Содержание

Благодарности

Я хотел бы поблагодарить всех авторов, чьи материалы вошли в состав данной книги. Это был огромный труд, но наконец-то все готово!

Также спасибо Кэрол Льюис, редактору издательства Newnes по новым проектам, и ее мужу Джеку, которые терроризировали меня целый год, пока я не пришел к такой концепции книги, которая, как мне кажется, должна быть понятна и читателям, и авторам. Кэрол положила начало моей авторской карьере более десяти лет назад, подвигнув меня на написание книги «*The Art of Programming Embedded Systems*» («Искусство программирования встраиваемых систем»). Спасибо ей за участие на протяжении всех этих лет.

И особая благодарность – моей жене Мэрибет за помощь и поддержку, а также за колоссальный объем работы по форматированию материалов и административному руководству проектом.

– Джек Ганссл,
пристань Анкоридж,
г. Балтимор, штат Мэриленд

Введение

Вот оно – «Руководство по микропрограммному обеспечению»! Эта книга заполняет важнейший пробел в литературе по встраиваемому ПО. Существует настоятельная потребность в сборнике идей и концепций, справочнике, настольной книге инженеров, куда они заглядывали бы, чтобы найти решение своих задач и освежить в памяти забытый материал. Главной темой этой книги является микрокод, однако суровая реальность мира встраиваемого ПО такова, что код и аппаратура взаимозависимы. Они не могут существовать в изоляции; ни в одной другой области программирования нет такой глубокой связи между реальным и виртуальным.

Аналоговые инженеры постоянно твердят, что у них прекрасная профессия. Конечно, очень здорово ворочать операционными усилителями. Но бедняги не ведают, как это увлекательно – сделать так, чтобы все двигалось, огоньки мигали, газ тек. Это мы, разработчики встраиваемого ПО, управляем работой моторов, перекачиваем кровь, приводим в действие автомобильные тормоза, контролируем развертку телевизионного изображения по горизонтали и по вертикали и выдвигаем компакт-диски из дисководов. Что может сравниться по притягательности с этой размытой границей между микрокодом и реальным миром?

Книга адресована разработчикам микрокода, пишущим те самые программы, на которых работают технологии XXI века. Эта книга создавалась не как учебник или вводный курс по написанию микрокода. Существует множество других пособий, цель которых – научить людей азам разработки встраиваемых систем. Не является она и введением в основы программирования. Каждый разработчик должен знать, что такое конструктивная стоимостная модель (COCOMO) Бёма, используемая для оценки затрат на разработку ПО; методы экстремального программирования; подходы Фагена и Гильба к инспектированию программного обеспечения; персональный метод разработки по Хамфри; модель оценки зрелости процессов разработки и сопровождения программного обеспечения, созданная в институте Software Engineering Institute, и ряд других хорошо известных и постоянно развивающихся методов. Тщательное программирование – ключевая составляющая успеха при создании любой большой системы, и мир встраиваемого ПО в этом не оригинален.

В данном руководстве очень мало информации по операционным системам реального времени, TCP/IP, цифровым сигнальным процессорам и другим подобным темам. Эти актуальные вопросы крайне важны для огромного множества встраиваемых приложений. Однако для освещения каждого из таких вопросов требуется отдельное издание – и таких книг уже очень и очень много.

РАЗДЕЛ СТРАНИЦА

I Основы оборудования

Введение

Первые электронные вычислительные машины были аналоговыми, в действительности они представляли собой совокупность операционных усилителей и ничего более. Того, что мы сейчас называем «программами», вообще не существовало, вместо них разработчики алгоритмов использовали массивы электронных компонентов, помещавшихся в петли обратной связи таких усилителей. Программирование сводилось в буквальном смысле к переключению проводов в машине. Только инженеры, хорошо разбиравшиеся в электротехнике, могли управляться с подобными монстрами.

С появлением в 1940-х гг. цифровых компьютеров, способных записывать информацию, программы превратились из электрических схем в биты, хранящиеся на различных носителях... хотя это было совершенно не похоже на то, что мы имеем сегодня! Менее очевидным преимуществом стало абстрагирование программиста от машины. Цифровая сущность вычислительных машин преодолела ограничения электрических параметров – больше не существовало ничего, кроме нуля и единицы. Возможности машинных вычислений открылись для огромной группы людей, гораздо более широкой, чем специалисты по электротехнике. Программирование стало самостоятельной дисциплиной со своими профессиональными приемами, ни один из которых не требовал даже самых элементарных знаний электроники и схемотехники.

Единственное исключение – встраиваемые системы. Изобретение микропроцессора корпорацией Intel (1971 г.) вернуло компьютеры назад – к самым истокам. Вычислительные машины сразу стали достаточно доступными по цене и небольшими по размерам, чтобы их можно было встраивать в какую-либо продукцию. Несмотря на низкую стоимость процессора, снижение общей стоимости систем снова оказалось проблемой для проектировщиков *и* программистов новой интеллектуальной продукции.

Это последний рубеж вычислительной техники, на котором оборудование и микропрограммное обеспечение (микрокод, firmware) не отделимы друг от друга. Зачастую можно понизить стоимость системы, используя компоненты с более интеллектуальной программой, или повысить производительность, применив другой способ крепления. Наши микропрограммы тесно связаны с особенностями периферийного оборудования и датчиков, а также со множеством явлений реального мира. Хотя и существует тенденция приглашать на работу «чистых» программистов, самыми лучшими разработчиками всегда оказываются те, кто имеет не только богатый опыт в разработке программного обеспечения, но и практические навыки в области электроники.

Каждый разработчик встраиваемого ПО должен – нет, *просто обязан*! – знать содержание этой главы. Вы не имеете права заявить: «Ну, поскольку я программист-разработчик микропрограммного обеспечения, мне не нужно знать, что такое резистор». Ваша задача как разработчика *встраиваемого* ПО – создание системы, которая будет больше, чем просто программным кодом.

Более глубокие сведения по всем аспектам электроники содержатся в замечательной книге «*The Art of Electronics*» (русский перевод: Хоровиц П., Хилл У. Искусство схемотехники: в 2 т. М.: Мир, 1986).

Джек Ганссл – автор ежемесячной колонки «Breakpoints» в разделе «*Embedded Systems Programming*» электронного еженедельника «Embedded Pulse», выходя-

щего на веб-сайте embedded.com; он написал четыре книги по встраиваемым системам и одну о своем неудачном опыте мореплавателя. Он начал заниматься встраиваемыми системами в начале 70-х гг. еще на процессоре 8008. С тех пор успел основать и продать три электронные компании, в том числе одно из крупнейших предприятий по созданию инструментальных средств для разработки встраиваемого ПО. Как программист и руководитель Джек принимал участие в работах по созданию более 100 встраиваемых продуктов: от глубоководного навигационного оборудования до системы безопасности Белого дома. В настоящее время он ведет семинары, посвященные наилучшим способам создания встраиваемых систем, для компаний по всему миру.

Глава 1. ОСНОВЫ ЭЛЕКТРОНИКИ

Джек Ганссл

Цепи постоянного тока

Постоянный ток (DC) – замечательный термин, обозначающий сигналы, не изменяющиеся со временем. Здесь отсутствуют пики, как на электроэнцефалограмме мертвого мозга или на выходе аккумуляторной батареи. Источник питания вашего ПК получает постоянный ток из переменного тока (AC) обычной электросети.

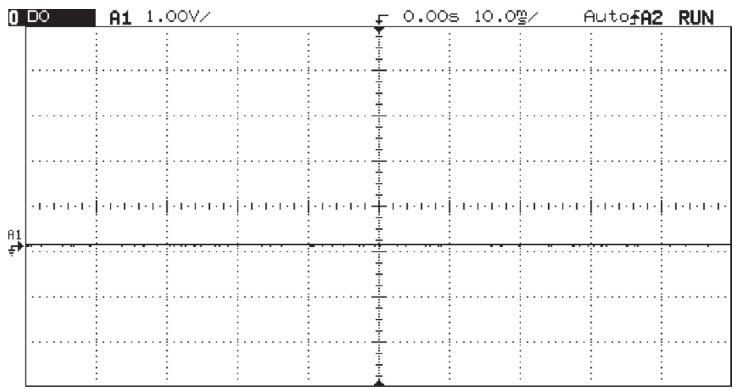

Рис. 1.1. Постоянный ток характеризуется сигналом постоянной, неменяющейся амплитуды

Напряжение и сила тока

Для характеристики электрического тока обычно пользуются напряжением и силой тока, не задумываясь, что за обеими величинами стоят фундаментальные закономерности физики. Атомы, имеющие недостаток или избыток электронов, называются *ионами*. Ионы могут иметь как положительный, так и отрицательный заряд. Два иона с противоположным зарядом (один положительный, что означает нехватку у него электронов, а другой отрицательный, обладающий одним или более дополнительным электроном) притягивают друг друга. Сила такого притяжения называется электродвижущей силой и обычно обозначается как ЭДС[1].

Заряды измеряются в кулонах (Кл), 1 кулон равен заряду $6{,}25\times10^{18}$ электронов (для отрицательных зарядов) или протонов (для положительных).

Ампер (А) – это заряд в 1 Кл, протекающий через данную точку за одну секунду. Напряжение 1 В – это разность потенциалов, благодаря которой проводник с текущим по нему током в 1 ампер совершит работу в 1 джоуль. Джоуль в секунду называется ваттом.

---

[1] Вообще говоря, сила взаимодействия точечных зарядов называется кулоновской, а под термином ЭДС понимается физическая величина, равная отношению работы сторонних сил по перемещению заряда вдоль замкнутого контура к величине этого заряда. – *Прим. пер.*

Следует заметить, что на самом деле редкие специалисты по электротехнике помнят эти определения и практически никто ими не пользуется.

Рис. 1.2. Даже старомодный недорогой аналоговый вольт-омметр, подобный изображенной на рисунке модели Radio Shack, измеряет постоянный ток, как минимум, не хуже осциллографа

Старинная, но очень удачная аналогия представляет электрический ток как воду, текущую по трубе: в этом случае сила тока будет количеством воды, пропускаемой трубой в единицу времени, а напряжение – давлением воды. Хотя обычно сила тока измеряется в амперах, для компьютера значение силы тока 1 А является крайне высоким. Большинству цифровых и аналоговых цепей требуются гораздо меньшие токи. В табл. 1.1 приведен перечень наиболее употребительных единиц силы тока.

Таблица 1.1. Единицы измерения силы тока

Название	Сокращенное обозначение	Значение [А]	Где встречается
Ампер	А	1	Источники питания; наиболее высокопроизводительным процессорам может потребоваться ток силой во многие десятки ампер
Миллиампер	мА	0,001	Логические схемы, процессоры (десятки или сотни мА), обычные аналоговые схемы
Микроампер	мкА (μА)	10^{-6}	Экономичные логические и аналоговые схемы, ОЗУ с резервным питанием от батарей
Пикоампер	пкА	10^{-12}	Очень чувствительные аналоговые входные цепи
Фемтоампер	фА	10^{-15}	Измерение аналоговых сигналов с помощью новейших технологий

Для большинства встраиваемых систем характерен не столь экстремальный диапазон напряжений. Типичные источники питания для логических схем и микропроцессоров обеспечивают напряжение от одного–двух до пяти вольт. Напряжение источников питания аналоговых устройств редко выходит за преде-

лы диапазона плюс или минус 15 В. Некоторые аналоговые сигналы от датчиков могут достигать значений милливольтового (0,001 В) диапазона. Радиоприемники способны обнаруживать сигналы микровольтового уровня, однако для этого используются довольно сложные методы шумоподавления.

Резисторы

Когда электроны движутся по проводам, внутри компонентов или – при несчастном случае – через тело человека, они встречают *сопротивление*, то есть стремление проводника ограничить ток электронов. Совершенным резистором является вакуум – через него электрический ток вообще не течет. Сходными характеристиками обладает и воздух, но влажный воздух может в некоторой степени проводить электрический ток, так как вода – достаточно хороший проводник.

Сверхпроводники, единственные материалы с нулевым сопротивлением, обладают этим свойством благодаря волшебной силе квантовой механики при крайне низких температурах – порядка температуры кипения азота или ниже. Остальные материалы, даже самые лучшие проводники, имеют сопротивление. Потрогайте шнур питания своего 1500-ваттного керамического нагревателя – он теплый, поскольку некоторое количество электрической мощности рассеивается в шнуре вследствие сопротивления провода.

Сопротивление измеряется в омах; чем больше эта величина, тем хуже проводник. Международное обозначение ома – большая греческая буква «омега» (Ω). Сопротивление, напряжение и сила тока связаны наиболее фундаментальным из всех законов электротехники – законом Ома:

$$V = I \times R,$$

где V – напряжение [В], I – сила тока [А], а R – сопротивление [Ом]. (Инженерам-электрикам нравится использовать букву «E» для обозначения напряжения, поскольку она может обозначать и электродвижущую силу).

Что это означает на практике? Подайте ток силой 1 А на нагрузку величиной 1 Ом и между ее контактами вы получите напряжение в 1 В. Увеличьте напряжение вдвое, и, если сопротивление осталось неизменным, сила тока удвоится. Хотя все электронные компоненты имеют сопротивление, *резисторы* используются специально для снижения проводимости. Они применяются везде. Регулятор громкости в стереосистеме (не в цифровой) – это сопротивление, значение которого меняется, когда вы вращаете ручку; при повышении сопротивления снижается уровень сигнала и, следовательно, уровень громкости динамиков.

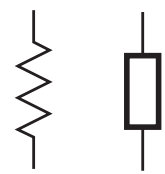

Рис. 1.3. Зигзагообразная линия слева – стандартное обозначение резистора на электрических схемах. Справа приведено обозначение, принятое в Великобритании. Как говаривал Черчилль, «мы – два народа, разделенных общим языком»

Таблица 1.2. Номиналы реальных резисторов

Название	Сокращенное обозначение	Значение [Ом]	Где встречается
Миллиом	мОм (mΩ)	0,001	Сопротивление проводов и других хороших проводников
Ом	Ом (Ω)	1	В источниках питания используются большие понижающие сопротивления с номиналом от нескольких ом до нескольких десятков Ом
Сотни ом			Во встраиваемых системах довольно часто можно обнаружить резисторы с номинальным сопротивлением в несколько сотен ом, которые используются для ограничения высокоскоростных сигналов
Килоом	кОм (kΩ, или просто k)	1000	Резисторы с номинальным значением от половины кОм до сотни или более кОм можно встретить во всех электронных приборах. Типичные значения «нагрузки» составляют от нескольких до десятков кОм
Мегаом	МОм (MΩ)	10^6	Аналоговые цепи с сигналом низкого уровня
Сотни мегаом		10^8 и более	Счетчики Гейгера и другие крайне чувствительные приборы; в резисторах встречается редко, поскольку близко по значению к сопротивлению воздуха

Что произойдет, если соединить несколько сопротивлений? Полное эффективное сопротивление системы последовательно соединенных резисторов будет равно сумме их значений:

$$R_{EFF} = R_1 + R_2.$$

Для двух резисторов, соединенных параллельно, эффективное сопротивление можно вычислить по формуле:

$$R_{EFF} = \frac{R_1 \times R_2}{R_1 + R_2}.$$

(Таким образом, два одинаковых резистора, соединенные параллельно, действуют как одно эффективное сопротивление, равное половине любого из них: если взять два сопротивления по 1 кОм, то эффективное сопротивление составит 500 Ом. Добавим третий резистор: это эквивалентно тому, что параллельно с резистором в 500 Ом подключен резистор 1 кОм, и эффективное сопротивление составит 333 Ом.)

Обобщенная формула для цепи, содержащей более двух параллельно соединенных резисторов:

$$R_{EFF} = \frac{1}{\frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} + \frac{1}{R_4} + \ldots}.$$

Для обозначения номинальных сопротивлений отдельных резисторов производители используют цветовую маркировку. Хотя на первый взгляд этот подход может показаться слишком загадочным, на практике он себя оправдывает – независимо от ориентации и способа установки резистора на печатной плате цветные полоски на нем всегда остаются видимыми.

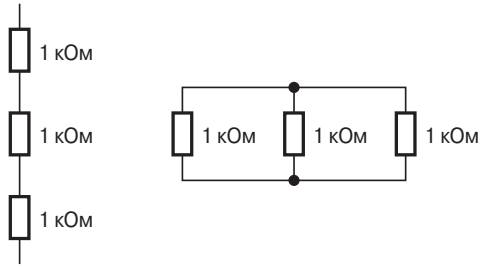

Рис. 1.4. Три последовательно соединенных резистора слева эквивалентны одному компоненту с сопротивлением 3000 Ом. Правые резисторы, соединенные параллельно, работают как одно сопротивление со значением 333 Ом

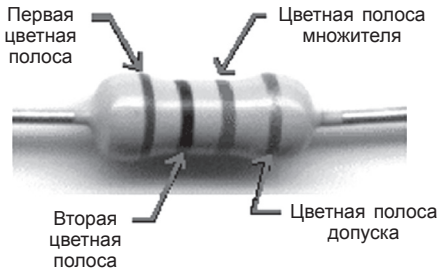

Рис. 1.5. На этой черно-белой фотографии цвет полосок не виден. Однако мы можем прочитать шифр слева направо: две первые описывают целую часть номинала, третья – множитель. Четвертая полоса может быть золотой (5%) или серебряной (10%), характеризуя допуск значения

Таблица 1.3. Цветовая маркировка резисторов. Все прежние мнемонические правила для запоминания порядка цветов теперь неприменимы. Единственная приемлемая, хотя и менее запоминающаяся, альтернативная фраза звучит так: «Big Brown Rabbits Often Yield Great Big Vocal Groans When Gingerly Slapped»[1]

Цвет полосы	Значение	Множитель
Черный	0	1
Коричневый	1	10
Красный	2	100
Оранжевый	3	1000
Желтый	4	10 000
Зеленый	5	100 000
Синий	6	1 000 000
Фиолетовый	7	Не используется
Серый	8	Не используется
Белый	9	Не используется
Золотой (третья полоса)		÷10
Серебряный (третья полоса)		÷100

[1] На самом деле запомнить порядок цветов довольно легко: после черного и коричневого идут все цвета радуги, кроме голубого, дальше – серый и белый. – *Прим. пер.*