9. Задание номинала резистора

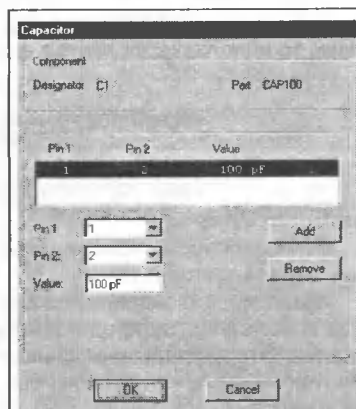


Рис. 10. Задание номинала конденсатора

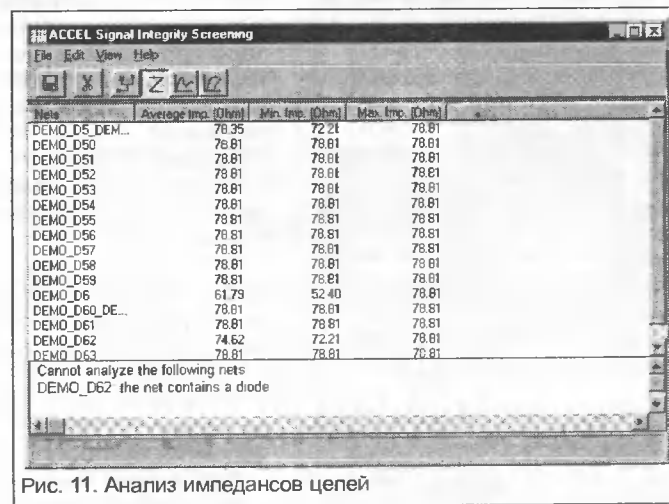


Рис. 11. Анализ импедансов цепей

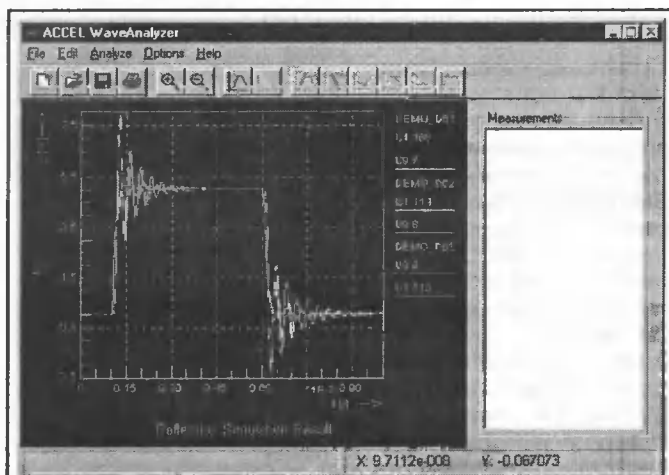


Рис. 12. Искажения тестового сигнала

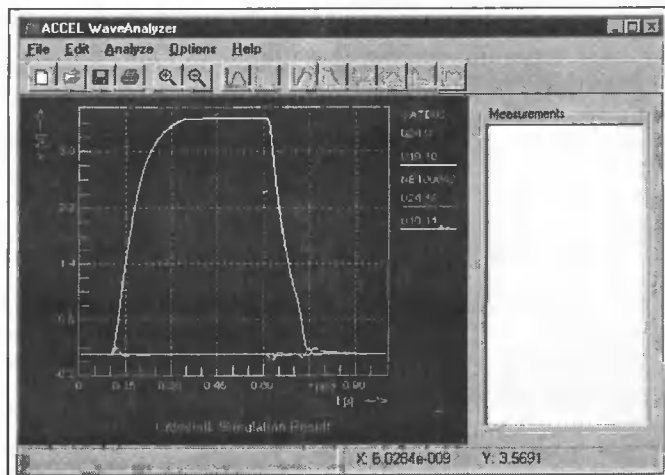


Рис. 15. Анализ перекрестных наводок



Рис. 13. Искажения тестового сигнала

стояния до сплошного слоя металлизации земли или питания. В основе расчета лежит методика расчета полосковых линий. Если сигнальный слой находится между двумя слоями металлизации, то при вычислении импеданса используется следующее выражение

$$\text{impedance} = \frac{60}{\sqrt{\text{diel_const}}} \times \log \left[\frac{4 \times \text{dist_between_planes}}{0,67 \text{pCu_width} \times A} \right],$$

$$\text{где } A = 0,8 + \frac{\text{Cu_height}}{\text{Cu_width}}.$$

Если дорожка находится над или под слоем металлизации, то используется другая формула:

$$\text{impedance} = \frac{87}{\sqrt{1,41 + \text{diel_const}}} \times \log \left[\frac{5,98 \times \text{dist_nearest_plane}}{0,8 \times \text{Cu_width} + \text{Cu_height}} \right]$$

Если плоскость металлизации отсутствует, то импеданс не определяется.

Известно, что современные высокоскоростные интерфейсы используют дифференциальную передачу и прием сигналов. Это накладывает определенные требования на проектирование печатных плат быстродействующих цифровых устройств.

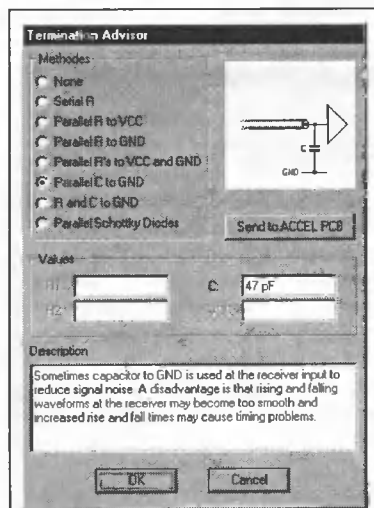
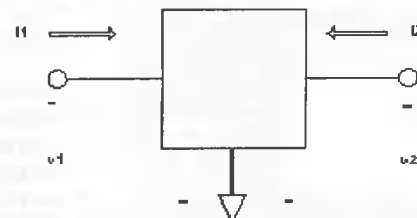


Рис. 14. Окно Termination Advisor

В случае использования дифференциальных интерфейсов сигнал передается по паре проводников, которые должны трассироваться на плате симметрично, чтобы не нарушить симметричности импедансов и не привести к рассогласованию линии.

Прежде чем рассмотреть основные особенности трассировки дифференциальных цепей, определим основные зависимости, характеризующие их электрические параметры.

Для любой схемы с двумя входами и общей



$$V_{\text{diff}} = V_1 - V_2$$

$$V_{\text{CommonMode}} = \frac{V_1 + V_2}{2}$$

$$Z_{\text{diff}} = \frac{V_{\text{diff}}}{I_1} \mid V_{\text{CommonMode}} = 0$$

$$Z_{\text{CommonMode}} = \frac{V_{\text{CommonMode}}}{I_1} \mid V_{\text{diff}} = 0$$

Рис. 16

землей можно определить как синфазное, так и дифференциальное напряжения и импедансы, описываемые приведенными на рис. 16 соотношениями.

Импедансы вычисляются как отношение напряжения к току в соответствующей цепи. Введем матрицы, характеризующие параметры линии.

$$C = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}, \text{ взаимные емкости } C_{12} = C_{21} \text{ для симметричных линий.}$$

$$L = \begin{bmatrix} L_{11} & L_{12} \\ L_{21} & L_{22} \end{bmatrix}, \text{ взаимные индуктивности } L_{12} = L_{21} \text{ для симметричных линий.}$$

$$Z = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{bmatrix}, \text{ взаимные импедансы } Z_{12} = Z_{21} \text{ для симметричных линий}$$

$$\begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = Z_0 \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}, \text{ для линии бесконечной длины без отражений.}$$

Рассмотрим дифференциальные импедансы. Если дифференциальное напряжение равно 1 В, синфазное равно 0 В имеем

$$\begin{bmatrix} 1/2 \\ -1/2 \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}.$$

Решая относительно I , получим

$$i_1 = \frac{1}{2(Z_{11} - Z_{12})} = -i_2$$

$$Z_{\text{diff}} = 2(Z_{11} - Z_{12})$$

Для синфазных импедансов аналогичные выкладки дают

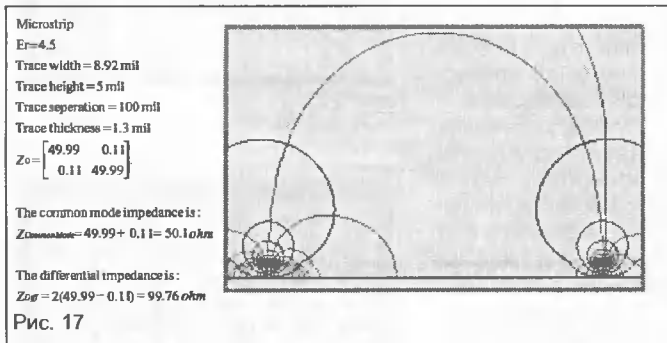
$$\begin{bmatrix} 1/2 \\ -1/2 \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \end{bmatrix}$$

Решая относительно i , получим

$$i_1 = \frac{1}{(Z_{11} + Z_{12})} = i_2$$

$$Z_{\text{comm}} = (Z_{11} + Z_{12})$$

Рассмотрим линии с сильной и слабой связью. На рис 17 приведен пример линий со слабой связью.



В этом случае собственные индуктивности и емкости линий оказывают незначительное влияние, поэтому матрица импедансов принимает вид

$$Z_0 = \begin{bmatrix} Z_{11} & 0 \\ 0 & Z_{22} \end{bmatrix} = \begin{bmatrix} Z_{11} & 0 \\ 0 & Z_{11} \end{bmatrix}$$

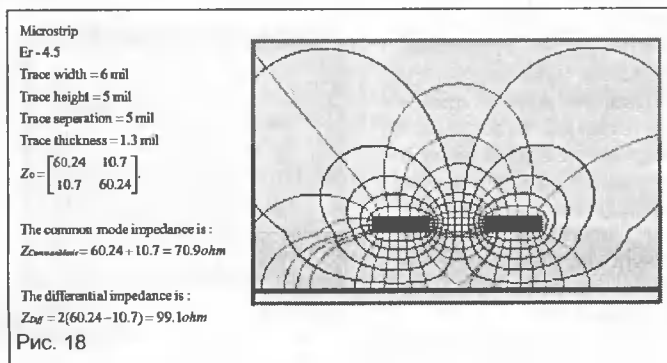
Синфазный импеданс

$$Z_{\text{comm}} = (Z_{11})$$

Дифференциальный импеданс

$$Z_{\text{diff}} = 2(Z_{11})$$

Рассмотрим пример сильной связи между линиями (рис. 18).



В этом случае возрастает влияние взаимных емкостей и индуктивностей, уменьшается дифференциальный импеданс, возрастает синфазный.

$$Z = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{12} & Z_{11} \end{bmatrix}$$

Синфазный импеданс

$$Z_{\text{comm}} = (Z_{11} + Z_{12})$$

Дифференциальный импеданс

$$Z_{\text{diff}} = 2(Z_{11} - Z_{12})$$

Нетрудно заметить, что если нагружать линию без учета взаимного влияния, дифференциальный сигнал может быть некорректно подавлен.

В табл. 1 приведены значения импедансов в зависимости от зазоров между трассами при характеристическом сопротивлении линии 50 Ом.

Таблица 1

Расстояние между дорожками, мкм	Собственный импеданс, Ом	Взаимный импеданс, Ом	Синфазный импеданс, Ом	Дифференц. импеданс, Ом
100	49,995	0,111	50,11	99,77
40	49,993	0,533	50,53	98,92
25	49,984	1,116	51,1	97,73
20	49,972	1,554	51,53	96,84
15	49,941	2,331	52,27	95,22
10	49,836	3,915	53,75	91,84
5	49,29	7,857	57,15	82,87

Начинающие разработчики делают традиционные ошибки, забывая о правилах согласования импедансов. В табл. 2 приведен рост коэффициента отражения в линии от уменьшения расстояния между дорожками при неправильном согласовании (линия 50 Ом, нагруженная на 100 Ом).

Таблица 2

Расстояние между дорожками	Синфазный импеданс, Ом	Дифференц. импеданс, Ом	Ошибка из-за неправильного согласования, %	Коэффициент отражения, %
100	50,11	99,77	0,2	0,12
40	50,53	98,92	1,1	0,54
25	51,1	97,73	2,3	1,15
20	51,53	96,84	3,2	1,61
15	52,27	95,22	4,8	2,45
10	53,75	91,84	8,2	4,25
5	57,15	82,87	17,1	9,37

Как видим, на плотных платах такие ошибки становятся непростительными. Поэтому при грамотном проектировании дифференциальных линий передач следует четко знать дифференциальные импедансы, характеристические сопротивления линий, сопротивления нагрузки возможное расстояние между проводниками на плате.

В близкорасположенных трассах нередко возникают наводки. Цепь «агрессор» — источник наводок и ее трассировка требует особого внимания, обычно «агрессорами» являются цепи тактовой синхронизации и им подобные с коротким временем нарастания фронтов. Следует уделять особое внимание трассировке рядом проходящих дифференциальных линий — надо использовать их корректное согласование.

Рассмотрим различные варианты согласования приемника сигнала с дифференциальной линией (Terminating Differential Pairs). Различные варианты согласования приведены на рис. 19.

Параллельное нагружение (Parallel termination) обеспечивает смещение по постоянному току и используется с открытыми выходами источника — открытый коллектор, эмиттер и т. п.

Дифференциальное нагружение (Differential termination) обеспечивает баланс токов между проводниками в паре. Однако, даже в случае небольшой асимметрии при трасси-

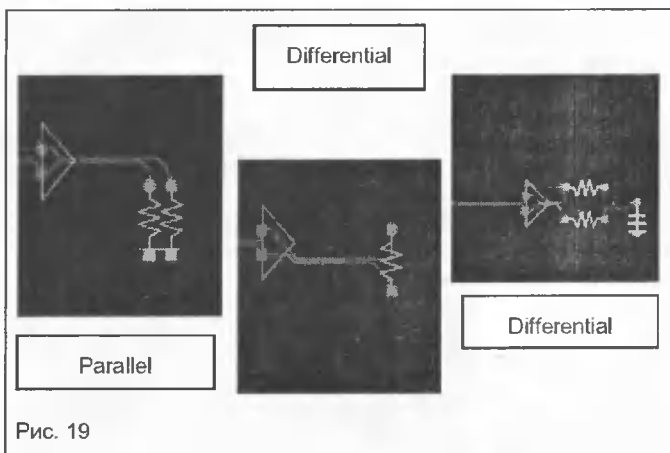


Рис. 19

ровке, такой способ нагружения приводит к рассогласованию по синфазной составляющей.

Комбинированное нагружение (*Differential tapped termination*) свободно от указанного недостатка.

Иногда дифференциальные линии трассируют на специально выделенном слое, находящемся между двух слоев общего провода (рис. 20).

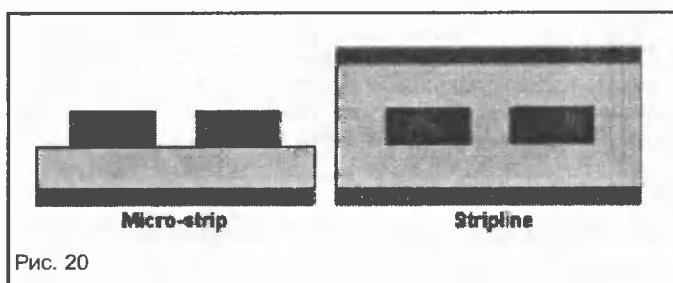


Рис. 20

Но этот способ имеет ряд недостатков, а именно: определенные требования к расстоянию от сбалансированной линии до переходных отверстий (ПО), которое не должно быть меньше определенной величины, чтобы внешние сигналы не создавали наводок на дифференциальной линии.

Это заставляет вводить разумные сетки расстановки ПО при использовании автотрассировщиков печатных плат.

При размещении на плате компонентов, содержащих дифференциальные цепи, следует придерживаться определенных правил (рис. 21).

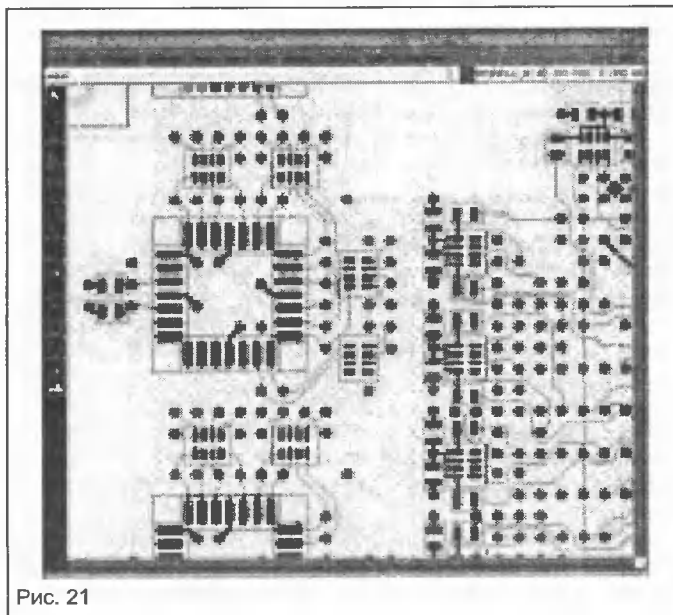


Рис. 21

Компоненты поверхностного монтажа должны располагаться симметрично относительно массива переходных отверстий, при расстановке компонентов шаг сетки следует установить равным шагу сетки ПО.

Следует использовать внешние слои платы для трассировки компонентов, не делая лишних переходов, то есть выполняя трассы во внешних слоях непрерывными.

При применении корпусов BGA значительно облегчает дело использование технологии микро ПО (mvia) непосредственно в контактной площадке (рис. 22).

Часто дифференциальные пары трассируют в нескольких слоях, проводя их между ПО попарно. При этом определяющую роль играет расположение переходных отверстий — сетка ПО не должна быть слишком плотной, чтобы между двумя ПО всегда можно было провести два проводника.

В первую очередь следует проводить трассировку таких дифференциальных пар по вертикали и горизонтали и лишь затем — по диагонали.

Пересечения дифференциальных пар в разных слоях для уменьшения наводок должны осуществляться под углом 90° (рис. 23).

На всем протяжении трассы проводники в дифференциальной паре не должны расходиться на расстояние, больше определенного, чтобы не нарушить симметричность линии. Исключение составляет только необходимость обхода узких мест.

При необходимости симметризовать несколько пар линий, особенно в случае использования многоконтактных соединителей, следует применять соответствующие меры — дополнительные петли (рис. 24). Введение дополнительных петель позволяет ввести дополнительную задержку и выровнять время распространения сигнала. Современные системы имеют соответствующие средства для описания таких приемов при трассировке.

При проектировании плат с использованием средств САПР необходимо всегда помнить, что сколь совершенными ни были алгоритмы автоматической трассировки, они никогда не заменят работу конструктора.

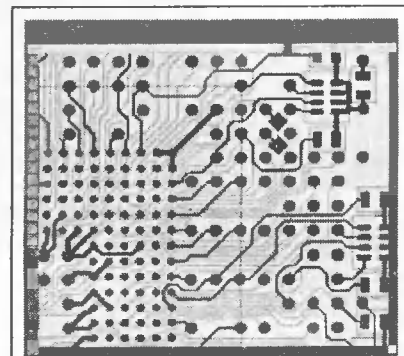


Рис. 22

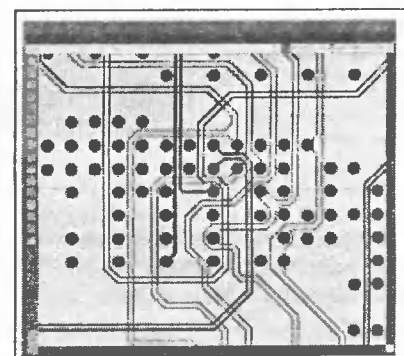


Рис. 23

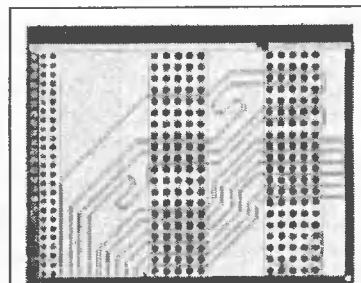


Рис. 24

Владимир Стешенко,
steshenk@sm.bmstu.ru

PSpice—модели цифровых устройств

Устройства интерфейса

Устройства интерфейса включаются между аналоговыми и цифровыми компонентами и выполняют две функции. Во-первых, с их помощью при моделировании электрических процессов в аналоговой части цепи задаются схемы замещения входных и выходных каскадов цифровых компонентов, соединенных непосредственно с аналоговыми. Во-вторых, они обеспечивают преобразование электрического напряжения в логический уровень и наоборот, чтобы осуществить обмен данными между подпрограммами моделирования электрических процессов в аналоговой части цепи и логического моделирования цифровой части. Устройства интерфейса подразделяются на устройства передачи данных от аналоговых компонентов на вход цифровых компонентов, называемых интерфейсом А/Ц (**Digital Output**) и на устройства передачи данных от цифровых на вход аналоговых компонентов, называемых интерфейсом Ц/А (**Digital Input**).

Если аналоговые и цифровые компоненты взаимодействуют в процессе моделирования, устройства интерфейса включаются в схему замещения цепи автоматически. Пользователь при этом не включает устройство сопряжения в файл описания цепи с расширением *.CIR. Для обеспечения такого режима предварительно в библиотеки цифровых компонентов вносятся ассоциируемые с каждым компонентом модели устройств интерфейса, оформленные в виде макромоделей. При расцеплении узла интерфейса на аналоговую и цифровую модели для автоматического включения устройства интерфейса программа Spice Schematics создает новый цифровой узел. Перечень этих узлов, созданных программой, помещается в выходном файле с расширением *.OUT в разделе **Generated AtoD and DtoA**. Заметим, что узел интерфейса характеризуется электрическим напряжением, а дополнительный цифровой узел — логическим состоянием. Пользователь имеет право непосредственно включить в описание цепи устройства интерфейса А/Ц и Ц/А, при этом программа новые устройства включать уже не будет.

Итак, каждому реальному цифровому компоненту ставятся в соответствие:

1) два устройства интерфейса для сопряжения с аналоговыми устройствами, которые могут подключаться к его входу и выходу, названные для краткости Ц/А и А/Ц. Они осуществляют обмен данными между подпрограммами моделирования аналоговых и цифровых устройств программы PSpice. А/Ц и Ц/А интерфейсы моделируются резисторами и конденсаторами, которые имеют полуфизический/полулогический смысл, в зависимости от интерпретации того или иного процесса, условия, требуемого состояния или просто здравого смысла. Имеются встроены модели интерфейсов, где заложены правила перевода сигналов (там же учитываются эффекты задержки и нелинейности);

2) модели вход/выход, отображающие его входные и выходные комплексные сопротивления;

3) модели динамики, учитывающие запаздывания сигналов. Модели цифровых компонентов и ассоциируемые с ними сопутствующие модели помещаются в специальные библиотеки.

Аналого-цифровой интерфейс

Назначение аналого-цифрового интерфейса: преобразование смешанного узла и аналогового сигнала в цифровой узел с цифровым сигналом, имитация входных каскадов цифровых ИС. Схема замещения А/Ц-интерфейса показана на рис. 3.

Информация о логическом уровне сигнала на выходе интерфейса А/Ц может распространяться в двух направлениях:

- в модуль логического моделирования программы PSpice;

- в файл (позднее он может быть просмотрен визуально или использован в качестве источника цифрового сигнала в следующем сеансе моделирования).

При взаимодействии аналоговых и цифровых компонентов в процессе моделирования описание интерфейса А/Ц (digital output: analog-to-digital) имеет формат:

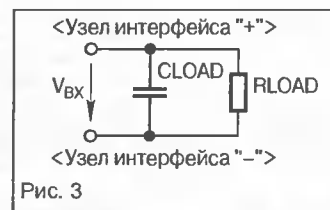


Рис. 3

Oxxx <+узел интерфейса> <-узел интерфейса> <имя модели А/Ц>

+ DGTNET=<имя цифрового узла> <имя модели вход/выход>

При записи логических уровней в файл описание интерфейса А/Ц имеет формат:

Oxxx <+узел интерфейса> <-узел интерфейса> <имя модели А/Ц>

+ [SIGNAME=<имя цифрового сигнала>]

Приведем примеры:

```
O12 ANALOG_NODE DIGITAL_GND DOUT_133
DGTNET=DIG_NODE IO_STD
OVCO170 TO_TTL SIGNAME=VCO_OUT; передача данных в файл
```

Модель аналого-цифрового интерфейса описывается предложением

.MODEL <имя модели А/Ц> DOUTPUT [<параметры модели>]

Параметры модели аналого-цифрового интерфейса приведены в табл. 1.

Каждому i-му логическому состоянию соответствует определенный диапазон напряжений **SiVLO...SiVHI**. До тех пор, пока входное напряжение интерфейса А/Ц V_{BX} не выходит за границы этого диапазона, логическое состояние на выходе интерфейса А/Ц не изменяется. В противном случае входное напряжение будет сравниваться с пороговыми уровнями, начиная с **SOVLO** до тех пор, пока оно не попадет в какой-нибудь интервал. Если оно не попадает ни в один интервал напряжений, логическому состоянию интерфейса А/Ц по умолчанию присваивается символ «?» (его можно изменить с помощью параметра SXNAME).

При взаимодействии с модулем логического моделирования программы PSpice имена логических состояний должны быть «0», «1», «X», «R», «F» или «Z» (однако состояние «Z» обычно не используется, так как состояние высокого импеданса не определяет уровень напряжения).

Параметры **TIMESCALE**, **FILE**, **CHGONLY** и **FORMAT** используются только при записи цифрового сигнала в файл. Параметр **TIMESTEP** определяет точность установления момента времени изменения логического состояния. При расчете переходных процессов в аналоговых цепях время интегрирования не может превышать значение **TIMESTEP**. По умолчанию этот параметр принимает меньшее из значений 1 нс или **1/DIGFREQ** (параметр **DIGFREQ** устанавливается по директиве **.OPTIONS**). Моменты времени, в которые записываются значения выборок сигнала, равны целым значениям величин

$$k = \text{TIMESCALE} \cdot \text{TIME} / \text{TIMESTEP},$$

где **TIME** — текущее время моделирования, **k** — номер отсчета. По умолчанию **TIMESCALE=1**. Для экономии машин-

Таблица 1. Параметры аналого-цифрового интерфейса

Идентификатор	Параметр	Значение по умолчанию	Единица измерения
RLOAD	Сопротивление нагрузки	1/GMIN	Ом
CLOAD	Емкость нагрузки	0	Ф
CHGONLY	Флаг преобразования: 0 — преобразование на каждом шаге по времени; 1 — преобразование при наличии изменения входного напряжения (только при записи в файл)		
SONAME	Имя логического состояния «0»	—	—
SOVLO	Нижний уровень напряжения лог. «0»	—	В
SOVHI	Верхний уровень напряжения лог. «0»	—	—
S1NAME	Имя логического состояния «1»	—	—
S1VLO	Нижний уровень напряжения лог. «1»	—	В
S1VHI	Верхний уровень напряжения лог. «1»	—	В
.....			
S19NAME	Имя логического состояния «19»	—	—
S19VLO	Нижний уровень напряжения логического состояния «19»	—	В
S19VHI	Верхний уровень напряжения логического состояния «19»	—	В
SXNAME	Имя логического состояния, когда напряжение на узле интерфейса находится вне заданных границ	—	—
FILE	Имя файла цифрового сигнала (только при записи в файл)	—	—
FORMAT	Код формата входного файла цифровых сигналов (только при записи в файл)	1	—
TIMESTEP	Интервал дискретизации по времени при записи в файл	10 ⁻⁹	с
TIMESCALE	Масштабный коэффициент при расчете интервала времени TIMESTEP (только при записи в файл)	1	—

ного времени следует назначать значение параметра **TIMESTEP**, исходя из скорости изменения напряжения.

При записи дискретизированного сигнала в файл его имя указывается в опции

SIGNAME= <имя файла>

Если эта опция опущена, то в качестве имени файла будет принято имя интерфейса А/Ц **Oxxx** без первого символа **O**.

Логическое состояние интерфейса А/Ц можно просмотреть с помощью программы Probe, указывая в качестве имени переменной **V(Oxxx)**, (см. следующий раздел «Цифро-аналоговый интерфейс») комментарий к интерфейсу Ц/А.

Моделирование входной цепи реального цифрового компонента с помощью пинейной RC-цепи, как показано на рис. 3, не всегда обеспечивает достаточную точность расчетов. Поэтому пользователь должен иметь в своем распоряжении несколько моделей входных цепей логических компонентов, оформляя их в виде макромоделей.

Цифро-аналоговый интерфейс

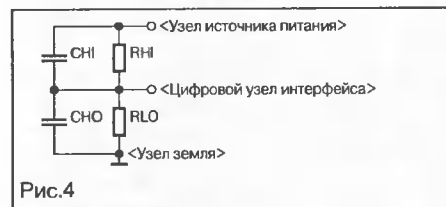


Рис.4

Назначение цифро-аналогового интерфейса: преобразование цифрового узла с цифровым сигналом («1», «0», «X», «R», «F» или «Z») в смешанный узел с аналоговым сигналом и

имитация выходных каскадов цифровых ИС. Схема показана на рис. 4.

Форма описания включения в схему:

Nxxx <узел интерфейса>

+ <узел источника опорного напряжения (-)>

+ <узел источника опорного напряжения (+)>

+ [<имя модели Ц/А>]

+ **DGTLNET**=<имя цифрового узла> <имя модели вход/выход>

+ [**IS**=<начальное состояние>]

Форма описания модели:

MODEL <имя модели Ц/А> **DINPUT** [<параметры модели>]

Устройства Ц/А интерфейса включают на входе аналоговых компонентов. Аналоговое напряжение образуется с помощью источника опорного напряжения и делителя на резисторах, сопротивления которых изменяются программно в соответствии с логическим уровнем цифрового сигнала. Информация о логическом уровне может быть получена из двух источников:

- в процессе расчета временной диаграммы логических устройств с помощью программы PSpice;
- из файла, который создан предварительно вручную или образован в результате моделирования в предыдущем сеансе работы с программой.

В связи с этим Ц/А интерфейс описывается двумя способами. При взаимодействии аналоговых и цифровых компонентов в процессе моделирования его описание имеет вид (digital input: digital-to-analog):

Nxxx <узел интерфейса> <-узел источника опорного напряжения>

+ <+узел источника опорного напряжения> <имя модели Ц/А>

+ **DGTLNET**=<UA<имя цифрового узла> <имя модели вход/выход>

+ [**IS**=<начальное состояние>]

При управлении из файла описание Ц/А интерфейса имеет формат

Nxxx <узелинтерфейса> <-узел источника опорного напряжения>

+ <+узел источника опорного напряжения> <имя модели Ц/А>

+ [**SIGNAME**=<имя цифрового сигнала>] [**IS**=<начальное состояние>]

Например:

```
N1 ANALOG DIGITAL GND
U_REFDIN_133 DGTNET=13
IO STD
```

N2 7 15 16 FROM TTL; Данные передаются из файла, имя которого указано в модели FROM_TTL

Модель Ц/А интерфейса описывается предложением

```
MODEL <имя модели Ц/А>
DINPUT [<параметры модели>]
```

Параметры модели цифро-аналогового интерфейса приведены в табл. 2.

Опция

```
DGTNET=<имя цифрового узла> <имя модели вход/выход>
```

указывает имя цифрового узла, к которому подсоединен интерфейс Ц/А, и имя модели входного/выходного сопротивления соответствующего цифрового компонента (см. ниже).

Начальное логическое состояние управляющего цифрового узла в момент времени $t=0$ определяется режимом схемы по постоянному току. Изменение этого состояния при необходимости производится с помощью необязательной опции

```
IS=<начальное состояние>.
```

Параметры модели **FILE**, **FORMAT** и **TIMESTEP** используются только при управлении из файла. При этом управление производится цифровым сигналом, имя которого указано в необязательной опции

```
SIGNAME=<имя цифрового сигнала>.
```

В ее отсутствие управление осуществляется сигналом, имя которого образовано символами **xxx**, находящимися справа от префикса **N** имени Ц/А интерфейса **Nxxx**.

Сопротивления резисторов в схеме замещения выходного каскада цифровых ИС (рис. 4) изменяются в соответствии с логическим уровнем управляющего цифрового узла. При его изменении сопротивления резисторов плавно изменяются по экспоненциальному закону с постоянной времени, которая определяется длительностью переключения, указанной в модели DINPUT. Эти сопротивления рассчитываются по формулам

$$RLO = R_{\text{вых}} \cdot E_n / (E_n - U_{\text{вых}}); \quad RHI = R_{\text{вых}} \cdot E_n / U_{\text{вых}}$$

где $R_{\text{вых}}$ и $U_{\text{вых}}$ — выходное сопротивление и уровень выходного напряжения в данном логическом состоянии; E_n — напряжение источника питания.

К управляющему цифровому узлу не должны подключаться какие-либо аналоговые компоненты, так как между ними автоматически будет включен аналого-цифровой интерфейс и цифровой узел отсоединится.

В процессе моделирования логические состояния управляющего цифрового узла получают имена «0», «1», «X», «R», «F» и «Z». Моделирование прерывается, если хотя бы одно из этих состояний не указано в перечне спецификаций **SnNAME**, **SnTSW**, **SnRLO** и **SnRHI**.

Логическое состояние входного цифрового сигнала можно просмотреть с помощью программы Probe на аналоговом экране, указывая в качестве имени переменной **B(Nxxx)**. Значение **B(Nxxx)** равно 0.0, если текущее состояние **SnNAME** равно 1.0 в состоянии **S1NAME** и т. д. вплоть до 19.0. Однако логические состояния входных цифровых узлов удобнее выводить на цифровой экран программы Probe.

Таблица 2. Параметры модели цифро-аналогового интерфейса

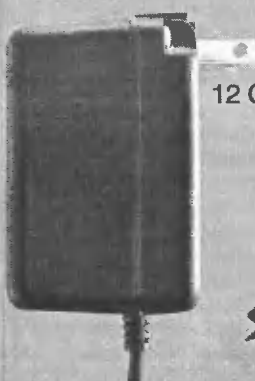
Имя	Параметр	Умолчение	Размерность
CLO	Емкость между входным узлом и "землей"	0	Ф
CHI	Емкость между входным узлом и источником питания	0	Ф
S0NAME	Имя логического состояния "0"		
S0TSW	Время переключения в состояние "0"		с
S0RLO	Сопротивление между входным узлом и землей в состоянии "0"		Ом
S0RHI	Сопротивление между выходным узлом и землей в состоянии "0"		Ом
S1NAME	Имя логического состояния "1"		
S1TSW	Время переключения в состояние "1"		с
S1RLO	Сопротивление между входным узлом и землей в состоянии "1"		Ом
S1RHI	Сопротивление между выходным узлом и землей в состоянии "1"		Ом
---	---	---	---
S19NAME	Имя логического состояния "19"		
S19TSW	Время переключения в состояние "19"		с
S19RLO	Сопротивление между входным узлом и землей в состоянии "19"		Ом
S19RHI	Сопротивление между выходным узлом и землей в состоянии "19"		Ом
FILE	Имя файла цифрового сигнала (только при записи в файл)		
FORMAT	Код формата выходного файла цифровых сигналов (только при записи в файл)	1	
TIMESTEP	Интервал дискретизации сигнала по времени (только при записи в файл)	10^{-9}	с

Олег Петраков
pertrakov@mtu-net.ru

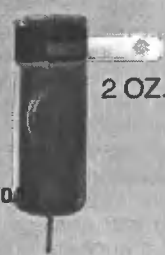
Продолжение следует



POWER
INTEGRATIONS, INC.



12 OZ.



2 OZ.

Разработать профессиональные импульсные источники питания позволяют микросхемы и программное обеспечение компании **Power Integrations**

- Время на оригинальную разработку — 2-3 недели
- Выигрыш в габаритах и количестве элементов
- КПД 90%
- Доступны по цене разработчика — около 1000 рублей



ATOC

125315, г. Москва, ул. Усиевича, 24/2, ООО «АТОС»
Телефоны и Факсы: (095) 787-48-05, 155-40-35
Интернет: www.atos.ru, info@atos.ru

Преобразователи уровней дискретных сигналов постоянного тока

В настоящей статье рассмотрены вопросы схемотехники преобразования уровней сигналов и гальванической развязки вход/выход, сформулированы основные задачи преобразования импульсных сигналов. Кроме того, в статье приведен краткий обзор импортных преобразователей и их общее схемотехническое решение, предложена принципиальная электрическая схема преобразователя, выполненного на отечественной элементной базе и указаны его основные технические характеристики.

Информация обычно бывает представлена в дискретном виде отдельными сигналами или их группами. Амплитуда входного сигнала зависит от характеристик источника и может достигать нескольких десятков вольт, форма из-за влияния помех может быть значительно искажена. Ввод и компьютерная обработка таких сигналов сопряжена с решением некоторых технических вопросов.

Вопрос первый — преобразование уровня входного сигнала до значений, применяемых в компьютерной технике. Другими словами, как выполнить масштабирование сигнала без потери его информационных свойств? Традиционно применяется уменьшение амплитуды от максимальных уровней 48, 36, 24 или 12 В до совместимых с ТТЛ или КМОП логикой. Чем выше амплитуда сигнала, тем больше энергия помехи для искажения его формы. Кроме того, уровень напряжения должен быть безопасным для человека.

Второй вопрос — гальваническое разделение входных и выходных цепей преобразователей уровней сигналов. Каким образом обезопасить компьютерную систему, если источник сформировал сигнал с амплитудой, в несколько раз превышающей принятое максимально допустимое значение? Например, возможно замыкание линий с напряжением в 220 В на входные линии, рассчитанные на 24 В. Для гальванического разделения применяют электромагнитное или оптическое разделение в среде передачи. В настоящее время удобнее и выгоднее применять второй вариант.

Третий вопрос связан с влиянием окружающей среды на информационное состояние сигналов. В частности, это дальность передачи сигнала, уровень электромагнитных и электростатических помех и т. д. Обычно применяют два способа защиты: исключение искажения сигнала (экранировка, простейшее разделение линий и т. п.) или допущение искажений в линии с последующей фильтрацией аппаратным или программным способом. С практической точки зрения рекомендуется применять оба способа. Комбинирование вариантов защиты снижает затраты и одновременно повышает надежность получения достоверной информации.

Учитывая изложенные выше основные вопросы, рассмотрим схемотехни-

ку устройств, позволяющих преобразовывать уровни сигналов постоянного тока, гальванически разделять входные и выходные цепи и отфильтровывать некоторые виды помех.

На рынке электронных компонентов России имеется достаточно широкий спектр преобразователей различных зарубежных фирм. К сожалению, их отечественных полнофункциональных аналогов в настоящее время не выпускается. В табл. 1 для примера приведены некоторые характеристики трех типов преобразователей по данным www.prosoft.ru. Номенклатура поставляемых этой фирмой преобразователей значительно шире. Есть и другие поставщики аналогичного оборудования, его технические характеристики схожи. Это объясняется стандартизацией и унификацией оборудования на мировом рынке.

Таблица 1

Тип	Уровень входного сигнала		Состояние выходного ключа		Напряжение питания логической части	Фирма изготовитель	Габариты HxLxB мм	Цена \$
	Лог. 1	Лог. 0	Лог. 1	Лог. 0				
70G-IDC5	3-32 В x 18 мА	0-3 В	выкл.	вкл.	5 В	Grayhill (США)	55x48x12	12,73
70M-IDC24	3-32 В x 18 мА	< 3 В	выкл.	вкл.	24 В	Grayhill (США)	32x43x10	12,08
PCLM-IDC5B	3-32 В x 12 мА	< 1 В	выкл.	вкл.	5 В	Advantech (Тайвань)	—	13

Модули устанавливают на специальные монтажные платы, снабженные нагрузочными резисторами, светодиодами и разъемами. Тип платы определяет логику выходных сигналов — прямую или инверсную. Выпускаются платы ввода дискретных сигналов, встраиваемые непосредственно в компьютер и имеющие гальваническую развязку. Для одинакового числа входных сигналов цена встраиваемых и модульных преобразователей соизмерима (первые немного дешевле).

Схемотехника преобразователей от различных фирм-изготовителей практически одинакова. На рис. 1 приведена упрощенная принципиальная схема модуля ввода. Резистор R1 ограничивает ток, протекающий через излучатель оптрона U1. Светодиод HL1 индицирует наличие питания логической части схемы. Преобразователь напряжения и регулятор тока обеспечивают работу транзистора VT1 в ключевом режиме. Следует заметить, что такое схемотехническое решение

весьма условно, но оно хорошо согласуется с данными табл. 1. Например, максимальный ток 18 мА достаточен для работы только излучателя оптрона и не допускает параллельного включения других стабилизирующих элементов. Более полных данных изготовители не предоставляют, за исключением расположения и функционального назначения выводов модулей.

Рассмотрим принципиальную электрическую схему преобразователя с использованием отечественных элементов. Кстати, стоимость одного такого преобразователя значительно ниже его «зарубежного коллеги». Входные цепи нашей схемы будут незначительно усложнены, но применение дополнительных элементов оправдано следующими предпосылками.

Импульс, поступающий на вход преобразователя, в большинстве случаев искажен помехами (рис. 2). В верхней части «площадки» и на фронте и срезе импульс наиболее незащищен. Значительных провалов в средней части импульса обычно не бывает, т. к. его энергия в этот момент значительно больше энергии помехи. А вот всплески иногда наблюдаются. Чтобы информация, содержащаяся в таком импульсе, была достоверной, необходимо подвергнуть анализу его центральную часть, как наиболее защищенную. Следовательно, устройство должно исключать верхнюю и нижнюю части сигнала с определенными уровнями напряжений. Значения $U_{огр}$, $U_{ср}$ — это уровни напряжения, выше и ниже значений которых сигнал

не должен регистрироваться. Ограничивая таким образом амплитуду сигнала, мы исключаем из рассмотрения переходные процессы и часть помех. Искажения на фронте и срезе сигнала удаляются смещением временного интервала для получения «чистой» информационной области. Сформированный таким образом новый импульс, а именно его часть с t_1 по t_2 , пригоден для дальнейшей обработки, т. е. гальванического разделения и масштабирования. Импортные преобразователи описанных выше действий, судя по схемотехнике, не выполняют.

Рассмотрим принципиальную электрическую схему преобразователя, рассчитанного на входной сигнал 24 В и с ТТЛ уровнями на выходе (рис. 3).



Рис. 1. Схема модуля ввода

Резистор R1 определяет входной ток преобразователя, необходимый для обеспечения режима стабилитронов VD1 и VD2. Совместно с конденсатором C1 резистор R1 образует интегрирующую цепочку, увеличивающую помехоустойчивость (на рис. 3 это фронт и срез сформированного импульса). Стабилитрон VD1 повышает порог срабатывания до $U_{cp} = 5,6$ В. При меньшем напряжении на входах преобразователь нечувствителен к изменениям сигналов. Наличие VD1 обязательно при использовании бесконтактных датчиков, имеющих высокий уровень остаточного напряжения. Элементы R1, VD1, VD2 образуют параметрический стабилизатор напряжения, питающий излучатель оптрона U1 и

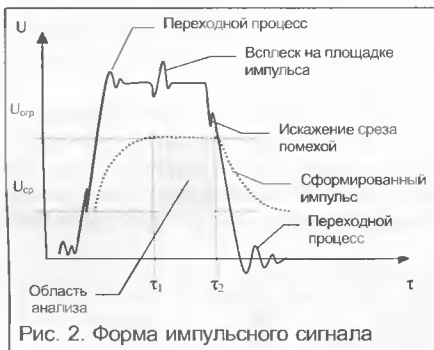


Рис. 2. Форма импульсного сигнала

светодиод HL1, индицирующий состояние входного сигнала. Значение $U_{огр}$ равно 5,6 В, начиная от U_{cp} . Кроме того, VD2 осуществляет защиту от пробоя излучателя оптрона и светодиода при наруше-

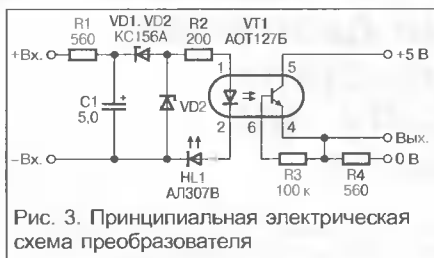


Рис. 3. Принципиальная электрическая схема преобразователя

нии полярности входного сигнала. Резистор R2 ограничивает ток излучателя и светодиода на уровне 10...15 мА. В качестве элемента гальванической развязки применен транзисторный оптрон U1. Следует заметить, что при отсутствии излучения в цепи коллектора фототранзистора протекает обратный (темновой) ток, аналогичный по происхождению и характеристикам току в обычных биполярных транзисторах. Обратный ток сильно зависит от температуры. При повышении температуры на 10°C он удваивается. Чтобы исключить ложные срабатывания и уменьшить темновой ток между выводами базы и эмиттера фототранзистора включен внешний резистор R3. Наличие темнового тока характерно для всех оптоэлектронных пар без исключения. Номинал R3 может изменяться в диапазоне 0,1 ... 1 МОм. Резистор R4 является нагрузкой фототранзистора. Преобразователь является неинвертирующим, т. е. высокому уровню сигнала на входе соответствует уровень логической единицы на выходе.

Технические характеристики преобразователя:

- уровни входного сигнала: высокий — $24 \pm 4,8$ В, низкий уровень — не более 5 В;
- максимальный выходной втекающий ток сигнала высокого уровня — не более 30 мА;
- минимальный входной втекающий ток сигнала высокого уровня — не менее 10 мА;
- электронная защита от обратной полярности входного напряжения;
- световая индикация входного сигнала;
- напряжение гальванической развязки при температуре 25°C — до 1500 В;
- задержка реакции на включение и выключение — 5 ± 2 мс.

С помощью преобразователя можно безопасно вводить сигналы в любое цифровое устройство. Время реакции можно уменьшить, исключив из схемы конденсатор C1. Все элементы можно заменить импортными аналогами. Схемотехническое решение применимо для ввода сигналов с различными максимальными уровнями, но для этого необходимо изменить номиналы некоторых элементов (в первую очередь R1). Область импульса, выбираемая для дальнейшего анализа, корректируется напряжением стабилизации стабилитронов и емкостью конденсатора C1. Напряжение стабилизации в этой связи может быть различным для VD1 и VD2.

Игорь Дьяков,
igord@sapr.tstu.ru

Открылся новый магазин

РАДИОДЕТАЛИ

на «Курской»

Цена зависит от количества!

Москва, ул. Земляной вал, д. 34,
метро «Курская»
Справочная служба:
(095) 916-0408
Оптовый отдел:
(095) 916-2321
promelec@co.ru
www.promelec.ru

Для **оптовых и розничных покупателей** действует система специальных и накопительных скидок

Рядом с м. «Курская»

Широкий, **более 40 тысяч наименований**, ассортимент радиодеталей, измерительного оборудования и инструмента

Специально подготовленный персонал

Большой **оптовый отдел**

Весь **товар сертифицирован, с гарантией качества**

Специальный **отдел технической литературы** по электронной тематике

ПРОМЭЛЕКТРОНИКА
ПОСТАВКИ ЭЛЕКТРОННЫХ КОМПОНЕНТОВ

РАДИОДЕТАЛИ

Быстродействующий диспетчер персонального компьютера RS-232C — RS-232C/RS-485 с оптической развязкой

Настоящая статья призвана ознакомить читателей со схемотехникой быстродействующего диспетчера персонального компьютера, его принципом работы и особенностями наладки.

В современной инженерной и радиолокационной практике достаточно часто бывает необходимо связать персональный компьютер с различным оборудованием, командно-информационными сетями на основе интерфейсов RS-232C или RS-485, измерительными приборами, программами и другими устройствами, оснащенными этими интерфейсами. Такие работы, особенно на стадии наладки и макетирования, связаны с риском вывода из строя ПК и требуют гальванической изоляции подключаемых изделий от цепей ПК. Эта проблема решается с помощью специальных узлов, обеспечивающих оптическую изоляцию входных цепей интерфейса RS-232C, гальванически связанного с цепями ПК, от выходных цепей интерфейсов RS-232C или RS-485, связанных с подключаемыми приборами. Такие узлы принято называть диспетчерами персонального компьютера (DPC — Dispatcher PC).

Выбор элементной базы

Важнейшей характеристикой современных DPC является их быстродействие. Хотя большинство промышленных приборов и контроллеров с вышеупомянутыми интерфейсами работают на достаточно низких скоростях передачи данных (обычно в диапазоне от 9600 до 38400 бод), обеспечение высокого быстродействия собственно DPC резко сокращает вероятность возникновения ошибок рассогласования. Это особенно важно при организации распределенных систем управления на базе командно-информационных сетей [1].

Оптическая изоляция обычно выполняется на микросхемах транзисторных оптронов. Однако большинство отечественных и зарубежных оптронов широкого применения характеризуется достаточно низким быстродействием. Наиболее важные параметры оптронов приведены в табл. 1.

Анализ приведенной таблицы показывает, что практически все отечественные и большинство зарубежных оптронов имеют время включения/выключения (ON/OFF) больше 1 мкс, что с трудом позволяет обеспечивать устойчивую работу на скоростях 19200 и 28800 бод. Кроме того, параметры оптронов зависят от партии и производителя. Как показали исследования, проведенные автором, параметры быстродействия многих отечественных оптронов, да и некоторых

импортных, значительно завышены. Фактически, все оптроны по быстродействию можно разделить на три группы: с высоким быстродействием (6N137, HCPL-2601, HCPL-2611, HCPL-2630, HCPL-2631), со средним быстродействием (6N136, CNW4502, HCPL-4502) и с низким быстродействием, к которым следует отнести все остальные оптроны. Оптроны с высоким быстродействием достаточно дороги и их трудно достать. Из оптронов средней группы наиболее доступен 6N136. Его испытания показали, что он устойчиво работает даже на

Таблица 1

Тип	Ток светодиода, мА	U _{ЭК} мккс, В	U _{НЭС} мккс, В	-N/-F _{макс} , МКС	Напряжение изоляции, кВ
3OT122	<15		1,5	100	0,1
3OT127	<15		1,5	100	0,5
3OT13	<10		1,5		1
3OT131A	<30		1,5	>20*	1
3OT135	<20		1,5	>20*	1
3OT138	<25		0,3		1
3OT142	<30		1,5		10
3OT144A	<25		0,3		3
3OT146	<30		0,4		
3OT150A	<20		1,5		3
4N25	<12	30	0,5	1,2/1,3	5,3
4N26	<12	30	0,5	1,2/1,3	5,3
4N27	<12	30	0,5	1,2/1,3	5,3
4N28	<12	30	0,5	1,2/1,3	5,3
4N35	<12	30	0,3	10/10	5,3
4N36	<12	30	0,3	10/10	5,3
4N37	<12	30	0,3	10/10	5,3
4N38	<12	80	1	—	5,3
6N135	16	15	0,4	1,5/1,5	2,5
6N136	16	15	0,4	0,8/0,8	2,5
6N137	5	7	0,6	0,075/0,075	2,5
6N138	1,6	7	0,4	10/35	2,5
6N139	0,5	18	0,4	1/7	2,5
CNG35	<12	30	0,4	20/20	5,3
CNG36	<12	30	0,4	20/20	5,3
CNW11AV-1	<12	70	0,4	15/15	4
CNW11AV-2	<12	70	0,4	15/15	4
CNW11AV-3	<12	70	0,4	15/15	4
CNW135	16	30	0,4	1,5/1,5	5
CNW136	16	30	0,4	0,8/0,8	5
CNW138	1,6	7	0,4	10/35	5
CNW139	0,5	18	0,4	1/7	5
CNW4502	16	30	0,4	0,8/0,8	5
CNW82		50	0,4	—	5,9
CNW83	<16	50	0,4	—	5,9
CNW84		80	0,4	—	5,9
CNW85	<16	80	0,4	—	5,9
CNX35U	<16	30	0,4	20/20	5,3
CNX36U	<16	30	0,4	20/20	5,3
CNX38U	<16	80	0,4	20/20	5,3
CNX82A		50	0,4	—	5,3
CNX83A	<16	50	0,4	3/3	5,3
CNY17	<12	70	0,4	10/10	5,3
CNY17(G/F)-(1-4)		70	0,4	10/10	5,3
H11A(1-5)	<12	30	0,4	—	5,3
H11AG1		30	0,4	—	5,3

скорости 115200 бод, что является пределом скорости большинства COM-портов современных компьютеров. В связи с этим, этот тип оптрона и был признан автором в качестве оптимального.

Очевидно, что в нашем изделии должны быть еще микросхемы драйверов RS-232C и RS-485. В настоящее время выпускается более 600 различных типов микросхем драйверов RS-232C и более 300 типов драйверов RS-485. Наиболее известные и распространенные из них производят фирмы Maxim и Analog Devices. В настоящее время выпускаются ИМС с различным количеством приемников и передатчиков в одном корпусе. Для нашего изделия необходимы микросхемы, содержащие два приемника и два передатчика интерфейса RS-232C и один приемопередатчик RS-485.

В табл. 2 приведены наиболее известные микросхемы интерфейса RS-232C, содержащие два приемника и два передатчика. В таблицу включены только те изделия, которые имеют встроенный

Окончание таблицы 1

Тип	Ток светодиода, мА	U _{ЭК МАКС} , В	U _{НАС МАКС} , В	-N/-Ff _{МАКС} , МКС	Напряжение изоляции, кВ
H11AG2		30	0,4	—	5,3
H11AG3		30	0,4	—	5,3
H11D(1-4)		300	0,4	—	5,3
HCPL-2503	16	15	0,4	0,8/0,8	2,5
HCPL-2530	16	30	0,5	1,5/1,5	2,5
HCPL-2531	16	30	0,5	0,8/0,8	2,5
HCPL-2601	5	7	0,6	0,075/0,075	2,5
HCPL-2611	5	7	0,6	0,075/0,075	2,5
HCPL-2630	5	7	0,6	0,075/0,075	2,5
HCPL-2631	5	7	0,6	0,075/0,075	2,5
HCPL-2730	1,6	7	0,4	20/35	2,5
HCPL-2731	0,5	18	0,4	2/10	2,5
HCPL-4502	16	15	0,4	0,8/0,8	2,5
IL1	<12	50	0,25	—	5,3
IL2	<12	70	0,25	—	5,3
IL201	<12	70	0,4	—	5,3
IL202	<12	70	0,4	—	5,3
IL203	<12	70	0,4	—	5,3
IL5	<12	70	0,25	—	5,3
IL74	<12	20	0,3	3/3	5,3
MCT2		30	0,4	1,2/1,3	5,3
MCT210		30	0,4	—	5,3
MCT2200		30	0,4	10/10	5,3
MCT2201		30	0,4	10/10	5,3
MCT2202		30	0,4	10/10	5,3
MCT270		30	0,4	10/10	5,3
MCT271		30	0,4	7/7	5,3
MCT272		30	0,4	10/10	5,3
MCT273		30	0,4	20/20	5,3
MCT274		30	0,4	25/25	5,3
MCT275		30	0,4	15/15	5,3
MCT276		30	0,4	3,5/3,5	5,3
MCT277		30	0,4	15/15	5,3
MCT2E		30	0,4	1,2/1,3	5,3
MCT5200		30	0,4	12/20	5,3
MCT5201		30	0,4	30/13	5,3
MCT5210		30	0,4	—	5,3
MCT5211		30	0,4	—	5,3
MCT6	<10	30	0,4	—	2,5
MCT61	<10	30	0,4	—	2,5
MCT62	<10	30	0,4	—	2,5
MCT9001	<10	55	0,4	—	2,5
MOC8100		30	0,5	20/20	7,5
MOC8111		30	0,4	10/10	5,3
MOC8112		30	0,4	10/10	5,3
MOC8113		30	0,4	10/10	5,3
MOC8204		400	0,4	—	7,5
MOC8205		400	0,4	—	7,5
MOC8206		400	0,4	—	7,5
PC111L	<12	35	0,2	4/3	5
PC112L	<12	70	0,2	4/3	5
PS2501-2	<10	80	—	3,5	5
PS2511-2	<10	40	—	3,5	5
PS2561-2	<10	50	—	3,5	5
PS2601		50	—	3,5	5
PS2651		50	—	3,5	5
PS2701-2	<10	80	—	3,5	3,75
SFH600-(0-4)		70	0,4	3/3	5,3
SFH601-(0-4)		100	0,4	3/2,3	5,3
SFH608-(2-5)		55	0,4	8/7,5	5,3
SFH640-(1-5)		300	0,4	5/6	7,5
SL5500		30	0,4	20/50	5,3
SL5501		30	0,4	20/50	5,3
SL5504		80	0,4	50/150	5,3
SL5511		30	0,4	20/50	5,3
SL5582		50	0,4	20/50	5,3
SL5582W		50	0,4	20/50	5,9
SL5583		50	0,4	20/50	5,3
SL5583W		50	0,4	20/50	5,3
TIL11	<12	30	0,4	10/10	5,3
TIL111	<12	30	0,4	5/5	1,5
TIL112	<12	20	0,5	2/2	1,5
TIL117	<12	30	0,4	5/5	2,5

преобразователь напряжения, выпускающиеся в корпусе DIP с количеством выводов не более 20 и работающие от одного напряжения питания +5 В.

Наилучшими качествами при приемлемой цене, а также максимальной доступностью обладают микросхемы MAX232E и ее аналоги.

Для микросхем интерфейса RS-485 ограничимся сведениями только о двух типах микросхем, содержащих один приемопередатчик, работающих в полудуплексном режиме, выпускающихся в корпусе DIP8. Эти микросхемы работают от одного напряжения питания +5 В (табл. 3).

Принципиальная схема

Принципиальная схема устройства приведена на рис. 1. Устройство содержит:

- входной приемопередатчик RS-232C D1 (MAX202, AMD202, MAX232, и т. д.), обеспечивающий связь с персональным компьютером; узел оптической развязки D2-D5 (6N136), осуществляющий развязку входных цепей, соединенных с персональным компьютером, от выходных цепей, связанных гальванически с локальной сетью или периферийным устройством;
- выходной приемопередатчик RS-232C D6 (MAX202, AMD202, MAX232, и т. д.), осуществляющий связь с выходным RS-232 каналом;
- выходной приемопередатчик RS-485 D7 (MAX485, MAX487).

Кроме этого, в состав устройства входят два изолированных друг от друга линейных пятивольтовых стабилизатора для раздельного питания входной и выходной части. Они реализованы на микросхемах A1, A2 (78L05).

Устройство обеспечивает умощнение с оптической развязкой двух каналов интерфейса RS-232C персонального компьютера. Первый канал составляют штатные линии приемопередатчиков (RxD и TxD), второй канал составляют вспомогательные линии интерфейса (CTS и RTS), которые можно использовать по прямому назначению для организации обмена или как дополнительный канал приема/передачи с программной эмуляцией обмена. Узел оптической развязки выполнен по классической схеме включения оптронов 6N136. В случае, если высокое быстродействие дополнительного канала не требуется, можно заменить оптроны D3 и D5 на более распространенные и дешевые оптроны 4N35. На рис. 2 показано расположение выводов оптронов 6N136 и 4N35. Оптрон 6N136 может быть заменен на 4N35 путем установки последнего в колодку DIP8 так, чтобы первый вывод оптрона 4N35 вошел во второй вход колодки. Очевидно, что при такой замене быстродействие второго канала заметно упадет. Оптоны 6N136 обеспечивают устойчивую передачу на скоростях до 115200 бод включительно, в то время как оптроны 4N35 с трудом работают на скорости 38400 бод, а устойчиво ведут себя только на скорости 28800 бод.

Таблица 2

Тип	Скорость, kbps	Напряжение питания, В	Ток потр., мА	Количество конденсаторов	Номинальная емкость, мкФ	±15кВ ESD	Корпус
MAX202, HIN202, AMD202, LT1181, SP232A	20	5	8	4	0,1		DIP16
MAX202E, HIN202E, SP202E, AMD202E	120	5	8	4	0,1	✓	DIP16
MAX203, HIN203, AMD203, SP233A	116	5	8				DIP20
MAX203E, HIN203E	120	5	8			✓	DIP20
MAX220	20	5	0,5	4	4,7/10		DIP16
MAX222, AMD222, LT1180	116	5	4	4	0,1		DIP18
MAX232, HIN232, LT1081, AMD232A	116	5	5	4	1		DIP16
MAX232A, HIN232A	116	5	4	4	0,1		DIP16
MAX232E, HIN232E, SP232E, LT1281, LT1381, DS14C232	120	5	5	4	1	✓	DIP16
MAX233, HIN233, AMD233L	116	5	5				DIP20
MAX233A, HIN233A	116	5	4				DIP20
MAX242, ICL3222, AMD242	116	5	4	4	0,1		DIP18
MAX243	116	5	4	4	0,1		DIP16
MAX250 Isolated	116	5	0,1				DIP14
MAX251 Isolated	116	5	0,1				DIP14
MAX3222	120	3 ÷ 5,5	0,3	4	0,1		DIP18
MAX3222E	250	3 ÷ 5,5	0,3	4	0,1	✓	DIP18
MAX3223	120	3 ÷ 5,5	0,3	4	0,1		DIP20
MAX3223E	250	3 ÷ 5,5	0,3			✓	DIP20
MAX3224	250	3 ÷ 5,5	0,3	4	0,1		DIP20
MAX3224E	250	3 ÷ 5,5	0,3	4	0,1	✓	DIP20
MAX3225	1000	3 ÷ 5,5	0,3	4	0,1		DIP20
MAX3225E	1000	3 ÷ 5,5	0,3	4	0,1	✓	DIP20
MAX3232, AMD3202	120	3 ÷ 5,5	0,3	4	0,1		DIP16
MAX3232E	250	3 ÷ 5,5	0,3	4	0,1	✓	DIP18
MAX3235E	250	5	0,3			✓	DIP20

Таблица 3

Тип	Скорость, Mbps	Напряжение питания, В	Ток потр., мА	Нагрузочная способность, количество приемопередатчиков	±15кВ ESD	Корпус
MAX485, ADM485, DS75176 / DS96176	2,5	5	0,5	32	✓	DIP8
MAX487	0,25	5	0,25	128	✓	DIP8

Напряжение изоляции входной и выходной частей изделия составляет 2,5 кВ при установке хотя бы одного оптрона 6N136 и 5,3 кВ при установке всех оптронов 4N35. Следует отметить, что напряжение изоляции определяется также используемым трансформатором. В устройстве рекомендуется использовать любой из трансформаторов ТПП213-ТПП218, ТПП222-ТПП225 (220/127-50),

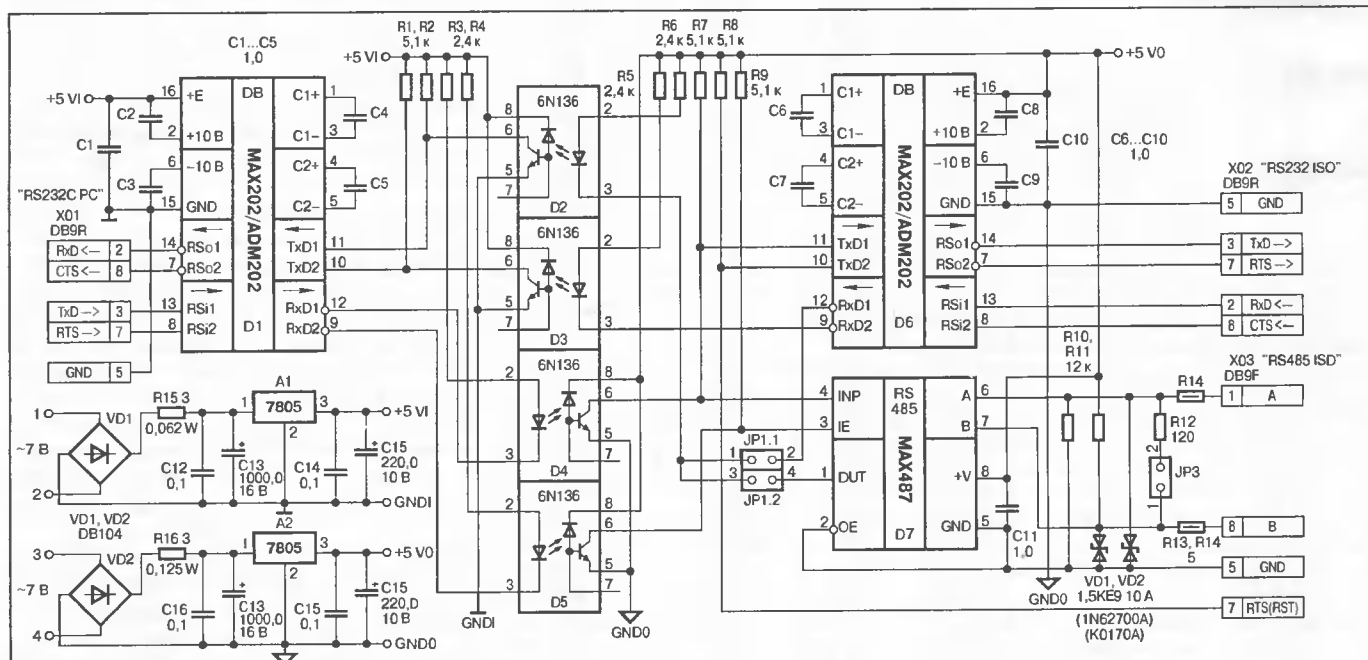


Рис. 1. Принципиальная схема быстродействующего диспетчера персонального компьютера.

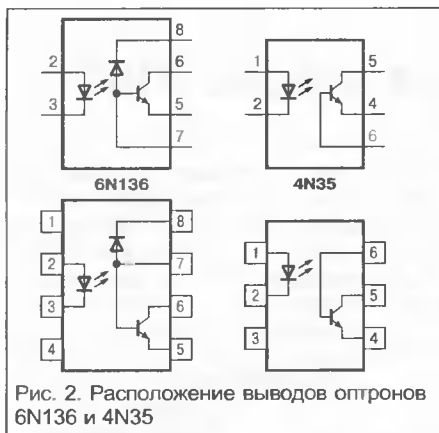


Рис. 2. Расположение выводов оптронов 6N136 и 4N35

которые обеспечивают напряжение изоляции между обмотками выше 7 кВ.

Особенностью выходной части является то, что передача информации от DPC к периферийному устройству (устройствам) не коммутируется, т. е. ведется одновременно по выходным основному каналу RS-232 и каналу RS-485. Прием информации — коммутируемый. Выбор канала поступления данных от периферийных устройств осуществляется с помощью установки перемычек JP1 или JP2. К штырям JP1/2 может быть подключен малогабаритный тумблер, установленный на передней панели и осуществляющий переключение выходных каналов RS-232/RS-485.

Канал интерфейса RS-485 защищен от перенапряжений с помощью супрессоров VD1, VD2 (1.5KE9.1CA или 1N6270CA) на напряжение 7,2 В. Супрессоры — это быстродействующие мощные стабилитроны (индекс CA указывает, что они двунаправленные), обеспечивающие срабатывание при напряжении выше 7,2 В и замыкающие на себя ток до 40 А. В лабораторных условиях, если подключаемая линия не очень длинная (до 10 м), их можно с успехом заменить на отечественные стабилитроны КС170/175А. Ограничение входного тока и роль предохранителей на линиях А и В интерфейса выполняют резисторы R13, R14 номиналом 3...5 Ом. Резистор R12 номиналом 120 Ом устанавливается на концах линии (концевых станциях сети) для согласования. Резисторы R10, R11 подтягивают свободно висящие линии и улучшают динамические характеристики сети. На разъем интерфейса RS-485 выведена также изолированная линия сигнала (RTS/RST), которая может служить сигналом сброса одновременно всех станций сети RS-485 для их последующей инициализации.

Входная часть устройства потребляет не более 10 мА, выходная — не более 15 мА. Поэтому в стабилизаторах используются микросхемы А1, А2 78L05. Резисторы R15, R16 выполняют роль предохранителей (рассеиваемая мощность не более 0,062 Вт). В особо ответственных изделиях следует устанавливать импор-

тные самовосстанавливающиеся предохранители на ток 50 мА. Входное напряжение для стабилизатора может быть от 7 до 12 В. В устройстве могут быть использованы диодные мосты DB10х с последними цифрами от 1 (100 В) до 7 (1 кВ), выполненные в корпусе DIP6, или любые другие малогабаритные мосты. Устройство выполнено на двухсторонней печатной плате с размерами 90х62 мм².

Описанное устройство может быть использовано в инженерной и радиолюбительской практике для оптической развязки персонального компьютера от различного рода сетей, а также промышленного и экспериментального оборудования. Устройство обеспечивает оптическую развязку персонального компьютера от подключаемого оборудования как минимум на 2,5 кВ, обеспечивает скорость передачи данных не ниже 115200 бод (при установленных оптронах 6N136), имеет высокую надежность и ремонтопригодность (все активные элементы установлены на панельках и в специальных колодках). Потребляемая мощность его не превышает 30 Вт.

Олег Николайчук,
onic@ch.moldpac.md

Литература:

1. О. Николайчук Командно-информационные сети — что это такое? — Схемотехника, 2001, №6, стр. 26-30; №7, стр. 37-41.

www.platan.ru ЭЛЕКТРОННЫЕ КОМПОНЕНТЫ ВАШЕГО УСПЕХА

ПЛАТАН

MITSUBISHI ELECTRIC

Семейство высокопроизводительных 16-ти разрядных микроконтроллеров **М16С62**

- высокоэффективное ядро, программная совместимость внутри семейства, 91 команда, тактовая частота 16МГц
- 128k...256k ROM (наличие Flash версий), 3k...20k RAM
- прямая адресация до 4М внешней Flash и 128k SRAM без подключения дополнительных элементов
- наличие встроенного контроллера ПДП и встроенного аппаратного умножителя
- АЦП 8 или 10 каналов по 10 бит, ЦАП 2 канала по 8 бит
- программирование по RS-232C
- до 87 линий программируемого ввода/вывода
- 6 выходных и 6 входных 16-битных таймеров
- 3 UART и 2 синхронных последовательных канала
- версии с CAN контроллером (M306N0FCTFP)
- наличие бесплатного ПО, включающего C++ компилятор

Наименование	Память ПЗУ кб ОЗУ кб	Порты ва/вы	Унит.В	Тип памяти	Корпус
M30820ECFP	128 10	87	2.7-5.5	ОПТ	100P6S-A
M30620ECFS	128 10	87	2.7-5.5	EPROM	100D0
M30620SFP	0 10	47	2.7-5.5	ROMLESS	100P6S-A
M30621ECGP	128 10	37	2.7-5.5	ОПТ	80P6S-A
M30622ECFS	128 5	87	2.7-5.5	EPROM	100D0
M30622ECGP	128 5	87	2.7-5.5	ОПТ	100P6Q-A
M30622SFP	0 3	47	2.7-5.5	ROMLESS	100P6S-A
M30623ECGP	128 5	87	2.7-5.5	ОПТ	80P6S-A
M30624FGGP	256 20	87	5	FLASH	100P6Q-A
M30624GLGP	256 20	87	3.3	FLASH	100P6S-A
M30624GLGP	256 20	87	3.3	FLASH	100P6Q-A
M306N0FCTFP	256 10	87	5	FLASH	100P6S-A

121351, Москва, ул. Ивана Франко, д. 40, к. 1, стр. 2
тел., факс: (095) 73-75-999

Почта: 121351, Москва, а/я 100
E-mail: mitsubishi@platan.ru

ARGUSSOFT ЭЛЕКТРОННЫЕ КОМПОНЕНТЫ
Департамент Микроэлектроники ОФИЦИАЛЬНЫЙ дистрибьютор фирм:

ANALOG DEVICES **ADSP-218xN**

Новые высокопроизводительные цифровые сигнальные процессоры

- интерфейс прямого доступа к памяти
- 2 быстрых последовательных порта
- небольшое энергопотребление
- удобные инструментальные средства
- приемлемая цена

- программная совместимость в рамках семейства
- возможность выполнения нескольких операций за один цикл

Наименование	Машинный цикл, нс	Напряжение питания, В	ОЗУ программ (24 разряда)	ОЗУ данных (16 разрядов)
ADSP-2184N	12.5	1.8	4 К	4 К
ADSP-2185M	13.3, 15	2.5	18 К	16 К
ADSP-2186N	12.5	1.8	16 К	16 К
ADSP-2186M	13.3, 15	2.5	8 К	8 К
ADSP-2188N	12.5	1.8	8 К	8 К
ADSP-2187N	12.5	1.8	32 К	32 К
ADSP-2188M	13.3, 15	2.5	48 К	56 К
ADSP-2189N	12.5	1.8	48 К	56 К
ADSP-2189M	13.3, 15	2.5	32 К	48 К
ADSP-2189N	12.5	1.8	32 К	48 К

Офис в Москве: Адрес: 129085, Москва, Проект Мира, 95
Тел. (095) 217-2487, 217-2519, 217-2505
Факс: (095) 216-86-42
E-mail: components@argussoft.ru

Офис в Санкт-Петербурге: Адрес: 191023, Санкт-Петербург, наб. кан. Грибоедова, 36
Тел. (812) 814-3808, тел. факс: (812) 310-6234
E-mail: spb@argussoft.ru

Интернет: WWW.ARGUSSOFT.RU

121351, Москва, ул. Ивана Франко, д. 40, к. 1, стр. 2
тел., факс: (095) 73-75-999

Почта: 121351, Москва, а/я 100
E-mail: mitsubishi@platan.ru

(Продолжение. Начало — №4/2002)

Генератор прямоугольных импульсов на основе AVR

Короткий и длинный режим

Как и в случае генерации меандра, разрядности таймера не хватает для перекрытия желаемого диапазона длительностей. Для того, чтобы расширить этот диапазон, в дополнение к 16-разрядному таймеру добавлен программный 8-битный счетчик PCNT. Это позволило получить диапазон длительностей импульсов до 1 с. Поскольку обработка программного счетчика требует дополнительных затрат времени, программа построена так, что выполнение обработчика зависит от формируемого временного интервала. При работе на коротких временах, когда справляется таймер, обработчик выполняет более короткую ветку. Это короткий режим работы. На больших временах, когда требуется дополнительный программный счетчик, обработчик прерывания выполняется гораздо дольше, что вполне допустимо, так как имеется достаточный запас времени. Это длинный режим работы. Короткий и длинный режим определяется индивидуально для лог. 1 и лог. 0, т. к. на скважность нет никаких ограничений.

Работа в коротком режиме

Наиболее критичной является ветка обработчика прерывания, которая выполняется в коротком режиме. Блок-схема обработчика прерывания показана на рис. 11. Для ускорения работы принят ряд мер. Обработчик прерывания размещен в памяти прямо начиная с вектора, таким образом экономится команда `jmp`.

Для того чтобы не вводить программный флаг текущего состояния выхода, который потребуется инвертировать на каждом полупериоде, используется чтение состояния ножки порта (т. е. в качестве флага используется аппаратный триггер схемы формирования выходного сигнала по совпадению). Это накладывает некоторые ограничения на нагрузку, подключаемую к ножке OC1: она не должна иметь слишком высокую емкость и не должна перегружать порт по току. После анализа текущего состояния выхода выполняется та ветка обработчика, которая отвечает за обнаруженный логический уровень. Ветки практически одинаковы, только они используют разные наборы параметров и учитывают различие логических уровней на выходе.

Первым делом в обработчике производится перезагрузка регистра сравнения OCR1A. Затем проверяется специальный флаг Sh (Short, короткий режим), который формирует основная программа при вычислении величины интервала. Единичное значение указывает на то, что интервал короткий, и программный счетчик использовать не нужно. Если это короткий режим, то кроме перезагрузки регистра сравнения ничего больше делать не нужно. В противном случае выполняется более длинная ветка обработчика. В короткой ветке прерывания не используются никакие команды, модифицирующие флаги, поэтому нет необходимости тратить время на сохранение регистра флагов SREG.

Если флаг Sh сброшен, то проверяется флаг короткого режима с внешней синхронизацией ShEx. Эта ветка обработчика будет обсуждаться ниже. Если флаг ShEx тоже сброшен, это означает длинный режим.

Работа в длинном режиме

При вызове обработчика в длинном режиме он начинает выполняться так же, как и в коротком: производит перезагрузку регистра сравнения. Однако далее проверяется состояние программного счетчика PCNT. Если это первый вызов для данного логического уровня на выходе, то содержимое PCNT будет равно нулю. В этом случае производится загрузка счетчика начальной величиной. Если вызов обработчика не первый, то счетчик уже загружен, и производится его декремент. Затем следует проверка содержимого счетчика на

нуль. Если нуль не достигнут, значит, формирование длинного интервала не закончено. В этом случае выход OC1 настраивается таким образом, чтобы события совпадения не изменяли его состояния до окончания формирования интервала. В данном примере выход нулевой, поэтому он настраивается на сброс при совпадении. Регистр сравнения снова перезагружается, на этот раз фиксированной величиной MaxW. С момента загрузки в начале обработчика совпадение произойти не успевает, поэтому определяющей будет как раз вторая загрузка. Величина MaxW определяет длительность интервала, соответствующую единице программного счетчика.

Если программный счетчик обнулится, значит пора заканчивать интервал. При этом выход OC1 настраивается на переключение по совпадению, и следующее совпадение изменит его состояние. В этом случае вторая перезагрузка регистра сравнения не производится, поэтому остается в силе величина, загруженная в начале обработчика. Таким образом, в длинном режиме интервал состоит из нескольких фрагментов дли-

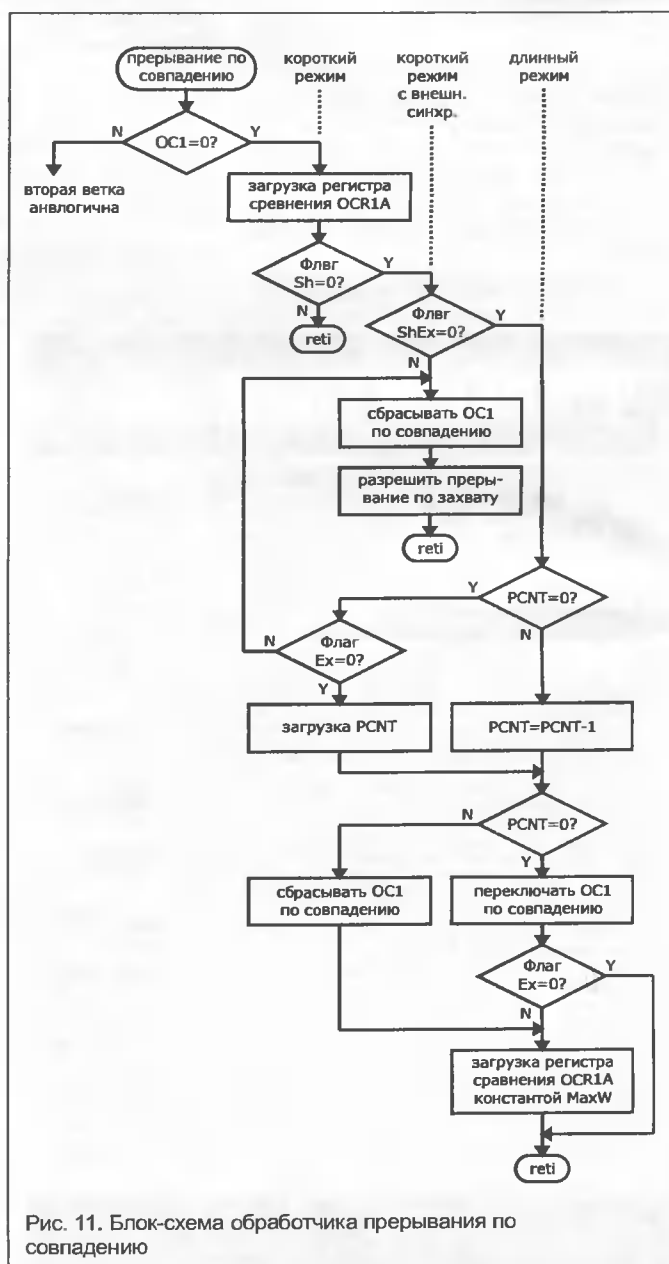


Рис. 11. Блок-схема обработчика прерывания по совпадению

тельностью MaxW, количество которых определяется загрузочной величиной программного счетчика PCNT. В конце интервала добавляется фрагмент, длительность которого представляет собой остаток от деления требуемого интервала на MaxW. Загрузочную величину программного счетчика и загрузочную величину регистра сравнения (длительность последнего фрагмента) заранее вычисляет основная программа.

Дополнительно в этой ветке обработчика проверяется флаг режима внешней синхронизации Ex.

Текст одной ветки обработчика прерывания (обработка логического нуля) приведен ниже:

;**** Обработчик прерывания по совпадению:

```
CMPARE: sbic PINB,OUT ;проверка состояния
выхода
        rjmp Phase1 ;переход на вторую ветку
; (здесь текст не приводится)
```

;Короткий режим:

```
Phase0: out OCR1AH,TN0 ;перезагрузка регистра
сравнения
        out OCR1AL,TN0
        sbrc FLAG0,Sh ;проверка флага Sh
        reti
```

;Короткий режим с внешней синхронизацией:

```
PhEx0: sbrc FLAG0,ShEx ;проверка флага ShEx
        rjmp PhLg0 ;переход на длинный режим
Exdo0: OutL ;сбрасывать OC1 при совпа-
дении
        IntCAP ;разрешение прерывания по захвату
        reti
```

;Длинный режим:

```
PhLg0: in XSREG,SREG ;сохранение SREG
        tst PCNT ;проверка программного счетчика
        brne Pcnt0 ;переход, если не ноль
        out SREG,XSREG ;восстановление SREG
        sbrc FLAG0,Ex ;проверка флага Ex
        rjmp Exdo0 ;переход на предыдущую ветку

        lds PCNT,TN0 ;загрузка программного
счетчика
        rjmp Pcnt0
Pcnt0: dec PCNT ;декремент программного
счетчика
        Pcnt0: tst PCNT ;проверка программного
счетчика
        breq Pcnt0 ;переход, если ноль
        OutL ;сбрасывать OC1 при совпадении
        rjmp Pcre0

Pcnt0: OutToggle ;переключать OC1 при со-
впадении
        sbrc FLAG0,Ex ;проверка флага Ex
        rjmp Pclg0

Pcre0: ldi tempR,high(MaxW) ;загрузка регистра
сравнения
        out OCR1AH,tempR
        ldi tempR,low(MaxW)
        out OCR1AL,tempR
Pclg0: out SREG,XSREG ;восстановление SREG
        reti
```

Формирование параметров

Основная программа вычисляет необходимые значения временных интервалов лог. 0 и лог. 1 для требуемой формы выходного сигнала. Таймер работает на частоте 10 МГц, что соответствует дискретности установки временных параметров 0,1 мкс. Поэтому именно в десятых долях микросекунды и представляется длительность нуля и единицы. Это трехбайтовые числа, два младших байта подлежат загрузке в регистр сравнения таймера, а старший байт — в про-

граммный счетчик. Однако при этом может возникнуть одна проблема. Дело в том, что период времени, через который возникает событие совпадения, определяется величиной, загруженной в регистр сравнения. Если эта величина будет очень малой, совпадение может возникнуть раньше, чем завершится обработка прерывания предыдущего совпадения. Это нарушит нормальную работу. А ведь два младших байта трехбайтового значения, меняющегося непрерывно, могут принимать любые значения. Для предотвращения нарушения работы используется специальный алгоритм формирования трехбайтового параметра, передаваемого в прерывание. Единица программного счетчика соответствует не 65536 тикам таймера, а 50000 (MaxW). Требуемое значение длительности делится на MaxW. Если получается нулевой результат, то данный интервал будет формироваться в коротком режиме без привлечения программного счетчика. Если результат ненулевой, то будет использоваться длинный режим, и это есть загрузочная величина TN программного счетчика. Дополнительно проверяется значение остатка от деления (величины, предназначенной для загрузки в регистр сравнения). Если это значение меньше 0xFFFF-MaxW, то к нему прибавляется MaxW, а TN уменьшается на единицу. При этом может оказаться, что TN станет равным нулю, тогда формирование будет производиться в коротком режиме. Таким образом, в длинном режиме загрузочная величина регистра сравнения не может быть меньше 15535. Это гарантирует, что промежуток времени между двумя событиями совпадения не будет слишком малым, и даже длинный обработчик прерывания выполниться успеет. В коротком режиме эта величина программно ограничена минимальной допустимой длительностью.

Леонид Ридико,
wubblick@yahoo.com

Продолжение следует



Программаторы "Стерх"

☑ Универсальный программатор ST-011

- программирование более 500 типов **BPROM, EPROM, FLASH, Serial EPROM, MPU/MCU, PAL, PLD** производства Россия, **Altera, AMD, Intel, Microchip, National, Phillips, Siemens, SST, SGS-Thomson, TI, Winbond, Zilog** и др.
- одна универсальная **DIP40** или **DIP42 ZIF**-панель
- определение правильности установки микросхем
- идентификация производителя и типа микросхемы
- быстросрабатывающая защита от перегрузок
- встроенный источник питания
- RS-232 со скоростью обмена до 115 кбод
- программное обеспечение с русскоязычным интерфейсом и поддержкой «мыши»
- программное обновление версий через Internet
- дополнительно: адаптеры для микросхем в корпусах **PLCC, SOP** и др.

☑ УФ-излучатель UV-01

- устройство стирания микросхем **EPROM**: таймер до 99 мин, звуковая сигнализация, до 16 микросхем одновременно.

Более подробную информацию об изделиях и последние версии ПО можно найти на нашем **WWW-сервере**:
<http://www.sterh.com>

Изготовитель: НПО «БОНД» г. Бердск
 ☎ (38341) 5-15-62, E-mail: pprog@bond.nsk.su

Москва: «Точка Опоры»
 Санкт-Петербург: «ЭФО»
 Екатеринбург: «Институт радиотехники»

☎ (095) 956-39-42/43
 ☎ (812) 247-89-00
 ☎ (3432) 74-58-61

(Окончание. Начало — №3/2002)

Микросхемы контроллеров ЖКИ фирмы Holtek и их применение

Формат команды

Три первых передаваемых бита определяют тип операции (запись или чтение), а также тип признака (команда или данные). В документации Holtek этот код операции называется ID. Всего существует четыре типа операций:

- запись команды — код 100;
- запись данных — код 101;
- чтение данных — код 110;
- чтение-модификация-запись данных — код 101.

Для последовательных кодов команд чтения или записи после кода признака команды следуют шесть разрядов адреса буферной памяти от A5 до A0, затем разряды записываемых или считываемых данных. Раз уж контроллер китайский, так что-нибудь должно быть необычным!

Обратите внимание, что при передаче адресов первым идет старший разряд, а при передаче/приеме данных — все наоборот, данные следуют младшим разрядом вперед. Но к этому можно привыкнуть, лишь бы первый раз не промахнуться!

Команда чтения Read, как и команда Read-Modify-Write, в большинстве случаев просто бесполезны! В памяти хоста почти всегда есть образ ОЗУ дисплея, и нет необходимости

хитрить. Чтение-модификация и запись просто увеличивает цикл работы с контроллером. Гораздо проще и быстрее эту процедуру выполнять в самом контроллере! Поэтому можно не использовать вход RD. Он имеет встроенный резистор подтяжки к питанию и при отсутствии связей будет неактивным. Так что реальных полезных команд из всего списка наберется не так уж много. Причем часть из них используется только при инициализации ЖК-дисплея. А во всех остальных случаях пользователь производит модификацию дисплейных данных у себя в контроллере и загрузку данных в дисплейное ОЗУ драйвера. Таким образом, при реализации загрузки данных в драйвер разумно использовать только команду записи. Ниже на реальном примере будут показаны фрагменты программы для инициализации драйвера, а также для его загрузки новыми данными.

Наличие всего двух частот ограничивает диапазон приложений встроенного контроллера зуммера. Система команд для управления счетчиком или сторожевым таймером в драйвере HT1621 весьма затейлива. Я бы не советовал им пользоваться совсем. Вся эта конструкция «счетчик/сторожевой таймер» состоит из последовательно включенных

Таблица 4. Коды основных операций для контроллера HT1621

Название операции	ID-код операции	Код команды	Функция
READ	110	A5A4A3A2A1A0D0D1D2D3	Чтение данных из дисплейного ОЗУ драйвера
WRITE	101	A5A4A3A2A1A0D0D1D2D3	Запись данных в дисплейное ОЗУ драйвера
Read-Modify-Write	101	A5A4A3A2A1A0D0D1D2D3	Чтение-модификация и запись данных в дисплейное ОЗУ драйвера
SYS DIS	100	0000-0000-X	Выключить внутренний генератор и отключить питание резистивного делителя bias
SYS EN	100	0000-0001-X	Разрешить работу внутреннего генератора и подать питание на делитель напряжения (bias)
LCD OFF	100	0000-0010-X	Отключить питание резистивного делителя
LCD ON	100	0000-0011-X	Включить питание резистивного делителя
TIMER DIS	100	0000-0100-X	Отключение выхода базового счетчика таймера
WDT DIS	100	0000-0101-X	Отключить ступень сторожевого таймера
TIMER EN	100	0000-0110-X	Подключить выход базового счетчика на 8 к выходной схеме формирователя прерывания
WDT EN	100	0000-0111-X	Подключить выход счетчика-делителя на 4 к выходной схеме формирователя прерывания
TONE OFF	100	0000-1000-X	Отключить выходы тонального генератора
TONE ON	100	0000-1001-X	Активизировать выходы тонального генератора
CLR TIMER	100	0000-0110-X	Сброс базового счетчика таймера
CLR WDT	100	0000-0111-X	Сброс делителя на 4 (старшей секции счетчика сторожевого таймера)
XTAL 32K	100	0000-0000-X	Включить режим тактирования от внутреннего кварцевого генератора 32768 Гц
RC 256K	100	0000-0000-X	Включить режим тактирования от внутреннего RC генератора 256 кГц
EXT 256K	100	0000-0000-X	Включить режим тактирования от внешнего генератора 256 кГц
BIAS 1/2	100	0010-abX0-X	Включить режим bias 1/2 Состояние восьмого бита определяет установку опции bias для любого режима мультиплексирования Для режима 1/2 это значение = 0 Ab — код определяет мультиплекс 00 — 1/2, используются выходы COM0 и COM1 01 — мультиплекс 1/3, используются выходы COM0..COM2 10 — мультиплекс 1/4, используются выходы COM0..COM3
BIAS 1/3	100	0010-abX1-X	Включить режим bias 1/3 Состояние восьмого бита определяет установку режима bias Для режима 1/2 это значение равно 0
TONE 4K	100	010X-XXXX-X	Установить частоту внутреннего тонального генератора 4 кГц
TONE 2K	100	011X-XXXX-X	Установить частоту внутреннего тонального генератора 2 кГц
IRQ DIS	100	100X-0XXX-X	Деактивировать выход прерывания IRQ

Примечания к табл. 4: X — состояние разряда не имеет значения;
A5...A0 — последовательные разряды адреса ячейки столбца;
D0...D3 — последовательные данные кода столбца.

счетчиков с программируемым коэффициентом деления на 1, 2, 4, 8, 16, 32, 64, 128; основного 8-разрядного счетчика и постскэйлера (делителя на 4). К выходной схеме прерывания может быть подключен либо выход старшего разряда основного счетчика (тогда сооружение будет называться таймер), либо выход постскэйлера — в этом случае такая конструкция уже называется сторожевым таймером! Хотя по сути ничего не меняется, просто у счетчика становится 10 разрядов вместо 8! При сбросе сторожевого таймера производится сброс только счетчика последней ступени. Команды TIMER EN или WDT EN работают в режиме toggle, т. е. когда проходит одна команда, например, включить таймер, при этом автоматически выключается выход секции делителя сторожевого таймера, и наоборот. Впрочем, если желание использовать это сооружение остается, то для управления таймером существуют дополнительные команды. В табл. 5 приведены коды команд для управления таймером.

Практическая реализация схемы управления ЖКИ на основе драйвера HT1621

Для тех приложений, где важно иметь меньшее потребление тока, нужно использовать внутренний кварцевый генератор на 32,768 кГц, а для питания логики драйвера — использовать напряжение 3 В вместо 5 В. На рис. 6 показан пример реализации интерфейса драйвера HT1621 с контроллером PIC16C717. Используемый в примере ЖКИ имеет 12 сегментов, и можно было бы применить микросхему HT1621D с меньшим числом выводов, однако укороченный вариант микросхемы не поддерживает мультиплекс 1:4, который используется в выбранном ЖКИ. В данной реализации режим чтения не применяется. Для питания драйвера и контроллера PIC16C717 используется один источник питания 3 В (как рекомендовано в технической документации на ЖКИ). Тональный генератор и таймер не используются.

Разработка программы управления модулем на основе заказного или готового низкомультимплексного ЖКИ упрощается, если будет доступна документация на ЖКИ. Из документации можно получить массу полезной информации: точные габариты, шаг выводов, назначение и расположение выводов, число мультиплексирования, размер рабочего поля, напряжение питания, рекомендуемый режим bias, рабочий угол обзора, диапазон рабочих температур и т. д.

В состав документации входит и топология разводки сегментных (столбцовых) и строчных (общих) электродов. Конструкторская документация существует для всех промышленно выпускаемых ЖКИ. Как правило, на нашем рынке индикаторы поставляются без какой-либо документации, поэтому не многие разработчики имеют представление, как она должна выглядеть. Для исправления этого пробела ниже, в качестве примера, приводится конструкторская документация на один из заказных ЖКИ фирмы Wintek.

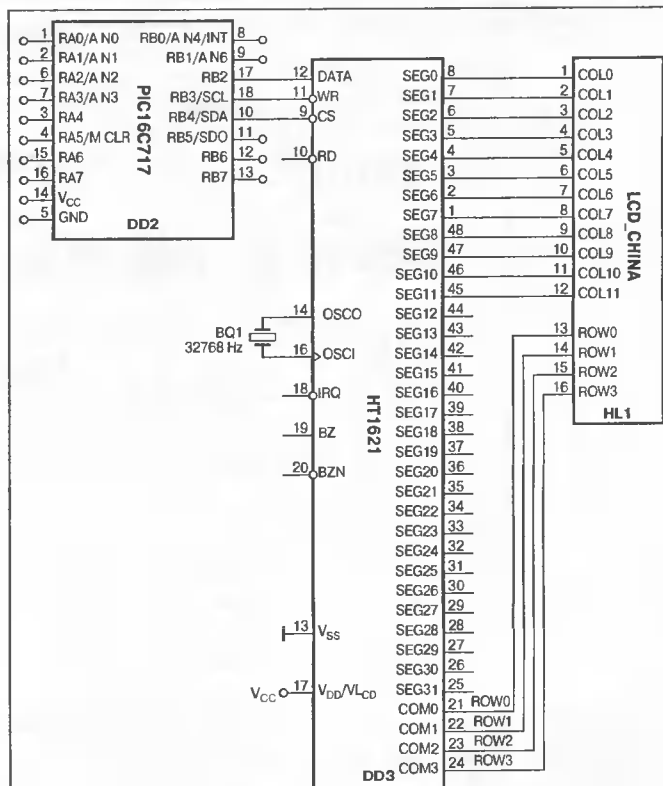


Рис. 6. Пример схемы управления ЖКИ

На одном листе приводится конструкция ЖКИ со всеми размерами, а также приведены основные характеристики индикатора: тип ЖК-материала, рабочий угол обзора, напряжение питания, рекомендуемая схема bias для управления и тип оптической схемы — на просвет, на отражение, на полупропускание. На двух других листах приведена топология сегментных (столбцовых) и строчных (общих) электродов.

- | | |
|-----------------------|---|
| 1. LCD TYPE : | TN |
| 2. VIEWING DIRECTION: | 6 O'CLOCK |
| 3. DRIVE METHOD: | 1/4 DUTY 1/3 BIAS |
| 4. OPERATING VOLTAGE: | 3,0 V |
| 5. FRONT POLARIZER: | TRANSMISSIVE/PASTED (38.0 mm × 23.0 mm) |
| | TRANSMISSIVE/PASTED (38.0 mm × 23.0 mm) |
| REAR POLARIZER: | |
| 6. DISPLAY MODE: | POSITIVE |
| 7. OPERATING TEMP.: | 0 °C ~ 50 °C |
| 8. STORAGE TEMP.: | -10 °C ~ 80 °C |

В документации на ЖКИ указывается:

- тип ЖК-материала — TN твист-нематик;
- рабочий угол обзора дисплея — 6 часов (для горизонтального положения);
- метод управления — мультиплексирование: 1/4 и bias 1/3;
- напряжение питания формирователей драйвера — 3 В;
- характеристики поляризаторов: Front — фронтальный (находится на верхней стеклянной подложке со стороны наблюдателя) с клеевым слоем (pasted), работающий на пропускание, и имеющий размеры 38x23 мм;
- Rear — задняя поляризационная пленка с клеевым слоем размерами 38x23 мм, работает на отражение;

Таблица 5. Коды команд для управления внутренним таймером

Название операции	ID-с Код операции	Код команды	Функция
IRQ EN	100	100X-1XXX-X	Разрешение выхода прерывания IRQ
F1	100	101X-X000-X	В режиме таймера частота сигнала IRQ — 1 Гц, а в режиме сторожевого таймера период — 4 с
F2	100	101X-X001-X	В режиме таймера частота сигнала IRQ — 2 Гц, а в режиме сторожевого таймера период — 2 с
F4	100	101X-X010-X	В режиме таймера частота сигнала IRQ — 4 Гц, а в режиме сторожевого таймера период — 1 с
F8	100	101X-X011-X	В режиме таймера частота сигнала IRQ — 8 Гц, а в режиме сторожевого таймера период — 0,5 с
F16	100	101X-X100-X	В режиме таймера частота сигнала IRQ — 16 Гц, а в режиме сторожевого таймера период — 0,25 с
F32	100	101X-X101-X	В режиме таймера частота сигнала IRQ — 32 Гц, а в режиме сторожевого таймера частота — 8 Гц
F64	100	101X-X110-X	В режиме таймера частота сигнала IRQ — 64 Гц, а в режиме сторожевого таймера частота — 16 Гц
F128	100	101X-X111-X	В режиме таймера частота сигнала IRQ — 128 Гц, а в режиме сторожевого таймера частота — 32 Гц

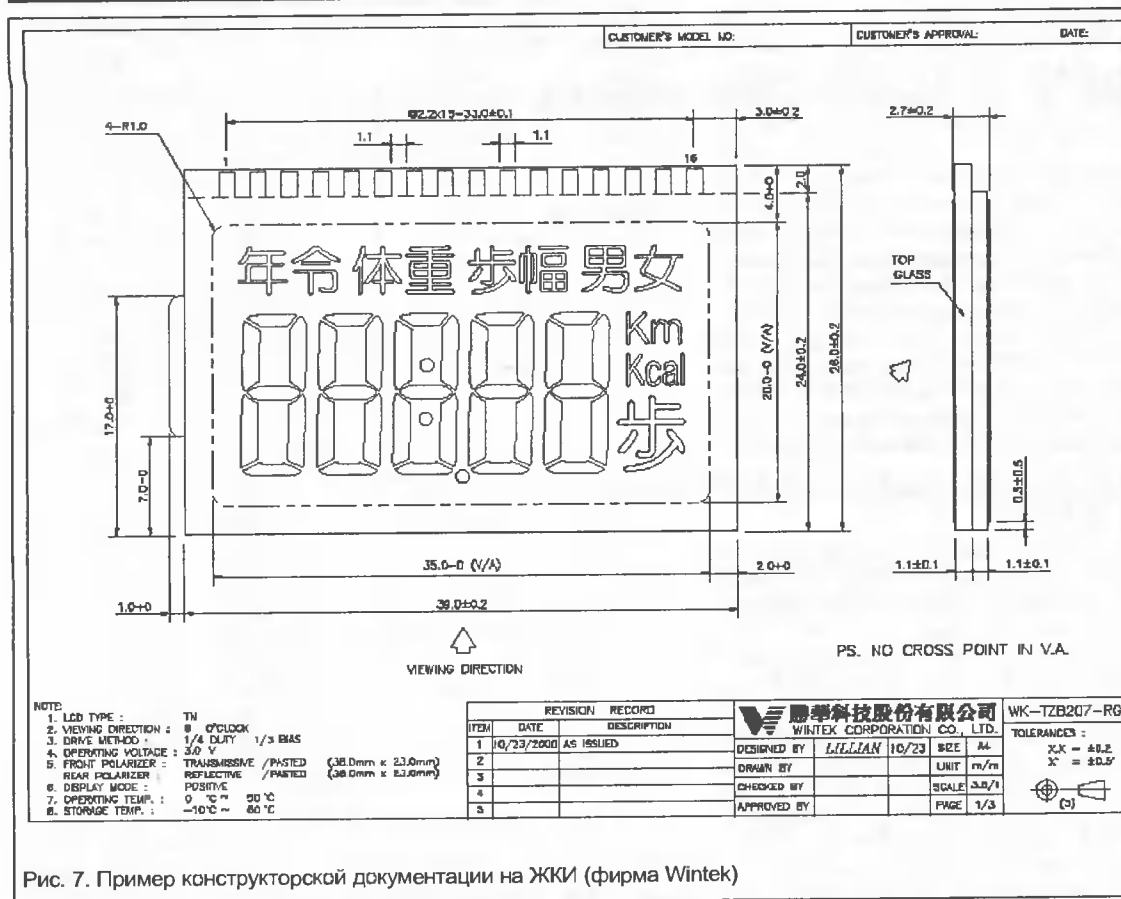


Рис. 7. Пример конструкторской документации на ЖКИ (фирма Wintek)

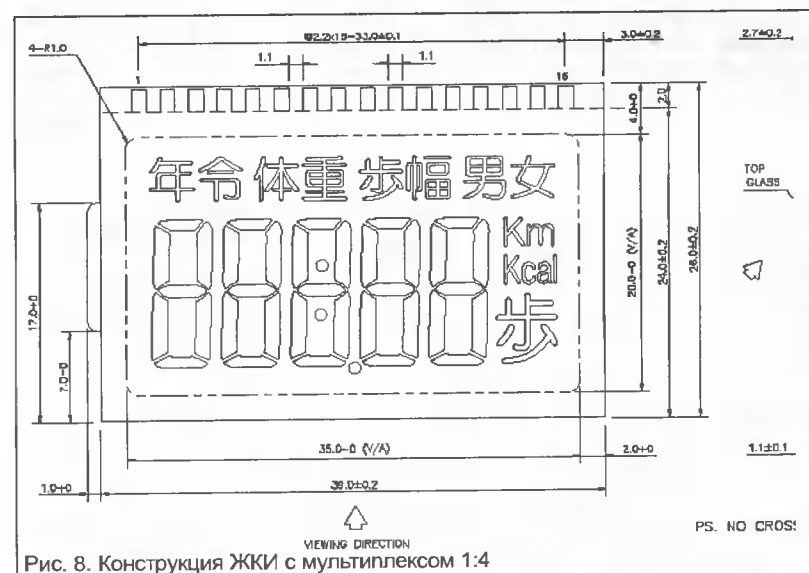


Рис. 8. Конструкция ЖКИ с мультиплексом 1:4

- режим дисплея — позитивный, т. е. черные (выбранные) элементы изображения на светлом фоне;
- диапазон рабочих температур: 0...50°C;
- диапазон допустимых температур при хранении: -10 ... 80°C.

Режим дисплея (позитивный/негативный) определяется соотношением вектора ориентации ЖК-материала и вектора ориентации поляризационных пленок: верхней и нижней. Векторы поляризационных пленок или совпадают или сдвинуты относительно друг друга на 90°. Вектор поляризации самого ЖК-материала для обоих режимов (позитивного и негативного) сдвинут на 45° относительно векторов поляризационных пленок. От ориентации ЖК-материала и угла наклона молекул ЖК-материала зависит и рабочий угол обзора дисплея. Этот параметр определяется технологическим процессом и в дальнейшем не может быть изменен. Поляризационная пленка на отражение отличается от плен-

ки на пропускание наличием тонкого дополнительного светоотражающего слоя (алюминиевая фольга).

Существуют конструктивные исполнения ЖК-дисплеев с накладной (не приклеенной к верхней стеклянной подложке) верхней поляризационной пленкой. В этом случае пользователь может изменить только режим индикации дисплея, например, изменить позитивный режим на негативный. Для негативного режима получим светлые (выбранные) элементы изображения на черном фоне. Обычно такие схемы используются в дисплеях, работающих с подсветкой (transmissive mode). Для этого достаточно перевернуть верхнюю поляризаци-

онную пленку так, чтобы получить темный фон.

В качестве подсветки может быть использован и внешний свет. За счет применения специальных зеркал или световодов можно «загнать» световой поток под ЖКИ. Но вот изменить рабочий угол обзора ЖКИ, манипулируя поляризаторами, уже невозможно. Если поляризационные пленки уже наклеены — отдирайте и пытайтесь изменить режим ЖКИ в домашних условиях не рекомендуется. Для этого потребуются найти новые поляризационные пленки взамен старых. Но даже если они есть, то нужно правильно разметить и вырезать их так, чтобы получить нужный угол ориентации. Если пленка с клеевым слоем, то перед ее наклейкой нужно тщательно очистить поверхность стекла. Наличие пылинок или остатков клея на стекле приведет к заметному воздушному пузырю при наклейке новой пленки.

На основании топологии разводки строчных и столбцовых электродов составляется карта адресации элементов изображения. Если документация на индикатор отсутствует, разработчик вынужден восстановить карту адресации сегментов и самостоятельно определить назначение выводов индикатора с помощью любого генератора сигналов. Для тестирования ЖКИ годится периодический сигнал любой формы с частотой от 50 до 500 Гц и амплитудой 4...5 В. В некоторых ЖКИ при наблюдении под определенным углом можно рассмотреть рисунок электродов. Наличие рисунка топологии электродов поможет определить хотя бы назначение выводов. Первым делом нужно определить выводы общих электродов. Далее для составления карты элементов изображения одним щупом генератора касаемся контактной площадки определенных заранее общих электродов, а другим производим сканирование сегментных площадок. Полученная информация (координаты строка-столбец для каждого элемента изображения) заносится в таблицу. Затем перемещаем первый щуп на контактную площад-

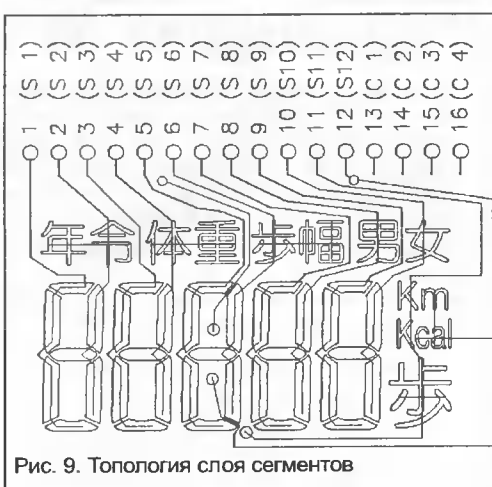


Рис. 9. Топология слоя сегментов

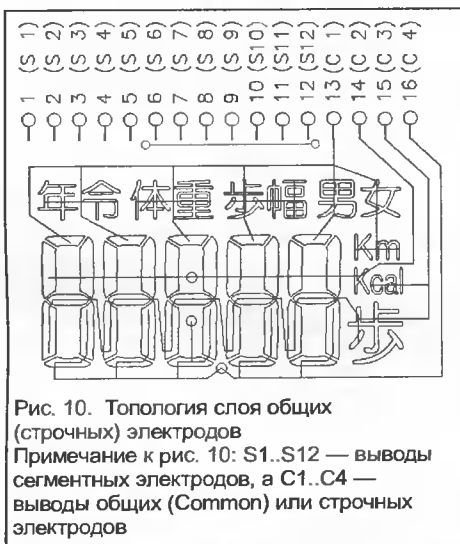


Рис. 10. Топология слоя общих (строчных) электродов

Примечание к рис. 10: S1..S12 — выводы сегментных электродов, а C1..C4 — выводы общих (Common) или строчных электродов

ражения лучше иметь документацию.

В табл. 6 приведен пример раскладки элементов изображения для индикатора Wintek в соответствии со схемой соединения, приведенной на рис. 6. Если в индикаторе используются семисегментные символы, желательно, чтобы они имели регулярную структуру — это заметно упростит программирование.

С помощью семисегментных символов можно индцировать не только цифры от 0 до 9, но и многие буквы алфавита! Ниже в качестве примера приведена таблица знакогенератора для синтеза 32 знаков для топологии ЖКИ WINTEK. На пятизнаковом семисегментном поле ЖКИ можно воспроизводить такие сообщения, как HELLO, SLEEP, Data, FILE, PASS и т. д.

Таблица 6. Карта раскладки элементов изображения в ОЗУ драйвера HT1621

	ROW0	ROW1	ROW2	ROW3
SEG0	A1	F1	E1	D1
SEG1	#1	B1	G1	C1
SEG2	A2	F2	E2	D2
SEG3	#2	B2	G2	C2
SEG4	A3	F3	E3	D3
SEG5	#3	B3	G3	C3
SEG6	A4	F4	E4	D4
SEG7	#4	B4	G4	C4
SEG8	A5	F5	E5	D5
SEG9	#5	B5	G5	C5
SEG10	K m	K (cal)	#6	P3
SEG11	m	P1	cal	P2

Примечание к табл. 6: Ai, Bi, Ci, Di, Gi, Ei, Fi — сегменты пяти цифровых знамен, имеющих, как можно заметить, регулярную топологическую структуру. #1..#6 обозначены позиции элементов изображения китайских иероглифов. P1, P2, P3 — сегменты точек.

ку другого общего электрода и повторяем сканирование. Процедура проста и почти очевидна. К сожалению, в большинстве современных зарубежных ЖКИ рисунок электродов обычно не удается рассмотреть. В стандартах на зарубежные ЖКИ существует требование — при отсутствии питания, рисунок электродов не должен быть замечен при любом угле наблюдения. Тем не менее, задача определения столбцовых и строчных выводов ЖКИ с помощью генератора решается однозначно. Но, повторяю, для составления карты адресации элементов изобра-

Теперь после всех подготовительных процедур можно приступить к написанию программы управления ЖКИ. Программа содержит несколько базовых фрагментов:

- look table — таблицу знакогенератора;
- программу начальной инициализации драйвера;
- программу загрузки файла данных в ОЗУ драйвера;
- программу преобразования двоично-десятичного кода цифры в семисегментный код.

Ниже приведены фрагменты программ, которые могут быть полезны для понимания работы с драйвером HT1621.

;знакогенератор 7сегментный

```

znak clrf    pclath
addwf    pc,1
retlw    h'af' ;0
retlw    h'a0' ;1
retlw    h'6d' ;2
retlw    h'e9' ;3
retlw    h'e2' ;4
retlw    h'cb' ;5
retlw    h'cf' ;6
retlw    h'al' ;7
retlw    h'ef' ;8
retlw    h'eb' ;9
retlw    h'e7' ;A
retlw    h'ce' ;b
retlw    h'0f' ;C
retlw    h'ec' ;d
retlw    h'4f' ;E
retlw    h'47' ;F
retlw    h'e6' ;H
retlw    h'e'  ;L
retlw    h'40' ;—
retlw    h'ce' ;o малое
retlw    h'67' ;P
retlw    h'44' ;r
retlw    h'4e' ;t
retlw    h'be' ;U
retlw    h'ea' ;Y
retlw    0     ;пусто
retlw    h'c'  ;(- скобка левая
retlw    h'88' ;) скобка правая
retlw    h'c4' ;n

```

Инициализация драйвера HT1621 — initlcd

```

=====
;xtal32k 100-0001-0100-0 выбираем опцию кварцевого гене-
ратора
;SYSEN 100 0000-0001-0 включаем системный генератор-
;bias 1/3 100 0010-1001-0
;lcd ON 100 0000-0011-0
initlcd
    movlw h'21' ;temp3
    movwf fsr
    call command ;ID 100
;передача команд — старшим битом вперед!
;0001-0100 — выбор кварцевого генератора
    movlw h'28';
    movwf f0
    call bytld
    call takt
    bsf portb,cs
    nop
;SYSEN 0000-0001-0 включаем системный генератор
    call command
    movlw h'80'
    movwf f0
    call bytld
    call takt
    bsf portb,cs
    nop
;bias 1/3 0010-1001-0
    call command
    movlw h'94'
    movwf f0
    call bytld
    call takt
    bsf portb,cs
    nop
;lcd ON 0000-0011-0
    call command
    movlw h'c0'
    movwf f0
    call bytld
    bsf portb,dat
    call takt

```

Таблица 7. Знакогенератор для ЖКИ Wintek

ЗНАК	D7	D6	D5	D4	D3	D2	D1	D0	Номер бита	
	С	Г	В	#	Д	Е	Е	А	Код	Номер в таблице
0 (O)	1	0	1	0	1	1	1	1	Afh	0
1 (I)	1	0	1	0	0	0	0	0	A0h	1
2	0	1	1	0	1	1	0	1	6dh	2
3	0	1	0	0	1	1	1	1	E9h	3
4	1	1	1	0	0	0	1	0	E2h	4
5 (S)	1	1	0	0	1	0	1	1	Cbh	5
6	1	1	0	0	1	1	1	1	Cfh	6
7	1	0	1	0	0	0	0	1	A1h	7
8	1	1	1	0	1	1	1	1	Efh	8
9	1	1	1	0	1	0	1	1	Ebh	9
A	1	1	1	0	0	1	1	1	E7h	Ah
b	1	1	0	0	1	1	1	0	Ceh	Bh
C	0	0	0	0	1	1	1	1	Fh	Ch
d	1	1	1	0	1	1	0	0	Ech	Dh
E	0	1	0	0	1	1	1	1	4fh	Eh
F	0	1	0	0	0	1	1	1	47h	Fh
G	1	0	0	0	1	1	1	1	8fh	10h
H	1	1	1	0	0	1	1	0	E6h	11h
J	1	0	1	0	1	1	0	0	ach	12h
L	0	0	0	0	1	1	1	0	Eh	13h
n	1	1	0	0	1	1	0	0	Cch	14h
o	1	1	0	0	1	1	0	0	Cch	15h
P	0	1	1	0	0	1	1	1	67h	16h
r	0	1	0	0	0	1	0	0	44h	17h
t	0	1	0	0	1	1	1	0	4eh	18h
Y	1	1	1	0	1	0	1	0	Ebh	19h
U	1	0	1	0	1	1	1	0	Beh	1ah
—	0	1	0	0	0	0	0	0	40h	1bh
<	0	0	0	0	1	1	0	0	Ch	1ch
>	1	0	0	0	1	0	0	0	88h	1dh
пусто	0	0	0	0	0	0	0	0	0h	1eh

```

bsf    portb,cs
nop
return
;=====
command
bsf    portb,dat
bcf    portb,cs
;команда
;это идентификатор — шапка для команд управления
call   takt    ;1
bcf    portb,dat
call   takt    ;0
goto   takt    ;0
;=====
nulik
;нулевой адрес
bcf    portb,dat
movlw  6
movwf  temp
tik    call   takt
decfsz temp,1
goto   tik
return
;
;загрузка байта
bytload movlw 8
movwf  temp1
movf   f0,0
movwf  temp3
loop   bcf    portb,dat
rrf    temp3,1
btfsz  STATUS,C
bsf    portb,dat
call   takt
decfsz temp1,1
goto   loop
return
;
takt   bcf    portb,wr
bsf    portb,wr
return
;=====

```

```

;=====
disp0  movf  ind1,0
call   znak
movwf  temp
bcf    temp,4
btfsz  lcd0,4
bsf    temp,4
movf   temp,0
movwf  lcd0
;=====
movf   ind2,0
call   znak
movwf  temp
bcf    temp,4
btfsz  lcd1,4
bsf    temp,4
movf   temp,0
movwf  lcd1
;=====
bsf    portb,dat
bcf    portb,cs
call   takt    ;1
bcf    portb,dat
call   takt    ;0
bsf    portb,dat
call   takt    ;1
call   nulik
;
movlw  h'57'
movwf  fsr
movlw  6
movwf  temp    ;счетчик байтов
lod    call   bytload
incf   fsr,1
decfsz temp,1
goto   lod
bsf    portb,dat
bsf    portb,cs
return

```

Подпрограмма загрузки дисплейного файла

Загрузка всегда производится с нулевого адреса. Ind0..ind5 — регистры пяти символьных знакомест. Сначала через таблицу знакогенератора производится преобразование кодов данных в семисегментные коды, которые помещаются в регистры lcd0..lcd5. Затем производится загрузка дисплейного файла lcd0..lcd5. В каждом регистре содержится коды для двух ячеек ОЗУ драйвера HT1621.

```

;=====
displ  movf  ind3,0
call   znak
movwf  temp
bcf    temp,4
btfsz  lcd2,4
bsf    temp,4
movf   temp,0
movwf  lcd2
;=====
movf   ind4,0
call   znak
movwf  temp
bcf    temp,4
btfsz  lcd3,4
bsf    temp,4
movf   temp,0
movwf  lcd3
;
movf   ind5,0
call   znak
movwf  temp
bcf    temp,4
btfsz  lcd4,4
bsf    temp,4
movf   temp,0
movwf  lcd4

```

Александр Самарин,
samara@zelax.ru

Литература:

1. HT1621 RAM Mapping 32 x 4 LCD Controller for I/O uC. Holtek datasheet.

FSK: сигналы и их демодуляция

FSK модуляция (частотная манипуляция, Frequency Shift Keying) является наиболее распространенным видом цифровой модуляции радиочастотных сигналов. Она нашла широкое применение в связи и телефонии. Эта статья представляет собой руководство по FSK модуляции во всем ее многообразии. Будут рассмотрены как схемы модуляции, так и схемы демодуляции.

Двоичная FSK

Двоичная FSK (нередко называемая просто FSK) представляет собой способ модуляции, обычно используемой для передачи информации между цифровыми устройствами, такими как телетайпы или компьютеры. Данные передаются путем сдвига частоты непрерывно передаваемой несущей попеременно на одну из двух дискретных частот. Одна из этих частот определена как частота «mark», а другая — как частота «space». Mark и space соответственно двоичной единице и нулю соответственно. Принято, что mark соответствует большей частоте несущей. Рис. 1 показывает зависимость между двоичными данными и передаваемым сигналом.

Основные параметры, описывающие FSK сигнал, показаны на рис. 2. Минимальная длительность условия mark или space называется длиной элемента. Типичное значение длины элемента лежит между 5 и 22 мс, однако, может использоваться длина элемента меньшая, чем 1 мкс, или большая, чем 1 с. Используемая полоса частот должна уместиться в те-

лефонном канале, а условия распространения сигнала в ВЧ каналах обычно требует длину элемента, большую 0,5 мс. Альтернативный способ определения длины элемента использует понятие скорости передачи. Скорость передачи в бодах равна обратной величине от длины элемента в секундах. Например, если длина элемента составляет 20 мс (0,02 с), это эквивалентно скорости 50 бод.

Частотные измерения FSK сигнала обычно основываются на понятиях сдвига и центральной частоты. Сдвиг представляет собой разницу между частотами mark и space. Сдвиг обычно лежит в интервале от 50 до 1000 Гц. Центральная частота представляет собой среднюю частоту между частотами mark и space. Иногда используется термин «девиация», применяемый для частотной модуляции. Девиация равна абсолютному значению разницы между центральной частотой и частотой mark или space. Девиация также численно равна половине сдвига.

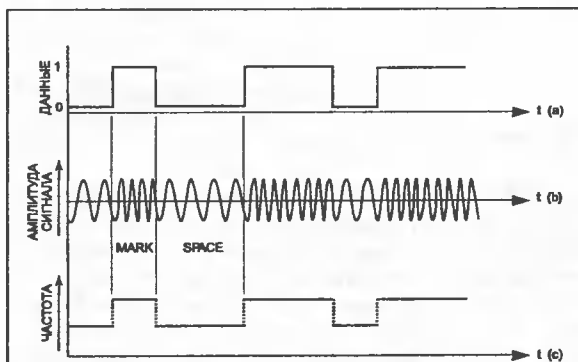


Рис. 1. FSK модуляция. Двоичные данные (а), несущая, модулированная частотно для получения FSK сигнала (б), которая имеет определенную частотную зависимость (с) м



Рис. 2. Параметры FSK сигнала

распространенным случаем когерентной FSK. Когерентная FSK имеет лучшие характеристики с точки зрения вероятности ошибки, однако, некогерентная FSK является более простой в реализации и используется в большинстве случаев. Некогерентная FSK не имеет определенной фазовой зависимости между последовательными посылками — при переходе от mark к space и обратно фаза меняется случайным образом.

Для передачи данных с помощью FSK существует много разных способов кодирования. Они могут быть классифицированы в две основные группы: синхронные и асинхронные. Для синхронных посылок переходы mark-space и space-mark синхронизированы с тактовой частотой. Асинхронные сигналы не требуют наличия тактовой частоты, однако, взамен они полагаются на определенные последовательности бит для управления

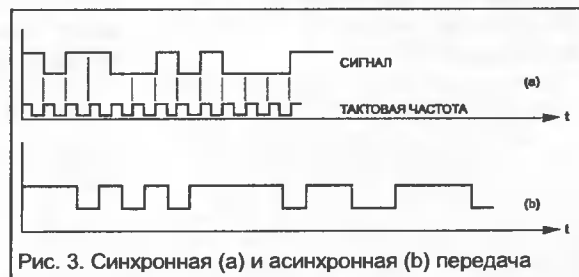


Рис. 3. Синхронная (а) и асинхронная (б) передача

временной диаграммой при декодировании. На рис. 3 приведено сравнение синхронной и асинхронной передачи.

Наиболее общая асинхронная система кодирования представляет собой 5-битный код Бодо со стартовым элементом (release) в начале и стоповым элементом (latch) в конце, изначально разработанный для использования с механическими телетайпами. Система «заперта» («latched»), пока не поступит «отпирающий» («release») элемент, который разрешает интерпретацию принтером следующих пяти элементов как информационных бит. Двоичные значения этих пяти бит соответствуют определенному символу. На рис. 4 показаны последовательности, соответствующие символам C и W соответственно. Непосредственно за пятым информационным битом следует стоп, (элемент «latch»), длительность которого составляет как минимум 1,42 длительности элемента. Этот бит останавливает дешифратор принтера до тех пор, пока элементом «release» не нач-

FSK может передаваться когерентно и некогерентно. Когерентность подразумевает, что фаза каждой посылки mark или space является неизменной по отношению к некоторому эталону. Это соответствует генерации FSK сигнала путем переключения между двумя генераторами, работающими на частотах mark и space. Несмотря на то, что этот метод иногда используется, существует ограничение: чтобы переход от mark к space и обратно происходил без разрыва фазы, значения сдвига и скорости передачи должны быть связаны между собой. Синхронный FSK сигнал, который имеет сдвиг в герцах, кратный скорости передачи в бодах, является

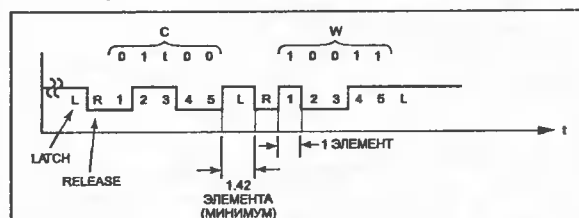


Рис. 4. Типичная старт-стоповая система с кодом Бодо

нется следующая посылка. Элемент «latch» между символами может быть очень длинным, специально для случая ручной генерации символов, когда оператор производит набор намного медленнее, чем система способна передавать. Нецелочисленная минимальная длительность элемента «latch» в 1,42 длительности элемента и случайная природа ручной генерации символов подчеркивает асинхронный тип этой схемы.

Распространенные синхронные системы используют ARQ кодирование Мура. Код Мура имеет 7 бит на символ и содержит стартовые и стоповые элементы. Синхронизация бит поддерживается благодаря использованию тактовой частоты, которая соответствует скорости передачи принимаемого сигнала. Синхронизация символов поддерживается путем передачи «пустых» символов между символами данных.

Частотное уплотнение (Frequency Division Multiplex, FDM)

Несколько FSK сигналов могут быть переданы одновременно в выделенной частотной полосе путем определения разных центральных частот для всех FSK сигналов. Этот метод одновременной передачи называется FDM. В радиодиапазоне несколько аудио FSK сигналов часто объединяются для передачи SSB передатчиком. Эту форму FDM часто называют телеграфией со звуковой несущей (Voice Frequency Telegraph, VFT). Для минимизации занимаемой полосы частот индивидуальные FSK каналы обычно имеют незначительный сдвиг в пределах от 50 до 200 Гц. Типичная FDM система показана на рис. 5. ВЧ радиосистема обычно передает 16 тональных каналов, однако возможна передача 24-х и более каналов на боковой полосе в 3 кГц.

ВЧ радиосистема обладает многолучевым затуханием сигнала. Относительно узкополосная природа этого явления приводит к тому, что в любой

момент времени оно влияет только на один из двух тональных каналов. Как защита от затухания используется передача дублирующих данных в тональных каналах той же FDM группы. Если дублирующие каналы разнесены примерно на 1 кГц, затухание в одном канале является относительно некоррелированным с затуханием в дублирующем канале. В результате, можно получить значительное улучшение характеристик системы в плане чувствительности к многолучевому затуханию. В 16-канальной ВЧ схеме канал 1 дублирован каналом 9, канал 2 — каналом 10 и т. д. Недостатком такой схемы является то, что половина каналов используется только для дублирования. Альтернативным подходом к решению проблемы является использование «перемежающейся» FDM.

Перемежающаяся FDM имеет преимущество в результате того, что длительность сигналов mark является комплементарной относительно длительности сигналов space. Это означает, что прием только частоты mark или только частоты space достаточен для восстано-

Двойная частотная манипуляция (Double Frequency Shift Keying, DFSK)

DFSK, иногда называемая DFS или «twinplex», является схемой передачи двух независимых потоков двоичных данных путем сдвига частоты одиночной несущей на одну из четырех дискретных частот.

На рис. 7 показана кодовая таблица для преобразования двух двоичных бит в одно из четырех выходных состояний. Например, если бит X и бит Y равны 1 и 0 соот-

Входные биты		Состояние выхода
X	Y	
0	0	A
0	1	B
1	0	C
1	1	D

Рис. 7

ветственно, будет выходное состояние C. Если каждое из четырех выходных состояний (от A до D) поставить в соответствие с определенной частотой FSK, станет возможным передать любое состояние двух бит как один элемент. На рис. 8 показан типичный DFSK сигнал. Так как два двоичных канала являются независимыми, они могут содержать любые значения для синхронных или асинхронных сигналов. Можно иметь телетайп в одном канале и код Морзе — в другом. Битовая синхронизация между каналами не требуется.

В примерах на рис. 7 и 8 выходные состояния от A до D представлены частотами от f_1 до f_4 соответственно. Существует 24 различных способа которыми четыре состояния

могут быть поставлены в соответствие четырем частотам. Поскольку в DFSK сигнале передается только одна частота в один момент времени, мощность передатчика используется намного эффективнее, чем при FDM передаче, которая предъявляет дополнительные требования к линейности для исключения появления интермодуляционных составляющих.

Демодуляция FSK

Методы демодуляции FSK могут быть разделены на две основных категории: демодуляция на основе частотного детектора и демодуляция на основе фильтров. Ранние разработки FSK демодуляторов основывались на частотном детекторе, поэтому эти демодуляторы будут рассмотрены первыми.

Демодуляторы на основе частотного детектора

Демодуляторы на основе частотного детектора обрабатывают FSK сиг-

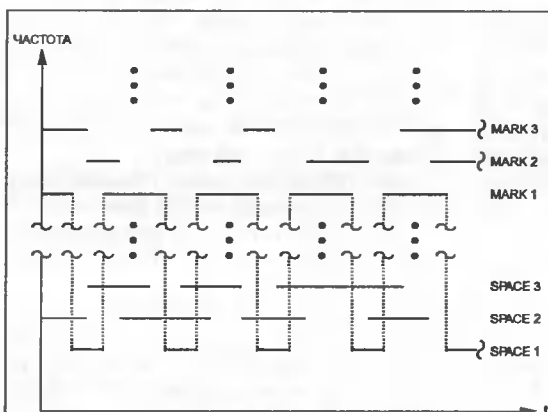


Рис. 6. Перемежающийся FDM FSK сигнал

ния переданных данных. Если частоты mark и space разнесены примерно на 1 кГц, они могут быть продетектированы отдельно. При объединении далеко разнесенных пар частот mark и space в одной боковой полосе FDM одна пара может переключаться с другими парами. Эта схема проиллюстрирована на рис. 6. Путем переключения пар частот FSK удается получить хорошую устойчивость к многолучевому затуханию без потери эффективности использования выделенной полосы частот. Несмотря на преимущества переключения, эта схема является менее популярной по сравнению с простым дублированием, так как она требует более сложного демодулятора.

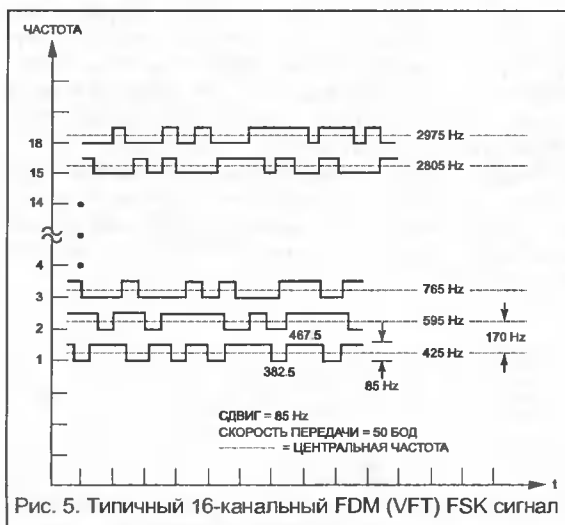


Рис. 5. Типичный 16-канальный FDM (VFT) FSK сигнал

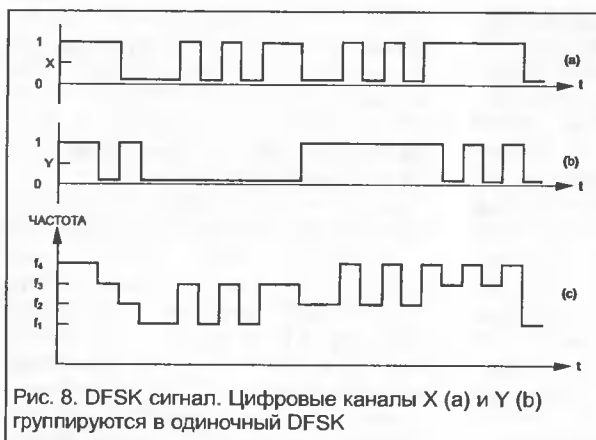


Рис. 8. DFSK сигнал. Цифровые каналы X (a) и Y (b) группируются в одиночный DFSK



Рис. 9. FSK демодулятор на основе частотного детектора

нал как обычный ЧМ сигнал с двоичной модуляцией. На рис. 9 показана блок-схема такого демодулятора. Вначале сигнал подвергается фильтрации для подавления внеполосных помех, затем сигнал ограничивается для устранения паразитной амплитудной модуляции. Ограниченный сигнал частотно детектируется для получения положительного выходного напряжения для частоты mark и отрицательно — для частоты space. Продетектированный сигнал подвергается низкочастотной фильтрации для подавления шумовых компонент на частотах выше скорости передачи данных, и, наконец, схема ограничителя (компаратор) преобразует положительное напряжение в уровень лог. 1 единицы, а отрицательное — в уровень лог. 0.

Данный тип демодулятора является очень популярным ввиду относительной простоты и не критичности в настройке. Демодуляторы на основе PLL являются более современными, однако, их характеристики являются очень близкими к характеристикам обычных частотных детекторов. Для небольшого класса сигналов, специально приспособленных к PLL демодуляции, такие демодуляторы имеют лучшие параметры по сравнению с обычными частотными детекторами. Демодуляторы на основе частотного детектора несколько усложняются при работе с FDM, а для работы с перемежающимися FDM сигналами вообще не подходят. Они также обычно не используются с DFSK сигналами.

Частотные детекторы не являются оптимальными в том смысле, что они обеспечивают худшие параметры при детектировании сигнала, чем это теоретически возможно. Для того чтобы убедиться в этом, нужно рассмотреть спектр типич-

ного FSK сигнала. Как показано на рис. 10, практически вся энергия посылок mark и space сосредоточена в полосе, ширина которой равна удвоенной скорости передачи, центрированной относительно частоты mark и space соответственно. На рис. 11 показан спектр при наличии помех. Сигнал помехи с частотой f_A подавляется, так как он не попадает в полосу пропускания входного фильтра частотного детектора. Сигнал помехи с частотой f_B подавляется благодаря действию ограничителя. В результате доминирует сигнал с максимальным уровнем (эффект «захвата» частотного детектора). Однако сигнал помехи с частотой f_C имеет высокий уро-

вень и лежит в полосе пропускания входного фильтра, в результате селективного затухания, меньше амплитуды внутриполосной помехи. Поэтому во время передачи сигнала mark демодуляция будет полностью нарушена. Намного труднее показать графически эффект действия широкополосной шумовой помехи. Когда огибающая шума достигает значения амплитуды FSK сигнала, демодуляция нарушается.

Сравнение спектра FSK сигнала со спектром сигнала помехи, к которому чувствителен частотный детектор, приводит к выводу, что характеристики частотного детектора значительно ухудшены из-за относительно широкой полосы пропускания. В полосу пропускания включена значительная часть спектра, которая не содержит энергии полезного сигнала, что, очевидно, не является оптимальным при демодуляции FSK.

В этом месте может возникнуть вопрос: «Какой же метод демодуляции является оптимальным?» Используя теорию детектирования сигналов, можно показать, что детектор на основе полосовых фильтров обеспечивает лучшие характеристики.

Демодуляция FSK на основе фильтров

Демодуляция FSK на основе фильтров является попыткой оптимально согласовать параметры FSK сигнала со структурой демодулятора для минимизации уровня ошибок демодуляции. Упрощенный спектр для демодулятора на основе фильтров показан на рис. 13. Правильный выбор характеристик фильтров зависит не только от параметров сигнала, но и от характера помех. Классический демодулятор на основе полосовых фильтров является оптимальным для когерентной FSK в условиях помех в виде белого гауссовского шума. Другие виды демодуляторов на основе фильтров используются для некогерентной FSK и/или нестационарного шума.

Блок-схема простейшего демодулятора на основе полосовых фильтров для когерентной FSK показана на рис. 14. В этом демодуляторе выходные сигналы фильтров сравниваются, и если уровень сигнала на выходе фильтра mark больше, чем на выходе фильтра space, принимается решение, что передается сигнал mark. Аналогично детектируется сигнал space. Демодулятор на основе полосовых фильтров является оптимальным, так как характеристики фильтров согласованы с характеристиками передаваемого сигнала. В результате ослабление полезного сигнала будет минимальным при максимальном ослаблении шумов. Для белого шума оптимальный фильтр имеет импульсную



Рис. 10. Упрощенный спектр типичного двоичного FSK сигнала

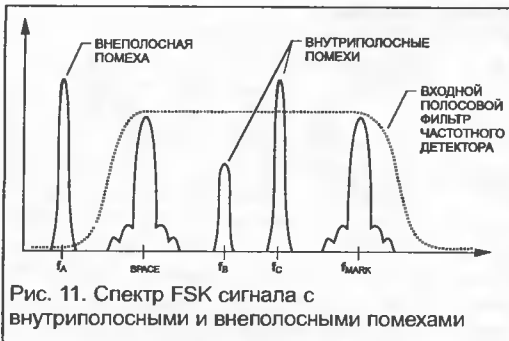


Рис. 11. Спектр FSK сигнала с внутриполосными и внеполосными помехами

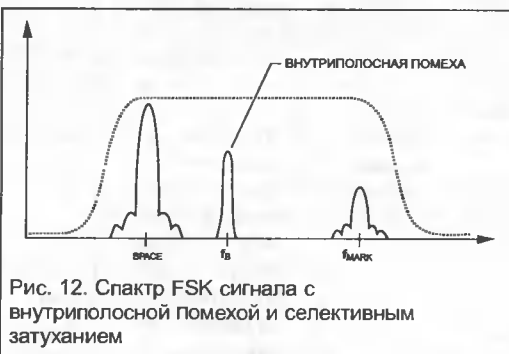


Рис. 12. Спектр FSK сигнала с внутриполосной помехой и селективным затуханием

вень и лежит в полосе пропускания входного фильтра, в результате он делает полностью невозможной демодуляцию полезного сигнала. В случае селективного затухания, как показано на рис. 12, несмотря на то, что сигнал space имеет больший уровень, чем внутриполосная помеха, амплитуда сигнала mark, ослаб-



Рис. 13. Упрощенный спектр FSK сигнала при демодуляции с помощью фильтров



Рис. 14. FSK демодулятор на основе фильтров

характеристику, равную обращенному во времени элементу входного сигнала. Поскольку некогерентная FSK является намного более распространенной по сравнению с когерентной FSK, требуется, чтобы демодулятор был нечувствительным к фазе.

Оптимальная демодуляция некогерентной FSK может быть осуществлена путем выделения огибающих выходных сигналов фильтров. Демодулятор этого типа показан на рис. 15. Выходы фильтров сигналов mark и space поступают на детекторы огибающей, а затем сравниваются для обнаружения сигнала с большим уровнем. Заметьте, что информация о фазе при этом не требуется. С «правильной» формой АЧХ фильтров характеристики этого типа демодуляторов приближаются к теоретически оптимальным для некогерентной FSK. «Правильной» АЧХ фильтров для условий помех в виде белого шума является та, которая имеет форму, соответствующую спектру передаваемого сигнала (рис. 10). Для «прямоугольной» FSK модуляции эта форма соответствует полосовому фильтру с АЧХ в виде функции sinc, центрированной по частоте сигнала mark или space. К сожалению, предположение о помехах, как о белом шуме, неверно в большинстве реальных случаев.

Для общего случая некогерентной FSK в условиях помех, отличных от бе-

ет и в условиях белого шума.

Для каждой формы АЧХ фильтра существует оптимальная полоса пропускания. Обычно, если полоса пропускания слишком широкая, будет восприниматься излишняя энергия шумовых помех. Если же полоса пропускания слишком узкая, последовательные элементы передаваемого сигнала будут искажать друг друга. Это называется межсимвольными помехами. В узкополосных фильтрах это вызвано «звоном» или неспособностью демпфирования энергии предыдущего элемента перед приемом последующего.

Для определения характеристик фильтра удобно использовать термин полосы пропускания по уровню -3 дБ. Оптимальная полоса пропускания по уровню -3 дБ напрямую зависит от скорости передачи информации или от обратной длительности элемента. Если мы рассматриваем фильтры с нормализованной полосой пропускания, полученной умножением полосы пропускания по уровню -3 дБ (В) на длину элемента (Т), можно изобразить единичную кривую качества фильтра для каждой формы АЧХ, не зависящую от скорости передачи данных. На рис. 16 показана связь нормализованной полосы пропускания и отношения сигнал/шум для постоянной вероятности ошибки в ус-

ловиях белого шума. Оптимальная нормализованная полоса пропускания для каждого фильтра находится в точке минимума отношения сигнал/шум. Слева от этого минимума доминируют межсимвольные помехи, справа — избыточная энергия белого шума.

Выбрав фильтр, который одновременно имеет хорошие характеристики при белом шуме и крутые склоны АЧХ, можно получить отличное качество демодуляции некогерентной FSK в условиях помех в виде белого и отличного от белого шума. Полоса пропускания фильтра и центральная частота могут варьироваться в широких пределах для наилучшего согласования с параметрами передаваемого сигнала: скоростью передачи и сдвигом частоты.

Выбор порога компаратора

Порог компаратора на рис. 15 может задаваться несколькими разными способами. В простейшем случае можно

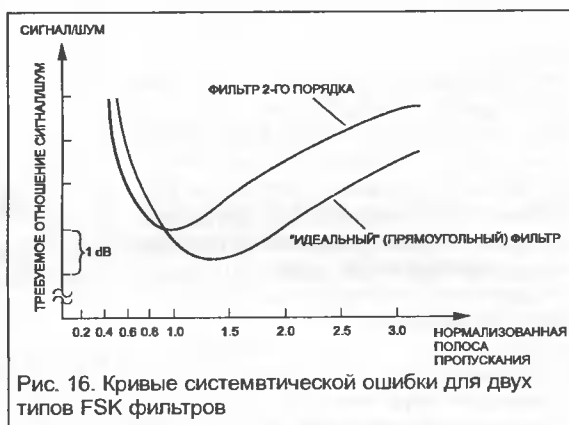


Рис. 16. Кривые систематической ошибки для двух типов FSK фильтров

принять решение о передаваемом сигнале mark, если выходной сигнал детектора mark превышает «шумовой» выходной уровень детектора space. При одинаковых уровнях передаваемых сигналов и при отсутствии селективного затухания этот метод является теоретически оптимальным методом принятия решения. Кривая ошибок для этого случая при двоичной некогерентной FSK в условиях белого шума показана на рис. 17. Если же FSK сигнал подвержен селективному затуханию (релее-

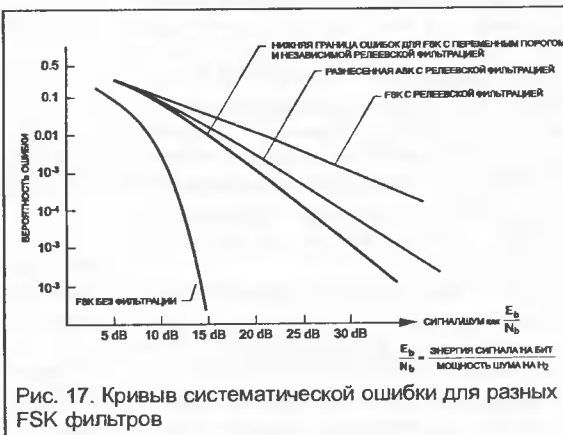


Рис. 17. Кривые систематической ошибки для разных FSK фильтров

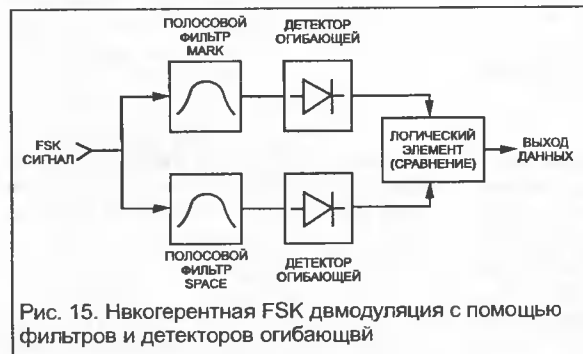


Рис. 15. Нekoгерентная FSK двомодуляция с помощью фильтров и детекторов огибающей

ская статистика амплитуд), то характеристики демодулятора существенно ухудшаются.

Если сигналы mark и space разделены по частоте, то они могут подвергаться независимым затуханиям. Можно просуммировать выходные сигналы mark и space с нужными знаками, чтобы получить разностный сигнал. На рис. 18, а по-

казан суммарный выходной сигнал для попеременной передачи mark и space. Огибающие на выходах детекторов mark и space изменяются при затухании сигналов. После суммирования сигналов компаратор сравнивает разностный сигнал с опорным напряжением, которое представляет собой порог. Если сумма превышает этот порог, детектируется mark, иначе — space. Если сигналы mark и space имеют одинаковые амплитуды, оптимальное значение порога равно нулю. Однако, из-за селективного затухания появляется необходимость динамически изменять порог для достижения оптимальной демодуляции. Эффект селективного затухания при фиксированном и переменном пороге продемонстрирован на рис. 18, б и с.

Характеристики детектора с переменным порогом можно сравнить с разновидностью демодуляции двух независимо затухающих сигналов с амплитудной манипуля-

цией (Amplitude Shift Keying, ASK). Естественно, благодаря дифференциальной природе сигналов mark и space, параметры с точки зрения вероятности ошибки будут лучше, чем для ASK. Характеристики ошибок для ASK и их нижние границы для FSK с переменным порогом показаны на рис. 17. Для получения наилучших характеристик параметры схемы детектора с переменным порогом должны быть оптимизированы по отношению к характеристикам сигнала.

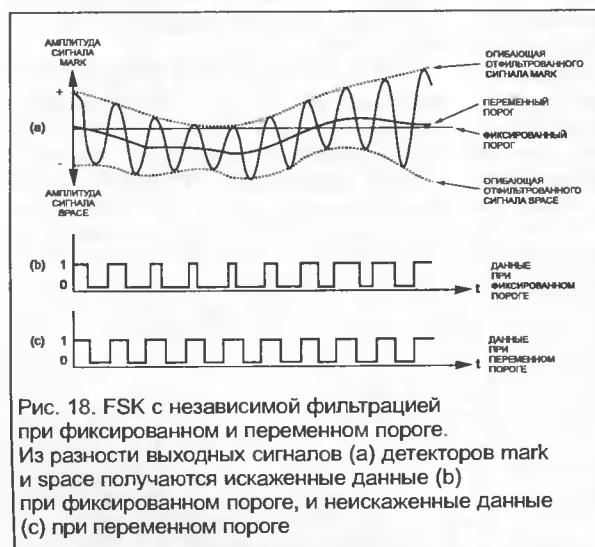
Заключение

FSK сигналы имеют много различных форм в зависимости от приложений, для которых они предназначены. По мере совершенствования электроники и теории детектирования сигналов становится ясно, какие FSK демодуляторы требуются для оптимальной демодуляции различных форматов сигналов.

Bob Watson

Watkins-Johnson Company
www.wj.com

перевод Леонида Ридико
wubblick@yahoo.com



ПОЛИСЭТ

оптовые поставки электронных компонентов

SAMSUNG

ELECTRO-MECHANICS

SMD компоненты

многослойные керамические SMD конденсаторы, танталовые SMD конденсаторы, SMD резисторы, резисторные сборки...



Конденсаторы электролитические

поверхностного монтажа (SMD), сверхминиатюрные и неполярные, с низкими импедансом и током утечки, для отклоняющих систем и пуска электромоторов...

Конденсаторы пленочные



GOOD-ARK

Выпрямительные диоды и мосты

SMD диоды и выпрямительные мосты, сверхбыстрые и мощные, высоковольтные и автомобильные, с барьером Шоттки и диоды Зенера...

Тел./факс: (095) 967-05-91, 967-05-92; <http://www.poliset.ru>; e-mail: info@poliset.ru

(Продолжение. Начало — №3/2002)

Автоматы световых эффектов от «А» до «Я»

Прежде, чем приступить к описанию программных автоматов световых эффектов, следует хотя бы поверхностно ознакомиться с принципами составления самых программ. Для этого необходимо знать, в каком виде эти программы составляются, и как примерно это делается.

При составлении программ следует учитывать, что самый верхний разряд счетчика, дешифратора и т. д. является самым младшим. Однако написание программ в двоичной системе счисления, в которой собственно и работает цифровая техника, весьма затруднительно. В восьми разрядах (типовой размер байта) информация может находиться в любой из 256 комбинаций и в двоичной системе счисления иметь вид от 00000000 до 11111111. Во-первых, это уже восьмизначное число и запомнить любую промежуточную комбинацию весьма непросто, во-вторых, подобные наборы двоичных чисел занимают много места при очень маленькой информативности. Именно поэтому для написания программ пользуются шестнадцатеричной системой счисления. Использование этой системы позволило существенно уменьшить количество разрядов в числе, и в этом отношении она даже «экономнее» десятичной системы. Описывая состояние тех же восьми разрядов информация может иметь вид от 00 до FF, последнее и означает число 255 (в цифровой технике принято за первую цифру считать цифру 0). Конечно, шестнадцатеричная система счисления трудновата в понимании, но освоившись и привыкнув, вполне можно писать программы для подобного рода устройств именно в ней. Для наглядности в табл. 1 сведено соответствие десятичных, двоичных и шестнадцатеричных чисел для первых четырех двоичных разрядов.

В случае, когда разрядов более четырех, младшая «четверка» пишется справа, а старшая — слева. Если общее количество разрядов не кратно четырем, то недостающие разряды понимаются ус-

ловно, как будто они есть. Например, разрядов 6, то максимальное число в двоичном виде будет иметь вид 11 1111, где первые две единицы показывают состояние старших разрядов, а остальные — младших (самый младший разряд располагается справа). В шестнадцатеричном виде это же число будет иметь вид 3F. Для более удобного понимания в случаях, когда требуется описать состояние более восьми разрядов, подобные коды komponуются парами.

Программные «бегущие огни» строятся на базе оперативных запоминающих устройств (ОЗУ) или постоянных запоминающих устройств (ПЗУ).

Построение на ОЗУ позволяет в любой момент времени изменить иллюминацию, но такие устройства требуют после каждого включения записывать все эффекты снова, а при больших объемах ОЗУ это занимает очень много времени. Собственно по этой причине подобные устройства не так популярны. Содержимое ПЗУ записывается один раз и на протяжении всего времени эксплуатации не меняется, а поскольку современная элементная база позволяет применять ПЗУ очень больших объемов, недостаток эффектов практически не замечается.

Для начала рассмотрим принципы

тор, набор счетчиков с количеством разрядов, равным количеству адресных входов микросхемы ППЗУ, и собственно само ППЗУ. После ППЗУ обычно идут силовые ключи, но об этом чуть позже. Принципиальная схема автомата приведена на рис. 1, на рис. 2 — чертеж печатной платы (а — сторона печатных проводников, б — сторона деталей), на рис. 3 — расположение деталей. Перечень используемых элементов сведен в табл. 2.

На микросхеме DD1 собран тактовый генератор (R1 регулирует скорость «бега»), на DD2, DD3 — счетчик, DD4 — ППЗУ. Тактовые импульсы с генератора поступают на вход EC DD2.1, и с приходом каждого спада (перехода от состояния лог. 1 в состояние лог. 0) содержимое счетчика увеличивается на единицу. Выходные сигналы счетчика

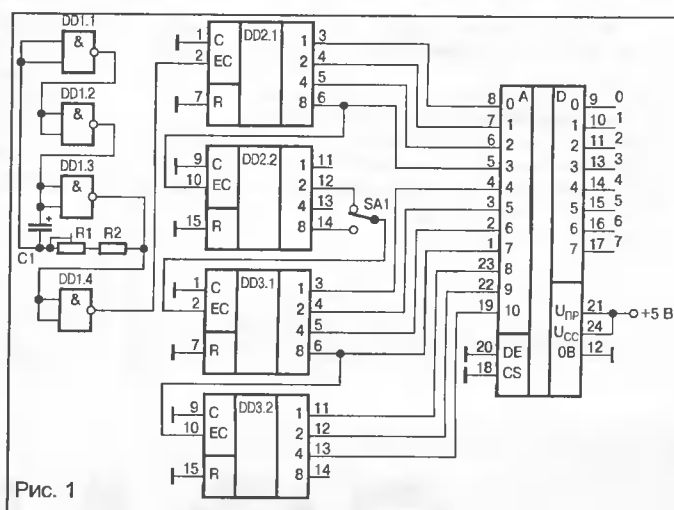


Рис. 1

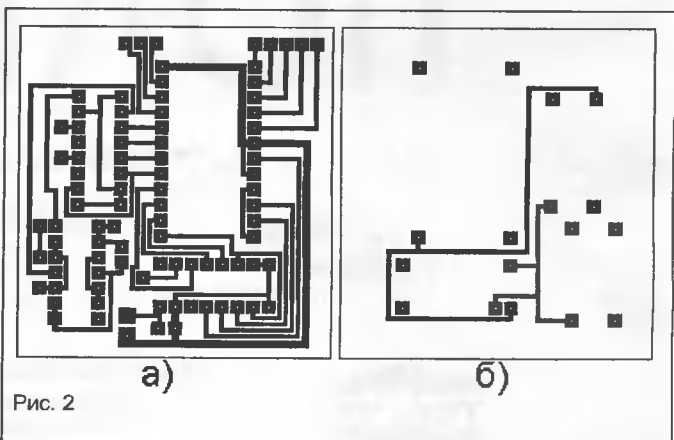


Рис. 2

простейшего автомата на базе перепрограммируемого постоянного запоминающего устройства (ППЗУ) с использованием микросхемы K573РФ2 (РФ5).

Как правило, подобные автоматы содержат тактовый генера-

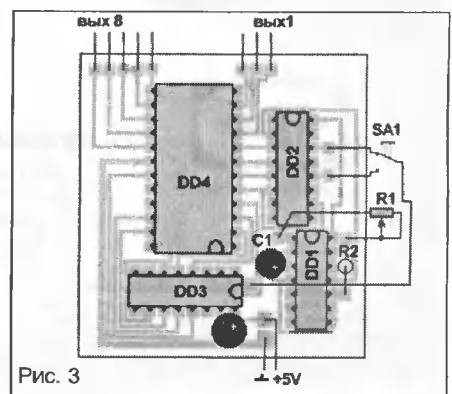


Рис. 3

Таблица 1

Десятичное число	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Двоичное число	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
	0	0	0	0	1	1	1	1	0	0	0	1	1	1	1	1
	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
Шестнадцатеричное число	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Таблица 2

Позиция	Наименование
DD1	K561ЛА7, K561ТЛ1, K561ЛЕ5
DD2, DD3	K561ИЕ10
DD4	K573РФ2, K573РФ5
C1	10 мкФ х 16В
R1	любого типа на 10...15 кОм
R2	4,3 кОм
SA1	любого типа

поданы на первые четыре адресных входа ППЗУ, в которое записана программа. Тут необходимо сделать небольшое отступление и подробнее рассмотреть методы составления программ для подобных устройств.

ППЗУ K573РФ2 имеет 11 адресных входов и 8 выходов данных. Комбинация на входах может иметь вид от 0000 до 07FF (2048 комбинаций). Поскольку выходов 8 (один выход — 1 бит, 8 бит — 1 байт), то записанная информация может иметь вид от 00 до FF. Таким образом, составив грамотно программу для ППЗУ и подав на адресные входы выходные сигналы считывающего тактовые импульсы счетчика, можно заставить информацию на выходах ППЗУ меняться в необходимой последовательности. В простейшем случае — это бегущие огни, т. е. перемещение ее от младшего разряда к старшему. Тут следует отметить, что подобная программа будет самой примитивной из того, что можно записать в ППЗУ. Имея восемь выходов, можно «зажигать» любой из огней в любой последовательности, тем самым создавая самые разнообразные эффекты.

Собственно по этому принципу и построен предлагаемый автомат. Счетчик на DD2, DD3 считает импульсы с генератора и постоянно увеличивает свое содержимое. Выходы счетчика подключены к адресным входам ППЗУ, причем разряды 4 и 6 не используются вообще, а для подачи на вход DD3.1 импульсы снимаются с разрядов 5 или 7 (переключатель SA1). Это позволяет через каждые 16 тактов повторять информацию, записанную в ППЗУ, что сделано для увеличения зрительного восприятия. Остается лишь добавить, что эта схема управления обеспечивает статическую индикацию, т. е. в любой момент времени может быть включены все восемь светоизлучающих приборов.

О том, как быстро составлять программы для подобных схем, впрочем, как и о программаторах, здесь упоминаться не будет. Для начинающих это тяжело, им предлагается готовый вариант «прошивки» ППЗУ.

Объем ППЗУ K573РФ2 не очень велик, но эта микросхема имеет вход выбора разрешения работы, позволяющий несложно объединять несколько микросхем, тем самым увеличивая объем информации. Для этого можно

напаять прямо друг на друга несколько корпусов ППЗУ и, подавая на вывод 20 лог. 0, разрешить работу той или иной микросхеме. При наличии лог. 1 на выводе 20 микросхема переходит в высокоимпедансное состояние, при котором ее выходы пассивны и позволяют работать другим микросхемам. Тут следует оговориться, что предлагаемые автоматы используют K573РФ2 (РФ5), имеющие объем 2048 байт, однако можно применять микросхемы с большими объемами, например K573РФ4 (РФ6) — 8192 байта, K573РФ8 — 32768 байт. Использование именно РФ2 объясняется их большой популярностью и распространенностью.

В качестве силовых можно использовать ключи любого типа из предыдущей части статьи («Схемотехника», №4/2002) или ключи, описанные ниже.

Поскольку информация может изменяться на выходах в необходимом программисту порядке, то, добавив несколько микросхем, можно управлять 64-мя индикаторами, причем используя всего 16 силовых ключей. Подобные характеристики возможны при динамической индикации, т. е. в любой момент времени горит не более восьми ламп. Поскольку зажигание той или иной восьмерки производится с большой частотой, благодаря инерционности спиралей ламп накаливания и человеческого глаза складывается впечатление полноценного свечения всех индикаторов.

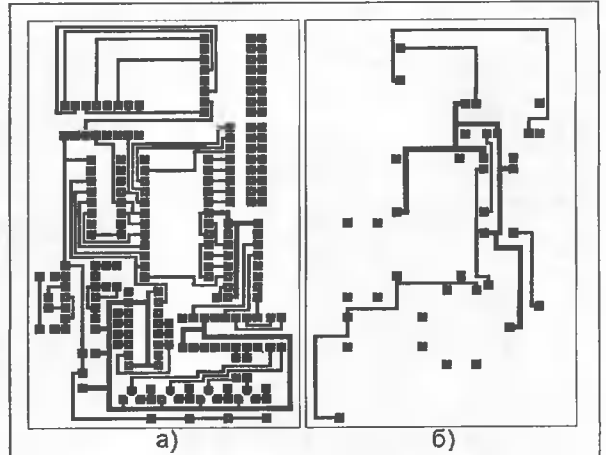


Рис. 5

Принципиальная схема устройства с динамической индикацией приведена на рис. 4, на рис. 5 — чертеж печатной платы (а — сторона печатных проводников, б — сторона деталей), на рис. 6 — расположение деталей (на схеме не показан оксидный конденсатор, установленный в цепи питания — 100 мкФ х 6,3 В). Перечень элементов сведен в табл. 3. На элементах DD1.1, DD1.2 и DD1.3, DD1.4 собраны два генератора. Первый служит для организации динамической индикации, второй — тактовый и определяет скорость «бега».

Для определенности примем схему выходных каскадов, приведенную на рис. 7 (список элементов сведен в табл. 4), расположение индикаторов в виде квадрата 8х8 (рис. 8), и пронумеруем столбцы латинскими буквами от А до Н, а строки — цифрами от 1 до 8. Теперь запишем первые коды программы в виде 01 02 04 08 10 20 40 80 (при-

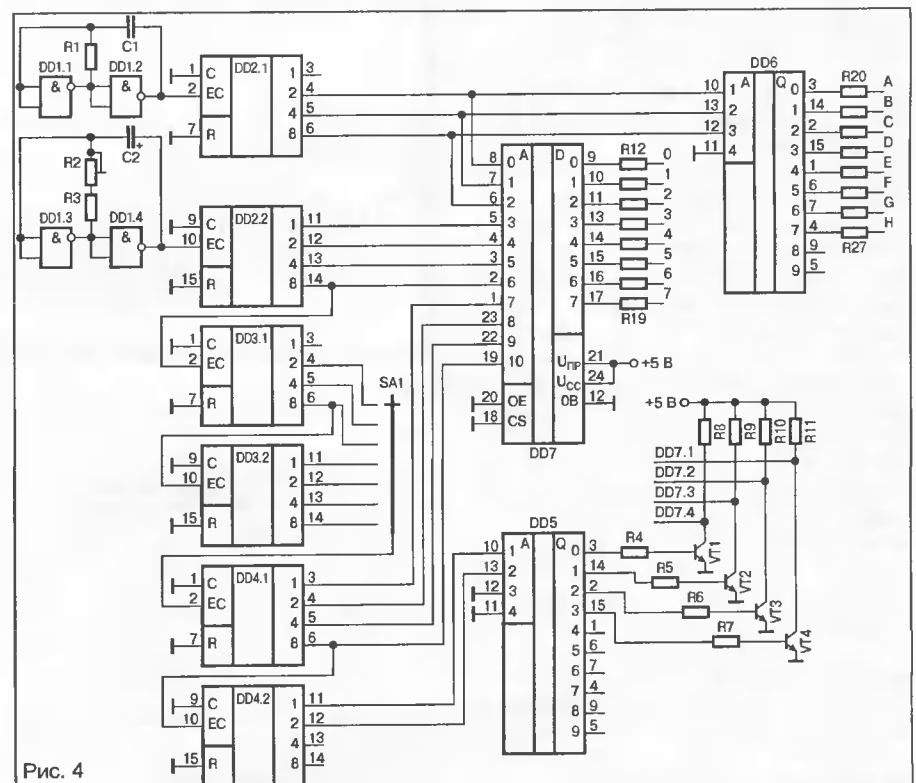


Рис. 4

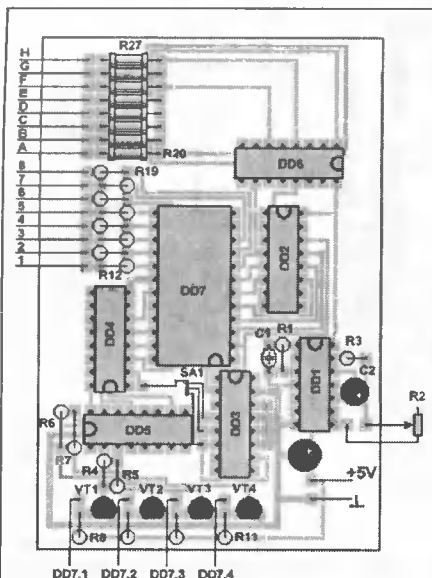


Рис. 6

Таблица 3

Позиция	Наименование
DD1	K561ЛА7, K561ТЛ1, K561ЛЕ5
DD2-DD4	K561ИЕ10
DD5, DD6	K561ИД1
DD7	K573РФ2, K573РФ5
R1	33 кОм
R2	любого типа на 10...15 кОм
R3	4,3 кОм
R4-R7, R12-R27	3,9...4,3 кОм
R8-R11	1,5...2 кОм
VT1-VT4	КТ3102 с любой буквой
C1	0,1 мкФ любого типа
C2	10 мкФ x 16В
SA1	любого типа

веденный фрагмент не более как пример для рассмотрения принципа работы устройства).

На микросхемах DD2.2-DD4 (рис. 4) собран 20-разрядный счетчик, причем три разряда элемента DD2.1 используются для дешифратора DD6, который осуществляет выбор столбцов матрицы. С каждым вторым спадом тактовых импульсов содержимое счетчика DD2.1 увеличивается на единицу, следовательно, на выходах дешифратора DD6 происходит передвижение лог. 1 от нулевого выхода до седьмого. Эта «плавающая» единица, усиленная по току транзисторами VT1 (рис. 7), открывает тот или иной силовой транзистор VT2, разрешая свечение столбца. Теперь, если на базу транзистора VT4 через токоограничивающий резистор подать открывающую лог. 1 он откроется, тем самым, открывая VT3. Поскольку транзисторы VT2 открываются поочередно в каждом столбце, то сначала загорится лампа A1, затем лампа B1 и т. д. до тех пор, пока не загорится лампа

H1, и цикл не начнет повторяться. В связи с тем, что открытие транзисторов VT2 происходит сравнительно быстро (тактовая частота генератора на DD1.1, DD1.2 составляет 300...500 Гц), естественно, что поочередное зажигание ламп не будет заметно, а появится ощущение их равномерного свечения. Теперь допустим, что генератор на DD1.3, DD1.4 остановлен, а на все, кроме первых трех, адресные входа DD5 подан лог. 0. Выше было оговорено, что первые коды программы имеют вид 01 02 04 08 10 20 40 80, следовательно при ситуации, когда на выходах счетчика DD2.1 будут все лог. 0, на самом верхнем выходе DD7 появится лог. 1. В то же время транзистор VT2 (рис. 7) будет открыт благодаря лог. 1 на нулевом выходе DD6 (вывод 3), и лампа A1 включится. С приходом следующего спада содержимое счетчика DD2.1 увеличится на единицу и изменит состояние на адресных входах DD7. Это повлечет появление лог. 1 на втором выходе

данных, откроется транзистор VT4 следующей строки. Выходы DD6 тоже изменят состояние, теперь лог. 1 будет на выходе В дешифратора, что в свою очередь откроет транзистор VT2 следующего столбца. В этой ситуации будет зажжена лампа B2. С приходом следующего спада тактового импульса снова адресная информация увеличится на единицу, соответственно на выходе С DD6 появится лог. 1, и теперь уже будет светиться лампа C3. Так будет происходить до тех пор, пока не загорится лампа H8, и видимая картина на матрице будет выглядеть диагональной линией, идущей от левого верхнего угла до правого нижнего (рис. 8).

Из вышесказанного следует, что для получения какой-либо «картинки» программа должна содержать последовательность из восьми шагов, которые собственно и образуют «картинку». Смена же самих «картинок» и будет составлять эффект «бега огня». Направление его движения возможно уже

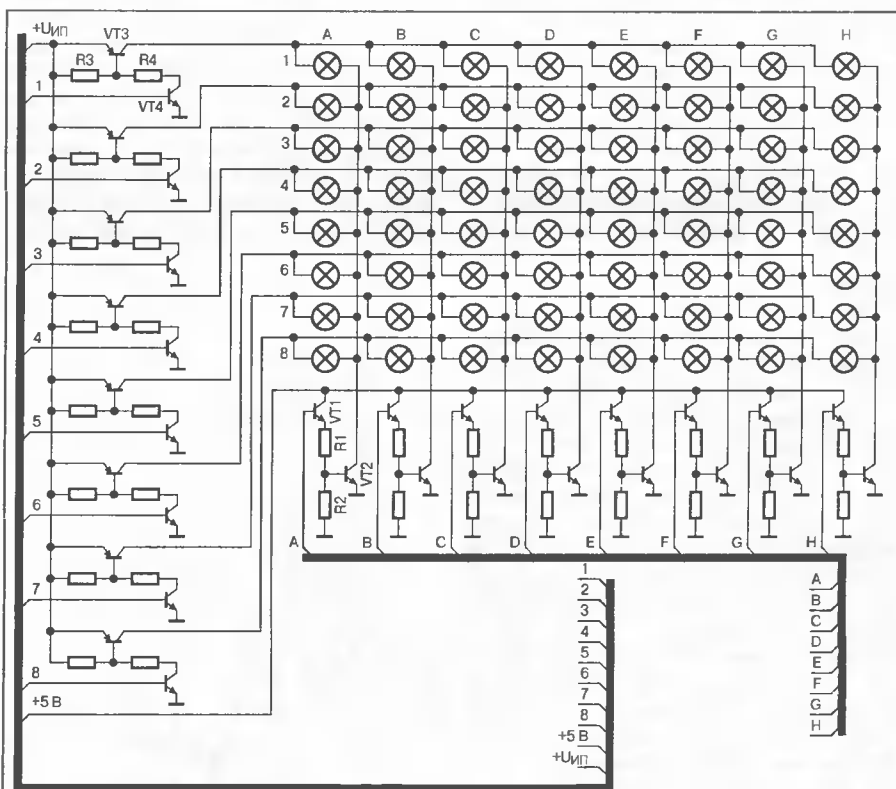


Рис. 7

Таблица 4

Позиция	Наименование
VT1	КТ503
VT2	КТ819
VT3	КТ503
VT4	КТ973
R1-R4	МЛТ-0,25 1 кОм

Примечание. Поскольку ключи управления столбцами и строками аналогичны, приведен список одного ключа для столбцов и одного для строк.

в двух плоскостях, т. е. помимо традиционных «влево — вправо» появилась возможность его перемещения и «вверх — вниз», а также комбинированного движения по диагоналям, периметру и т. д. Скорость смены «картинок» зависит от частоты генератора на DD1.3, DD1.4.

Переключатель SA1 определяет через сколько повторов «картинка» сменится на следующую. Последний же счетчик DD4.2, совместно с дешифратором DD5 образуют управление выбором микросхем ППЗУ DD7.1-DD7.4 по

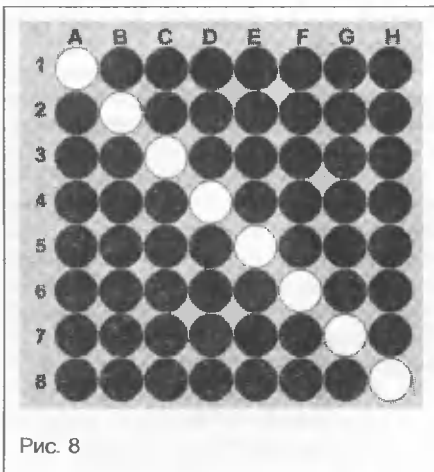


Рис. 8

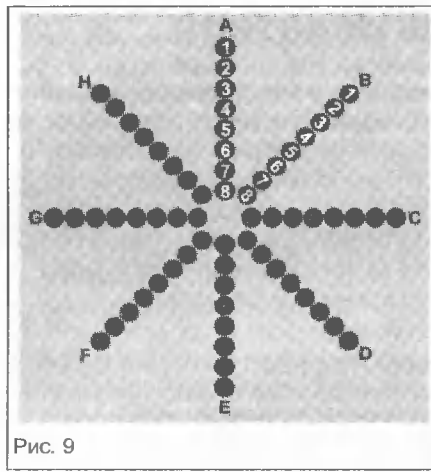


Рис. 9

их входам CS. Использование нескольких ППЗУ позволяет существенно увеличить набор эффектов, но и требует немало усидчивости при программировании, поэтому увеличивать объем памяти можно постепенно, уже эксплуатируя готовое устройство. Если одной микросхемы достаточно, то DD7.2-D7.4 и следующие за ним инверторы на транзисторах VT1-VT4 можно не запаивать в плату вообще.

Лампы можно расположить и в виде восьмиконечной звезды, в этом случае столбцы лучше сделать лучами, пример приведен на рис. 9. Надо будет

учесть, что построение программ несколько изменится, поскольку зрительное восприятие одного и того же эффекта для матрицы и звезды весьма разнятся. Для примера, если на матрице создать эффект поднятия горизонтальной линии вверх, то для звезды это будет выглядеть как движение «огня» от центра к краям.

Поскольку питание на лампы подается кратковременно, то, несмотря на ровность свечения, они будут гореть с неполной яркостью, поэтому при выборе ламп следует использовать лампы на напряжение, по крайней мере в два

раза меньше, чем напряжение питания (при напряжении питания силовой части 24 В нужно использовать лампы на 12 В). Переживать за их исправность не стоит — действующее значение подаваемого на лампы напряжения составляет $1/\sqrt{2} = 0,35$ от напряжения питания, поэтому они все равно гореть на полную мощность не будут.

Андрей Тишкунов,
root@amr.novshah.donpac.ru

Продолжение следует

**Редакция журнала
«Схемотехника»
приглашает авторов
к сотрудничеству**

По всем вопросам обращаться:
e-mail: editor@dian.ru
телефон: (095) 777-1215

Гонорары выплачиваются
авторам, проживающим
на территории СНГ

ПЕЧАТНЫЕ ПЛАТЫ ОТ ЦПТА

www.zao-cpta.ru ЦЕНТР ПЕРСПЕКТИВНЫХ ТЕХНОЛОГИЙ И АППАРАТУРЫ

12 лет

на рынке производства печатных плат

Вниманию разработчиков и производителей изделий электроники!

предлагаем Вам качественное и быстрое изготовление односторонних и многослойных печатных плат.

Прием заказов на производство производится по адресу:

115230, г. Москва,
Варшавское шоссе, д. 42, к. 2,
ЗАО "ЦПТА", метро "Нагатинская"
(территория АО "СЧЕТМАШ")
тел./факс 111-50-66
Тел. 111-92-98 (диспетчерская),
111-93-49, 787-41-06, 787-41-07
(монтажно-сборочное производство)
E-mail:
внутренний телефон 2-98
111-86-50, 111-92-98
(заказ печатных плат)

По Вашему желанию:

- ✓ Покрытие: ПОС, никель, золото, паяльная защитная маска
- ✓ Возможна разнообразная механическая обработка изделий: фрезеровка, перфорация сверлением, обрезка алмазным кругом и гильотиной
- ✓ Сертифицированное ISO 9001 монтажно-сборочное производство (SMD-монтаж от макетов до крупных серий)

Специальное предложение!

При заказе свыше 1000 шт. стоимость изготовления печатных плат снижается на **10%!!!**

pcb@zao-cpta.ru, zao-cpta@mtu-net.ru

Автомат для водонагревателя

Предлагаемое устройство, будучи встроено в электроводонагреватель, обеспечивает поддержание необходимой температуры воды, не позволяет включить нагревательный элемент при ее отсутствии и индицирует ее уровень. Стабилизация температуры осуществляется включением и выключением сетевого напряжения, подаваемого на нагреватель, в зависимости от температуры датчика — термистора. Включение симистора происходит в момент перехода сетевого напряжения через нуль, что исключает помехи радиоприему.

Электроводонагреватель «Эл-терм» ЭВБО-15/1.25 (15 литров, 1,25 кВт) очень удобен для установки на кухне садового дома, а также, при частых перебоях в подаче горячей воды и в городской квартире, однако ему свойственны некоторые недостатки. Во-первых, биметаллический контактный регулятор имеет очень большой гистерезис по температуре — вода или очень горячая или едва теплая. Установлен этот регулятор на корпусе довольно высоко и, когда уровень воды понижается и тепловой контакт между водой и регулятором нарушается, вода нагревается еще больше. Во-вторых, аварийный выключатель, который отключает нагревательный элемент при отсутствии воды, имеет более чем посредственный тепловой контакт с нагревателем. В результате до его срабатывания нагреватель успевает разогреться так, что пластмассовый корпус в месте крепления нагревателя размягчается и коробится. Третий недостаток — отсутствие указателя уровня воды. Предлагаемый автомат позволяет устранить эти недостатки.

Схема устройства приведена на рис. 1. Она состоит из термостабилизатора (верхняя половина схемы) и светодиодного индикатора уровня (нижняя половина). Блок питания автомата собран на трансформаторе Т1, диодном мосте VD1, конденсаторах С1 и С2 и микросхемном стабилизаторе DA1 на 6 В. Диод VD2 совместно с транзисторами VT1 и VT2 образуют формирователь импульсов в момент прохождения сетевого напряжения через нуль. Компаратор DA2 служит для сравнения напряжения, поступающего с подстраиваемого резистивного делителя R6R7R8, с напряжением на термисторе R10. Выходной сигнал компаратора через оптрон U1 управляет симистором VS1, который и коммутирует нагрузку.

Работает термостабилизатор так. Пока мгновенное напряжение сети по абсолютной величине превышает 25 В, транзистор VT1 включен и насыщен, транзи-

стор VT2 закрыт. На базе транзистора VT3 напряжение близко к напряжению питания (+6 В), он закрыт. Светодиод оптрона U1 погашен, ток на управляющий электрод симистора VS1 не поступает.

При снижении мгновенного напряжения сети до 25 В транзистор VT1 закрывается, VT2 открывается и подает на базу транзистора VT3 отпирающее напряжение. Теперь ток через светодиод оптрона определяется состоянием компаратора DA2, которое, в свою очередь, определяется температурой термистора R10. Если его температура ниже необходимой, установленной подстроечным резистором R7, напряжение на инвертирующем входе 4 компаратора выше напряжения на неинвертирующем входе 3, и на выходе 9 компаратора появится напряжение, близкое к нулю. Через резистор R13, транзистор VT3 и светодиод

светодиода оптрона за счет тока базы транзистора VT3.

В схеме устройства отсутствует ограничительный резистор, обычно включаемый последовательно с фотодином оптрона для ограничения тока через него и управляющий электрод симистора. Дело в том, что, как указывалось выше, ток через светодиод оптрона может течь только при мгновенном напряжении сети менее 25 В. Поэтому ток через фотодином оптрона и управляющий электрод симистора не может превышать $25 \text{ В} / 38 \text{ Ом} = 0,67 \text{ А}$, что допустимо как для оптрона, так и для управляющего электрода симистора (сопротивление нагревателя мощностью 1,25 кВт равно 38 Ом).

Симистор включается в самом начале каждого полупериода напряжения сети, что исключает помехи радиоприему. Первый раз включение симистора происходит в конце полупериода, в котором выполнилось условие его включения.

Индикатор уровня воды состоит из четырех однотипных узлов на транзисторах VT4-VT7 и светодиодах HL3-HL6. К контакту X1 платы подключен датчик максимального уровня, к X4 — минимального. Когда датчик находится в воде, ее проводимости оказывается достаточной для открывания соответствующего транзистора и включения зеленого светодиода в его коллекторной цепи. Если уровень

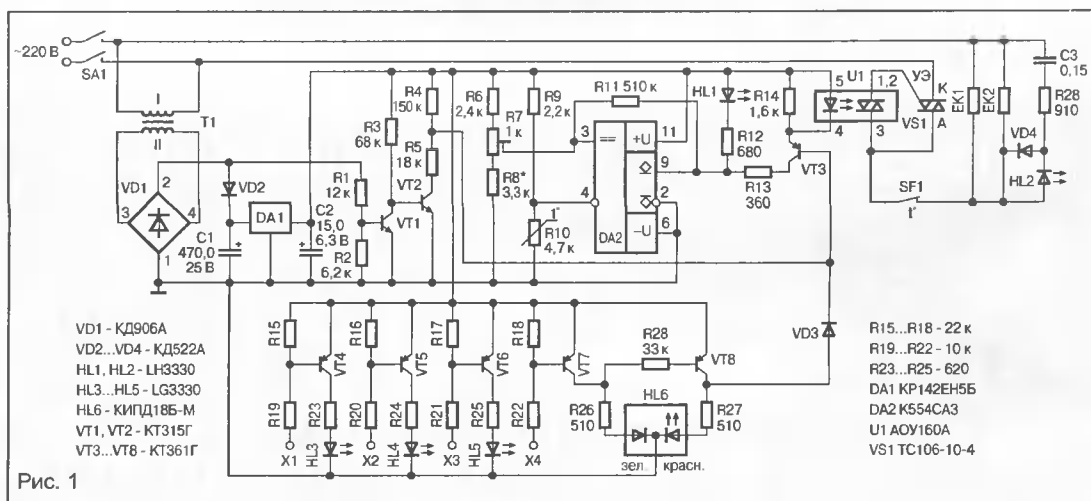


Рис. 1

оптрона U1 течет ток, включающий фотодином оптрона и симистор VS1. На нагреватель ЕК1ЕК2 подается напряжение сети, температура воды повышается. Включенное состояние водонагревателя индицируется светодиодами HL1 и HL2 красного цвета.

После того, как температура воды достигнет необходимой величины, напряжение на термисторе уменьшится, и симистор перестанет включаться. Когда вода слегка остынет, снова включится нагреватель, поддерживая нужную температуру.

Резистор R11 обеспечивает некоторый гистерезис по температуре включения и выключения нагревателя, а резистор R14 исключает включение

воды ниже расположения датчика, транзистор закрыт, и светодиод не светится. Общим электродом для датчиков служат корпус нагревательных элементов ЕК1 и ЕК2, с которыми соединен минус источника питания.

Датчик нижнего уровня дополнен еще одним транзистором — VT8. Если уровень воды будет ниже нижнего датчика, транзистор VT7 закроется, а VT8 — включится. Двухцветный светодиод HL6 начнет светиться красным, предупреждая о том, что воды в водонагревателе осталось мало. Кроме того, напряжение, близкое к напряжению питания, через диод VD3 поступит на базу транзистора VT3 и закроет его. Оптод U1, симистор VS1 и

нагреватель EK1EK2 не будут включаться.

SF1 — штатный аварийный выключатель электроводонагревателя.

Почти все элементы автомата, включая трансформатор Т1 и симистор VS1 на теплоотводе, смонтированы на печатной плате размерами 55×187,5 мм (рис. 2). Плата рассчитана на установку резисторов МЛТ, импортного аналога конденсатора К50-35 (С1), К53-16 (С2), который можно заменить на любой другой оксидный конденсатор, К73-17 (С3) на напряжение 400 В. Подстроечный резистор R7 — СП3-19а.

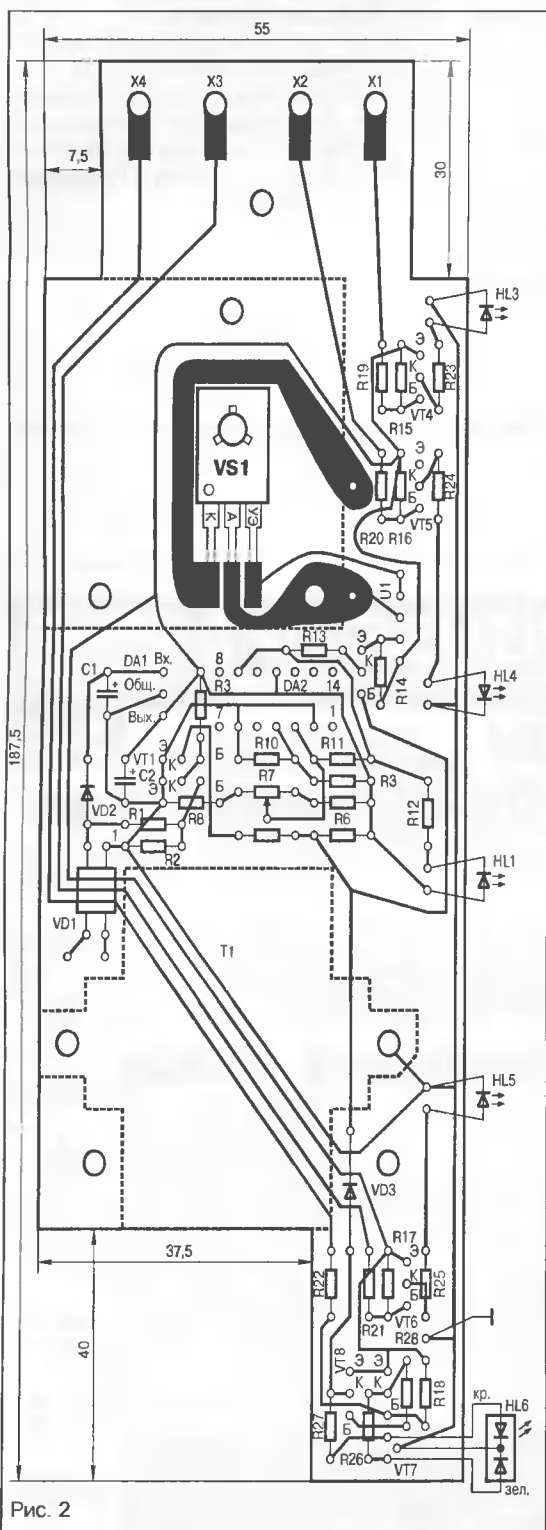


Рис. 2

Диоды VD2-VD4 — любые маломощные кремниевые, диодный мост VD1 — любой подходящий на ток не менее 100 мА, например КЦ407А, или четыре диода, однотипных с VD2-VD4.

Микросхема DA1 — стабилизатор напряжения на 6 В, например, КР142ЕН5Г, КР1180ЕН6, КР1157ЕН6 с любыми суффиксами, КР1180ЕН6, 78L06, 78M06, 7806. Микросхему К554СА3 (DA2) можно заменить на К521СА3 с учетом различия в цолевке. Следует иметь в виду, что почти во всех справочниках прямой и инверсный входы этих микросхем перепутаны. Транзисторы могут быть любыми кремниевыми маломощными соответствующей структуры с h_{213} не менее 50, например серий КТ3102 (VT1 и VT2) и КТ3107 (VT3-VT8).

Оптрон U1 можно использовать серии АОУ160 с любым буквенным индексом, но при установке АОУ160Б и АОУ160В сопротивление резистора R13 следует уменьшить до 220 Ом. Светодиоды HL1-HL5 — любые указанного цвета свечения, например, АЛ307НМ (зеленые) и АЛ307БМ (красные). Двухцветный светодиод HL5 может быть типа АЛС331А, КИПД18А-М, КИПД19А-М, КИПД19Б-М, КИПД37А-М, КИПД37А1-М, ЛНГ3392, можно также применить и два отдельных светодиода.

Трансформатор Т1 использован от блока питания (адаптера) «Электроника Д2-37». Он имеет вторичную обмотку 2×9 В, которая использована целиком. Напряжение на конденсаторе С1 составляет около 22 В при номинальном напряжении сети. Столь высокая его величина обеспечивает работоспособность устройства при значительном снижении сетевого напряжения, типичном для сельской местности. Можно использовать любой маломощный трансформатор с напряжением на вторичной обмотке 12...18 В, однако на сколько вольт относительно верхнего предела этого диапазона будет ниже напряжение на вторичной обмотке, на столько же килоом следует снизить сопротивление резистора R1. Это необходимо для того, чтобы длительность импульсов на коллекторе VT2 была равна требуемой. Интересно отметить, что при напряжении питания 220 В она составляет около 500 мкс (±250 мкс относительно момента перехода через ноль) и увеличивается обратно пропорционально сниже-

нию напряжения сети. При этом начало и конец импульса соответствуют мгновенному напряжению сети 25 В (по абсолютной величине).

Симистор VS1 может быть использован любой на необходимый рабочий ток и класса не ниже 4 (рабочее напряжение 400 В и более), например, ТС112-16-4, фирмы Motorola серий MAC8, MAC9, MAC12, MAC15, MAC16 с любыми буквенными индексами. Размеры ребристого теплоотвода симистора 40×45×30 мм, число ребер равно 6. Выключатель SA1 — тумблер ТЗ, он установлен на имеющемся в нижней части водонагревателя кронштейне.

Элементы индикатора включения нагревателя С3, R28, VD4, HL2 установлены на небольшой плате, закрепленной около нижнего нагревательного элемента.

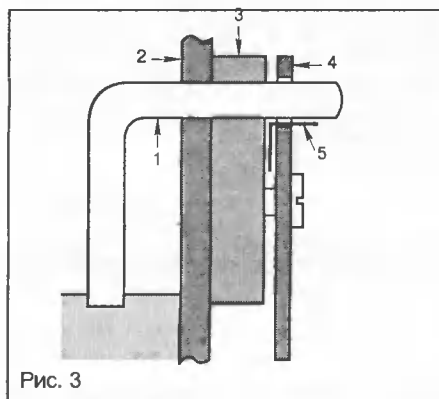


Рис. 3

Термистор R10 автор использовал ММТ-4Б. Его можно заменить на ММТ-4а. При установке R10 указанного типа, но другого номинала (до 33 кОм) сопротивление резистора R9 должно быть чуть меньше половины номинального сопротивления термистора при температуре +20°C. Термистор вставлен в рассверленный сверлом 4,2 мм латунный болт-держатель биметаллического регулятора температуры водонагревателя. Болт закреплен на стенке бака на высоте 10...15 мм выше нагревателя и в стороне от него (на уровне нижнего датчика). Корпус термистора должен быть соединен с общим проводом.

Все элементы устройства установлены под штатным кожухом водонагревателя. Для установки печатной платы сетевой выключатель и биметаллический регулятор температуры были сняты, а крепежное отверстие регулятора заглушено резиновой пробкой. К вертикальным ребрам жесткости корпуса водонагревателя винтами М3 привернуты горизонтальные бруски 8×30×42 и 8×8×65 мм из органического стекла, а к ним тремя винтами М2,5 — печатная плата. Трансформатор на плате закреплен хомутиком из латуни толщиной 0,5 мм. В нижней части кожуха просверлены 20 отверстий диаметром 7 мм, служащих для вентиляции элементов устройства. Такие же 12 отверстий выполнены в верхней части кожуха,

оставшееся от штатного выключателя отверстие ничем не закрыто.

Конструкция датчиков уровня проиллюстрирована на рис. 3. Они представляют собой Г-образные отрезки прутка диаметром 3,5 мм из нержавеющей стали. Четыре отрезка 1 разной длины пропущены в отверстия в стенке 2 бака, просверленные выше максимального уровня воды, и плотно вставлены в отверстия упомянутого выше бруска 3 (8×30×42 мм). Контакт с токоведущими дорожками печатной платы 4 осуществляется за счет узких латунных полосок 5, припаянных к плате. Высота установки нижнего среза датчиков примерно соответствует положению светодиодов на плате.

Светодиоды HL3-HL6 расположены в одну вертикальную линию, а HL1 и HL2 — вынесены из нее. Вообще говоря, один из двух последних светодиодов избыточен, но HL1 удобен при настройке, а переключение HL2 демонстрирует работоспособность всех элементов термостабилизатора и полезно для контро-

ля его исправности. Напротив светодиодов в кожухе просверлены отверстия диаметром 5 мм.

Проверяют и градуируют регулятор без симистора. Термистор помещают в воду с температурой +45°C. Его изолятор не должен быть в воде. При вращении оси резистора R7 и подходе к крайнему левому положению светодиод HL1 должен загораться. Если это не так, подбирают резистор R8. При такой регулировке нижний предел температуры составит около 25°C.

Для окончательной регулировки устройства в него наливают воду с температурой, которую необходимо стабилизировать, и подбирают положение движка R7 на грани включения светодиода HL1.

Описанный выше автомат разрабатывался для установки в электроводонагреватель с двумя нагревательными элементами — EK1 и EK2. В устройствах более позднего выпуска установлен один нагревательный элемент. Когда статья уже была подготовлена к публикации,

был изготовлен автомат для такого водонагревателя. Поскольку в устройстве с одним нагревательным элементом полезный объем воды больше (ее допустимый уровень может быть ниже), печатная плата по своей длине была увеличена до 240 мм. Рисунок платы в основном сохранен, лишь увеличен шаг установки светодиодов HL3-HL6 с 50 до 67,5 мм. Кроме того, использован трансформатор питания ТП-121-4, сопротивление резистора R1 уменьшено до 10 кОм. Из имевшихся в наличии термисторов выбран KMT-46 сопротивлением 22 кОм, при этом R6=8,2 кОм, R7=4,7 кОм, R8=13 кОм, R9=8,2 кОм, R11=1 МОм. Элементы цепи индикации включения нагревательного элемента C3, R28, VD4, HL2 установлены между светодиодами HL4 и HL5. Светодиод HL1 не выведен на лицевую сторону кожуха и после настройки его можно удалить и установить на его место перемычку.

Сергей Хвастин,
editor@dian.ru

www.platan.ru
ПЛАТАН

ЭЛЕКТРОННЫЕ КОМПОНЕНТЫ
ВАШЕГО УСПЕХА

FIGARO

ДАТЧИКИ ГАЗА (Япония)

- Обнаружение утечки токсичных и взрывоопасных газов
- Определение концентрации газов

Полупроводниковые датчики серий TGS 2, 8, 2000

- Чувствительный элемент (ЧЭ) изготовлен на основе оксида олова с использованием поверхностных эффектов
- Токсичности структуры
- TGS4160 (углекислый газ)

• ЧЭ — ячейка с твердым электролитом

Серия KE (исполнители)

- ЧЭ — гальваническая ячейка с жидким электролитом

Диапазон технических характеристик для разных моделей датчиков:

- Напряжение питания цепи: 5 В или 15 В (серия TGS 2000), 12 В или 24 В (серия TGS 2, 8)
- Потребляемая мощность нагревательного элемента: 210 - 830 мВт (серия TGS 2000), 680 - 1000 мВт (серия TGS 2, 8)

Газ	Содержание газа в окружающей среде, ppm	Тип датчика (серия 2, 8)	Тип датчика (серия 2000)
Горючие газы			
Пропан	500 - 10000 ppm	TGS813	TGS2610
Природный газ (метан)	500 - 10000 ppm	TGS842	TGS2611
Газовая смесь природных газов	500 - 10000 ppm	TGS813	TGS2610
Водород	50 - 1000 ppm	TGS821	TGS2620
Токсичные газы			
CO - угарный газ	50 - 1000 ppm	TGS203	TGS2442
NH ₃ - аммиак	30 - 300 ppm	TGS826	
H ₂ S - сероводород	5 - 100 ppm	TGS825	
Органические растворители			
Алкоголь	50 - 5000 ppm	TGS822	TGS2620
Другие летучие органические пары		TGS822	TGS2620
Фреоны			
R-22, R-113	100 - 3000 ppm	TGS830	
R-21, R-22	100 - 3000 ppm	TGS831	
R-134a, R-22	100 - 3000 ppm	TGS832	
Загрязнители воздуха внутри помещений			
CO ₂ - двуокись углерода		TGS4160	
Другие загрязнители воздуха	<10 ppm	TGS800	TGS2100
Пищевые испарения			
Летучие пищевые испарения (алкоголь)		TGS880	TGS2181
Водяной пар		TGS883T	TGS2180
Летучие пары			TGS2281
Кислород			
0-100% - задержка 12 с, отклик 90% - срок службы 5 лет		KE-25	
0-100% - задержка 96 с, отклик 90% - срок службы 10 лет		KE-50	

121351, Москва,
ул. Ивана Франко, д. 40, стр. 2
Тел./факс: (095) 73-75-999

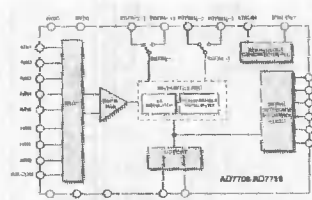
Почта: 121351,
Москва, а/я 100
E-mail: platan@aha.ru

ARGUSSOFT
Департамент Микроэлектроники

ЭЛЕКТРОННЫЕ КОМПОНЕНТЫ
ОФИЦИАЛЬНЫЙ
дистрибутор фирм:

ANALOG
DEVICES

Сигма-дельта аналого-
цифровые преобразователи



Микроконтроллер	Разрядность, бит	Макс. частота вычисления, кГц	Кол-во АЦП в корпусе	Кол-во выходов	Исходный диапазон, В	Формат вых. данных	ИОН
AD7731	24	0.4	1	5	±VREF/Gain	по ш. вых.	внешн.
AD7718	24	1.3	1	10	±VREF/Gain	по ш. вых.	внешн.
AD7736	24	1.2	1	2	±VREF/Gain	по ш. вых.	внешн.
AD1555	24	1	1	1	±2.5	по ш. вых.	внешн.
AD7714	24	1	1	5	±VREF/Gain, ±10	по ш. вых.	внешн.
AD7710/AD7712	24	1	1	2	±VREF/Gain, ±10	по ш. вых.	внешн.
AD7713	24	0.2	1	3	±VREF/Gain, 10	по ш. вых.	внешн.
AD7785	24	0.02	1	2	±VREF, ±VREF/16	по ш. вых.	внешн.
AD7782	24	0.03	1	2	±VREF, ±VREF/16	по ш. вых.	внешн.
AD7716	22	0.67	4	4	±2.5	по ш. вых.	внешн.
AD7718	16/24	0.1	2	4	±VREF/Gain	по ш. вых.	внешн.
AD7721	18	470	1	1	VREF, ±VREF/2	по ш. вых.	внешн.
AD7722	18	220	1	1	2.5, ±12.5	по ш. вых.	внешн.
AD7724	16	200	2	2	2.5, ±12.5	по ш. вых.	внешн.
AD7726	16	200	1	1	2.5, ±12.5	по ш. вых.	внешн.
AD77350	16	64	8	8	1.3±VREF	по ш. вых.	внешн.
AD7708	18	1.3	1	8	±2.5	по ш. вых.	внешн.
AD7709/6	16	0.5	1	2/3	±VREF/Gain	по ш. вых.	внешн.
AD7707	16	0.5	1	3	±VREF/Gain, ±10	по ш. вых.	внешн.
AD7718	16	0.5	1	1	±VREF/Gain	по ш. вых.	внешн.
AD7709	16	0.1	1	4	±VREF/Gain	по ш. вых.	внешн.
AD7729	15	270	2	2	±2VREF	по ш. вых.	внешн.

Хотите попробовать? Закажите бесплатные образцы у нас:

Офис в Москве:
Адрес: 129085, Москва, Пролетария Мира, 96
Тел.: (095) 217-2487, 217-2519, 217-2605
Факс: (095) 216-66-42
E-mail: info@argusssoft.ru

Офис в Санкт-Петербурге:
Адрес: 1, 1023, Санкт-Петербург, наб. кан. Гр. Талова, 36
Тел.: (812) 314-3808, тел. факс: (812) 310-8234
E-mail: info@argusssoft.ru

Интернет: WWW.ARGUSSOFT.RU

ANALOG
DEVICES

ALMEL

TRACO
POWER

BOURNS

CLARE

sumida

JE-AN

HANTRONIX

Honeywell

SII
Seko Instrument

muRata

Простая всеволновая антенна

Простую всеволновую логопериодическую антенну (ЛПА) можно собрать из четырех телескопических антенн («усов») для диапазонов МВ и антенн «Истра» или «Дельта» для диапазона ДМВ.

Схематическая конструкция такой антенны показана на рис. 1 (для диапазонов МВ — I, II, III).

Вибратор 1 принимает передачи в диапазоне МВ-I, II (ОПТ и ТВЦ) и пред-

ставляет собой телескопические «усы». Для приема передач в диапазоне МВ-III (ТВ-6, НТВ и РТР) используются три телескопических «уса», включенных по схеме ЛПА (вибраторы 2, 3, 4).

При этом вибратор 3 конструктивно видоизменен — его «усы» прививаются с противоположных сторон пластмассового основания. Это позволяет соединить все три вибратора прямой собирательной линией «а-б» из алюминиевой проволоки диаметром не больше 3 мм в укороченную ЛПА для МВ-III. «Усы» вибраторов 2, 3 и 4 раздвинуты на угол не больше 170° для того, чтобы проложить проволоки «а» и «б» собирательной линии прямо.

Штатные ленточные кабели от «усов» отсоединяют. С левого конца линии

«а-б» присоединяется кабель РК-75-4-11. По изоляционной рейке АБ размером 3х3 см (например, деревянной), на которой крепятся «усы» 1-4, кабель идет к телевизору или к сумматору МВ+ДМВ.

ЛПА диапазонов МВ-III и ДМВ («Истра») имеют усиление не меньше 4 дБ, что значительно улучшает прием ТВ-станций ВЧ-диапазона по сравнению с антенной «Диполь». В диапазоне МВ-I, II «Диполь» работает лучше, т. к. длинные волны лучшегибают поверхность земли и обычного резервного вибратора вполне достаточно.

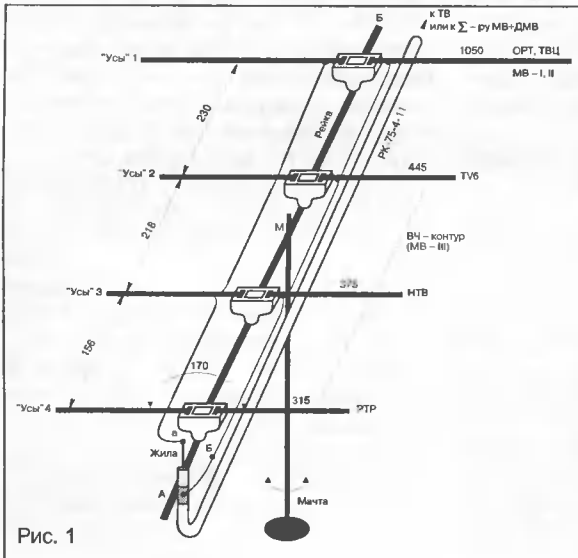
При установке антенного малошумящего усилителя SWA-7 (или SWA-9), включенного по штатной схеме питания, можно получить еще лучшие результаты по всем диапазонам. Антенна крепится к поворотной мачте в точке М.

Если удлинить собирательную линию «а-б» влево (рис. 1), то можно на ней собрать ЛПА ДМВ, размеры которой приведены в [1].

Николай Авдюнин,
editor@dian.ru

Литература

1. Канчинский Л. М. Телевизионные антенны. — М., Энергия, 1972.

**справочный листок**

(Окончание. Начало — № 4/2002)

Отечественные ИС стандартной логики. Серии 5554 и 5514

Интегральные схемы (ИС) серий 5554 и 5514 построены на основе типовых базовых матричных кристаллов (БМК). Это обеспечивает им ряд ценных качеств:

- предельно сокращает (до 1...1,5 месяцев) длительность цикла «заказ/поставка» при создании новых ИС;
- обеспечивает экономическую возможность разработки ИС при их относительно невысокой потребности, что характерно в настоящее время для ИС стандартной логики;
- исключает необходимость длительной и дорогостоящей процедуры аттестации каждой ИС на устойчивость внешним воздействием (достаточно аттестовать базовый кристалл).

Технология БМК разбивает процесс производства ИС на два этапа:

- разработка и изготовление БМК (наиболее длительная и дорогостоящая часть производства).

- разработка и изготовление ИС на основе БМК (более короткая и недорогая часть).

При разработке БМК формируется библиотека стандартных элементов (логических, аналоговых, элементов ввода/вывода и т. п.), необходимых для построения ИС определенного класса. Эти элементы определенным образом распределяются по поверхности кристалла. Выполняется полный цикл проектирования ИС кроме межсоединений элементов верхними слоями металлизации. На основе полученного таким образом БМК выполняются тестовые ИС, используемые для аттестации БМК по всем необходимым параметрам и характеристикам, включая устойчивость к внешним воздействиям. При разработке заказной ИС нужно только сформировать схему и топологию соединений стандартных элементов

верхним слоем металлизации кристалла в соответствии с требуемыми функциями ИС.

В производстве изготавливаются полуфабрикаты — кремниевые пластины с сформированными полупроводниковыми структурами библиотечных элементов.

При изготовлении определенной ИС берется соответствующее число подготовленных пластин, и на их поверхности формируются два верхних слоя металлизации, соединяющие ранее изготовленные библиотечные элементы в соответствии со схемой изготавливаемой ИС. Процесс завершается стандартными процедурами проверок и корпусирования ИС.

При изготовлении ИС серий 5554/5514 в настоящее время используется три типа БМК:

- БМК-1 содержит 88 четырехтранзисторных логических вентилей и 18 схем ввода/вывода с тремя состояниями и током нагрузки до 12 мА (цифра 1 после букв БЦ в обозначении ИС);
- БМК-2 содержит 96 четырехтранзисторных логических вентилей и 18 схем ввода/вывода с тремя состоя-

ниями и током нагрузки до 24 мА (цифра 2 после букв БЦ в обозначении ИС);

- БМК-3 содержит 200 четырехтранзисторных логических вентилях, 22 схемы ввода/вывода с тремя состояниями и током нагрузки до 24 мА (цифра 3 после букв БЦ в обозначении ИС).

БМК производятся на кремниевых пластинах диаметром 150 мм по современной КМОП технологии с кремниевыми затворами и двумя карманами с двухуровневой коммутацией из сплавов алюминия. Конструкция транзисторов и технология производства обеспечивают высокую устойчивость схем серии к внешним воздействиям.

Предлагаемые ИС отличаются от серий 54АС/54НС (1554/1564) следующими характеристиками:

- быстродействием — серия 5514 превосходит зарубежные и отечественные аналоги, серия 5554 не уступает им;
- более высокой нагрузочной способностью — серия 5514 превосходит зарубежные и отечественные аналоги, серия 5554 не уступает им;
- возможностью внутреннего согласования заданных входов ИС с ТТЛ уровнями входных сигналов;
- возможностью реализации заданных входов ИС с триггером Шмитта;

- возможностью реализации заданных входов и выходов ИС с триггерной петлей;
- возможностью реализации заданных входов ИС без диода на шину питания.

При заказе новых ИС потребитель имеет возможность использовать эти дополнительные свойства в различных их сочетаниях.

Возможность внутреннего (без дополнительных внешних элементов) согласования по уровням входных сигналов ИС серий 5554/5514 с ТТЛ приборами обеспечивает их совместное применение в аппаратуре. Новые ИС могут либо полностью, либо по части входов соответствовать микросхемам серий 54АС/54НС (1594/5564).

При наличии на входе ИС триггера Шмитта существенно повышается помехоустойчивость, что важно при приеме управляющего сигнала, поступающего с длинной линии.

Высокоомная триггерная петля, подключенная ко входу или к мощному выходу с третьим состоянием, позволяет поддерживать шину (магистраль), соединенную с этими элементами, в предвещаемом квазиактивном состоянии. При этом обмен по шине производится в штатном режиме. Такое решение, при котором все управляющие шиной выходы находятся в состоянии высокого

импеданса, снимает проблему с появлением на шине лишних импульсов из-за внешних наводок.

Отключение диода на шине питания, обычно присутствующего на входах КМОП ИС, позволяет использовать данный вход (входы) в режиме преобразования уровней для понижения напряжения питания и амплитуды выходных сигналов для группы ИС.

Большой опыт изготовления ИС на заказ показал, что на указанных кристаллах БМК реализуются блоки компараторов напряжения с чувствительностью не хуже 50 мВ и временем реакции не более 10 нс. Такие блоки вместе со стандартными логическими элементами могут упростить разработку аппаратуры с не очень жесткими требованиями к обработке аналоговых сигналов.

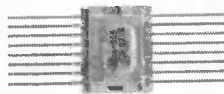
В настоящее время в состав серии 5554 входит 37 ИС, серии 5514 — 24 ИС, однако это только начало. Потребитель имеет возможность заказать необходимую ему ИС как имеющую аналог в сериях 54АС/54НС (1554/1564), так и оригинальную, и по истечении 1...1,5 месяцев он получит первую партию разработанных и изготовленных для него ИС.

Борис Малашевич,
Валерий Краюшкин
mbm@angstrom.ru



Отечественные ИС стандартной логики

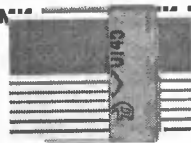
Серия 5554



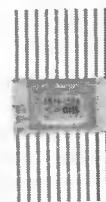
Серия 5514

Две новые серии быстродействующих КМОП логических ИС — аналогов всемирно известных, выпускаемых многими мировыми фирмами:

54АС (1554)



54НС (1564)



Состав серий 5554/5514:

- полные аналоги ИС серий 54АС/54НС (1554/1564),
- новые ИС, не имеющие аналогов в выпускаемых сериях.

Диапазон рабочих температур $-60 \div +125^\circ\text{C}$,
Допустимое значение потенциала статического электричества — 2000 В.

Современная система схемотехнического, топологического и технологического автоматического проектирования, стандартные технология производства и верификации проектов, аттестация предприятия на соответствие требованиям ISO 9001

позволяют Ангстрему сократить цикл «заказ — поставка» до $1 \div 1,5$ месяцев и оперативно выполнять заказы на разработку и поставку новых ИС стандартной логики, как оригинальных, так и имеющих аналоги в сериях 54АС/54НС (1554/1564).

103460, Москва, Зеленоград, ОАО Ангстрем, т. (095) 531-49-06, ф: (095) 532-96-21, E-mail: market@angstrom.ru, http://www.angstrom.ru

Таблица 1

Параметры контроллеров	NM6011	NM6012
Напряжение питания	4...6 В	5...30 В
Ток потребления в режиме ожидания	<0,001мА	<0,15 мА
Кратковременный ток потребления в режиме открывания	<100 мА 2...6 с	<3 А 0,4 с или <1,5 А 10 с
Тип ключа	DS1990	
Количество комбинаций кодов ключей	280 000 000 000 000	
Количество одновременно зарегистрированных ключей	до 16	
Размер печатной платы	30х80 мм	

Таблица 2

Позиция	Номинал	Примечание	Колич.
R1, R5	100 Ом	0,125 Вт	2
R2, R4, R6, R8, R9	4,7 кОм	0,125 Вт (R8 нет в NM6012)	5
R3, R10	1 кОм	0,125 Вт (R10 нет в NM6012)	2
R7	100 кОм	0,125 Вт	1
R11	150 Ом	0,125 Вт	1
R6, R9	47 кОм	0,125 Вт	2
R7	100 кОм	0,125 Вт	1
C1	1000 мкФ/35В	Размер 1220 или 1320	1
C2, C4	0,1 мкФ	Керамический	2
C3, C5	33 пФ	Керамический 27...33 пФ	2
C6	100 мкФ/16В	100 мкФ/25В	1
DA1	LP2950 Z5	1187EH1 (нет в NM6011)	1
DD1	AT90S2313-10PI	Прошивка 6011.x	1
L1	1 мГн	Дроссель малогабаритный (нет в NM6012)	1
KN1, KN2	TS-A2PS-130	Микрокнопки TS-A1PS-130, TS-A3PS-130 (нет в NM6012)	2
VT2	BS170	(нет в NM6012)	1
VT3	BC557B	BC558B	1
VT4	BC547B	BC548B	1
VD1	1N4001	1N4002-1N4007	1
VD2	1N4933	1N4934, 1N4935 (VD2 нет в NM6012)	1
VD3, VD4	1N4148		2
VD5	BZX55C 33V	Стабилитрон 33В (нет в NM6012)	1
HL1	SC05-11	Индикатор сегментный KGB	1
BQ1	4 МГц	Резонатор кварцевый	1
Панель цанговая	TRS-10	10-и контактная	1
Плата печатная	A6011	30х80 мм	1

ровать ключи. Использование микропроцессорного управления питанием электромагнита замка обеспечивает надежную его работу при снижении напряжения питания до 3,5...4 В.

Контроллер может работать также и с другими типами замков и источниками питания. Для этого предусмотрено несколько вариантов исполнения, отличающихся установленными компонентами и алгоритмом работы. Подробное описание вариантов дается далее в разделах «Конструкция» и «Работа устройства».

Конструкция

В обеих модификациях контроллеров NM6011 и NM6012, отличающихся по комплектации, подаче питания и алгоритму функционирования используются одинаковые печатные платы и прошивки процессора. Алгоритмы функционирования изменяются в зависимости от установленных компонентов, что определяется микропроцессором автоматически. Контроллеры выполнены на односторонней печатной плате из фольгированного стеклотекстолита. Детали, за исклю-

чением индикатора и микрокнопок, устанавливаются обычным образом. Микрокнопки устанавливаются со стороны печати после монтажа остальных деталей. Индикатор устанавливается со стороны деталей, но его установка имеет некоторые особенности. Так как выводы индикатора находятся под его корпусом, то непосредственно распаять его на печатную плату весьма сложно. Он может монтироваться одним из двух способов, показанных на рис. 4. При установке по способу а — в отверстия для индикатора со стороны печати вставляются цанговые зажимы, извлеченные из панели, входящей в набор. Затем снаружи они аккуратно пропаиваются, и в них вставляется индикатор. При установке по способу в — в отверстия вставляются кусочки проволоки, согнутые в виде буквы Г. Затем они припаиваются к печатной плате, после этого устанавливается индикатор, и его ножки припаиваются к проволочкам.

При установке платы контроллера в замок типа EL370 (EL301) необходимо удалить по перфорации угол печатной платы и использовать для крепления контроллера два отверстия ЖЗ в верхней части платы. Контроллер крепится при помощи двух саморезов ЖЗ×12 и двух шурупов ЖЗ×5 на стандартной пластмассовой колодке замка. Для крепления контроллера при его использовании с другими типами замков может использоваться дополнительное отверстие в углу платы. Пример установки платы в замок типа EL370 показан на рис. 5.

Считыватель ключей представляет собой два обычных контакта. Можно приобрести готовый узел, или сделать его самостоятельно. Один из вариантов считывателя показан на рис. 6.

Работа устройства

Работа контроллера различается в зависимости от его варианта.

NM6011. Этот вариант предназначен для режима батарейного питания со сверхмалым потреблением и режимом наработки на отказ. Контроллер собирается по схеме рис. 1, но в этой модификации не устанавливается микросхема DA1. Напряжение питания $V_{cc1} = 4...6$ В подается на контакты X1, X7. Электромагнит открывания замка подключается к контактам X3, X6. К контактам X5, X8 можно подключить светодиод или динамическую головку для индикации режима открывания. К контактам X9, X4 подключается считыватель ключей.

После подачи питания микропроцессор опрашивает состояние внешних цепей, определяет конфигурацию устройства и в зависимости от этого настраивает

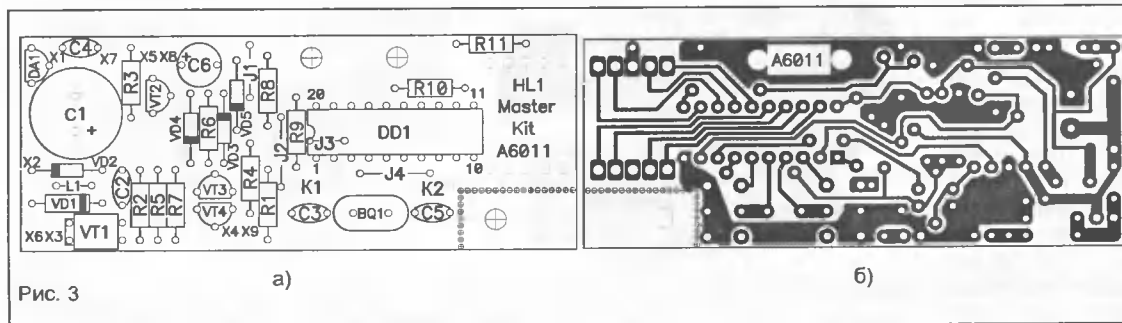


Рис. 3

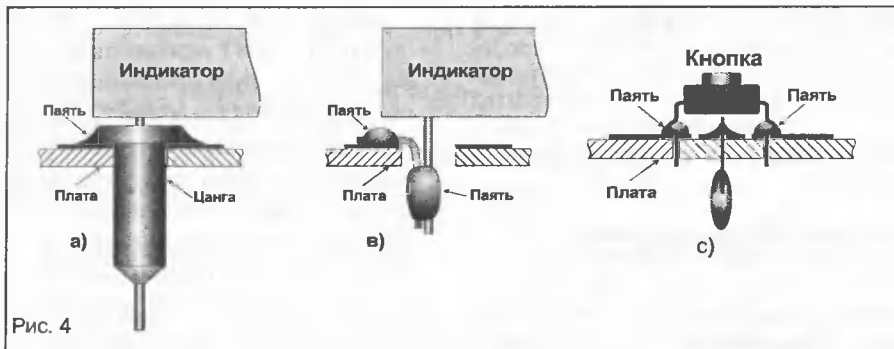


Рис. 4

режимы своих внешних портов и переходит в спящий режим. В этом режиме тактовый генератор микропроцессора выключен, и все узлы находятся в режиме микропотребления. Схема контроллера построена таким образом, что остальные узлы также не потребляют тока питания. В таком состоянии весь микроконтроллер потребляет ток порядка 1 мкА.

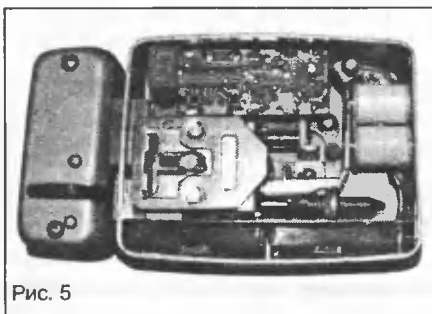


Рис. 5

При подключении ключа к контактам считывателя, он вырабатывает «импульс присутствия» — короткий отрицательный импульс, извещающий микропроцессор о подключении ключа. Этот импульс переводит триггер, собранный на транзисторах VT3 и VT4, в состояние лог. 0. В этом состоянии триггер находится в течение времени, необходимого для выхода микропроцессора из спящего состояния. После перехода в активное состояние микропроцессор производит чтение кода подключенного ключа, проверяет его на верность и сравнивает его с кодами разрешенных ключей. Если определен разрешенный ключ, то микропроцессор переходит к процессу накачки энергии в накопительный конденсатор C1. Процесс накачки состоит из ряда циклов. В цикле накачки на короткий интервал времени открывается ключ накачки на полевом транзисторе VT2. Происходит линейное нарастание тока через дроссель L1. Затем ключ закрывается, и накопленная энергия из катушки L1 через диод VD2 переходит в накопительный конденсатор C1. После этого микропроцессор через вывод 11 (PD6) проверяет уровень накопленного напряжения. Если он недостаточен,

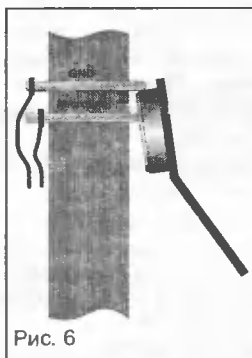


Рис. 6

то цикл накачки повторяется. Когда напряжение на конденсаторе C1 достигнет значения 35 В, открывается стабилитрон VD5, и на выводе 11 появляется напряжение лог. 1. Микропроцессор прерывает процесс накачки и на 0,4 секунды открывает силовой ключ на полевом транзисторе VT1. Энергия, накопленная в конденсаторе, поступает в электромагнит открывания замка, и замок разблокируется. Из-за использования режима накачки с контролем величины заряда, энергия, подаваемая в электромагнит, не зависит от состояния батареи. Кроме того в микроконтроллере отсутствует эффект внезапного отказа при истощении батареи. Среднее значение тока потребления в процессе накачки не превышает 0,1 А. Время накачки зависит от состояния батареи и составляет от 1,5 для 6 В до 8 с для 3,5 В. Для пользователя время накачки — это время от момента прикосновения ключа к считывателю до момента разблокирования замка. Это время показывает состояние батареи, и при его увеличении до 5...6 с необходимо заменить батарею или зарядить аккумуляторы. Для удобства пользования можно подключить к контактам X5, X8 контрольный светодиод и вывести его к считывателю. Этот светодиод будет гореть во время процесса накачки, индицируя работу устройства.

Вместо светодиода можно подключить миниатюрную динамическую головку, тогда работа контроллера будет подтверждаться звуковым сигналом.

При нажатии на кнопку «Select» микропроцессор также выходит из спящего состояния и переходит в режим установки. В этом режиме после выхода из спящего состояния, микропроцессор пытается произвести чтение кода

ключа. Если ключ присутствует и код прочитан верно, производится его сравнение с кодами, хранящимися в памяти. Если такой код уже присутствует в памяти, то микроконтроллер выдает на индикатор HL1 номер ключа, а если такого ключа в памяти нет, то на короткое время на индикаторе высвечивается знак «—», после чего процессор переходит в режим ожидания регистрации ключа. Подробнее порядок работы с

контроллером в режиме установки описан в разделе «Инструкция по использованию».

Элементы R1, VD3, VD4 обеспечивают защиту входных цепей считывателя от статических разрядов. Диод VD1 защищает ключевой транзистор VT1 от выбросов напряжения на катушке электромагнита замка.

Расчет показывает, что при десяти открываниях в день за год контроллер израсходует около 0,25 А/ч емкости батареи. При типичной емкости батарей размера AAA в 1 А/ч очевидно, что время работы комплекта батарей будет ограничиваться временем их хранения, поэтому при выборе батарей для использования в электромеханическом замке рекомендуются выбирать батареи с максимальным допустимым сроком хранения. Для подстраховки на случай отказа батарей можно вывести контакты питания наружу, например, под гвозди обивки. Тогда появляется возможность подключить батарею снаружи и открыть замок.

NM6012. Этот вариант предназначен для режима обычного питания с малым потреблением. Контроллер собирается по схеме на рис. 1, но в этой модификации не устанавливаются следующие элементы: транзистор VT2, диоды VD2 и VD5, дроссель L1, резисторы R8 и R10. Резистор R4 устанавливается в зависимости от требуемого режима работы контроллера. Если резистор не установлен, то при срабатывании контроллер подает на отпирающий электромагнит замка импульс тока длительностью 0,4 секунды (для замков со спусковым устройством). Если резистор R4 установлен, то при срабатывании контроллер подает на механизм замка напряжение в течение 10 с, одновременно подавая сигнал подтверждения с частотой около 2 кГц на выход X5. К этому выходу можно подключить контрольный светодиод или динамическую головку. Этот режим удобен для замков с электромуфтами.

Питание $V_{cc2} = 5...30$ В подают на контакты X2, X7. В остальном работа контроллера NM6012 не отличается от NM6011.

Инструкция по использованию

Тот факт, что в микропроцессорных устройствах большинство функций реализуется на программном уровне, потребовал введения в статью раздела, описывающего процедуры использования электронного замка.

Общие положения

Все операции по регистрации, контролю, активации/деактивации ключей выполняются при помощи двух кнопок «Select» и «Set», расположенных на плате контроллера, установленной внутри замка. Расположение кнопок на плате показано на рис. 2, б. Все операции необходимо выполнять с паузами между нажатиями на кнопки не более

2 с. В противном случае контроллер сбрасывается, и операцию требуется начинать сначала.

Как зарегистрировать новый ключ

Для регистрации нового ключа приложите ключ к считывателю, затем, нажимая кнопку «Select», выберите номер для ключа и нажмите кнопку «Set». Признаком правильной регистрации — загорание десятичной точки на индикаторе.

Как деактивировать/активировать ключ

Для активации/деактивации ключа, нажимая на кнопку «Select» выберите номер ключа, который надо активировать/деактивировать, затем нажмите кнопку «Set». Признаком активного ключа — свечение точки внизу справа от номера ключа. При каждом последующем нажатии ключ будет активироваться/деактивироваться. Если ключ не активируется, значит под этим номером не зарегистрировано какого-либо ключа.

Как проверить номер ключа

Для того, чтобы проверить, под каким номером зарегистрирован ключ, нужно приложить ключ к считывателю

и нажать кнопку «Select». На индикаторе высветится номер, под которым зарегистрирован ключ. Если ключ не зарегистрирован, то на 0,4 с высветится знак «-», а затем номер 0, что означает приглашение зарегистрировать ключ.

Как стереть номер ключа

Если Вы хотите по каким-либо причинам стереть в процессоре информацию о ключе, то для этого необходимо под тем же номером зарегистрировать любой другой ключ. Для стирания допускается один и тот же ключ регистрировать в контроллере под несколькими номерами.

Заключение

Так как описанный контроллер является элементом системы безопасности, то для повышения степени ее защищенности было принято решение не распространять его внутреннюю микропрограмму. Но в магазинах по продаже электронных компонентов по индивидуальному заказу радиолюбители смогут приобрести отдельно микропроцессоры с уже зашитой программой, защищенной от считывания.

Вячеслав Чулков,
Slava1355@mail.ru

МАСТЕР КИТ подготовил два набора для радиолюбителей NM6011 и NM6012, которые позволят всем желающим сделать самостоятельно электронные замки. Каждый набор включает в себя печатную плату с нанесенной маркировкой, все необходимые компоненты, микропроцессор с зашитой программой и подробную инструкцию по сборке.

Данная разработка — еще один пример реализации концепции: превратить «системы для профессионалов» в «системы для всех».

Подробную информацию по всему ассортименту смотрите на сайте www.masterkit.ru и в каталоге «МАСТЕР КИТ».

Спрашивайте электронные наборы и модули в магазинах радиодеталей Вашего города.

НАШ АНОНС

Издательский дом «Скимен» готовит к выпуску книгу Александра Фрунзе «Микроконтроллеры? Это же просто!». Данное издание, в первую очередь, ориентировано на тех, кто только начинает или собирается начать знакомство с микроконтроллерной техникой, и тех, кому еще предстоит стать потребителем чипов, КИТов, программаторов и иных средств поддержки разработки. Для них — первый том, в котором последовательно и доступно для начинающих рассказано о внутреннем устройстве микроконтроллера, о принципиальных схемах сопряжения его с внешними микросхемами, АЦП, ЦАП, устройствами индикации, кнопками управления, о системе команд и технике программирования на ассемблере. В книге приводится большое количество отработанных практических примеров, причем как отдельных подпрограмм, так и полностью законченных программ. Все программы снабжены комментариями и тщательным объяснением принципов их построения, а также используемых в них характерных приемов программирования. Не оставлены без внимания техника отладки программ и их занесения отлаженных программ в микроконтроллеры. Подобных книг, в которых на доступном для начинающих уровне рассмотрены абсолютно все аспекты, необходимые для работы с микроконтроллерной техникой, в нашей стране не выпускалось уже около 20 лет.

На более подготовленных специалистов ориентированы второй и третий тома книги. Во втором томе приводится большое количество обзорной информации по абсолютному большинству x51-совместимых микроконтроллеров, о модернизации систем на их основе, о том, как связать микроконтроллерные системы друг с другом и с компьютером, о технике использования программных симуляторов и внутрисхемных эмуляторов и о некоторых практических аспектах использования новейших микроконтроллеров семейств AduC, Sigнал и др. Дополнительно приведена подробная информация о таких сложных для программиста контроллерно-ориентированных микросхемах, как сигма-дельта АЦП семейства AD77xx от Analog Devices.

Третий том целиком посвящен практически не описанным в отечественной литературе принципам построения программ целочисленной многобайтовой беззнаковой и знаковой арифметики, а также арифметики с плавающей запятой (ПЗ). Приведены тщательно описанные и снабженные большим количественным комментарием отлаженные подпрограммы, формирующие законченные пакеты целочисленной арифметики и арифметики с ПЗ. Отдельно рассмотрены специальные быстрые алгоритмы расчета различных функций, общие методы аппроксимации с заданной точностью любых разумных функциональных зависимостей, что позволяет вести быстрые расчеты даже на микроконтроллерах с производительностью в единицы MIPS. Книг такого уровня, посвященных этой теме, в бывшем СССР не издавалось вообще. Несмотря на то, что упомянутые программы написаны на языке микроконтроллеров семейства x51, тщательное описание их устройства позволяет подготовленным специалистам легко перенести программы на любой микроконтроллер.

Отдельно отметим, что приведенные в книге практические примеры используют, помимо микроконтроллеров, большое количество аналоговых и цифровых микросхем таких известных производителей, как Analog Devices (AD77xx, AD7894, AD855x, REF19x и ряд других), Maxim (MAX202/232, MAX680, MAX619, MAX187/189/1241, 1243 и им подобные), Vup-Brown (ADS7816 и им подобные, DDC112 и т. д.).

ПОДПИСКА - 2002

Журнал «КОМПОНЕНТЫ и ТЕХНОЛОГИИ»

www.compitech.ru

Через Объединенный каталог «Пресса России» (зеленый) — индекс 41734
 для жителей России, стран Балтии и СНГ.
 Через каталог Агентства «Роспечать» (красный) — индекс 80743.
 Жители Украины могут подписаться через каталог агентства
 KSS (044-212-0050, 464-0220) — индекс 10358.

Журнал «СХЕМОТЕХНИКА»

www.dian.ru

Через Объединенный каталог «Пресса России» (зеленый) — индекс 41733
 для жителей России, стран Балтии и СНГ.
 Через каталог Агентства «Роспечать» (красный) — индекс 80724.
 Жители Украины могут подписаться через каталог агентства
 KSS (044-212-0050, 464-0220) — индекс 10540.

НАШИ РЕКВИЗИТЫ:
ООО «ИД СКИМЕН»
ОСБ КИЕВСКОЕ 5278 ИНН 7731195492
Р/с 40702810338190104019
в Сбербанк России в г. Москва
К/с 30101810400000000225
БИК 044525225
ОКОНХ 84500 ОКПО 52744508

Комплект журнала «Схемотехника» №№3-12 за 2001 г. (стоимость доставки включена) — 275 р., в т. ч. НДС — 10% (РФ); 25 у. е. (страны СНГ, Балтии) *с НДС	Комплект журнала «Компоненты и Технологии» №№5-8 за 2001 г. (стоимость доставки включена) — 176 р., в т. ч. НДС — 10% (РФ); 12 у. е. (страны СНГ, Балтии) *с НДС
Комплект журнала «Схемотехника» №№ 1-12 за 2002 г. (стоимость доставки включена) 330 р., в т. ч. НДС — 10% (РФ); 30 у. е. (страны СНГ, Балтии) *с НДС	Комплект журнала «Компоненты и Технологии» №№ 1-9 за 2002 г. (стоимость доставки включена) — 495 р., в т. ч. НДС — 10% (РФ); 33 у. е. (страны СНГ, Балтии) *с НДС

УСЛОВИЯ ПОДПИСКИ ЧЕРЕЗ РЕДАКЦИЮ:

- перечислите деньги на наш расчетный счет через Сбербанк по квитанции;
- отправьте квитанцию об оплате (или копию) и свой точный почтовый адрес (индекс обязательно) в редакцию по адресу: 121351, г. Москва, ул. Ивана Франко, дом 40, стр. 2, отдел подписки или по факсу (095) 777-1215, E-mail: podpiska@dian.ru
- можно также подписаться на журнал непосредственно в редакции по адресу: г. Москва, ул. Обручева, 29, м. «Калужская», вн. тел: 3629 и на специализированных выставках.

Информация о подписчике:

Название организации _____ Ф.И.О. _____
 Почтовый адрес, индекс _____
 Телефон/факс _____ E-mail _____
 Подписка на журнал _____ №№ _____ за 200 _____ г.

Внимание! К сведению подписчиков:

Отдел подписки не несет ответственности, если подписка оформлена через другие фирмы.
 В случае отмены заказчиком произведенной подписки деньги за подписку не возвращаются.