

Практическая работа Тема 7.

Особенности работы с аналоговыми интерфейсами

Цель работы: изучение особенностей обработки и преобразования аналоговых сигналов. Изучение аналоговых интерфейсов, представленных в микроконтроллере ATMega328P.

Используемое оборудование:

Плата с МК Atmel AVR ATMega 328P 1 шт.

Программатор USBISP 1 шт.

Потенциометр 1 шт.

Используемое ПО: Интегрированная среда разработки Atmel Studio 7.0.

Программа для загрузки программного кода в микроконтроллер AVRDUDEPROG

Теоретическая часть

Большинство окружающих человека процессов характеризуются непрерывностью своих характеристик во времени. Сигнал, имеющий непрерывную характеристику, т.е. непрерывное множество возможных значений во времени называется аналоговым. Многие датчики предоставляют информацию о параметрах среды именно в аналоговой форме в виде напряжения. Но практически все современные вычислители работают с дискретным цифровым сигналом.

Дискретный сигнал - сигнал, который является прерывистым (в отличие от аналогового), который изменяется во времени и принимает любое значение из списка возможных значений. Сигнал может быть дискретным по уровню (принимать только значения из определенного множества), по времени (изменения могут произойти только в определённые моменты времени), а также и по уровню и во времени. Цифровой сигнал – дискретный сигнал (как во времени, так и по значению), который может принимать только значения «1» и «0». Соответственно, для обработки информации полученной от аналоговых датчиков необходимо преобразовать его в цифровую форму.

Компаратор

Компаратор аналоговых сигналов (от лат. comparare «сравнивать») — сравнивающее устройство, принимающее на свои входы два аналоговых сигнала и выдающее сигнал высокого уровня, если сигнал на неинвертирующем входе («+» на рисунке 7.1) больше, чем на инвертирующем (инверсном) входе («-» на рисунке 7.1), и сигнал низкого уровня, если сигнал на неинвертирующем входе меньше, чем на инверсном входе. Значение выходного сигнала компаратора при равенстве входных напряжений, в общем случае не определено. Сигналу высокого уровня приписывается значение логической 1, а низкому — логического 0.

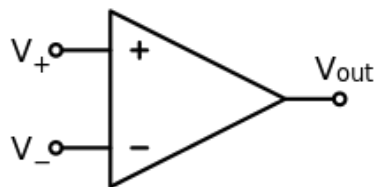


Рисунок 7.1 – Аналоговый компаратор

Аналоговый компаратор сравнивает напряжения на двух выводах (пинах) микроконтроллера. Результатом сравнения будет логическое значение, которое может быть прочитано из программы. Аналоговый компаратор может служить источником прерывания. Пользователь в этом случае может установить срабатывание прерывания по нарастающему или спадающему фронту, или по переключению.

ACSR – Регистр компаратора

7	6	5	4	3	2	1	0
ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0

Бит **ACD** - включение компаратора. 0 - включен, 1 - выключен. По умолчанию 0, (при старте компаратор включен).

Бит **ACBG** — подключение к прямому входу компаратора внутреннего источника опорного напряжения (ИОН). Если 0, то ИОН не подключен.

Бит **ACO** — бит результата. Собственно, это и есть выход компаратора.

Бит **ACI** — флаг прерывания. Устанавливается по событию, сбрасывается после ухода на обработчик либо программно записью в него 1.

Биты **ACIS1:ACIS0** определяет условие возникновения прерывания от компаратора:

Таблица 7.1 – События вызывающие прерывания

ACIS1	ACIS0	Условие прерывания
0	0	Любое изменение на входе
0	1	Резерв
1	0	Переход с 1 на 0
1	1	Переход с 0 на 1

Бит **ACIE** — разрешаем прерывания от компаратора. 1 - разрешаем прерывания от компаратора. По умолчанию 0.

Бит **ACIC** — подключение компаратора к схеме захвата таймера.

Биты **AIN0** (считается положительным – positive) и **AIN1** (считается отрицательным – negative). Если значение сигнала (напряжения) на линии **AIN0** становится больше сигнала (напряжения) на линии **AIN1** – выходной сигнал компаратора **ACO** устанавливается в 1. Выход компаратора может быть подключен ко входу.

Аналогово-цифровой преобразователь. Модуль АЦП

АЦП микроконтроллеров измеряет напряжение на своём входе. Чтобы произвести измерение других физических величин, их нужно вначале преобразовать в напряжение. Сигнал измеряется относительно величины, которую называют опорным напряжением (V_{ref} – reference voltage). Эта величина является максимумом, который можно измерить. Напряжение внутреннего источника у разных контроллеров может быть разным, но обычно лежит в диапазоне 2,3-5В. Однако его использовать не рекомендуется, по причине низкой стабильности. Источники внешнего опорного напряжения обычно обозначаются как **AVCC** или **Aref**. Максимальное значение, которое может измерить микроконтроллер даже при использовании внешнего источника, редко может быть более 5В. Чтобы расширить диапазон значений используются делители напряжения.

Одна из важнейших характеристик АЦП - **разрешающая способность**. Она оказывает прямое влияние на точность измерения. Весь диапазон

измерения разбивается на части. Минимум - ноль, максимум - опорное напряжение.

Для 10-битного АЦП это $2^{10}=1024$, для 12-битного это $2^{12}=4096$. Следовательно, чем выше разрядность, тем точнее можно измерять сигнал. Например, при оцифровке сигнала, изменяющегося в диапазоне от 0 до 5В, при опорном напряжении 5В, а также 12-разрядном АЦП фиксируемое изменение аналогового сигнала равно $5В / 2^{12} = 5/4096 = 0,00122В$. Однако при программировании АЦП следует учитывать, что младшие 1-2 разряда могут быть не очень устойчивыми, и поэтому по возможности их необходимо отбрасывать. Поэтому реальная практическая точность 12-разрядного АЦП $5В / 2^{10} = 5/1024 = 0,00488В$.

Ещё один параметр АЦП - **частота дискретизации**. Она определяет, как часто выполняется преобразование аналогового сигнала. Является характеристикой быстродействия АЦП и измеряется в герцах или количестве выборок в секунду (SPS – samples per second).

В схеме АЦП используется делитель частоты, который позволяет управлять частотой дискретизации. Сам АЦП обычно может иметь прямой доступ к памяти для того, чтобы не оказывать влияние на АЛУ и процессорное ядро МК в целом.

Принцип преобразования сигнала с помощью АЦП показан на Рисунке 7.2.

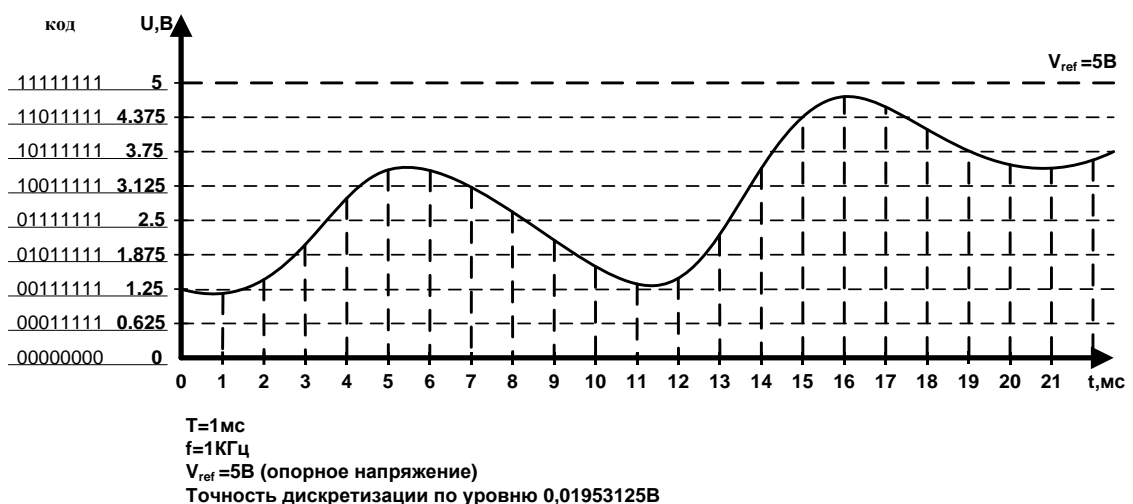


Рисунок 7.2 - Аналогово-цифровое преобразование

Предполагается, что используется восьмиразрядный АЦП с текущей частотой дискретизации сигнала 1КГц и опорным напряжением 5В. Зная разрядность АЦП и опорное напряжение, можно рассчитать точность преобразования $5 / 2^8 = 0.01953125\text{В}$. Рассмотрим несколько значений, полученных с помощью преобразователя в разные моменты времени:

- при $t=1$, $U \approx 1,171875$, код на выходе преобразователя 00111011;
- при $t=7$, $U \approx 3,125$, код на выходе преобразователя 10011111;
- при $t=19$, $U \approx 3,75$, код на выходе преобразователя 10111011.

В микроконтроллере ATmega328P имеется 10 разрядный АЦП, который находится в PORTC. Выходное напряжение конвертируется в 10-ти разрядное двоичное значение. Минимальное значение соответствует 0, а максимальное - опорному напряжению. В качестве источника опорного напряжения можно использовать внутренний или внешний источники. Как правило выбирается наиболее стабильный источник - это нужно для того, чтобы общий диапазон значений не менялся вместе со скачками опорного напряжения.

В микроконтроллере ATmega328p в корпусе TQFP-32 есть возможность выбрать один из 7 входов для преобразования, а также ИОН или датчик температуры микроконтроллера (только для ATmega328P). В качестве опорного напряжения также можно выбирать внутренний ИОН и внешний ИОН.

Таблица 7.2 – Характеристики модуля АЦП

Разрешение преобразования	10 бит
Абсолютная погрешность	2
Интегральная нелинейность	0.5
Скорость преобразования	13 - 260 микросекунд
Входное напряжение	от 0 до V питания
6 мультиплексированный каналов преобразования	
Встроенный датчик температуры	
Возможность подключить внутренний ИОН 1.1 В	
Независимость от ЦП	
Генерация прерывания по завершению преобразования	
Режим подавления шумов	

Регистры АЦП

ADMUX

ADMUX управляет выбором опорного напряжения и мультиплексором.

Регистр ADMUX:

7	6	5	4	3	2	1	0
REFS1	REFS0	ADLAR		MUX3	MUX2	MUX1	MUX0

Биты REFS1 (7) и REFS0 (6) устанавливают какой источник опорного напряжения будет выбран:

REFS1	REFS0	Тип источника напряжения
0	0	Подключен внешний опорный ион, внутренний ион выключен. Опорное напряжение на входе AREF
0	1	Впитания (вход AREF должен быть отключен или к нему можно подключить фильтрующий конденсатор)
1	0	Резерв
1	1	Внутренний ИОН 1.1 В (к входу AREF можно подключить фильтрующий конденсатор)

Бит ADLAR (5) позволяет выравнивать результат преобразования по левому краю при записи в него 1.

Биты MUX3 - MUX0 (3 - 0) - управляют мультиплексором:

MUX3	MUX2	MUX1	MUX0	Активный вход
0	0	0	0	Вход ADC0 (23 ножка)
0	0	0	1	Вход ADC1 (24 ножка)
0	0	1	0	Вход ADC2 (25 ножка)
0	0	1	1	Вход ADC3 (26 ножка)
0	1	0	0	Вход ADC4 (27 ножка)
0	1	0	1	Вход ADC5 (28 ножка)
0	1	1	1	Резерв
1	0	0	0	Датчик температуры
1	0	0	1	Резерв
...				
1	1	0	1	
1	1	1	0	1.1 В
1	1	1	1	0 В (земля)

ADCSRA

Регистр ADCSRA - регистр управления АЦП А:

7	6	5	4	3	2	1	0
ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0

Бит **ADEN** (7) регистра ADCSRA включает или выключает АЦП. 1 – включен. 0 – выключен.

Бит **ADSC** (6) регистра ADCSRA запускает преобразование если в него записать 1 (для многократного режима запуск первого преобразования).

Бит **ADATE** (5) регистра ADCSRA позволяет запускать преобразование по прерыванию от периферийных устройств микроконтроллера если установить в 1.

Бит **ADIF** (4) регистра ADCSRA - флаг прерывания от АЦП.

Бит **ADIE** (3) регистра ADCSRA - разрешает прерывания от АЦП если установлен в 1.

Биты **ADPS2 - ADPS0** (2 - 0) регистра ADCSRA выбирают режим работы определителя тактовой частоты:

ADPS2	ADPS1	ADPS0	Тактовая частота
0	0	0	CLK/2 – без деления должно быть
0	0	1	CLK/2
0	1	0	CLK/4
0	1	1	CLK/8
1	0	0	CLK/16
1	0	1	CLK/32
1	1	0	CLK/64
1	1	1	CLK/128

ADCSRБ

Регистр ADCSRB - регистр управления АЦП В:

7	6	5	4	3	2	1	0
	ACME				ADTS2	ADTS1	ADTS0

Бит **ACME** (6) регистра ADCSRB позволяет использовать мультиплексор АЦП в качестве входов аналогового компаратора при установке 1 (при этом АЦП должен быть выключен).

Биты **ADTS2 - ADTS0** (2 - 0) регистра ADCSRB выбирают источник сигнала, по которому будет начинаться преобразование АЦП:

ADTS2	ADTS1	ADTS0	Источник сигнала
0	0	0	Непрерывное преобразование
0	0	1	Прерывание от аналогового компаратора
0	1	0	Внешнее прерывание INT0
0	1	1	Прерывание по совпадению таймера/счетчика T0 с A
1	0	0	Прерывание по переполнению таймера/счетчика T0
1	0	1	Прерывание по совпадению таймера/счетчика T1 с B
1	1	0	Прерывание по переполнению таймера/счетчика T1
1	1	1	Прерывание по захвату таймера/счетчика T1

ADCL и ADCH

ADCL и ADCH - регистры с результатом преобразования АЦП. Результат преобразования может быть выровнен относительно этих двух регистров двумя способами. Первый способ (бит ADLAR регистра ADMUX равен 0): результат преобразования выравнивается по младшему регистру: 8 младших бит результата преобразования записываются в регистр ADCL, 2 старших бита результата в младшие биты ADCH.

Второй способ (бит ADLAR регистра ADMUX равен 1): результат преобразования выравнивается по младшему регистру: 2 младших бита результата записываются в старшие биты регистра ADCL, старшие биты результата записываются в регистр ADCH.

Второй способ удобен когда младшие биты не имеют значения, например сильно шумят или не участвуют в передаче.

DIDR0

7	6	5	4	3	2	1	0
ADC7D	ADC6D	ADC5D	ADC4D	ADC3D	ADC2D	ADC1D	ADC0D

Регистр DIDR0 запрещает использование входов. Биты ADC7D - ADC0D (7 - 0) регистра DIDR0 запрещают использование входов ADC7 - ADC0 если в них установить 1.

Сводная таблица описанных регистров представлена ниже.

Адрес в ОЗУ	Адрес в I/O Reg	Имя регистра	Назначение битов регистра [7:0]							
			7	6	5	4	3	2	1	0
0x50	0x30	ACSR	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0
0x78		ADCL	ADC7(1)	ADC6(0)	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0
0x79		ADCH	ADC(9)	ADC(8)	ADC(7)	ADC(6)	ADC(5)	ADC(4)	ADC9(3)	ADC8(2)
0x7A		ADCSRA	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0
0x7B		ADCSRB		ACME				ADTS2	ADTS1	ADTS0
0x7C		ADMUX	REFS1	REFS0	ADLAR		MUX3	MUX2	MUX1	MUX0
0x7E		DIDR0	ADC7D	ADC6D	ADC5D	ADC4D	ADC3D	ADC2D	ADC1D	ADC0D
0x7F		DIDR1							AIN1D	AIN0D

Для работы с регистром ACSR используются команды IN/OUT, он расположен в адресном пространстве регистров ввода/вывода.

Для работы с регистрами ADCL, ADCH, ADCSRA, ADCSRB, ADMUX, DIDR0, DIDR1 используются команды `sts`, `lds`, они расположены в адресном пространстве дополнительных регистров ввода/вывода.

Потенциометр

Потенциометр — регулируемый вручную переменный резистор с 3 выводами. Два его вывода подключены к обоим концам резистивного элемента, а третий вывод (движок) подключен к скользящему контакту, который перемещается по резистивному элементу. Выходное напряжение потенциометра зависит от текущей позиции скользящего контакта и может находиться в диапазоне от нуля до наибольшей величины, равной входному напряжению на потенциометр.

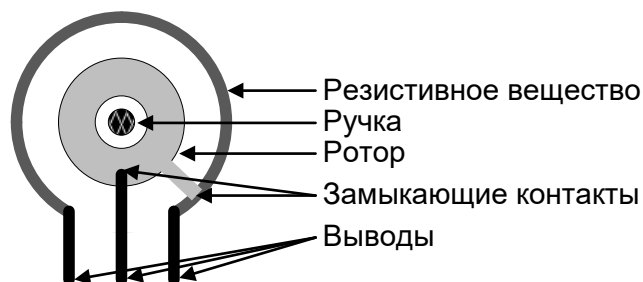


Рисунок 7.3 – Устройство потенциометра

Практическая часть

Задание 1. Исследование работы с аналоговым компаратором

Подготовьте проект для контроллера на языке assembler. Наберите программу представленную ниже. Выполните сборку программы.

```

1  .include "m328Pdef.inc"
2      ldi r16,0x40 ;загрузка константы для работы компаратора в r16
3      out ACSR,r16 ;загрузка константы в регистр управления компаратора
4      ldi r16,0x20 ;формирование константы для настройки PORTB5 на вывод
5      out DDRB,r16 ;PORTB5 – режим «вывод» (светодиод L)
6  main:                                ;главная подпрограмма
7      in r16, ACSR ;загрузка значения ACSR в r16
8      bst r16, 5   ;сохранение значения 5 бита из r16 (ACO)
9      BRTC ledon   ;если t-clear, то значение выше порога, переход к ledon
10     cbi PORTB,5; ;сброс бита 5 регистра PORTB – выключение светодиода
11     jmp main     ;переход к метке main
12 ledon:          ;метка – включение диода
13     sbi PORTB,5; ;установка бита 5 регистра PORTB – включение светодиода
14     jmp main     ;переход к метке main

```

Данная программа обеспечивает инициализацию работы аналогового компаратора. В данной программе к входу AIN1 подключен источник опорного напряжения (ИОН) 1.1В. AIN0

Подключите устройства согласно схеме рисунок 7.4. Загрузите программу в контроллер. Проанализируйте работу системы при изменении положения потенциометра. Зафиксируйте полученные результаты в отчёте.

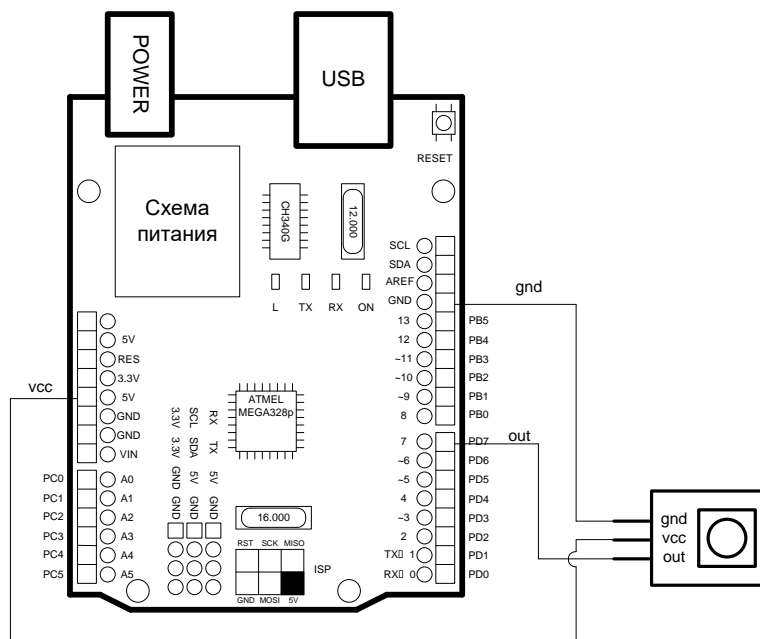


Рисунок 7.4 - Схема подключения потенциометра

Задание 2. Обработка прерываний компаратора

Работу с компаратором, как и с любым периферийным устройством, рационально осуществлять по прерыванию. Благодаря использованию данного механизма, можно разгрузить процессор микроконтроллера от постоянного анализа состояния компаратора. В таблице 7.1. представлены признаки, которые могут вызвать прерывание. Все настройки осуществляются через регистр ACSR. Подготовьте проект для контроллера на языке assembler. Наберите программу, представленную ниже, заменив знаки вопроса на необходимые константы. Программа выполняет те же действия, что и в задании 1.

```
1  .include "m328Pdef.inc"
2  .org 0x0000      ;адрес вектора reset
3  jmp reset        ;переход к метке reset
4  .org ?????       ;адрес вектора аналогового компаратора
5  jmp acomp        ;переход к метке acomp
6  acomp:           ;обработчик прерывания компаратора
7      in r16, ACSR ;чтение регистра ACSR
8      bst r16, 5   ;записать 5ый бит регистра R16 в SREG - бит T
9      BRTC ledon   ;t-clear
10     cbi PORTB,5  ;выключить светодиод PORTB5
11     jmp intexit  ;переход к выходу из обработчика
12  ledon:          ;метка "включение светодиода"
13     sbi PORTB,5  ;включить светодиод PORTB5
14  intexit:        ;метка выхода из обработчика
15     reti         ;возврат из процедуры обработки прерывания
16  reset:          ;метка обработчика reset (точка входа)
17     ldi r16, ???? ;константа для настройки ACSR
18     out ACSR,r16  ;Загрузка константы из R16 в ACSR
19     ldi r16,0x20  ;Загрузка константы 0x20 в R16
20     out DDRB,r16  ;Загрузка константы 0x20 в DDRB, PORTB5-на вывод
21     sei          ;глобальное разрешение прерываний
22  main:           ;метка основной подпрограммы
23     jmp main      ;переход к метке основной подпрограммы
```

Задание 3. Перепишите программы из заданий 1 и 2 на языке C. Выполните загрузку кода в контроллер. Проанализируйте работу устройства. Выводы зафиксируйте в отчёте.

Задание 4. Передача состояния компаратора по UART

Подготовьте проект для контроллера на языке assembler. Наберите программу, представленную ниже. Выполните сборку проекта. Загрузите программу в контроллер. Подключите отладочную плату к компьютеру через USB кабель. Откройте программу Terminal. Выберите соответствующий плате COM-порт, скорость передачи 9600бит/с, нажмите кнопку «Connect». Микроконтроллер должен постоянно передавать слово «Normal».

```

1  .include "m328Pdef.inc"
2  .equ CLK=16000000
3  .equ BAUD=9600
4  .equ UBRR0_value = (CLK/(BAUD*16)) - 1
5  .org 0
6      jmp Reset
7  Reset:
8      ;настройка UART
9      ;установка частоты приёма/передачи
10     ldi r16, high(UBRR0_value)
11     sts UBRR0H, r16
12     ldi r16, low(UBRR0_value)
13     sts UBRR0L, r16
14     ;разрешение передачи
15     ldi r16, (1<<TXEN0)
16     sts UCSR0B, r16
17     ;длина слова 8 бит
18     ldi r16, (1<< UCSZ00)|(1<< UCSZ01)
19     sts UCSR0C, r16
20     sei ;разрешение всех прерываний
21 main:
22     ldi z1, low(NorV<<1); метка указывает на слово (16 бит) в памяти, для правильного
23     ;чтения необходим сдвиг на 1 бит влево, младший бит указывает на часть слова
24     ldi zh, high(NorV<<1); 0 - младший байт, 1 - старший
25     ldi r18, 0x07 ;количество байт в сообщении
26 send:
27     lpm r16, Z+ ;чтение символа слова из таблицы (ПЗУ)
28     sts UDR0, r16 ;отправка символа по uart
29     subi r18, 0x01 ;вычитание 1 из счетчика букв
30 repeat:
31     lds r17, UCSR0A; чтение содержания регистра
32     bst r17, 5 ;запись 5 бита (udre) регистра (ucsr0a) в бит t регистра sreg
33     brtc repeat ;если t=0
34     cpi r18, 0x00 ;r18=0?
35     breq main ;если да, то повторяем сообщение
36     jmp send ;если нет, то следующий символ
37 .cseg ;работа с ПЗУ
38 LowV: .db "Low Voltage", '\n' ;размещение сообщения в памяти ПЗУ
39 NorV: .db "Normal", '\n' ; размещение сообщения в памяти ПЗУ

```

Используя программы, полученные в заданиях 1 и 2, измените представленный пример так, чтобы при уровне сигнала, превышающем порог, микроконтроллер передавал значение «Normal», при низком «Low Voltage». Выполните сборку проекта. Отключите питание от платы. Подключите потенциометр согласно рисунку 7.4. Восстановите питание. Загрузите программу в контроллер. Проанализируйте работу системы с помощью программы Terminal. Проанализируйте работу системы при изменении положения потенциометра. Зафиксируйте полученные результаты в отчёте. Перепишите программу на языке C. Проанализируйте работу устройства. Выводы зафиксируйте в отчёте.

Задание 5. Аналогово-цифровой преобразователь

Подготовьте проект для контроллера на языке assembler. Наберите программу, представленную ниже. Выполните сборку проекта. Подключите потенциометр к отладочной плате, согласно рисунку 7.5. Подключите отладочную плату к компьютеру через USB кабель. Загрузите программу в контроллер. Откройте программу Terminal. Выберите соответствующий плате COM-порт, скорость передачи 9600бит/с, нажмите кнопку «Connect». Микроконтроллер должен постоянно передавать слово «Normal».

```
1  .include "m328pdef.inc"
2  .def temp=r16 ;объявление логического имени temp для регистра r16
3  .def adcvL=r17 ;объявление логического имени adcvL для регистра r17
4  .def adcvH=r18 ;объявление логического имени adcvH для регистра r18
5  .equ CLK=16000000 ;тактовая частота генератора
6  .equ BAUD=9600 ;скорость передачи данных UART
7  .equ UBRR0_value = (CLK/(BAUD*16)) - 1 ;подготовка константы для UBRR
8  .org 0 ;вектор прерывания reset
9      jmp Reset ;переход к метке reset
10 .org 0x002A ; вектор прерывания АЦП
11      jmp ADC_conv ;переход к метке ADC_conv
12 Reset: ;точка входа в программу
13      ldi temp, (1<<REFS0)|(1<<ADLAR) ;Вопорное = Vпитания, выравнивание результата
14 ;преобразования по «левому краю» старшие 8бит будут записаны ADCH, канал ADC0
15      sts ADMUX,temp ;запись значения в ADMUX
16      ldi temp,(1<<ADEN)|(1<<ADSC)|(1<<ADIE)|(1<<ADPS2)|(1<<ADPS1)|(1<<ADPS0) ;
17 ;включение АЦП, циклическое преобразование, разрешение прерывания, частота - CLK/128
18      sts ADCSRA,temp ; запись значения в ADCSRA
19      ldi temp, high(UBRR0_value) ;старшая часть константы для UBRR в temp
20      sts UBRR0H, temp ;загрузка значения в UBRR0H
21      ldi temp, low(UBRR0_value) ; младшая часть константы для UBRR в temp
22      sts UBRR0L, temp ;загрузка значения в UBRR0L
23      ldi temp, (1<<TXEN0) ;разрешение передачи UART
24      sts UCSR0B,temp ;запись значения в UCSR0B
25      ldi temp,(1<< UCSZ00)|(1<< UCSZ01) ;размер пакета данных
26      sts UCSR0C,temp ; запись значения в UCSR0C
27      sei ;глобальное разрешение прерываний
28 main: jmp main
29 ;---обработчик прерывания АЦП -----
30 ADC_conv:
31      lds adcvH, ADCH ;чтение значения регистра ADCH в регистр r18
32 repeat: ;метка repeat обеспечивает цикл ожидания готовности UART к передаче
33      lds r17, UCSR0A ;чтение содержания регистра
34      bst r17, 5 ;запись 5 бита (UDRE) регистра UCSR0A в бит t регистра SREG
35      brtc repeat ;если t=0 - возврат к метке repeat
36      sts UDR0, adcvH ;передача старших 8бит АЦП по UART
37      reti ;возврат из прерывания
38 ;---обработчик прерывания АЦП -----
```

Проанализируйте работу системы при изменении положения потенциометра. Зафиксируйте полученные результаты в отчёте.

Перепишите программу на языке C. Проанализируйте работу устройства. Выводы зафиксируйте в отчёте.

Температурный датчик будет возвращать 10 битное значение [0:1023], выбранное опорное напряжение 1.1В позволяет сопоставлять код, полученный от АЦП, с ИОН практически один к одному. Например, полученному коду 900 будет соответствовать напряжение примерно 900мВ. В техническом описании представлена формула, по которой напряжение датчика можно сопоставить с температурой по формуле $t=V/10$, следовательно, напряжению 900мВ соответствует температура 90 градусов Цельсия. Однако, необходимо учитывать, что по UART в данной работе передаётся только часть слова – старшие 8 бит, т.е. код, принятый компьютером, будет в 4 раза меньше реального значения, полученного АЦП.

Используя данную информацию, проанализируйте работу системы. Результаты зафиксируйте в отчете.

Контрольные вопросы

1. Чем аналоговый сигнал отличается от дискретного?
2. Что такое АЦП?
3. Назовите основные характеристики АЦП?
4. Что такое аналоговый компаратор?
5. Для чего необходимо выравнивание регистров ADC?
6. Перечислите особенности аналогово-цифрового преобразователя.
7. Для чего используется источник опорного напряжения?
8. Сигналы с каких выводов могут сравниваться с помощью компаратора?
9. Назовите характеристики АЦП микроконтроллера Atmega328p.
10. Какие настройки АЦП необходимо выполнить для запуска циклического преобразования?