

Бегущая строка на микроконтроллере для начинающих

Принцип работы устройства

Динамическая индикация – это процесс вывода информации, изменяющейся во времени. Т.е. на небольшом участке пространства (в нашем случае – платы) можно отобразить целую строку символов. Сделать это достаточно просто – необходимо лишь сдвигать информацию (строку) в одном направлении с течением времени. Другими словами – в 1-ю секунду мы отображаем "ПРИ", во вторую – "РИВ", в третью – "ИВЕ", и четвертую – "ВЕТ". Если промежуток времени достаточно малый, то человек воспринимает все это как бегущую надпись "ПРИВЕТ". Однако, если нам необходимо вывести этот текст на светодиодной матрице, скажем, 5 строк * 10 столбцов, то минимальное количество отводов будет $5 \cdot 10 = 50$ шт. +1 общий катод (анод). Согласитесь, достаточно много для такой небольшой матрицы! Поэтому было придумано управлять такими сборками по принципу СТОЛБЕЦ - СТРОКА. Т.е. для включения любого одного сегмента (светодиода) вам необходимо подать "-" питания на строку, в которой находится данный сегмент, и "+" на столбец с тем же сегментом (рис. 1).

Конфигурация (+/-) зависит от того, что вы считаете за строку/столбец. Теперь посчитаем: 5 "-" и 10 "+" = 15 отводов, вместо 51! Однако не все так просто. Нам в момент времени надо вывести конкретный набор символов,

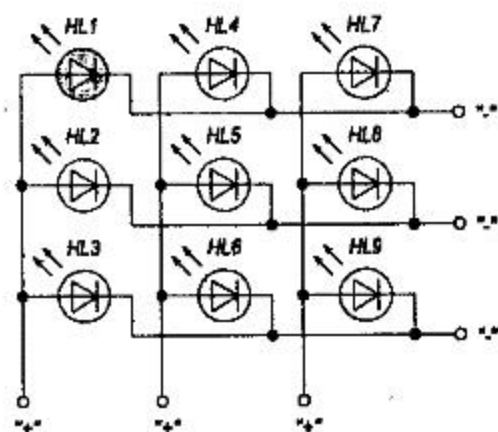


Рис. 1

а не сегмент. И опять нам на помощь приходит "человеческий фактор". Когда вы смотрите на быстро мерцающую лампочку, то для вас она кажется просто горячей. Аналогично и с мониторами – частота обновления заставляет нас воспринимать набор мерцающих пикселей как картинку. Значит для получения простых и неподвижных символов нам необходимо в короткий промежуток времени включать определенные сегменты строк на соответствующем столбце (например, у нас матрица 3*3 светодиода, нам на нее необходимо вывести +, это делается так: сначала зажигаем середину первого столбца, ждем небольшое время (видим -), выключаем столбец, включаем 2 столбец и зажигаем 3 сегмента (видим I), опять ждем и выключаем, потом переходим на последний, включаем середину (видим -) и опять переходим на начало. Так при повторении с достаточной частотой мы воспримем это как неподвижный символ -I-). Значит, для вывода бегущей строки в таком режиме, нам необходимо выводить неподвижную часть символов с достаточной частотой обновления, меняя их через определенное время для эффекта бега.

Практическая реализация

Для такого непростого управления нам понадобится микроконтроллер (МК) PIC16F84A, счетчик K561IE8 (он необходим для расширения выводов МК) и самое главное – светодиодная матрица. Я собирал собственную, состоящую из 50 обычных светодиодов. Получилось не так дорого, зато паял не так быстро, как покупал... В итоге была составлена схема, представленная на рис. 2.

Как видно, столбцы управляются счетчиком DD2 K561IE8 (4017 – зарубежный аналог), импульс на который подается с выв. 11 DD1. Выв. 12 DD1 необходим для правильного запуска счетчика DD2, т.е. он при первом

старте схемы сбрасывает DD2 в 0. Микроконтроллер DD1 включен по классической схеме с кварцевым резонатором ZQ1 4 МГц. Транзисторы VT1...VT10 служат для разгрузки микросхемы DD2. В качестве VT1...VT10 подойдут любые маломощные транзисторы (например, КТ3102БМ). Резисторы R1...R5 – 470 Ом; их сопротивление желательно подобрать под конкретный тип светодиодов и под необходимую яркость, однако стоит помнить и о токе выходов DD1 (для нормальной работы не более 20...25 мА). DA1 – стабилизатор +5 В. Все конденсаторы рассчитаны на напряжение от 25 В и выше. Светодиоды HL1...HL50 – любые, одного типа.

Программа

В программе [1], написанной для этой схемы, есть комментарии по вводу данных. Прошивка [2] выводит текст "Pic16f84 + k561ie8 'бегущая строка'" Внеся свои изменения в программу (для этого не обязательно быть программистом), можно воспроизводить любой небольшой текст. Вот только символы надо будет вводить вручную, т.е. собирая из сегментов.

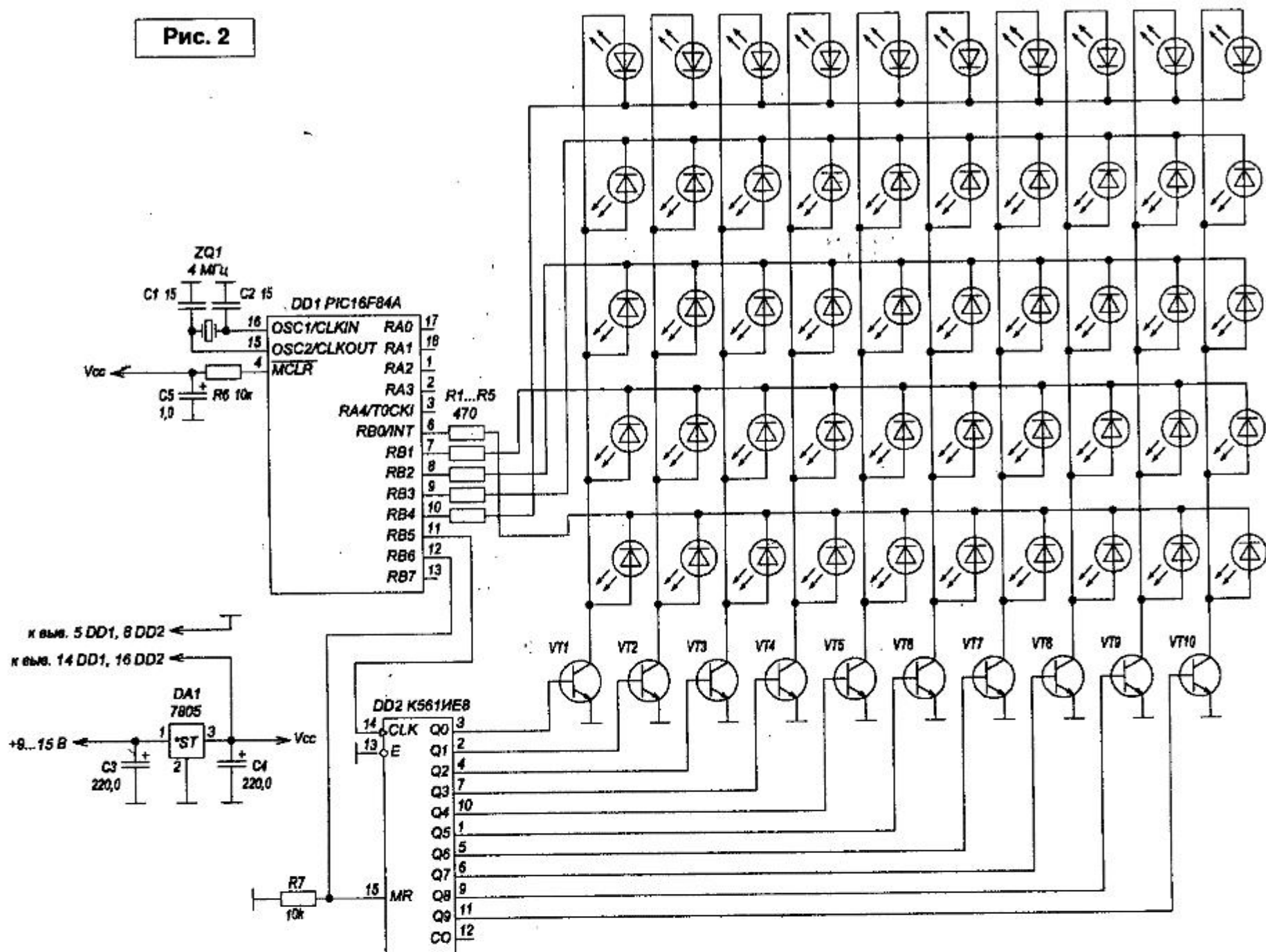
Как и чем прошивать?

Как показывает практика, если с деталями обычно проблем нет, то вот с прошивкой МК у многих возникают проблемы. Мне хотелось бы внести ясность в это дело и дать пару советов: данными советами я лишь вас "удовольствия" лазить по многочисленным страницам Интернета, где не все правильно и не все качается.

Для прошивки вам потребуется:

- программатор с управляющей программой;
- ассемблер (он же компилятор) – переводит коды программ в понятный для МК цифровой код (для прошивки он собственно не нужен, а вот для изменения программы понадобится).

Рис. 2



Теперь по порядку. Я советую для PIC программатор JDM (он недорогой и простой, подключается к COM порту компьютера) – схема, плата, статья по нему и управляющая программа здесь [3]. Можно использовать и PopuProg, но он будет подороже. Пару слов об установке и использовании: сначала в настройках программы выберите программатор JDM, потом в опциях включите галочку “включить NT/XP драйвер”. Если этого не сделать, то толку с этого программатора на операционной системе XP будет мало, т.к. программировать он будет с ошибками. Для прошивки подключите саму схему к компьютеру, запустите управляющую программу (ICprog), выберите необходимый тип МК (для нас PIC16F84A), откройте файл прошивки и нажмите “Программировать микросхему”. Через несколько секунд процесс завершится, и вы получите уведомление об успешной проверке. Теперь микроконтроллер прошит! Что касается конфигурации (в свободное время можно ее

изучить), то в данном типе МК можно поставить бит CP (это галочка справа, говорящая о том, что прошивку вами программой нельзя считать), WDT – сторожевой таймер (нам он не нужен, не ставить!), XT – это тип резонатора (в нашем случае кварцевый), PWRT – запуск программы при появлении +Vcc на выв. 4 DD1, по умолчанию он стоит в программе.

Ассемблер берем в поставке с программой MPLAB от официального производителя Microchip на русскоязычном сайте [4]. После того, как вы переделаете программу на свою строку, необходимо будет ее откомпилировать (представить в форме кодов). Это делается так: запускаете MPASM, выбираете тип

микросхемы – Processor (16F84A), все остальные опции – в положение Default (при первом запуске необходимо только выбрать тип МК, все остальное менять не надо). Потом открываете файл с программой *.asm и нажимаете “Assemble”. Если все правильно, то вы увидите зеленую линию с надписью 100% и отчет по программе (замечания, количество строк...). В той папке, где находится исходная программа, должен появиться файл с тем же названием, что и программа, но с расширением *.hex. Это и есть прошивка. Если же вылетела красная линия, то в программе есть ошибки. Их количество содержится в отчете, а что за ошибки, смотрим в файле с расширением *.err.

Программу (файл *prog_run.zip*) вы можете загрузить с сайта нашего журнала <http://www.radioliga.com> (раздел “Программы”)

Ресурсы

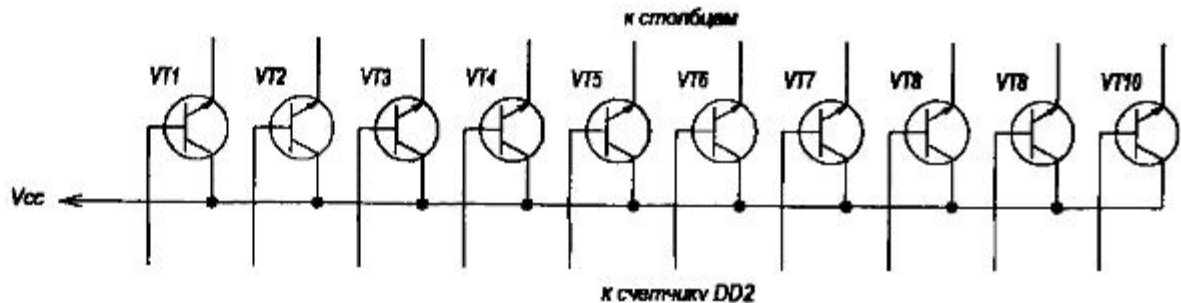
1. http://mdn-project.narod.ru/prog_run.asm
2. http://mdn-project.narod.ru/prog_run.HEX
3. <http://mdn-project.narod.ru/JDM.rar>
4. <http://www.microchip.ru/>

Бегущая строка на микроконтроллере для начинающих

Возвращаясь к напечатанному
("РЛ", №5/2007, с. 54-55)

При переносе схемы в электронный вариант была допущена следующая ошибка: на схеме (рис. 2) транзисторы эмиттером подключены к корпусу, а коллектором – к столбцам с диодами.

Необходимо: транзисторы VT1...VT10 эмиттером подключить к столбцам, а коллектором – к "+" питания после стабилизатора.



Приносим извинения читателям
за допущенную ошибку.

Дмитрий Мигаль
Беларусь, п. Ждановичи