ФИЛИАЛ ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО

ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ «МЭИ» в г. СМОЛЕНСКЕ

Кафедра: вычислительной техники

Направление: 09.03.01 «Информатика и вычислительная техника»

Профиль подготовки: «Программное обеспечение»

Курсовая работа

по дисциплине: «Схемотехника»

Тема**:** «Секундомер на основе светодиодной матрицы**»**

Студент \_\_\_\_ПО1-19\_\_\_ \_\_\_\_\_\_\_\_\_\_ \_Милославский С. А.\_

группа подпись фамилия И.О.

Руководитель \_\_к. т. н.\_\_\_\_\_ \_\_доцент\_\_ \_\_\_\_\_\_\_\_\_\_ \_\_\_Аверченков О. Е.\_\_

учен. степень должность подпись фамилия И.О.

Смоленск, 2021 г.

АННОТАЦИЯ

Пояснительная записка содержит 36 листов печатного текста, 9 рисунков, 1 таблицу, список литературы из 6 наименований и 7 формул.

Данный курсовой проект состоит из введения, двух разделов: проектирование и разработки курсовой работы, заключения, списка используемых источников и трех приложений.

В первом разделе курсовой работы выполняется проектирование курсовой работы, а именно анализ технического задания, описание структурной схемы, временные диаграммы и вывод основных формул.

Во втором разделе осуществляется разработка курсовой работы, а именно описание принципиальной схемы и используемых алгоритмов, оценка нормируемых параметров.

Результатом проекта является разработанное устройство – секундомер на основе светодиодной матрицы. Устройство позволяет подсчитывать время с точностью до 1/100 секунды. Оно способно отображать сотые доли секунды, секунды, минуты или часы в зависимости от режимов показа, который можно переключать с помощью кнопки. Также секундомер можно запускать, останавливать и сбрасывать.

Ключевые слова: секундомер, цифровой секундомер, x51, ОВМ, микроконтроллер, AT89C2051.

ANNOTATION

The explanatory note contains 36 sheets of printed text, 9 figures, 1 table, a bibliography of 6 titles and 7 formulas.

This course project consists of an introduction, two sections: design and development of a course work, a conclusion, a list of sources used and three applications.

In the first section of the course work, the design of the course work is carried out, namely the analysis of the technical assignment, the description of the structural diagram, timing diagrams and the derivation of the basic formulas.

In the second section, the development of the term paper is carried out, namely, the description of the schematic diagram and the algorithms used, the assessment of the normalized parameters.

The result of the project is a developed device – a stopwatch based on an LED matrix. The device allows you to calculate the time with an accuracy of 1 / 100th of a second. It is capable of displaying hundredths of a second, second, minute or hour, depending on the display modes, which can be switched using the button. Also, the stopwatch can be started, stopped and reset.

Keywords: stopwatch, digital stopwatch, x51, OBM, microcontroller, AT89C2051.

СОДЕРЖАНИЕ

[введение 5](#_Toc89358665)

[1. ПРоектирование курсовой работы 7](#_Toc89358666)

[1.1 Анализ технического задания 7](#_Toc89358667)

[1.2 Структурная схема 8](#_Toc89358668)

[1.3 Описание принципа действия устройства по структурной схеме 9](#_Toc89358669)

[1.4 Временные диаграммы и вывод основных формул 10](#_Toc89358670)

[1.5 Выбор, обоснование и расчет отдельных элементов и узлов 12](#_Toc89358671)

[2. Реализация курсовой работы 15](#_Toc89358672)

[2.1 Принципиальная схема 15](#_Toc89358673)

[2.2 Схемы алгоритмов 15](#_Toc89358674)

[2.3 Оценка нормированных параметров 21](#_Toc89358675)

[2.3.1 Оценка общей потребляемой мощности 21](#_Toc89358676)

[2.3.2 Оценка потребляемого тока 21](#_Toc89358677)

[2.3.3 Оценка рабочих температур 21](#_Toc89358678)

[заключение 22](#_Toc89358679)

[список использованных источников 23](#_Toc89358680)

[приложение А – перечень элементов 24](#_Toc89358681)

[Приложение Б – принципиальная схема 25](#_Toc89358682)

[приложение в – код программы 26](#_Toc89358683)

введение

Данная курсовая работа направлена на изучение проектирования устройств на базе ОВМ семейства x51. В рамках работы решается конкретная прикладная задача: разработка изделия, выполняющего функцию секундомера, способного подсчитывать время с точностью до 1/100 секунды.

Актуальность работы заключается в том, что секундомер является важным устройством с широкой областью применения. Он может использоваться в научных лабораториях, в заводских лаборатория, в учебных лабораториях ВУЗов, техникумов и школ, в спорте, в военном деле, а также в быту. Электронные секундомеры отличаются от механических повышенной точностью (1/100 сек) и более широким функционалом. Значимость работы подтверждается также и тем, что на рынке присутствуют сотни различных электронных секундомеров, цена на которые зачастую неоправданно завышена. В Интернете же возможно найти большое количество схем секундомеров на основе цифробуквенных индикаторов, но не на основе светодиодной матрицы, поэтому разработка дешевого устройства, в котором дисплей будет заменен на небольшую светодиодную матрицу. В качестве микроконтроллера будет взята недорогая микросхема их семейства x51.

Цель курсовой работы – разработка секундомера на основе светодиодной матрицы.

Для достижения поставленной цели были поставлены следующие задачи:

Изучить, используя литературу и Интернет, схемы других секундомеров и сделать на них обзор.

1. Нарисовать структурную схему и описать ее работу;
2. На основе структурной схемы нарисовать принципиальную схему;
3. Разработать общий алгоритм функционирования системы в словесном виде;
4. На основе составленного алгоритма написать программу на языке программирования C;
5. Откомпилировать программу и проверить ее работоспособность;
6. В случае наличия ошибок внести необходимые изменения, получив тем самым рабочий вариант программы.

Объектом исследования является проектирование устройств на базе ОВМ семейства x51.

Предметом исследования является разработка секундомера на основе светодиодной матрицы.

1. ПРоектирование курсовой работы
   1. Анализ технического задания

Электронный секундомер можно разработать на базе различных микроконтроллеров. В частности, предлагается использовать 8-ми разрядный МК Attiny2313, серии Tiny фирмы Atmel[4]. Также можно использовать PIC16F628[5] или даже ATmega16A, на основе которого строятся часы с использованием светодиодной матрицы[6]. В рамках данной работы предполагается использовать ОВМ семейства x51. Учитывая выше сказанное, а также то, что к микроконтроллеру в данном случае предъявляются минимальные требования (например, несущественно наличие большого количества ножек), для реализации секундомера будем использовать микросхему AT89C2051.

Для подсчета времени также можно применить различные подходы. Например, можно использовать микросхему часов реального времени[4], задействовать паузу[5] или воспользоваться внутренним таймером микроконтроллера[6]. Воспользуемся последним способом в силу его надежности и простоты реализации.

Выводить время можно с помощью жидкокристаллического индикатора[4], с помощью цифробуквенного индикатора[5] или с помощью светодиодной матрицы[6]. Применим последний способ, поскольку:

* во-первых, именно он обозначен в теме работы;
* во-вторых, применение дисплея неоправданно дорого, светодиодные матрицы же сейчас очень дешевы, и их стоимость не выше, а то и ниже, чем у семисегментных индикаторов такого же размера.

Кроме того, на основе светодиодной матрицы строится модуль с матричным индикатором[3], конструкция которого может послужить основой конструкции разрабатываемого секундомера.

Для управления устройством можно использовать матричную клавиатуру[4] или отдельные кнопки[5][6]. Для построения секундомера на основе светодиодной матрицы достаточно всего три кнопки, поэтому применение полноценной клавиатуры в данном случае неоправданно.

* 1. Структурная схема

Структурная схема электронного секундомера должна удовлетворять основным требованиям:

* точность измерения (1/100 секунды согласно ТЗ);
* стабильность питания;
* удобство использования.

Исходя из вышесказанного, а также из того, что устройство должно включать микроконтроллер, светодиодную матрицу 5×7, дешифратор, три кнопки и разъем для подключения питания, то структурная схема будет иметь следующий вид, представленный на рисунке 1.



Рисунок 1 – Структурная схема электронного секундомера

Опишем каждый элемент в отдельности:

* МК – микроконтроллер AT89C2051;
* кварцевый резонатор с частотой , работает как тактирующее устройство микроконтроллера. Резонатор обеспечивает наиболее удобную, подходящую частоту тактового генератора;
* светодиодная матрица предназначена для вывода времени. В зависимости от режима показа должны выводиться сотые доли секунды, секунды, минуты или часы;
* дешифратор предназначен для выбора текущего столбца в матрице;
* кнопка 1 запускает и останавливает секундомер;
* кнопка 2 предназначена для переключения режима показа;
* кнопка 3 используется для сброса секундомера;
* разъем питания связан со всеми кнопками, кроме этого через разъем к микросхеме должно подключаться питание для микроконтроллера и дешифратора.
  1. Описание принципа действия устройства по структурной схеме

При подключении питания секундомер должен находиться в незапущенном состоянии, то есть на светодиодной матрице должны показываться две цифры 0, а также устройство должно поддерживать четыре режима работы:

* показ сотых долей секунды;
* показ секунд;
* показ минут;
* показ часов.

Первая кнопка предназначена для запуска устройства, кроме этого секундомер должен находиться в режиме показа минут, после нажатия на нее секундомер начинает отсчет времени. Повторное нажатие на эту кнопку должно остановить устройство, а следующее нажатие возобновить работу и так далее.

Вторая кнопка служит для переключения режима показа. Нажатие на нее должно приводить к смене режима, как в запущенном состоянии, так и в остановленном.

Третья кнопка используется для сброса секундомера, то есть нажатие на нее должно возвращать секундомер в исходное состояние (два нуля на индикаторе, ожидание запуска).

Правильная работа динамической индикации должна обеспечиваться за счет двух типов кодов:

* первый код, отвечающий за то, какие светодиоды должны гореть в текущем столбце, микроконтроллер должен подавать непосредственно на матрицу;
* второй код, указывающий, какой столбец в матрице является текущим, должен подаваться на матрицу через дешифратор.
  1. Временные диаграммы и вывод основных формул

Рассмотрим временные диаграммы светодиодной матрицы. Микроконтроллер формирует информационный сигнал на светодиодах всех столбцов, но светиться будет только тот столбец, катод которого подключен к общей точке, чтобы обеспечить слитное изображение всех столбцов, необходимо организовать поочередное и периодическое их включение с высокой частотой.

Пусть сигналы на катодах 1 – 7 столбцов, тогда временные диаграммы примут следующий вид (рисунок 2, 3).



Рисунок 2 – Временные диаграммы светодиодной матрицы (общая и сигналов )



Рисунок 3 – Временные диаграммы светодиодной матрицы (сигналов )

Оценим частоту «смены кадра». Пусть  (). Время включения одного столбца должно быть в семь раз меньше периода , то есть  [1, с. 406].

В схеме будут применяться транзисторные ключи на основе p-n-p транзистора для усиления выходного тока ножек микроконтроллера, выведем формулы для их расчета. Вначале найдем сопротивление резистора для ограничения тока нагрузки:

, (1)

где  – падение напряжения на ограничивающем резисторе;

 − ток в цепи коллектора.

, (2)

где  – напряжение питания;

 – падение напряжения на светодиоде;

 – падение напряжения на транзисторе.

В итоге получим:

. (3)

Теперь найдем значение сопротивления резистора в цепи базы:

, (4)

где  − падение напряжения на искомом резисторе;

 − базовый ток.

, (5)

где  − напряжение входного сигнала;

 − падение напряжения на переходе база-эмиттер.

, (6)

где  − ток коллектора,

 − коэффициент усиления транзистора по току.

В итоге получим:

. (7)

* 1. Выбор, обоснование и расчет отдельных элементов и узлов

Как уже было сказано выше, в качестве микроконтроллера выберем микросхему AT89C2051. Внешний кварцевый резонатор подберем согласно рекомендациям из пособия[2, c. 227][2, с. 294] для упрощения подсчета времени с частотой . Такую частоту имеет кварцевый резонатор HC-49S. Величина внешних конденсаторов для него указана в документации и равна , поэтому будем использовать в устройстве два керамических конденсатора такой емкости. Также нам понадобится еще один танталовый конденсатор с емкостью равной для организации длинного единичного сигнала сброса[2, c. 229].

Исходя из того, что в модуле с матричным индикатором[3] применяется светодиодная матрица 757AS, будем использовать ее, тогда из документации получим следующие параметры , . Для построения транзисторных ключей применим биполярные p-n-p транзисторы KT3107А c коэффициентом передачи тока , падением напряжения между базой и эмиттером  и падением напряжения между коллектором и эмиттером .

Рассчитаем сопротивления ограничивающих резисторов и резисторов цепи базы:

 Ом.

 Ом.

Исходя из расчетов, применим в устройстве металлопленочные резисторы величиной и .

В схеме должен присутствовать дешифратор, будем использовать микросхему К155ИД10, так как у нее подходящее значение выходного тока низкого уровня:

.

Рассчитаем также отдельно ток базы, который является входным для AT89C2051:

.

Для переключения режимов, запуска/остановки, а также сброса секундомер должен иметь три переключателя. В качестве них будем использовать тактовые кнопки KLS7-TS3601-4.3-180 (ТС-0120). Для подключения внешнего питания в схеме устройства должен присутствовать разъем.

Номинальное значение тока в цепях со светодиодами:

.

Номинальное значение тока в базовых цепях:

.

Номинальное значение тока в эмиттерных цепях:

.

1. Реализация курсовой работы
   1. Принципиальная схема

Основой полной принципиальной схемы секундомера является микроконтроллер АT89C2051. К нему подключен, обрамленный внешними конденсаторами  и , кварцевый резонатор HC-49S с частотой . Ножки P1.0 − P1.4 ОВМ присоединяются через резисторы  к базам p-n-p транзисторов. На них программа должна подавать код, определяющий какие светодиоды должны гореть в текущем столбце. Это возможно за счет того, что коллекторы транзисторов подключены через токоограничивающие резисторы  к ножкам 1, 5, 8, 13 и 14 светодиодной матрицы. Эмиттеры транзисторов при этом подключены к источнику питания.

Ножки P1.5 − P1.7 микроконтроллера присоединяются к входу дешифратора К155ИД10. На них должен подаваться код, определяющий какой столбец светодиодной матрицы является текущим, поэтому выход дешифратора подключен к катоду матрицы.

Для управления устройством на схеме присутствуют три кнопки KLS7-TS3601-4.3-180 (ТС-0120). Первая кнопка подключена к Res микросхемы и предназначена для сброса секундомера. Для организации длинного единичного сигнала сброса к входу Res подключен также конденсатор . Вторая и третья кнопки подключены к ножкам P3.0 и P3.1 и служат для запуска/остановки и переключения режима работы соответственно.

* 1. Схемы алгоритмов

Для дальнейшего описания принципа действия устройства приведем основные алгоритмы, написанной для микроконтроллера программы.



Рисунок 4 – Схема функции main

После включения питания AT89C2051 инициализирует таймер для срабатывания прерывания каждую сотую долю секунды. Затем происходит начальная установка цифр (двух нулей). Далее в бесконечном цикле вначале происходит проверка нажатия кнопки (смены режима или запуска/остановки), а затем после паузы происходит переключение на следующий столбец светодиодной матрицы.



Рисунок 5 – Схема функции buttonsCheck

Функция проверки нажатия кнопки вначале считывает значение бита P3.0. Если бит равен нулю, то нажата кнопка запуска/остановки. В этом случае программа проверяет в каком состоянии находится секундомер, и в зависимости от этого состояния запускает или останавливает устройство. Затем функция проверяет значение бита P3.1 и в случае равенства его нулю (нажата кнопка переключения режима) меняет режим показа на следующий. При этом в работе функции учитывается возможный дребезг кнопки. Для его игнорирования применяются паузы в 10 мс.



Рисунок 6 – Схема функции T0\_int

Переменные time10ms, timeSec, timeMin и timeHour хранят значения сотых долей секунды, секунд, минут и часов соответственно. В прерывании таймера происходит увеличение общего времени на одну сотую секунды.



Рисунок 7 – Схема функции setDigits

Данная функция записывает число сотых долей секунды, секунд, минут или часов в зависимости от режима (переменная mode) в два массива first и second с помощью функции setData. Первый массив служит для кодирования первой цифры, а второй – для второй.



Рисунок 8 – Схема функции setData

Функция записи в массив принимает в качестве параметров цифру, которую необходимо записать и указатель на массив. Она вносит в массив нужные коды, которые потом сможет использовать функция nextColumn для формирования нужного значения в регистре P1.



Рисунок 9 – Схема функции nextColumn

Переменная ptrColumn указывает на текущий столбец. Используя ее и заранее подготовленные массивы функция nextColumn записывает правильный код в регистр P1, а затем переключает столбец путем инкремента ptrColumn.

* 1. Оценка нормированных параметров
     1. Оценка общей потребляемой мощности











* + 1. Оценка потребляемого тока



* + 1. Оценка рабочих температур

Таблица 1 – Рабочие температуры элементов

|  |  |  |
| --- | --- | --- |
| Элемент | Нижняя граница, °C | Верхняя граница, °C |
| AT89C2051 |  |  |
| К155ИД10 |  |  |
| 757AS |  |  |

Итоговый рабочий диапазон температур: °C.

заключение

В рамках данной курсовой работы было изучено проектирование устройств на базе ОВМ семейства x51. В процессе работы была подготовлена структурная схема, подробно описана ее работа. После этого были подобраны элементы и составлена принципиальная схема. Далее была написана программа, которую удалось протестировать и отладить.

В результате было создано актуальное и полезное устройство – секундомер на основе светодиодной матрицы. Он способен считать время с точностью до 1/100 секунды, а также работать в различных режимах показа.

Следует отметить, что полученное устройство полностью согласуется с техническим заданием. Все составные части, а именно микроконтроллер, светодиодная матрица , дешифратор, три кнопки, и разъем для подключения питания присутствуют. Схема также соответствует техническим параметрам:

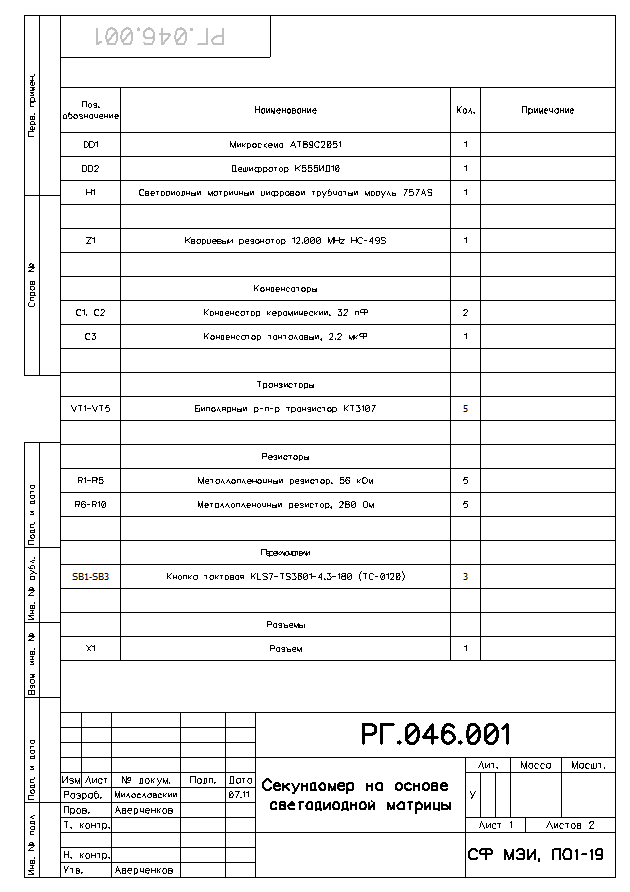
* ;
* ;
* итоговый рабочий диапазон температур  лучше указанного в ТЗ ().

Принцип работы схемы полностью соответствует указанному в ТЗ. Программа для микроконтроллера написана на языке программирования C, протестирована и отлажена.

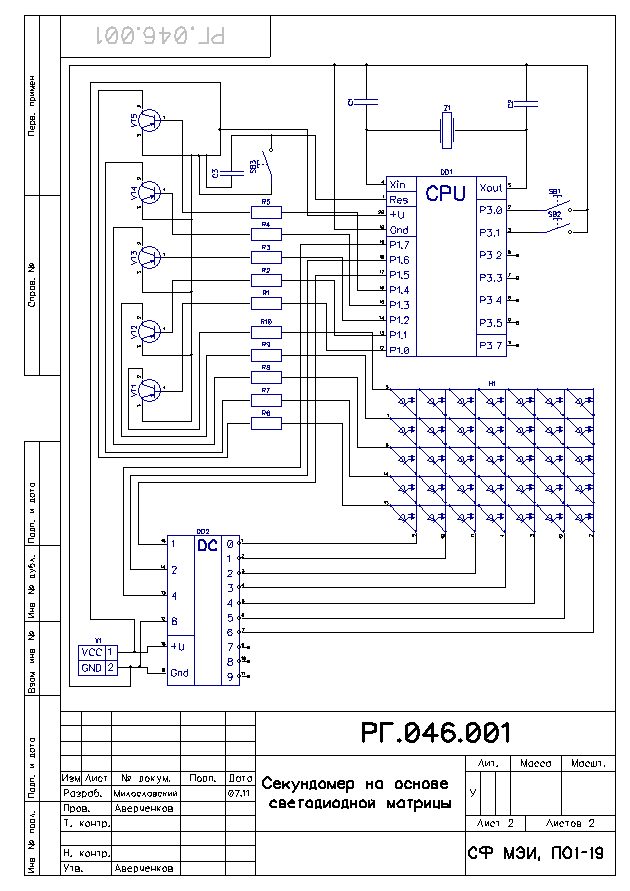
список использованных источников

1. АВЕРЧЕНКОВ О.Е. Схемотехника: аппаратура и программы. ДМК Пресс, 2012. – 588 с.
2. АВЕРЧЕНКОВ О.Е. Сборник лабораторных работ по курсу «Схемотехника». СФ МЭИ, 2013.
3. АВЕРЧЕНКОВ О.Е. Сборник лабораторных работ по курсу «Микропроцессорные системы». СФ МЭИ, 2014.
4. Прошин А. А., Бростилов С. А., Горячев Н. В. Разработка цифрового секундомера // Молодой ученый. – 2015. – №2. – С. 187-190. – URL https://moluch.ru/archive/82/15066/ (дата обращения: 29.11.2021).
5. Курсовая работа: Электронный секундомер URL https://www.bestreferat.ru/referat-248590.html (дата обращения: 29.11.2021).
6. Простые часы на светодиодных матрицах. URL https://www.radiokot.ru/circuit/digital/home/103/ (дата обращения: 14.11.2021).

приложение А – перечень элементов



Приложение Б – принципиальная схема



приложение в – код программы

//Подключаем библиотеку

#include<at89x51.h>

//Тик таймера

#define tik (10000)

//Число столбцов в матрице

#define countColumn 7

//Ширина цифры

#define digitWidth 3

//Пауза 1мс

#define delay1ms delayFOR(99)

//Запущен или нет

unsigned char started = 0;

//Режим работы

//0 - показывать 1/100 секунды

//1 - показывать секунды

//2 - показывать минуты

//3 - показывать часы

unsigned char mode = 1;

//Указатель на столбец

unsigned char ptrColumn = 0;

//Время в 1/100 секунды

unsigned char time10ms = 0;

//Время в секундах

unsigned char timeSec = 0;

//Время в минутах

unsigned char timeMin = 0;

//Время в часах

unsigned char timeHour = 0;

//Массивы для хранения цифр

data unsigned char first[digitWidth];

data unsigned char second[digitWidth];

//Функция записи цифры в массив

void setData(unsigned char digit, unsigned char data \*array)

{

    switch(digit)

    {

        //(\*): если цира 0, то

        case 0:

            //В первый столбец записываем 1 + 2 + 4 + 8 + 16 = 1F = \*\*\*\*\*

            array[0] = 0x00;

            //Во второй столбец записываем 1 + 16 = 11 = \*---\*

            array[1] = 0x0E;

            //В третий столбец записываем 1 + 2 + 4 + 8 + 16 = 1F = \*\*\*\*\*

            array[2] = 0x00;

            //Итого получаем \*\*\*\*\*

            //               \*---\*

            //               \*\*\*\*\*

            //Повернуть по часовой стрелке; "\*" - светодиод горит, "-" - светодиод не горит

            break;

        case 1:

            array[0] = 0x00;

            array[1] = 0x1D;

            array[2] = 0x1B;

            break;

        case 2:

            array[0] = 0x08;

            array[1] = 0x0A;

            array[2] = 0x04;

            break;

        case 3:

            array[0] = 0x00;

            array[1] = 0x0A;

            array[2] = 0x0A;

            break;

        case 4:

            array[0] = 0x00;

            array[1] = 0x1B;

            array[2] = 0x18;

            break;

        case 5:

            array[0] = 0x02;

            array[1] = 0x0A;

            array[2] = 0x08;

            break;

        case 6:

            array[0] = 0x02;

            array[1] = 0x0A;

            array[2] = 0x00;

            break;

        case 7:

            array[0] = 0x1C;

            array[1] = 0x1A;

            array[2] = 0x06;

            break;

        case 8:

            array[0] = 0x00;

            array[1] = 0x0A;

            array[2] = 0x00;

            break;

        case 9:

            array[0] = 0x00;

            array[1] = 0x0A;

            array[2] = 0x08;

            break;

    }

}

//Функция переключения столбца

void nextColumn(void)

{

    //Последние 3 бита регистра P1 отвечают за то, какой столбец показывать

    //Они подаются на дешифратор, который выдает значения из множества {0,1,2,3,4,5,6} (Последние три ножки дешифратора не используются)

    //Исходя из сказанного, в P1 нужно записать указатель на столбец смещенный на 5 битов влево побитно сложенный со значением из массива, хранящего цифру, по указателю

    if (ptrColumn < 3)

        P1 = (ptrColumn << 5) | second[ptrColumn];

    //Если промежуток, то ничего не выводим

    if (ptrColumn == 3)

        P1 = (ptrColumn << 5) | 0x1F;

    //Если 2 цифра, то берем из массива second

    if (ptrColumn > 3 && ptrColumn < 7)

        P1 = (ptrColumn << 5) | first[ptrColumn - 4];

    //Переходим к следующему столбцу

    ptrColumn++;

    //Если столбец был последним, то переходим опять к первому

    if (ptrColumn == countColumn)

        ptrColumn = 0;

}

//Функция записи цифр в зависимости от режима

//Работает аналогично примеру из книги

void setDigits()

{

    switch(mode)

    {

        case 0:

            setData(time10ms/10, &first[0]);

            setData(time10ms%10, &second[0]);

            break;

        case 1:

            setData(timeSec/10, &first[0]);

            setData(timeSec%10, &second[0]);

            break;

        case 2:

            setData(timeMin/10, &first[0]);

            setData(timeMin%10, &second[0]);

            break;

        case 3:

            setData(timeHour/10, &first[0]);

            setData(timeHour%10, &second[0]);

            break;

    }

}

//Инициализация таймера

void T0init(void)

{

    TR0=0;

    TMOD=(TMOD&0xf0)|0x1;

    TL0=(~tik);

    TH0=(~tik)>>8;

    TR0=1;

    ET0=1;

    EA=1;

}

//Прерывание таймера;

void T0\_int (void) \_\_interrupt (TF0\_VECTOR)

{

    TR0=0;

    TL0=(~tik);

    TH0=(~tik)>>8;

    TR0=1;

    if (started)

    {

        time10ms++;

        if (time10ms == 100)

        {

            time10ms = 0;

            timeSec++;

            if (timeSec == 60)

            {

                timeSec = 0;

                timeMin++;

                if (timeMin == 60)

                {

                    timeMin = 0;

                    timeHour++;

                    if (timeHour == 24)

                        timeHour = 0;

                }

            }

        }

        setDigits();

    }

}

//Пауза с помощью FOR

void delayFOR(unsigned char kodF)

{

    unsigned char ml;

    for(ml = 0; ml < kodF; ml++){}

}

//Пауза с заданным числом мс

void DelayMS(unsigned char KodMS)

{

    do delay1ms;

    while(--KodMS);

}

//Обработка нажатия кнопок

void buttonsCheck(void)

{

    //Пауза и старт секундомера

    if (!P3\_2)

    {

        //Выключаем матрицу

        P1 = 0x1F;

        //Игнорируем дребезг

        DelayMS(10);

        //Ждем отпускания

        while(!P3\_2);

        //Игнорируем дребезг

        DelayMS(10);

        //Запускаем или останавливаем секундомер

        if (started)

        {

            started = 0;

        } else

        {

            started = 1;

        }

    }

    //Переключение режимов

    if (!P3\_3)

    {

        //Выключаем матрицу

        P1 = 0x1F;

        //Игнорируем дребезг

        DelayMS(10);

        //Ждем отпускания

        while(!P3\_3);

        //Игнорируем дребезг

        DelayMS(10);

        //Меняем режим

        mode++;

        if (mode == 4)

            mode = 0;

        //Устанавливаем правильные цифры в зависимости от режима

        setDigits();

    }

    //Сброс секундомера

    if (!P3\_4)

    {

        //Сброс значений

        DelayMS(10);

        time10ms=0;

        //Время в секундах

        timeSec = 0;

        //Время в минутах

        timeMin = 0;

        //Время в часах

        timeHour = 0;

        DelayMS(10);

        setDigits();

    }

}

void main(void){

    //Инициализируем таймер

    T0init();

    //Устанавливаем цивры (0, 0)

       setDigits();

    while(1){

        //Проверяем не нажал ли пользователь на кнопку

        buttonsCheck();

        //Делаем паузу

        delayFOR(10);

        //Меняем столбец

        nextColumn();

    }

}