

Лабораторная работа N6

Клавиатура

Оглавление

Введение	2
Механические клавиатуры	2
Пленочные клавиатуры	3
Клавиатура стенда SDK-2.0	3
Описание драйвера	3
Схема подключения	4
Дребезг	4
Обработчик прерывания	5
Литература	6

Введение

Несмотря на бурное развитие цифровых технологий, клавиатура продолжает оставаться чуть ли не единственным устройством ввода символьной информации. Любую интерактивную систему невозможно себе представить без клавиатуры или мышки. Поэтому спрос на такие, казалось бы, простейшие устройства ввода не снижается и в настоящее время.

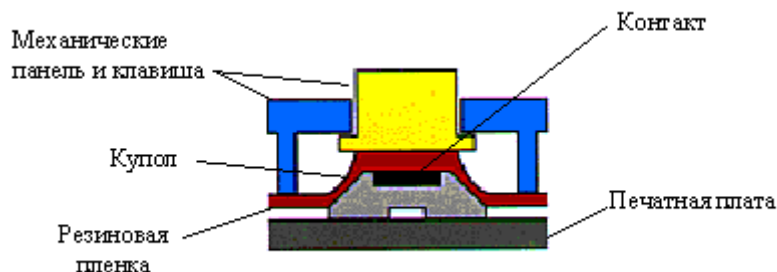


Рисунок 1 Элемент матрицы клавиатуры

Механические клавиатуры

В состав стандартной клавиатуры входят: матрица механических клавиш из пластика или металла, резиновая пленка, печатная плата с контактными площадками. Механизм действия такой клавиатуры следующий: при нажатии клавиши происходит замыкание контакта, расположенного на внутренней поверхности резиновой пленки и металлизированных контактных площадок печатной платы. Возврат нажатой клавиши осуществляется с помощью резинового купола. Контакт клавиатуры обычно выполняется в виде диска или кольца (для клавиатур с подсветкой), типовой диаметр которого составляет 4мм. Контакты на печатной плате могут быть луженые, никелированные или золоченые. В таблице 1 приведены электрические характеристики и общая спецификация стандартных клавиатур ACCORD.

Таблица 1 Электрические характеристики и общая спецификация

Сопротивление контактов Ом, не более	200
Сопротивление изоляции, МОм, не менее	100
Коммутируемое напряжение, В, не более	24
Коммутируемый ток, мА, не более	100
Сила нажатия, г, типовое значение	100±25
Ресурс, циклов/контакт, не менее	1 000 000
Дребезг контактов, мс, не более	10
Ход клавиши, мм, типовое значение	0.8 ... 1.2
Диапазон рабочих температур, °C	-20 ... +60
Температура хранения, °C	-40 ... +65

Пленочные клавиатуры



Пленочная клавиатура на полиэфирных пленках с тактильным эффектом. На внутреннюю сторону лицевой пленки нанесен многоцветный рисунок с изображением клавиш и специальных символов. Тактильный эффект по технологии mylar_dom. Внешняя сторона антибликовая, устойчивая к истиранию и воздействию агрессивных сред. На внутренней стороне клавиатуры нанесен клеевой слой для монтажа на корпус прибора.

Таблица 2 Условия эксплуатации

Рабочий температурный диапазон	от -40 до +50 гр.С
Относительная влажность воздуха при 25гр.С	не более 93%
Пылевлагозащищенность	IP65
Устойчивость лицевой пленки к агрессивным средам	растворители, масла и смазки, бензин и диз. топливо, моющие средства
Синусоидальная вибрация (работоспособность во время и после воздействия)	5 - 80 Гц с амплитудой ускорения 2 g и амплитудой перемещения 2 мм
Многочисленные механические удары (работоспособность во время и после воздействия)	10 g в 3-х направлениях с частотой 40 - 60 уд/мин. 1000 ударов

Клавиатура стенда SDK-2.0



Клавиатура AK1604A-WWB подключена через порты ввода-вывода микроконтроллера. Клавиатура организована в виде матрицы 4x4. Доступ к колонкам и рядам организован как чтение/запись определенного бита порта внешней памяти (4 бита соответствуют 4 колонкам, другие 4 бита - рядам). Ряды ROW1..ROW4 подключены к плюсу питания через резисторы. Это обеспечивает наличие логической единицы при отсутствии нажатия. На столбцы клавиатуры подают логический ноль. При нажатии на кнопку, происходит изменение значения сигнала на входе

соответствующего ряда с единицы на ноль.

Описание драйвера

Драйвер клавиатуры должен работать по прерыванию от таймера. Во время прерывания производится опрос регистра клавиатуры, и если нажата какая-либо клавиша, ее код заносится в буфер драйвера. Чтение символа из буфера драйвера производится с помощью API функции чтения ReadKeyBuffer(), которая в случае удачного завершения возвращает 1 и передает считанный из буфера байт, а в случае, если буфер пуст, возвращает 0. Буфер клавиатуры организован аналогично буферам драйвера последовательного. При инициализации необходимо указать задержку перед повтором символа (первый параметр) и скорость повтор символа (второй параметр).

Схема подключения

Клавиатура SDK-2.0 организована в виде матрицы 4x4. Один вывод каждой кнопки соединяется с горизонтальной линией, второй вывод соединяется с вертикальной линией. Каждая горизонтальная линия одним концом соединяется с источником питания, а вторым подключается к входному порту. Вертикальная линия подключается только с одной стороны к отдельным разрядам выходного порта.

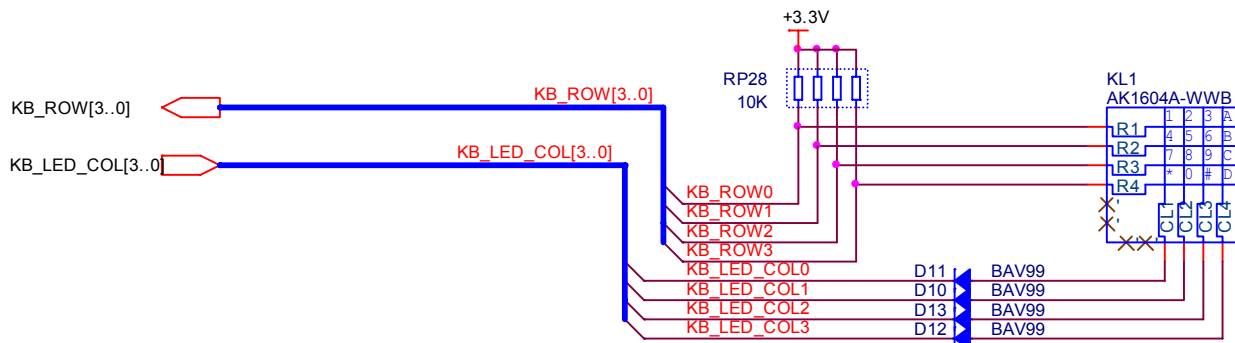


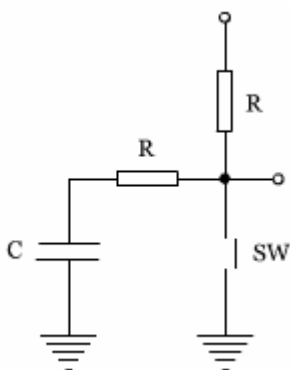
Рисунок 2 Схема подключения клавиатуры

Если все ключи разомкнуты, то потенциал на выходных портах ROW остается высоким, т.е. соответствует логической единице. Однако если замкнуть один из ключей и на соответствующую вертикальную линию подать сигнал низкого уровня, то потенциал горизонтальной линии, с которой соединена эта кнопка, также станет низким. Иначе говоря, если записать в порт COL значение 1101, то это позволит сканировать состояние кнопок на этого “столбца”. Например, если при этом будет нажата кнопка K₂₆, то на горизонтальной линии будет низкий потенциал и в соответствующий разряд выходного порта примет значение логического нуля, т.е. ROW=1101.

Таким образом, полный опрос матричной клавиатуры включает последовательное обнуление четырех битов ROW с анализом на каждом шаге разрядов COL.

Дребезг

При нажатии на кнопку напряжение не сразу устанавливается на уровне 0 В, а “скачет” в течение некоторого времени (1-10 мс), пока цепь надежно не замкнется. После того, как клавиша будет отпущена, напряжение также “скачет”, пока не установится на уровне логической “1”.



При замыкании и размыкании переключателей в цепи возникают импульсные помехи (которые называются “шум” или “звон”), вызванные дребезгом контактов. Это явление часто возникает в системах на базе микроконтроллеров, где для ввода данных используется клавиатура, и дребезг может восприниматься как многократное нажатие клавиши. Звон возникает при установке и разрыве контакта путем нажатия на клавишу.

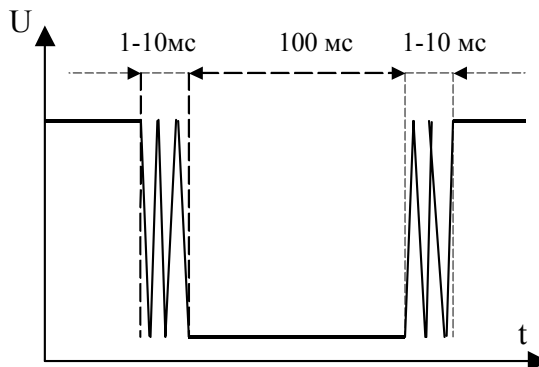
Помехи, вызванные дребезгом контактов, могут интерпретироваться как несколько размыканий и замыканий ключа. Если клавиша используется для управления

устройством, то это устройство воспримет возникающие помехи как многократное нажатие клавиши, что вызовет значительные затруднения при его использовании. Чтобы устранить данный эффект, используются специальные схемы или программные методы для подавления дребезга.

Один из простых схемотехнических способов устранения звона состоит в подключении RC-цепи. В этой схеме время, требуемое для заряда/разряда конденсатора до порогового напряжения, маскирует “дребезг” контактов при переключении. Можно также установить

триггер Шмитта между схемой ключа и микроконтроллером, чтобы усилить эффект подавления звона. Недостатками этого метода являются дополнительные затраты на компоненты, которые должны быть установлены на плате, и дополнительное время, требуемое для заряда/разряда RC-цепи.

Значительно лучший способ избавиться от звона - сделать это программно. Если уровень напряжения на выходе ключа не изменяется в течение 20мс, то можно считать что дребезг окончился, и больше изменений состояния не ожидается.



Поскольку процессор обладает высоким быстродействием, то он может воспринять эти скачки напряжения за несколько нажатий. Для программного устранения влияния “дребезга” используется задержка. После того, как в результате сканирования обнаружится “0” в регистре ROW, сканирование прекращается и производится задержка на некоторое время. После этого сканируется тот же столбец и, если на том же месте регистра ROW обнаружен “0”, то фиксируется нажатие клавиши. После этого через некоторое время, достаточное для отпускания клавиши, еще раз проверяется тот же столбец. Если состояние линии изменилось, то фиксируется отпускание клавиши и продолжается сканирование клавиатуры. Если клавиша все еще нажата, то производится задержка на время перед повтором символа, и если состояние регистра не изменилось, то в буфер клавиатуры повторно заносится символ. После этого, пока клавиша не будет отпущена, в буфер заносится код клавиши через промежутки времени, определяемые скоростью повтора символа.

Обработчик прерывания

В связи с тем, что сканирование клавиатуры и обработка нажатия клавиши производится в теле обработчика прерывания, необходимо следить, чтобы время его выполнения не превышало времени между соседними прерываниями. Это приведет к повторному входу в обработчик прерывания, последствия этого могут быть непредсказуемы. Также, увеличение времени выполнения обработчика приведет к уменьшению времени выполнения основной программы, т.е. может оказаться, что процессор большую часть времени будет занят обработкой прерываний.

Исходя из этого, все задержки в обработчике были реализованы не на прямую, а через счетчик прерываний. После фиксации нажатия устанавливается флаг нажатия KeyPress, и сканирование запрещается до тех пор, пока клавиша не будет отпущена. Теперь при каждом вызове обработчика увеличивается значение счетчика и проверяется его состояние. Если счетчик равен определенному значению, то выполняются связанные с ними действия (например: count=1, прошло время достаточное для пропуска “дребезга”; count=3, можно фиксировать отпускание клавиши; count=3+DelayBeforeRepeat, пропущена задержка перед повтором символа).

Перед сканированием клавиатуры производится проверка на нажатие, не тратить время на бесполезное сканирование, если не одна клавиша не нажата. Если фиксируется нажатие, то производится проверка буфера на заполнение, и если буфер переполнен, то работу обработчика необходимо завершить. Далее производится сканирование клавиатуры, пока не будет идентифицирована нажатая клавиша. После этого устанавливается флаг, и

работа обработчика завершается. Когда приходит следующее прерывание, то поскольку установлен флаг нажатия, клавиатура не сканируется, а сразу осуществляется переход к обработке нажатия на клавишу. Поскольку между текущим прерыванием и прерыванием, во время которого было зафиксировано нажатие клавиши, прошло некоторое время то “дребезг “ был пропущен. Теперь можно повторно проверить состояние клавиши отмеченной переменными COL и ROW. Если клавиша еще нажата, то соответствующий символ в буфер, в противном случае флаг нажатия клавиши KeyPress сбрасывается, обработчик завершается и продолжается сканирование клавиатуры.

После записи кода клавиши в буфер осуществляется задержка достаточная для отпускания клавиши и еще раз опрашивается состояние клавиши. Если состояние клавиши изменилось, то фиксируется отпускание клавиши (флаг нажатия клавиши KeyPress сбрасывается) и продолжается сканирование клавиатуры. Иначе осуществляется задержка перед повтором символа, и если клавиша еще не отпущена, повторно заносится код символа. После этого через промежутки, определяемые скоростью повтора, код клавиши заносится в буфер до тех пор, пока не будет зафиксировано отпускание клавиши.

Литература

1. Учебный стенд SDK-2.0. Инструкция по эксплуатации
2. LPC2119/2129/2194/2292/2294 USER MANUAL. Philips Semiconductors.
3. <http://www.itpuls.com/wrt/drebesg.html>