МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
Белорусский национальный технический университет

Кафедра «Техническая эксплуатация автомобилей»

ПРАКТИКУМ ПО МИКРОПРОЦЕССОРНЫМ СИСТЕМАМ УПРАВЛЕНИЯ АВТОМОБИЛЕМ

Практикум
для обучающихся по специальностям 1-37 01 06
«Техническая эксплуатация автомобилей (по направлениям)»,
1-37 01 07 «Автосервис»

Рекомендовано учебно-методическим объединением по образованию в области транспорта и транспортной деятельности

Минск
БНТУ
2022

УДК 629.33.05:004.31-181.48 (076.5)
ББК 39.33-04я7
П69

А в т о р ы :
В. С. Ивашко, *А. С. Гурский*,
В. С. Смольская, *И. А. Серебряков*

Р е ц е н з е н т ы :
доцент кафедры «Технология и организация технического сервиса» Белорусского государственного аграрного технического университета, канд. техн. наук *А. С. Сай*;
кафедра моделирования и проектирования учреждения образования Белорусский государственный аграрный технический университет

Практикум по микропроцессорным системам управления автомобилем : практикум для обучающихся по специальностям 1-37 01 06 «Техническая эксплуатация автомобилей (по направлениям)», 1-37 01 07 «Автосервис» / А. М. Ивашко [и др.]. – Минск : БНТУ, 2022. – 45 с.
П69
ISBN 978-985-550-793-3.

Лабораторный практикум состоит из четырех лабораторных работ. В них отражено устройство и назначение стенда НТЦ-02.05.1 «Электроника с МПСО». Практикум предназначен для проведения лабораторных работ по дисциплинам «Электронные системы управления автомобилем», «Интеллектуальные системы управления автомобилем».

УДК 629.33.05:004.31-181.48 (076.5)
ББК 39.33-04я7

ISBN 978-985-550-793-3

© Белорусский национальный технический университет, 2022

Лабораторная работа № 1

НАЗНАЧЕНИЕ, УСТРОЙСТВО И ПРИНЦИП РАБОТЫ МИКРОПРОЦЕССОРА

Цель работы: изучить назначение, устройство и принцип работы микропроцессора.

Оборудование: стенд НТЦ-02.05.1 «Электроника с МПСО».

Основные сведения

Внешний вид лабораторного стенда представлен на рис. 1.1.

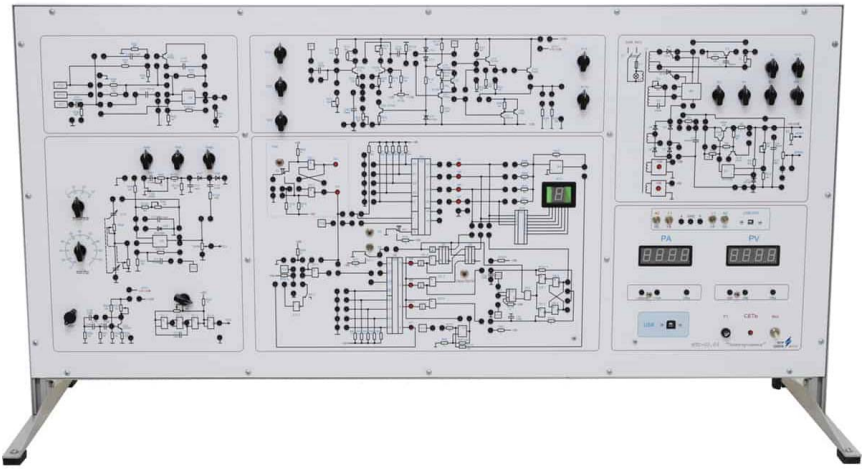

Рис. 1.1. Внешний вид лабораторного стенда

На лицевой панели изображены электрические схемы объектов исследования. Все схемы, представленные на панели, разбиты на группы в соответствии с тематикой проводимых лабораторных работ. На панели установлены коммутационные гнезда, цифровые индикаторы, коммутационная аппаратура, а также органы управления, позволяющие изменять параметры элементов при проведении лабораторных работ.

Функционально лицевая панель разбита на шесть блоков:

1. Блок питания. В пределах этого блока можно выполнить лабораторные работы, целью которых является исследование напряжения, компенсационного стабилизатора напряжения, активных и пассивных сглаживающих фильтров, управляемого тиристорного выпрямителя.

2. Блок для исследования схем на биполярных транзисторах. Блок позволяет исследовать классические типы усилительных каскадов: однокаскадный усилитель (с общим эмиттером), дифференциальный усилитель, усилитель мощности и эмиттерный повторитель (схема с общим коллектором). Все схемы являются составными частями схемы усилителя мощности. В качестве источника входного сигнала служит выходной сигнал RC-автогенератора на операционном усилителе с мостом Вина. Напряжение питания платы стабилизированное и может быть как фиксированным 12 В, так и регулируемым в пределах 0÷16 В.

3. Блок для исследования схем на основе операционных усилителей. Блок позволяет исследовать схемы на ОУ: стандартные схемы усиления гармонических сигналов (инвертирующего и неинвертирующего усилителя); повторителя сигнала; суммирования постоянных напряжений; дифференцирования и интегрирования входного сигнала; сравнения постоянных напряжений (компаратор); триггера Шмидта. Входными сигналами могут быть постоянный сигнал с регулируемого источника напряжения или сигнал с RC-автогенератора на ОУ.

4. Блок для исследования генераторов сигналов. Блок позволяет проводить исследования: RC-генератора на биполярном транзисторе, RC-автогенератора на ОУ с мостом Вина с АРУ и без АРУ, мультивибратора на ОУ, ждущего мультивибратора на ОУ, RC-генератора на логических элементах.

5. Блок для исследования схем цифровой электроники. Блок позволяет исследовать: логические элементы НЕ, И-НЕ, И, синхронный RS-триггер, D-триггер, JK-триггер, T-триггер

на логических элементах, схемы регистров, счетчиков и дешифраторов в интегральном исполнении, ЦАП с матрицей резисторов, АЦП последовательного приближения.

6. Блок для исследования микропроцессора и программируемой логической интегральной микросхемы (ПЛИС). Блок предоставляет возможность исследовать: микропроцессор (изучение архитектуры процессора, программирование микропроцессора с ПК, работа со светодиодной четырехразрядной индикацией, работа в качестве таймера, совместная работа с ПЛИС и т. д.); программируемую логическую интегральную микросхему (ПЛИС) (изучение архитектуры ПЛИС, программирование с персонального компьютера или ноутбука, реализация устройств двоичной логики, в том числе устройств, используемых в стенде на дискретных элементах, работа с микропроцессором).

На рис. 1.2 представлена блок-схема наиболее распространенного микроконтроллера. На рис. 1.3 отображены варианты расположения выводов микроконтроллеров. В табл. 1.1 и 1.2 перечислены соответственно команды микроконтроллера и назначение его выводов.

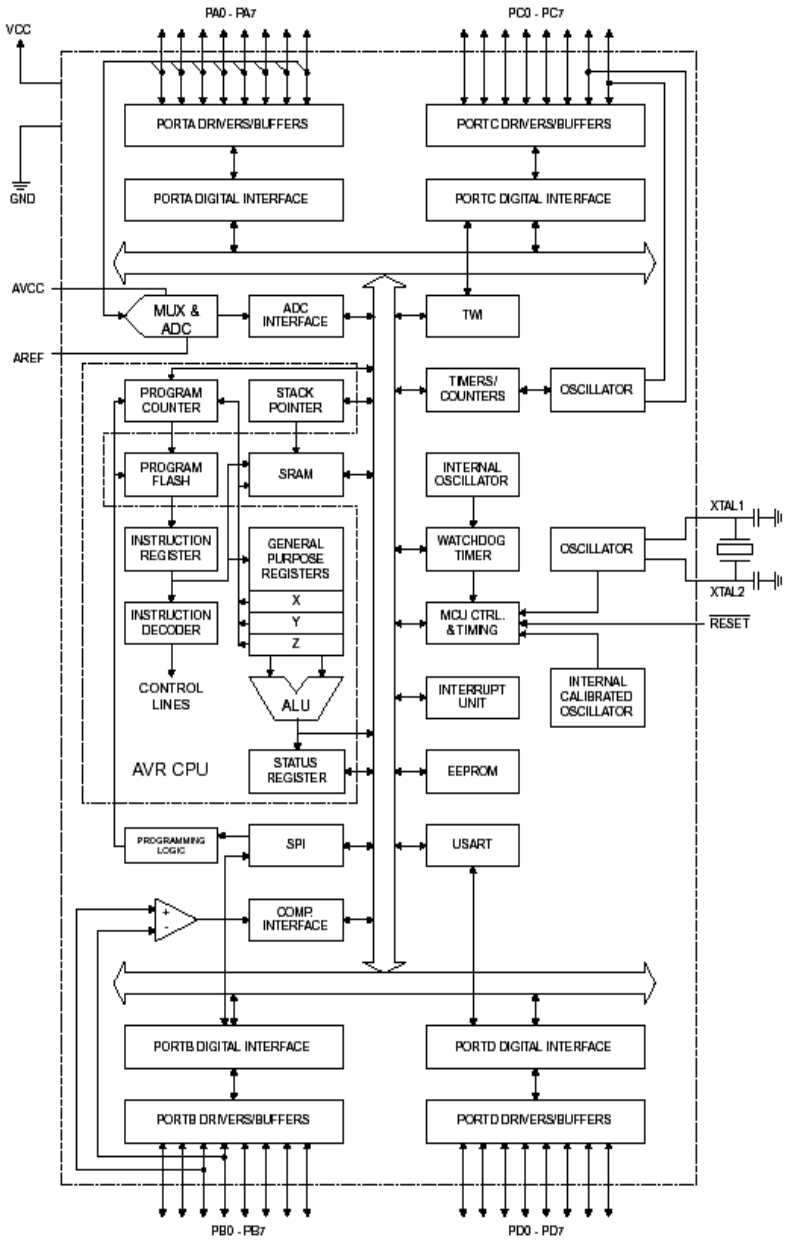

Рис. 1.2. Блок-схема RISC микроконтроллера семейства AVR

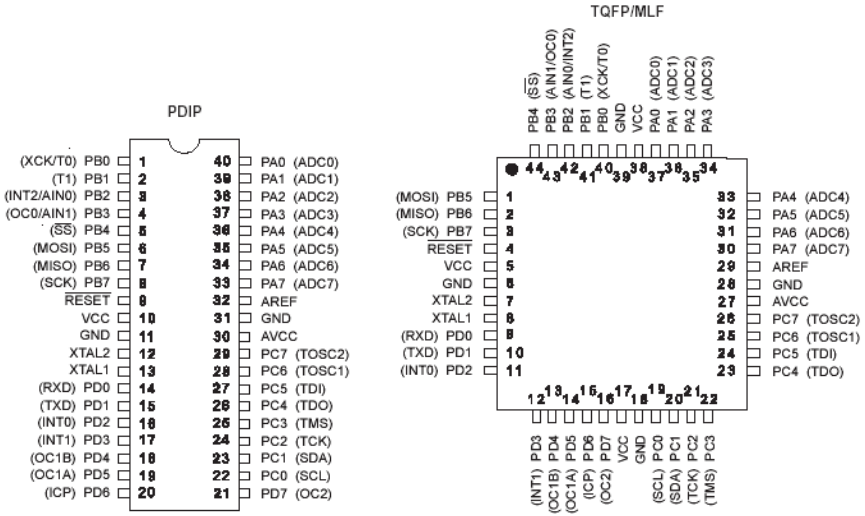

Рис. 1.3. Варианты расположения выводов микроконтроллеров

Таблица 1.1

Система команд 8-разрядных RISC-микроконтроллеров семейства AVR: Принятые обозначения

Обозначение	**Функция**
SREG:	Регистр статуса
C:	Флаг переноса
Z:	Флаг нулевого значения
N:	Флаг отрицательного значения
V:	Флаг-указатель переполнения дополнения до двух
H:	Флаг полупереноса
T:	Флаг пересылки, используемый командами BLD и BST
I:	Флаг разрешения/запрещения глобального прерывания
Регистры и операнды	
Rd:	Регистр назначения (и источник) в регистровом файле

Продолжение табл. 1.1

Обозначение	Функция
Rr:	Регистр источник в регистровом файле
R:	Результат выполнения команды
K:	Литерал или байт данных (8 бит)
k:	Данные адреса константы для счетчика программ
b:	Бит в регистровом файле или I/O регистр (3 бита)
s:	Бит в регистре статуса (3 бита)
X, Y, Z:	Регистр косвенной адресации (X = R27:R26, Y = R29:R28, Z = R31:R30)
P:	Адрес I/O порта
q:	Смещение при прямой адресации (6 бит)
I/O регистры	
RAMPX, RAMPY, RAMPZ:	Регистры связанные с X, Y и Z регистрами, обеспечивающие косвенную адресацию всей области СОЗУ микроконтроллера с объемом СОЗУ более 64 Кбайт
Стек	
STACK:	Стек для адреса возврата и опущенных в стек регистров
SP:	Указатель стека
Флаги	
+:	Флаг, на который воздействует команда
0:	Очищенный командой Флаг
1:	Установленный командой флаг
–:	Флаг, на который не воздействует команда
ADC	Сложить с переносом
ADD	Сложить без переноса
ADIW	Сложить непосредственное значение со словом
AND	Выполнить логическое AND
ANDI	Выполнить логическое AND с непосредственным значением
ASR	Арифметически сдвинуть вправо
BCLR	Очистить флаг
BLD	Загрузить флаг T в бит регистра

Продолжение табл. 1.1

Обозначение	Функция
BRBC	Перейти, если бит в регистре статуса очищен
BRBS	Перейти, если бит в регистре статуса установлен
BRCC	Перейти, если флаг переноса очищен
BRCS	Перейти, если флаг переноса установлен
BREQ	Перейти, если равно
BRGE	Перейти, если больше или равно (с учетом знака)
BRHC	Перейти, если флаг полупереноса очищен
BRHS	Перейти, если флаг полупереноса установлен
BRID	Перейти, если глобальное прерывание запрещено
BRIE	Перейти, если глобальное прерывание разрешено
BRLO	Перейти, если меньше (без знака)
BRLT	Перейти, если меньше чем (со знаком)
BRMI	Перейти, если минус
BRNE	Перейти, если не равно
BRPL	Перейти, если плюс
BRSH	Перейти, если равно или больше (без знака)
BRTC	Перейти, если флаг T очищен
BRTS	Перейти, если флаг T установлен
BRVC	Перейти, если переполнение очищено
BRVS	Перейти, если переполнение установлено
BSET	Установить флаг
BST	Переписать бит из регистра во флаг T
CALL	Выполнить длинный вызов подпрограммы
CBI	Очистить бит в регистре I/O
CBR	Очистить биты в регистре
CLC	Очистить флаг переноса
CLH	Очистить флаг полупереноса
CLI	Очистить флаг глобального прерывания
CLN	Очистить флаг отрицательного значения
CLR	Очистить регистр
CLS	Очистить флаг знака
CLT	Очистить флаг T
CLV	Очистить флаг переполнения

Продолжение табл. 1.1

Обозначение	Функция
CLZ	Очистить флаг нулевого значения
COM	Выполнить дополнение до единицы
CP	Сравнить
CPC	Сравнить с учетом переноса
CPI	Сравнить с константой
CPSE	Сравнить и пропустить, если равно
DEC	Декрементировать
EOR	Выполнить исключающее OR
ICALL	Вызвать подпрограмму косвенно
IJMP	Перейти косвенно
IN	Загрузить данные из порта I/O в регистр
INC	Инкрементировать
FMUL	Дробное незнаковое умножение
FMULS	Дробное умножение со знаком
FMULSU	Дробное умножение знакового с незнаковым
JMP	Перейти
LD Rd,X	Загрузить косвенно
LD Rd,X+	Загрузить косвенно, инкрементировав впоследствии
LD Rd,–X	Загрузить косвенно, декрементировав предварительно
LDI	Загрузить непосредственное значение
LDS	Загрузить непосредственно из СОЗУ
LPM	Загрузить байт памяти программ
LSL	Логически сдвинуть влево
LSR	Логически сдвинуть вправо
MOV	Копировать регистр
MUL	Перемножить
NEG	Выполнить дополнение до двух
NOP	Выполнить холостую команду
OR	Выполнить логическое OR
ORI	Выполнить логическое OR с непосредственным значением
OUT	Записать данные из регистра в порт I/O

Продолжение табл. 1.1

Обозначение	Функция
POP	Загрузить регистр из стека
PUSH	Поместить регистр в стек
RCALL	Вызвать подпрограмму относительно
RET	Вернуться из подпрограммы
RETI	Вернуться из прерывания
RJMP	Перейти относительно
ROL	Сдвинуть влево через перенос
ROR	Сдвинуть вправо через перенос
SBC	Вычесть с переносом
SBCI	Вычесть непосредственное значение с переносом
SBI	Установить бит в регистр I/O
SBIC	Пропустить, если бит в регистре I/O очищен
SBIS	Пропустить, если бит в регистре I/O установлен
SBIW	Вычесть непосредственное значение из слова
SBR	Установить биты в регистре
SBRC	Пропустить, если бит в регистре очищен
SBRS	Пропустить, если бит в регистре установлен
SEC	Установить флаг переноса
SEH	Установить флаг полупереноса
SEI	Установить флаг глобального прерывания
SEN	Установить флаг отрицательного значения
SER	Установить все биты регистра
SES	Установить флаг знака
SET	Установить флаг T
SEV	Установить флаг переполнения
SEZ	Установить флаг нулевого значения
SLEEP	Установить режим SLEEP
ST X,Rr	Записать косвенно
ST Y,Rr	Записать косвенно из регистра в СОЗУ с использованием индекса Y
ST Z,Rr	Записать косвенно из регистра в СОЗУ с использованием индекса Z
STS	Загрузить непосредственно в СОЗУ

Окончание табл. 1.1

Обозначение	Функция
SUB	Вычесть без переноса
SUBI	Вычесть непосредственное значение
SWAP	Поменять нибблы местами
TST	Проверить на ноль или минус
WDR	Сбросить сторожевой таймер

Таблица 1.2

Назначение выводов

Обозначение	Назначение
VCC	Напряжение питания
GND	Земля
Port A (PA7..PA0)	8-разрядный двунаправленный порт I/O. К выходам порта могут быть подключены встроенные нагрузочные резисторы (отдельно к каждому разряду). Выходные буферы обеспечивают втекающий ток 20 мА и способны напрямую управлять LED индикатором. При использовании выводов порта в качестве входов, а также при установке внешним сигналом в низкое состояние ток будет вытекать только при подключенных встроенных нагрузочных резисторах. Порт А, при наличии внешней SRAM, используется в качестве мультиплексируемой шины адреса/данных
Port B (PB7..PB0)	8-разрядный двунаправленный порт I/O со встроенными нагрузочными резисторами. Выходные буферы обеспечивают втекающий ток 20 мА. При использовании выводов порта в качестве входов, а также при установке внешним сигналом в низкое состояние ток будет вытекать только при подключенных встроенных нагрузочных резисторах. Порт В используется также при реализации различных специальных функций

Продолжение табл. 1.2

Обозначение	Назначение
Port C (PC7..PC0)	8-разрядный порт выхода. Выходные буферы обеспечивают втекающий ток 20 мА. Порт С используется так же, как выходы адреса при использовании внешней SRAM
Port D (PD7..PD0)	8-разрядный двунаправленный порт I/O со встроенными нагрузочными резисторами. Выходные буферы обеспечивают втекающий ток 20 мА. При использовании выводов порта в качестве входов и установке внешним сигналом в низкое состояние ток будет вытекать только при подключенных встроенных нагрузочных резисторах
Port E (PE7..PE0)	8-разрядный двунаправленный порт I/O со встроенными нагрузочными резисторами. Выходные буферы обеспечивают втекающий ток 20 мА. При использовании выводов порта в качестве входов и установке внешним сигналом в низкое состояние, вытекающий через них ток обеспечивается только при подключенных встроенных нагрузочных резисторах
Port F (PF7..PF0)	8-разрядный порт входа. Входы порта используются так же, как аналоговые входы аналого-цифрового преобразователя
RESET	Вход сброса. Для выполнения сброса необходимо удерживать низкий уровень на входе в течение двух машинных циклов
XTAL1	Вход инвертирующего усилителя генератора и вход схемы встроенного генератора тактовой частоты
XTAL2	Выход инвертирующего усилителя генератора
TOSC1	Вход инвертирующего усилителя генератора таймера/счетчика
TOSC2	Выход инвертирующего усилителя генератора таймера/счетчика
WR	Строб записи внешней SRAM
RD	Строб чтения внешней SRAM

Окончание табл. 1.2

Обозначение	Назначение
ALE	Строб разрешения фиксации адреса, используемый для разрешения внешней памяти. Строб ALE используется для фиксации младшего байта адреса в защелках адреса в течение первого цикла обращения; в течение второго цикла обращения, при обращении к данным, используются выводы AD0–AD7
AVCC	Напряжение питания аналого-цифрового преобразователя. Вывод подсоединяется к внешнему VCC через низкочастотный фильтр
AREF	Вход аналогового напряжения сравнения для аналого-цифрового преобразователя. На этот вывод, для обеспечения работы аналого-цифрового преобразователя, подается напряжение в диапазоне между AGND и AVCC
AGND	Этот вывод должен быть подсоединен к отдельной аналоговой земле, если плата оснащена ею. В ином случае вывод подсоединяется к общей земле
PEN	Вывод разрешения программирования в низковольтном последовательном режиме программирования. При удержании этого вывода на низком уровне во время сброса по включении питания, прибор перейдет в режим программирования по последовательному каналу

Конфигурация памяти

Микроконтроллеры ATmega32 поддерживают две конфигурации, как показано в табл. 1.3.

Таблица 1.3

Конфигурация памяти

Конфигурация	Встроенная SRAM данных	Внешняя SRAM данных
A	4000 байт	Нет
B	4000 байт	До 64 Кбайт (1)

По первым 4096 адресам памяти данных размещаются регистровый файл, пространство памяти I/O и встроенная SRAM данных. Из них первые 96 адресов занимают регистровый файл и пространство памяти I/O, в следующих 4000 адресов размещается встроенная SRAM (рис. 1.4).

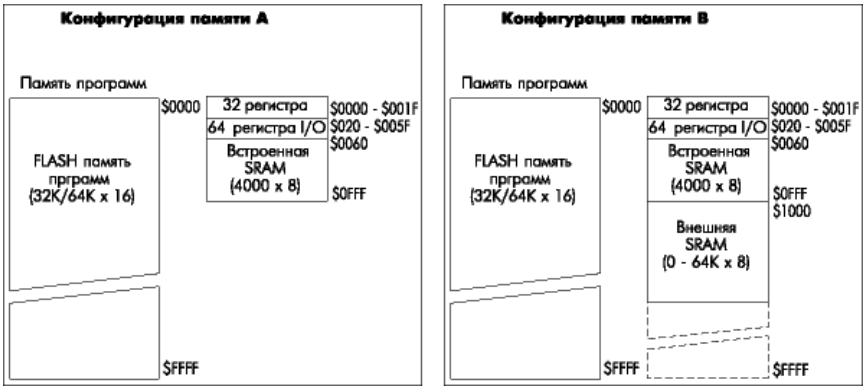

Рис. 1.4. Конфигурация памяти

Микроконтроллеры конфигурации B позволяют использовать дополнительную внешнюю память данных. Внешняя память будет адресоваться оставшимся до 64K пространством адресов, т. е. оно будет начинаться следом за пространством адресов встроенной SRAM. При использовании внешней SRAM емкостью 64K будут потеряны 4K внешней памяти, поскольку адреса этого объема будут заняты встроенной памятью.

При обращении по адресам памяти данных за пределами встроенной SRAM используются те же команды, что для обращения к встроенной SRAM. При обращении к встроенной памяти данных выводы стробов управления внешней памятью данных (RD и WR) остаются неактивными во время всего цикла обращения.

Работа внешней SRAM разрешается установкой бита SRE в регистре MCUCR. По сравнению с обращением к встроен-

ной памяти данных, обращение к внешней памяти данных требует дополнительного цикла на каждый байт. Это означает, что для выполнения команд LD, ST, LDS, STS, PUSH и POP требуется дополнительный тактовый цикл. Если стек размещен во внешней SRAM, то прерывания, вызов подпрограмм и возвраты потребуют двух дополнительных циклов, поскольку в стеке будет опускаться и подниматься содержимое двухбайтового счетчика команд. Если интерфейс с внешней SRAM используется с состоянием ожидания, то на каждый байт необходимо еще два дополнительных тактовых цикла. Это приводит к следующему эффекту. Командам пересылки данных необходимо два дополнительных тактовых цикла, тогда как при обработке прерывания, вызове подпрограммы и при возврате из подпрограмм потребуется на четыре тактовых цикла больше, чем это указано в описании системы команд.

При адресации памяти данных используются пять режимов адресации: непосредственная адресация, косвенная со смещением, косвенная, косвенная с преддекрементом и косвенная с постдекрементом. Регистры с R26 по R31 регистрового файла работают как X, Y и Z регистры-указатели косвенной адресации.

Косвенной адресации со смещением доступны 63 адреса относительно базовых адресов, находящихся в регистрах Y или Z. При использовании косвенной адресации с автоматическим преддекрементом и постдекрементом автоматически декрементируются и инкрементируются адреса, записанные в регистры X, Y и Z. Всеми этими режимами перекрывается все адресное пространство данных, включая 32 регистра общего назначения и 64 регистра I/O. Подробное описание всех режимов адресации приведено ниже.

Режимы адресации памяти программ и данных

При обращении к Flash-памяти программ и памяти данных (SRAM, регистровому файлу и памяти I/O) AVR Enhanced RISC микроконтроллерами ATmega603/103 используются мощ-

ные и эффективные режимы адресации. В данном разделе описываются режимы адресации, поддерживаемые AVR-архитектурой. На рисунках OP обозначает часть слова команды, соответствующую операционному коду.

Время обращения к памяти и тактирование выполнения команд

В данном разделе описаны основные принципы тактирования обращений при выполнении команд и обращений к встроенной памяти.

AVR CPU тактируется системным тактовым сигналом SystemClock O, формируемым посредством внешнего кварцевого кристалла. Внутреннее деление не используется.

На рис. 1.5 представлен процесс параллельных выборки и выполнения команд, обеспечиваемые Гарвардской архитектурой, и концепция регистрового файла быстрого доступа. Это базовый принцип конвейерной обработки, обеспечивающий удельную производительность 1 MIPS/МГц при соответствующих результатах стоимости функции, количества функций на один такт и количества функций на единицу потребляемой мощности.

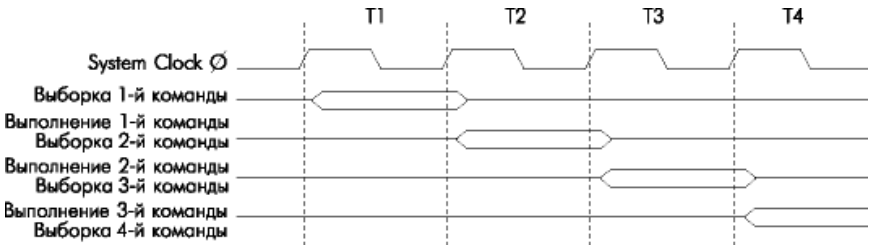

Рис. 1.5. Параллельные выборка и выполнение команд

На рис. 1.6 представлен принцип внутреннего тактирования регистрового файла. В течение одного тактового цикла выполнения операции ALU использует два операнда регистров и результат возвращает в регистр назначения.

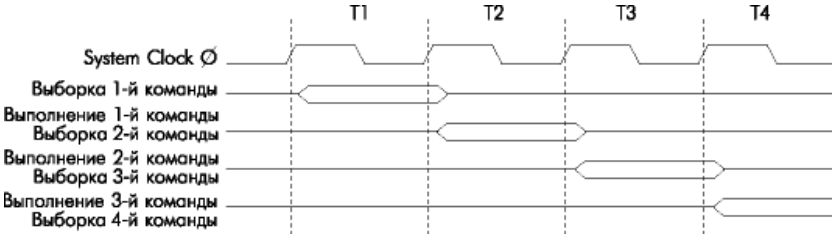

Рис. 1.6. Одноцикловая работа ALU

На рис. 1.7 показано обращение к встроенной SRAM данных за два тактовых цикла.

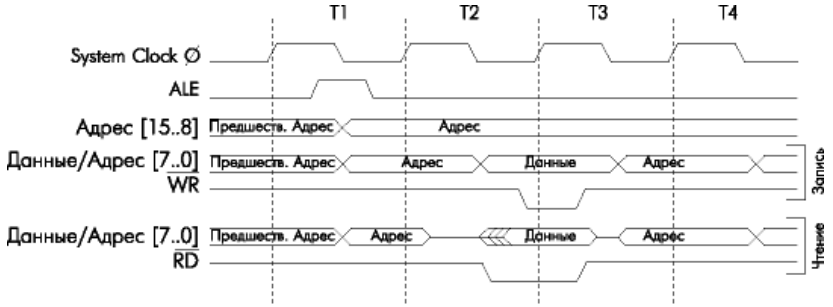

Рис. 1.7. Циклы обращения к внешней SRAM данных
без состояния ожидания (WaitState)

На рис. 1.8 показано обращение к внешней SRAM данных при установленном бите состояния ожидания.

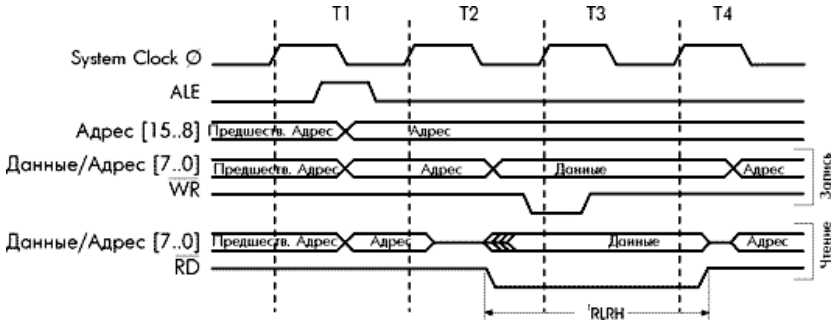

Рис. 1.8. Циклы обращения к внешней SRAM данных
с состоянием ожидания (WaitStateactive)

Выполнение работы

В комплекте стенда поставляется МК ATmega32-16AI (микроконтроллер 8-bit, архитектура AVR RISC, память 32 кБт, EEPROM 1 кБт, SRAM 2 кБт, частота 0–16 МГц, V_{cc} = 4,5–5,5 В), содержащий одну базовую программу для демонстрации своих возможностей. Микроконтроллер стенда может работать в двух режимах: «ПРОГРАММИРОВАНИЕ» и «РАБОТА ПРОГРАММЫ».

Для запуска подпрограмм базовой программы в начале следует подключить стенд к сети и включить стенд. Для включения стенда необходимо включить тумблер питания «Вкл», при этом должен загореться светодиод «СЕТЬ» (см. рис. 1.9).

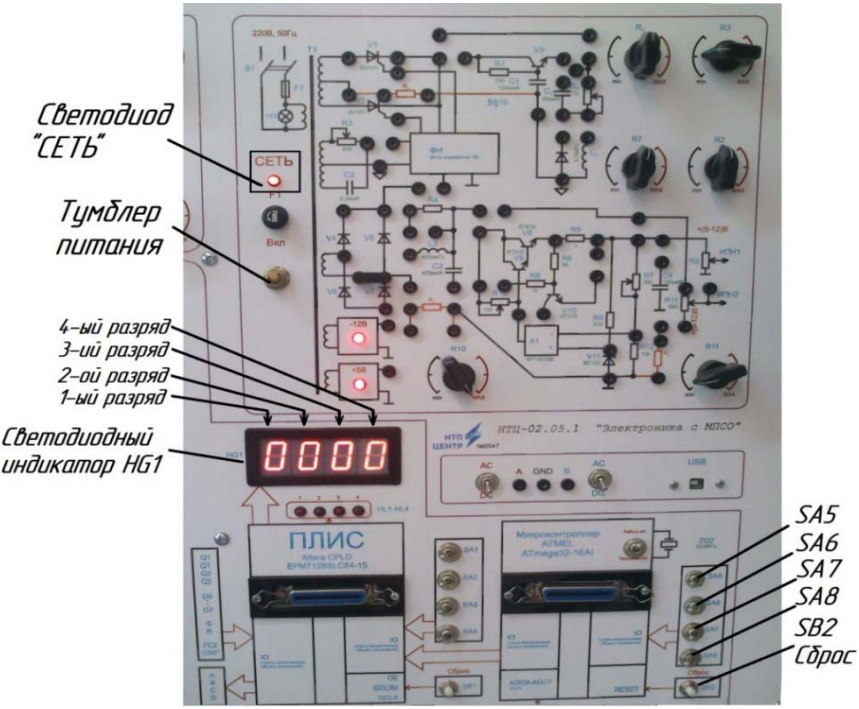

Рис. 1.9. Расположение основных компонентов стенда

Чтобы вернуться к выбору подпрограммы, необходимо перезапустить микроконтроллер, для чего нажать на кнопку SB2 (Сброс).

Перечень демонстрационных подпрограмм отображен в табл. 1.3.

Таблица 1.3

Перечень подпрограмм

<table>
<tr><th>№ п/п</th><th>Подпрограмма</th></tr>
<tr><td>0</td><td>Отображает состояние тумблеров SA5-SA8 на индикатор HGI</td></tr>
<tr><td>1</td><td>Демонстрирует возможности четырехразрядного светодиодного индикатора HG1</td></tr>
<tr><td>2</td><td>Таймер с интервалом счета 1 секунда, в пределах [–128; +128]. Направление счета соответствует след. таблице истинности:
<table>
<tr><th>SA6</th><th>SA5</th><th>Направление</th></tr>
<tr><td>0</td><td>0</td><td>СТОП</td></tr>
<tr><td>0</td><td>1</td><td>ВВЕРХ</td></tr>
<tr><td>1</td><td>0</td><td>ВНИЗ</td></tr>
<tr><td>1</td><td>1</td><td>СТОП</td></tr>
</table></td></tr>
<tr><td>3</td><td>Счетчик, изменяющий значение переменной на 1. Счет ВВЕРХ производится по изменению положения тумблера SA5 из положения лог.0 в лог.1. Счет ВНИЗ по изменению положения тумблера SA6 из положения 0 в 1</td></tr>
</table>

После включения стенда индикатор HG1 начинает светиться, отображает состояние тумблеров SA5–SA7 на первых трех разрядах в виде значений 0 или 1 (см. рис. 1.10). Необходимо измерить время загрузки процессора и занести в журнал наблюдений (табл. 1.4).

Далее измеряем время перезагрузки процессора: для этого нажимаем сброс и записываем измерение в журнал наблюдений.

Измеряем частоту миганий индикатора: количество миганий в минуту делим на 60. По техническим условиям индикатор должен мигать с частотой 4 Гц.

Рис. 1.10. Соответствие тумблеров SA5–SA7 разрядам светодиодного четырехсегментного индикатора.

Таблица 1.4

Журнал наблюдений

Параметр	Результат измерений	Технические условия
Время загрузки процессора		
Время перезагрузки процессора		
Частота мигания индиктора		
Соответствие символов в двоичной и десятичной системе исчисления		

Последний четвертый разряд отображает число в десятичной системе исчисления, соответствующее числу в двоичной системе счисления, отображенному на предыдущих трех. Не-

обходимо занести в журнал наблюдений эти числа и установить их соответствие.

Первые три разряда горят постоянно, четвертый мигает. Таким образом, на четвертом разряде отображается номер демонстрационной подпрограммы. Чтобы начать выполнение подпрограммы, необходимо изменить состояние тумблера SA8 (переключить тумблер в другое положение).

Чтобы вернуться к выбору подпрограммы, необходимо перезапустить микроконтроллер, для чего необходимо нажать на кнопку SB2 (Сброс).

Лабораторная работа № 2

ВЗАИМОДЕЙСТВИЕ МИКРОПРОЦЕССОРА С ПЕРИФЕРИЙНЫМИ УСТРОЙСТВАМИ НА ПРИМЕРЕ СВЕТОДИОДНОГО ИНДИКАТОРА

Цель работы: изучить взаимодействие микропроцессоров с периферийными устройствами на примере светодиодного индикатора.

Оборудование: стенд НТЦ-02.05.1 Электроника с «МПСО».

Основные сведения

К периферийным устройствам относятся преобразователи, компараторы, системы обмена информацией и устройства отображения информации (рис. 2.1).

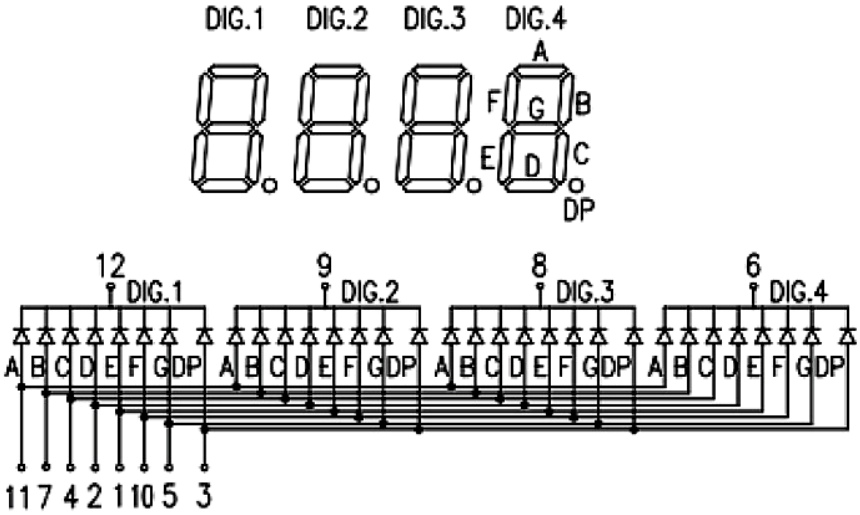

Рис. 2.1. Описание 4-разрядного 7-сементного индикатора

Аналоговый компаратор

Аналоговый компаратор сравнивает уровни на положительном выводе PE2 (AC+) и отрицательном выводе PE3 (AC–). При напряжении на положительном выводе PE2 (AC+) большем, чем напряжение на отрицательном выводе PE3 (AC–), выход аналогового компаратора ACO устанавливается в состояние 1. Выход компаратора может быть использован для управления входом захвата таймера/счетчика. Кроме того, компаратор может формировать свой запрос прерывания. Пользователь может задать формирование запроса на прерывание по нарастающему или падающему фронту или по переключению. Блок-схема аналогового компаратора, со схемами обрамления, показана на рис. 2.2.

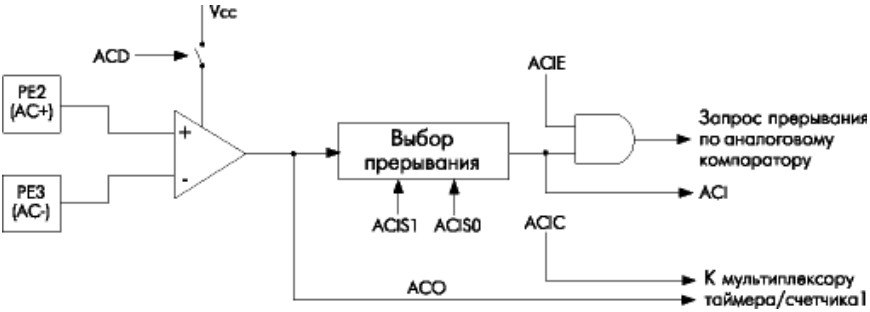

Рис. 2.2. Блок-схема аналогового компаратора

Аналого-цифровой преобразователь (Analog to Digital Converter)

Основные характеристики:

– разрешение 10 разрядов;
– точность ±1/2 LSB;
– время преобразования 70...280 мс;
– 8 мультиплексируемых каналов входа;
– режимы циклического и однократного преобразования;

– прерывание по завершению ADC-преобразования;

– устройство подавления шумов Sleep-режима.

Микроконтроллеры ATmega603/103 оснащены 10-разрядным ADC последовательного приближения. ADC подсоединен к 8-канальному аналоговому мультиплексору, позволяющему использовать любой вывод порта F в качестве входа ADC. ADC содержит усилитель выборки/хранения, удерживающий напряжение входа ADC во время преобразования на неизменном уровне. Блок-схема ADC представлена на рис. 2.3. Для питания ADC используются два отдельных вывода: AVCC и AGND. Вывод AGND должен быть подсоединен к GND, а напряжение AVCC не должно отличаться от напряжения VCC более, чем на 0,4 В.

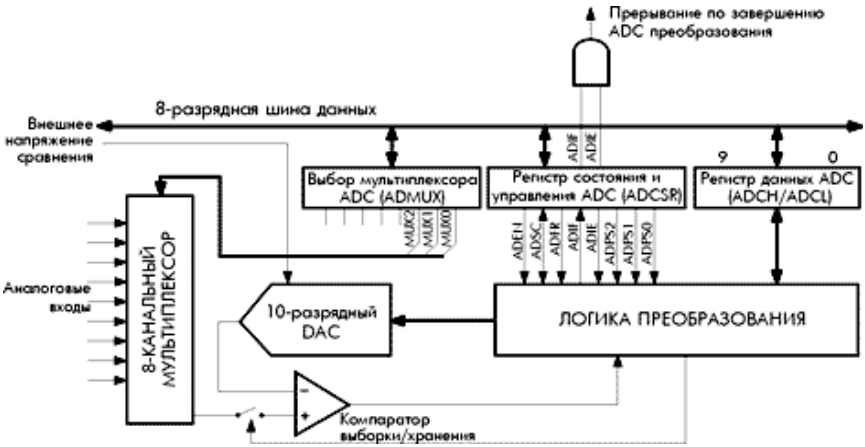

Рис. 2.3. Блок-схема аналого-цифрового преобразователя

Внешнее напряжение сравнения подается на вывод AREF и должно быть в диапазоне от 2,7 В до AVCC.

Порт А.

Порт А является 8-разрядным двунаправленным I/O портом, который оснащен встроенными нагрузочными резисторами.

Взаимодействие с портом А осуществляется тремя расположенными в пространстве I/O памяти данных регистрами: регистром данных – PORTA, $1B($3B), регистром направления данных – DDRA, $1A($3A) и регистром адресов выводов входа – PINA, $19($39). Регистр адресов выводов входа порта А обеспечивает возможность только чтения, регистры данных и направления данных порта А обеспечивают возможность и чтения, и записи. Все выводы порта А оснащены индивидуально подключаемыми встроенными нагрузочными резисторами.

Выходные буферы выводов порта А обеспечивают втекающий ток до 40 мА, что достаточно для прямого управления LED-дисплеями. Если выводы с PA0 по PA7 используются в качестве входов и внешним сигналом удерживаются на низком уровне, то вытекающий ток обеспечивается подключением внутренних нагрузочных резисторов. Выводы порта А могут выполнять дополнительную к основной функцию обеспечения взаимодействия с внешней дополнительной SRAM данных – они могут быть сконфигурированы как младшие разряды шины адреса/данных внешней SRAM данных.

Дополнительная функция включается установкой бита SRE (разрешение внешней SRAM) в регистре управления MCU (MCUCR), при этом установки регистра направления данных игнорируются. PINA-адрес выводов входа порта А не является регистром в полном смысле этого слова, эти адреса обеспечивают считывание физического состояния каждого вывода порта. При считывании PORTA считывается состояние фиксаторов данных порта А, а при считывании PINA считываются непосредственно логические состояния выводов.

Схемотехника порта А представлена на рис. 2.4.

Каждый из выводов порта синхронизирован, однако на рисунке защелка синхронизации не показана.

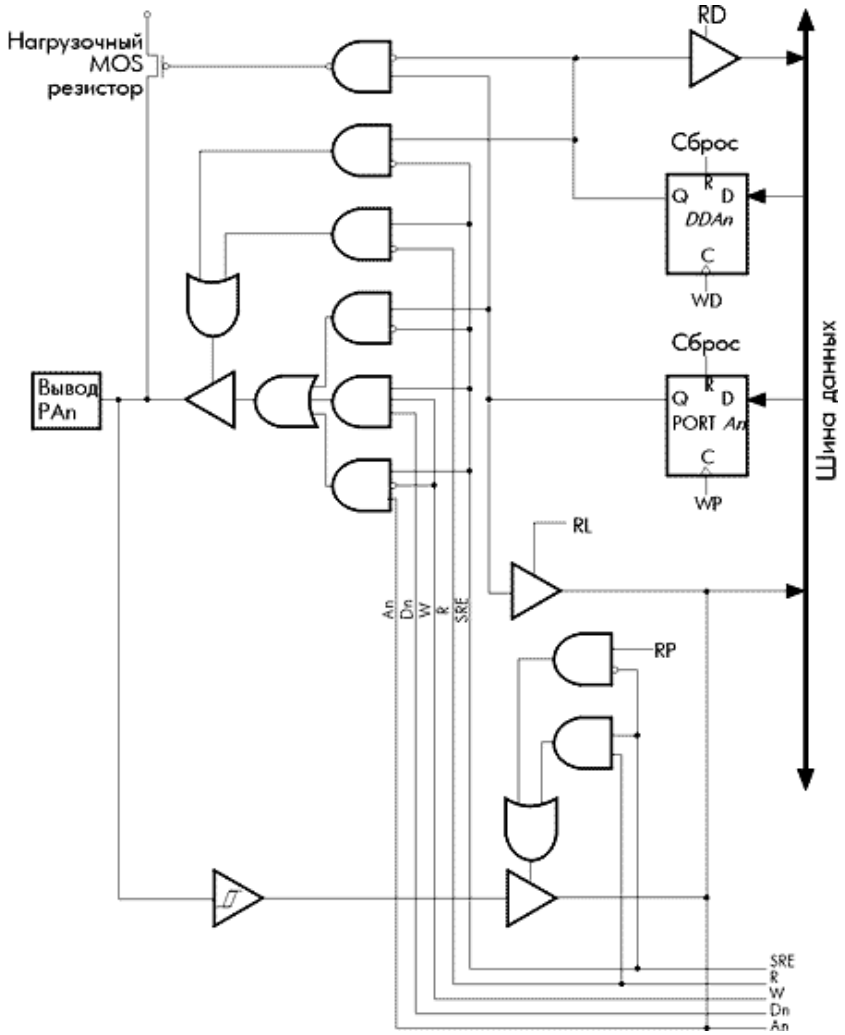

Рис. 2.4. Схема организации вывода порта А (выводы PA0–PA7)

Последовательный периферийный интерфейс – SPI (Serial Peripheral Interface)

Последовательный периферийный интерфейс (SPI) обеспечивает высокоскоростной синхронный обмен данными между

микроконтроллерами ATmega603/103 и периферийными устройствами или между несколькими микроконтроллерами ATmega603/103.

Основные характеристики SPI-интерфейса:

– полнодуплексный 3-проводный синхронный обмен данными;

– режим работы ведущий или ведомый;

– обмен данными с передаваемым первым старшим или младшим битами;

– четыре программируемые скорости обмена данными;

– флаг прерывания по окончании передачи;

– активация из Idle режима (только в режиме ведомого).

Соединения между ведущим и ведомым CPU, использующими SPI-интерфейс, показаны на рис. 2.5. Вывод PB1(SCK) является выходом тактового сигнала ведущего микроконтроллера и входом тактового сигнала ведомого. По записи ведущим CPU данных в SPI-регистр начинает работать тактовый генератор SPI, и записанные данные сдвигаются через вывод выхода PB2(MOSI) ведущего микроконтроллера на вывод входа PB2 (MOSI) ведомого микроконтроллера. После сдвига одного байта тактовый генератор SPI останавливается, устанавливая флаг окончания передачи (SPIF). Если в регистре SPCR будет установлен бит разрешения прерывания SPI (SPIE), то произойдет запрос прерывания. Вход выбора ведомого PB0(SS), для выбора индивидуального SPI устройства в качестве ведомого, устанавливается на низкий уровень. При установке высокого уровня на выводе PB0(SS) порт SPI деактивируется, и вывод PB2(MOSI) может быть использован в качестве вывода входа. Режим ведущий/ведомый может быть установлен и программным способом, установкой или очисткой бита MSTR в регистре управления SPI.

Два сдвиговых регистра ведущего и ведомого микроконтроллеров можно рассматривать как один разнесенный 16-разрядный циклический сдвиговый регистр. При сдвиге данных из ведущего микроконтроллера в ведомый одновременно про-

исходит сдвиг данных из ведомого микроконтроллера в ведущий, т. е. в течение одного цикла сдвига происходит обмен данными между ведущим и ведомым микроконтроллерами.

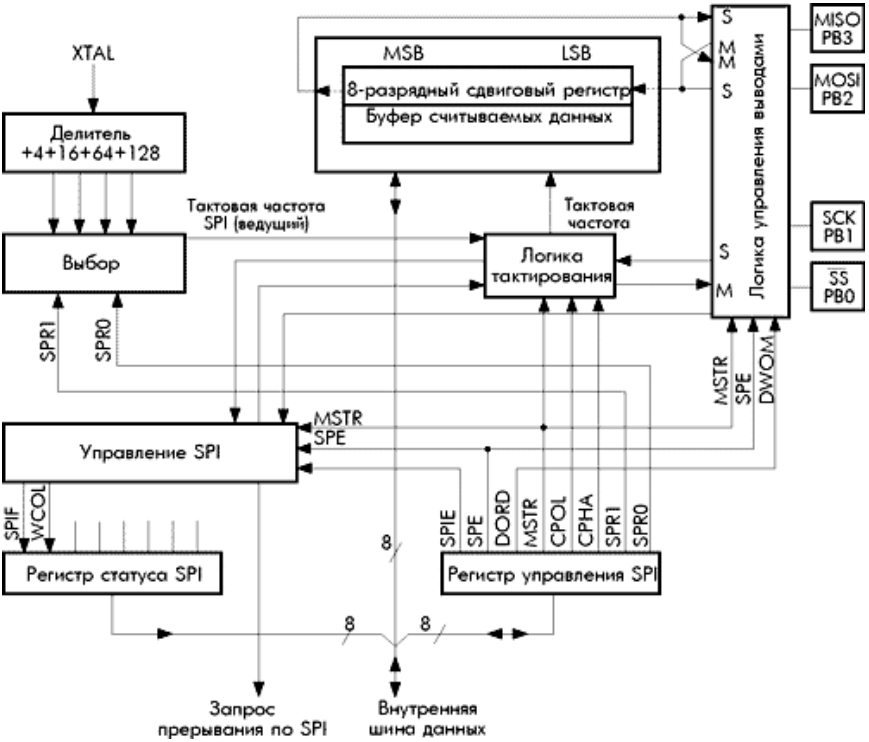

Рис. 2.5. Блок-схема SPI

В системе организовано одиночное буферирование передающей стороны и двойное буферирование на приемной стороне. Это означает то, что передаваемые символы не могут быть записаны в регистр данных SPI прежде, чем будет полностью завершен цикл сдвига.

С другой стороны, при приеме данных принимаемый символ должен быть считан из регистра данных SPI прежде, чем будет завершен прием следующего символа, в противном случае предшествовавший символ будет потерян.

Выполнение работы

Выбор подпрограммы:

1. Нажать кнопку SB2 (Сброс) для сброса предыдущей подпрограммы.

2. Перевести тумблеры SA5, SA6 в нулевое состояние (рычаг тумблера смотрит влево), а тумблер SA7 в состояние «1» (рычаг тумблера смотрит вправо).

3. Для выполнения подпрограммы необходимо перевести тумблер SA8 в другое положение.

После запуска подпрограммы светодиодный индикатор HG1 (рис. 2.6) произведет демонстрацию своих возможностей в виде поочередного включения и выключения различных сегментов индикаторов.

Рис. 2.6. Светодиодный индикатор HG1

Измерить время активации программы с момента сброса записать в журнал наблюдений (табл. 2.1).

Измерить частоту переключения и занести в журнал

Зарисовать символы, отображаемые на индикаторе в журнал наблюдений

Таблица 2.1

Журнал наблюдений

Время активации программы	
Частота переключения	
Символы на индикаторе	

Лабораторная работа № 3

РАБОТА ТАЙМЕРА МИКРОПРОЦЕССОРА

Цель работы: изучить назначение и принцип работы таймера микропроцессора.

Оборудование: стенд НТЦ-0.2.05.1 «Электроника с МПСО».

Основные сведения

В автомобиле очень много устройств, призванных работать временно, то есть не постоянно, а время от времени. Это и различные подогреватели, и указатели поворотов (ленивый указатель поворотов), и турботаймеры, и устройства, включающие камеры заднего хода не сразу, а через какое-то время, то есть с задержкой. В этих случаях используется таймер, который и задает для исполняющего устройства период его работы или отключения.

Таймеры/счетчики

Микроконтроллеры ATmega32 оснащены тремя таймерами/счетчиками общего назначения – двумя 8-разрядными и одним 16-разрядным. Таймер/счетчик, в дополнение к обычному режиму, может тактироваться асинхронно от внешнего генератора. Этот генератор оптимизирован под использование кварцевого кристалла на частоту 32,768 кГц, что позволяет использовать таймер/счетчик как часы реального времени (Real Time Clock – RTC).

Тактовый генератор

XTAL1 и XTAL2 являются, соответственно, входом и выходом инвертирующего усилителя, который с использованием кварцевого кристалла или керамического резонатора работает

как встроенный генератор, как показано на рис. 3.1. При использовании внешнего источника тактовой частоты вывод XTAL2 должен остаться свободным, сигнал подается на вывод XTAL1, как показано на рис. 3.2.

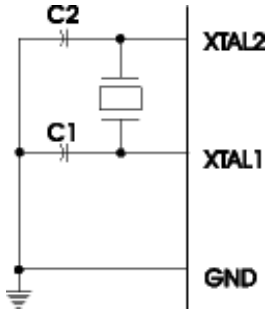

Рис. 3.1. Подсоединение тактового генератора

Рис. 3.2. Подсоединение внешнего источника тактового сигнала

Кварцевый кристалл генератора таймера подсоединяется непосредственно к выводам OSC1 и OSC2. Внешние конденсаторы не требуются. Генератор оптимизирован под часовой кварц с частотой 32,768 кГц. Внешний тактовый сигнал, подаваемый на эти выводы, поступает на усилитель с полосой пропускания 256 кГц.

Таким образом, частота внешнего сигнала должна находиться в диапазоне от 0 до 256 кГц.

Таймер/счетчик оснащен своим собственным предварительным делителем. Таймеры/счетчики 1 и 2 используют выходы ступеней деления общего 10-разрядного предварительного делителя. Эти два таймера/счетчика можно использовать как таймеры со встроенной временной базой или как счетчики, переключаемые по состоянию на внешнем выводе. Содержимое этих 8-разрядных регистров является состоянием таймеров/счетчиков. Оба таймера/счетчика реализованы как счетчики по нарастанию или реверсивные (в ШИМ режиме) счетчики с возможностью чтения/записи. Если в таймер/счетчик записано некоторое значение и выбран источник тактового сигнала, то он продолжит счет с записанного значения с тактовой частотой счетчика.

Обработка прерываний и сброса.

Микроконтроллеры ATmega603/103 используют 23 источника прерывания. Эти прерывания и вектор сброса располагают отдельными программными векторами в пространстве памяти программ. Каждому прерыванию присвоен свой бит разрешения, который должен быть установлен совместно с битом I регистра статуса.

Младшие адреса пространства памяти программ автоматически определяются как векторы сброса и прерываний.

Источники сброса.

Микроконтроллеры ATmega603/103 располагают тремя источниками сигнала сброса:

1. Сброс по включению питания (Power-OnReset). MCU сбрасывается при подключении питания к выводам VCC и GND.

2. Внешний сброс (ExternalReset). MCU сбрасывается, если низкий уровень присутствует на входе более двух циклов XTAL.

3. Сброс по сторожевому таймеру (WatchdogReset). MCU сбрасывается, если истекает период сторожевого таймера и сторожевой таймер разрешен.

В течение сброса все регистры I/O, за исключением регистра статуса MCU, устанавливаются в их начальные состояния,

и программа начинает работу с адреса $0000. По этому адресу должна находиться команда JMP – команда абсолютного перехода к подпрограмме обработки сброса. Если программа никогда не разрешает прерывания, то векторы прерываний не используются, и по этим адресам могут располагаться коды программы.

Логика сброса продемонстрирована на рис. 3.3.

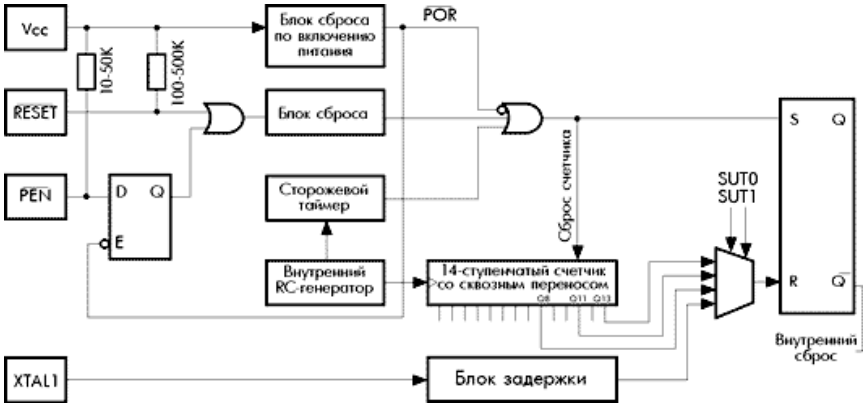

Рис. 3.3. Логика сброса

Обработка прерываний

Микроконтроллеры ATmega603/103 содержат два специальных 8-разрядных регистра масок прерываний: регистр масок внешних прерываний EIMSK (ExternalInterruptMask) и регистр масок прерываний по таймеру/счетчику TIMSK (Timer/CounterInterruptMask). Кроме того, в регистрах управления периферией могут быть организованы и другие биты разрешения и биты масок.

При возникновении прерывания бит I разрешения глобального прерывания (GlobalInterruptEnable) очищается и все прочие прерывания запрещаются. Чтобы разрешить вложенные прерывания, пользовательское ПО может установить бит I внут-

ри подпрограммы обработки прерывания. Выход из подпрограммы обработки прерывания происходит по команде RETI, при этом бит I устанавливается в состояние 1.

Когда счетчик команд указывает вектор подпрограммы обработки прерывания, соответствующий флаг, вызвавший прерывание, аппаратно очищается. Некоторые флаги прерываний можно очистить, записав в соответствующий бит(ы) очищаемого флага логическую единицу.

Внешние прерывания 7 – 4 активируются по выводам INT7 – INT4, если установлен флаг I в SREG и установлена соответствующая маска в EIMSK. Запрос прерывания по логическому уровню или фронтам определяется в табл. 3.1.

Таблица 3.1

Управление опознаванием прерывания

SCX1	ISCX0	Описание
0	0	Запрос прерывания генерируется низким уровнем на INTX
0	1	Зарезервирован
1	0	Запрос прерывания генерируется падающим фронтом на INTX
1	1	Запрос прерывания генерируется нарастающим фронтом на INTX

При изменении битов ISC11/ISC10 прерывание должно быть запрещено путем очистки бита разрешения в регистре GIMSK. В ином случае может произойти прерывание. Запрос прерывания по логическому уровню, если он разрешен, будет генерировать запрос прерывания до тех пор, пока на входе будет наблюдаться низкий уровень.

Выполнение работы

Выбор подпрограммы «Таймер»:

1. Нажать кнопку SB2 (Сброс) для сброса предыдущей подпрограммы, после перевести тумблеры SA5 и SA7 в нулевое состояние (рычаг тумблера смотрит влево), а тумблер SA6 в состояние «1» (рычаг тумблера смотрит вправо).

2. Для выполнения подпрограммы необходимо изменить состояние тумблера SA8 (переключить тумблер в другое положение).

Измеряем время запуска программы. После запуска подпрограммы на светодиодном индикаторе HG1 появится таймер. Требуется измерить частоту переключений (по ТУ шаг 1 секунда). Измеренный шаг занести в журнал наблюдений.

Измерить диапазон изменений (по ТУ в пределах [–128; +128]). Занести измерения в журнал наблюдений.

Управление счетом (стоп, вверх, вниз) осуществляется тумблерами SA5 и SA6 в соответствии с таблицей истинности (см. табл. 3.1, подпрограмма № 2).

При направлении счета «ВВЕРХ», счет начинается с 0; дойдя до 128, счет будет производиться от –128 до 0, после чего цикл повторится (рис. 3.3). Измеряем количество интервалов и заносим его в журнал наблюдений.

Таблица 3.2

Журнал наблюдений

Величина	Измерение	ТУ
Частота переключений		
Диапазон изменений		
Количество интервалов		

Лабораторная работа № 4

СЧЕТЧИКИ ИМПУЛЬСОВ НА БАЗЕ МИКРОПРОЦЕССОРА

Цель работы: изучить назначение и принцип действия счетчика импульсов.

Оборудование: стенд НТЦ-02.05.1 «Электроника с МПСО».

Основные сведения

Подсчет числа импульсов является наиболее распространенной операцией в устройствах цифровой обработки информации. Повышенный интерес к таким устройствам объясняется их высокой точностью, возможностью применения регистрирующих приборов с непосредственным цифровым представлением результата, а также возможностью осуществления связи с компьютером. В результате цифровой обработки информации от различных датчиков измеряемый параметр (угол поворота, перемещение, скорость, частота, время, температура и т. д.) преобразуется в импульсы напряжения, число которых в соответствующем масштабе характеризует значение данного параметра. Эти импульсы подсчитываются счетчиками и выражаются в виде цифр. По целевому назначению счетчики бывают *простые* и *реверсивные*. Простые счетчики, в свою очередь, бывают *суммирующие* и *вычитающие*.

Суммирующий счетчик предназначен для выполнения счета в прямом направлении, т. е. для сложения. С приходом счетного импульса на вход счетчика его показание увеличивается на единицу.

Вычитающий счетчик служит для осуществления счета в обратном направлении, т. е. для вычитания. Каждый счетный импульс, поступающий на вход вычитающего счетчика, уменьшает его показание на единицу.

Реверсивные счетчики предназначены для выполнения операции счета как в прямом, так и в обратном направлении, т. е. они могут и суммировать, и вычитать. Основными показателями счетчиков являются модуль счета (коэффициент счета K) и быстродействие. Быстродействие счетчика характеризуется максимальной частотой следования счетных импульсов $f_{сч}$ и связанным с ней временем установки счетчика $t_{уст}$. Счетчики импульсов выполняются на основе триггеров. Счет числа поступающих импульсов производится с использованием двоичной системы счисления.

Двоичные суммирующие счетчики с непосредственной связью

Двоичные счетчики производят счет поступающих импульсов в двоичной системе счисления. Основным узлом двоичного счетчика (служащим также его разрядом) является триггер со счетным запуском, осуществляющий подсчет импульсов по модулю 2. Многоразрядные двоичные суммирующие счетчики с непосредственной связью выполняются путем последовательного соединения счетных триггеров. Счетные импульсы подаются на счетный вход первого триггера. Счетные входы последующих триггеров связаны непосредственно с прямыми выходами предыдущих триггеров: вход второго триггера соединен с выходом первого триггера, вход третьего – с выходом второго и т. д.

Принцип действия двоичного счетчика с непосредственной связью. Рассмотрим на примере 3-разрядного счетчика, показанного на рис. 4.1, *а*. Счетчик выполнен на ТТ-триггерах с внутренней задержкой. Работу схемы иллюстрируют временные диаграммы, приведенные на рис. 4.1, *б*.

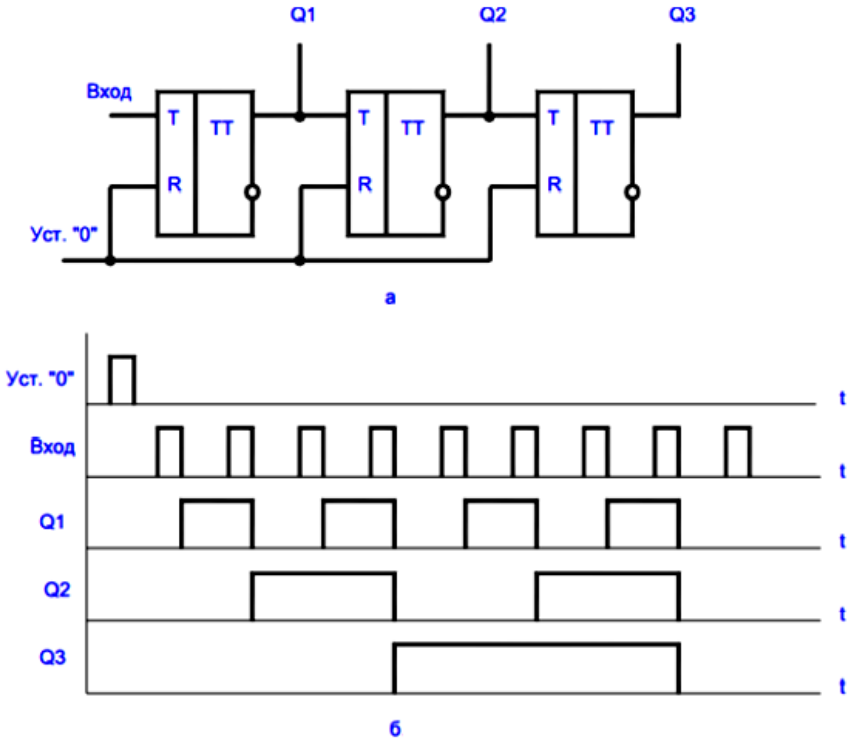

Рис. 4.1. Принцип действия двоичного счетчика
с непосредственной связью

Перед поступлением счетных импульсов все разряды счетчика обнуляются (Q1 = Q2 = Q3 = 0) подачей импульса на вход Уст. «0». При поступлении 1-го счетного импульса 1-й разряд подготавливается к переключению в противоположное состояние и по заднему фронту входного импульса переходит в состояние Q1 = 1. В счетчик записывается число «1». Высокий уровень с выхода Q1 воздействует на счетный вход второго разряда, подготавливая его к переключению. По окончанию 2-го счетного импульса 1-й разряд счетчика переходит в состояние «0», а 2-й разряд – в состояние «1». То есть в счетчик будет записано число «2» (двоичный код «010»). Подобным

образом осуществляется работа схемы с приходом последующих импульсов. Как видно из рис. 4.1, *б*, 1-й разряд счетчика переключается с приходом каждого входного импульса, 2-й разряд – каждого второго, а 3-й разряд срабатывает на каждый 4-й счетный импульс. В процессе работы двоичного счетчика частота следования импульсов на выходе каждого последующего триггера уменьшается вдвое по сравнению с частотой его входных импульсов (рис. 4.1, *б*). Это свойство схемы используют для построения делителей частоты. При этом входной сигнал подают на счетный вход 1-го триггера, а выходной снимают с последнего триггера. В результате выходная и входная частоты связаны соотношением

$$f_{вых} = f_{вх} / К_{сч}.$$

Десятичные счетчики

Счетчики с $К_{сч}$ = 10 называют десятичными, или декадными. Они нашли широкое применение для регистрации числа импульсов с последующим визуальным отображением результата. Для построения счетчика с $К_{сч}$ = 10 необходимо иметь 4-разрядный двоичный счетчик, число состояний которого следует уменьшить с 16 до 10. Счетная последовательность десятичного счетчика может быть представлена в двоично-кодированном десятичном коде (Q4 Q3 Q2 Q1 –8 4 2 1), в котором каждая десятичная цифра кодируется 4-разрядным числом. Счетная последовательность суммирующего десятичного счетчика в этом случае совпадает с двоичной последовательностью от 0000 до 1001, после чего следует 0, и последовательность повторяется. Последовательное соединение двух схем десятичного счета дает пересчет на 100, трех – на 1000 и т. д. Первая декада производит счет единиц входных импульсов от 0 до 9. Десятый импульс устанавливает разряды первой декады в состояние «0», а формируемый на его выходе импульс записывает «1» во вторую декаду, что соответствует числу 10.

Вторая декада считает десятки (от 10 до 90), третья – сотни (от 100 до 900) и т. д.

Вычитающие и реверсивные двоичные счетчики

В вычитающих счетчиках с приходом очередного счетного сигнала предыдущий результат уменьшается на единицу. В вычитающем двоичном n-разрядном счетчике реализуется счетная последовательность, начинающаяся с числа $(2n - 1)$ и заканчивающаяся числом 0. Очередное число в этой последовательности получается вычитанием единицы из предыдущего числа. После получения значения 0 последовательность повторяется. Вычитающий счетчик, в отличие от суммирующего, строится так, что вход каждого последующего триггера соединяется с инверсным выходом предыдущего триггера. Схема вычитающего счетчика с последовательной передачей переносов приведена на рис. 4.2.

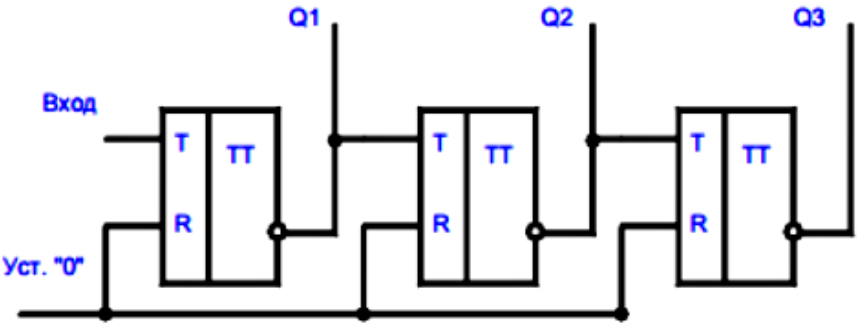

Рис. 4.2. Схема вычитающего счетчика

В *реверсивном* счетчике объединяются схемы суммирующего и вычитающего счетчиков. Кроме того, существует возможность управления направлением счета, для чего предусматриваются дополнительные управляющие входы. Для счетных сигналов в этом счетчике предусмотрены два входа. Если нужен суммирующий счетчик, сигналы счета подают на вход

«+1». Для вычитающего счетчика сигналы счета подают на вход «–1». На выходе счетчика «15>» сигнал появляется при переходе счетчика в состояние с номером 15, в котором все триггеры установлены в состояние 1. На этом выходе формируется сигнал переноса в следующий счетчик. На выходе «<0» сигнал появляется при заполнении счетчика нулями. Это происходит в схеме вычитающего счетчика. Условное обозначение реверсивного счетчика приведено на рис. 4.3.

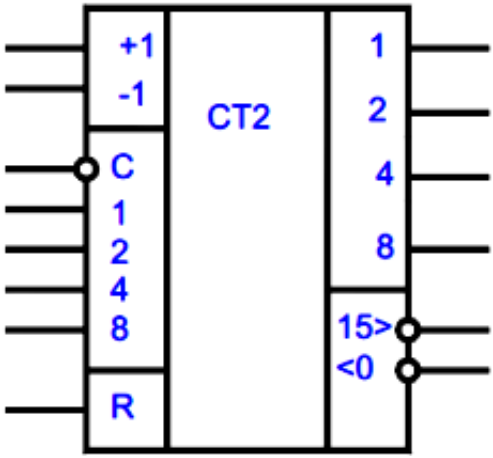

Рис. 4.3. Обозначение реверсивного счетчика

Выполнение работы

Выбор подпрограммы «Счетчики»:

1. Нажать кнопку SB2 (Сброс) для сброса предыдущей подпрограммы.

2. Перевести тумблеры SA6 и SA7 в нулевое состояние (рычаг тумблера смотрит влево), а тумблер SA5 – в состояние «1» (рычаг тумблера смотрит вправо).

3. Для выполнения подпрограммы необходимо изменить состояние тумблера SA8 (переключить тумблер в другое положение).

4. После запуска подпрограммы на светодиодном индикаторе HG1 появится счетчик с шагом 1.

Ход счета представлен на рис. 4.4.

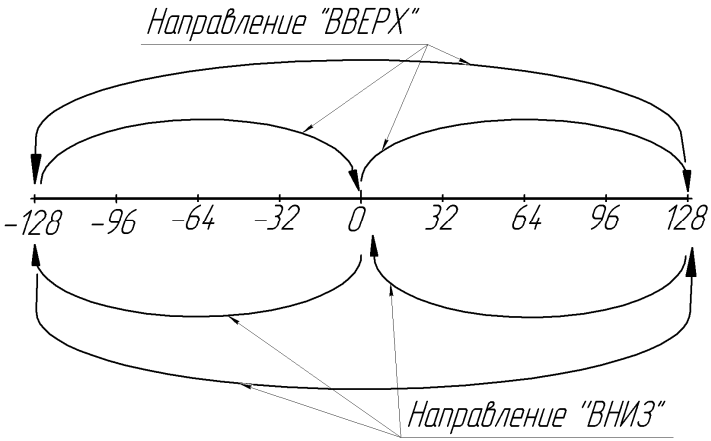

Рис. 4.4. Ход счета при направлениях «ВВЕРХ» и «ВНИЗ»

Управление счетом (вверх, вниз) осуществляется тумблерами SA5, SA6.

Счет «ВВЕРХ» производится по изменению положения тумблера SA5 из положения «0» (рычаг тумблера смотрит влево) в «1» (рычаг тумблера смотрит вправо).

Счет «ВНИЗ» по изменению положения тумблера SA6 из положения «0» в «1».

Провести проверку адекватности счета и занести его в журнал наблюдений (табл. 4.1).

Таблица 4.1

Журнал наблюдений

Измеряемая величина	Направление счета	
	Вверх	Вниз
Крайнее значение		
Время отсчета		

Литература

1. Электронные системы управления автомобилем: лабораторные работы (практикум) для студентов специальностей 1-37 01 06 «Техническая эксплуатация автомобилей» 1-37 01 07 «Автосервис»: в 3 ч. / сост.: Е. Л. Савич, А. С. Гурский. – Минск : БНТУ, 2007. – Ч. 1. – 80 с.

2. Электронные системы управления автомобилем: лабораторные работы (практикум) для студентов специальностей 1-37 01 06 «Техническая эксплуатация автомобилей» 1-37 01 07 «Автосервис»: в 3 ч. / сост.: Е. Л. Савич, А. С. Гурский. – Минск : БНТУ, 2012. – Ч. 3: Диагностирование электронных блоков управления автомобильных систем. – 63 с.

3. Электронные системы управления автомобилем: лабораторные работы (практикум) для студентов специальностей 1-37 01 06 «Техническая эксплуатация автомобилей» 1-37 01 07 «Автосервис»: в 3 ч. / сост.: Е. Л. Савич, А. С. Гурский. – Минск : БНТУ, 2007. – Ч. 2: Устройство и диагностирование датчиков и исполнительных механизмов автомобильных электронных систем управления. – 80 с.

4. Лабораторный практикум по цифровым вычислительным устройствам Л12 на базе стендов НТЦ-02.05.1 «Электроника с МПСО» и персонального компьютера / сост. С. Н. Сазонов. – Ульяновск : УлГТУ, 2017. – 172 с.

5. Инструкция к стенду НТЦ-02.05.1 «Электроника с МПСО».

Содержание

Лабораторная работа № 1
НАЗНАЧЕНИЕ, УСТРОЙСТВО И ПРИНЦИП РАБОТЫ МИКРОПРОЦЕССОРА 3

Лабораторная работа № 2
ВЗАИМОДЕЙСТВИЕ МИКРОПРОЦЕССОРА С ПЕРИФЕРИЙНЫМИ УСТРОЙСТВАМИ НА ПРИМЕРЕ СВЕТОДИОДНОГО ИНДИКАТОРА 23

Лабораторная работа № 3
РАБОТА ТАЙМЕРА МИКРОПРОЦЕССОРА 31

Лабораторная работа № 4
СЧЕТЧИКИ ИМПУЛЬСОВ НА БАЗЕ МИКРОПРОЦЕССОРА 37

Литература 44

Учебное издание

ПРАКТИКУМ ПО МИКРОПРОЦЕССОРНЫМ СИСТЕМАМ УПРАВЛЕНИЯ АВТОМОБИЛЕМ

Практикум
для обучающихся по специальностям 1-37 01 06
«Техническая эксплуатация автомобилей (по направлениям)»,
1-37 01 07 «Автосервис»

ИВАШКО Виктор Сергеевич
ГУРСКИЙ Александр Станиславович
СМОЛЬСКАЯ Валентина Станиславовна и др.

Редактор *Н. А. Костешева*
Компьютерная верстка *Н. А. Школьниковой*

Подписано в печать 02.02.2022. Формат 60×84 $^{1}/_{16}$. Бумага офсетная. Ризография.
Усл. печ. л. 2,67. Уч.-изд. л. 2,09. Тираж 100. Заказ 252.

Издатель и полиграфическое исполнение: Белорусский национальный технический университет.
Свидетельство о государственной регистрации издателя, изготовителя, распространителя
печатных изданий № 1/173 от 12.02.2014. Пр. Независимости, 65. 220013, г. Минск.