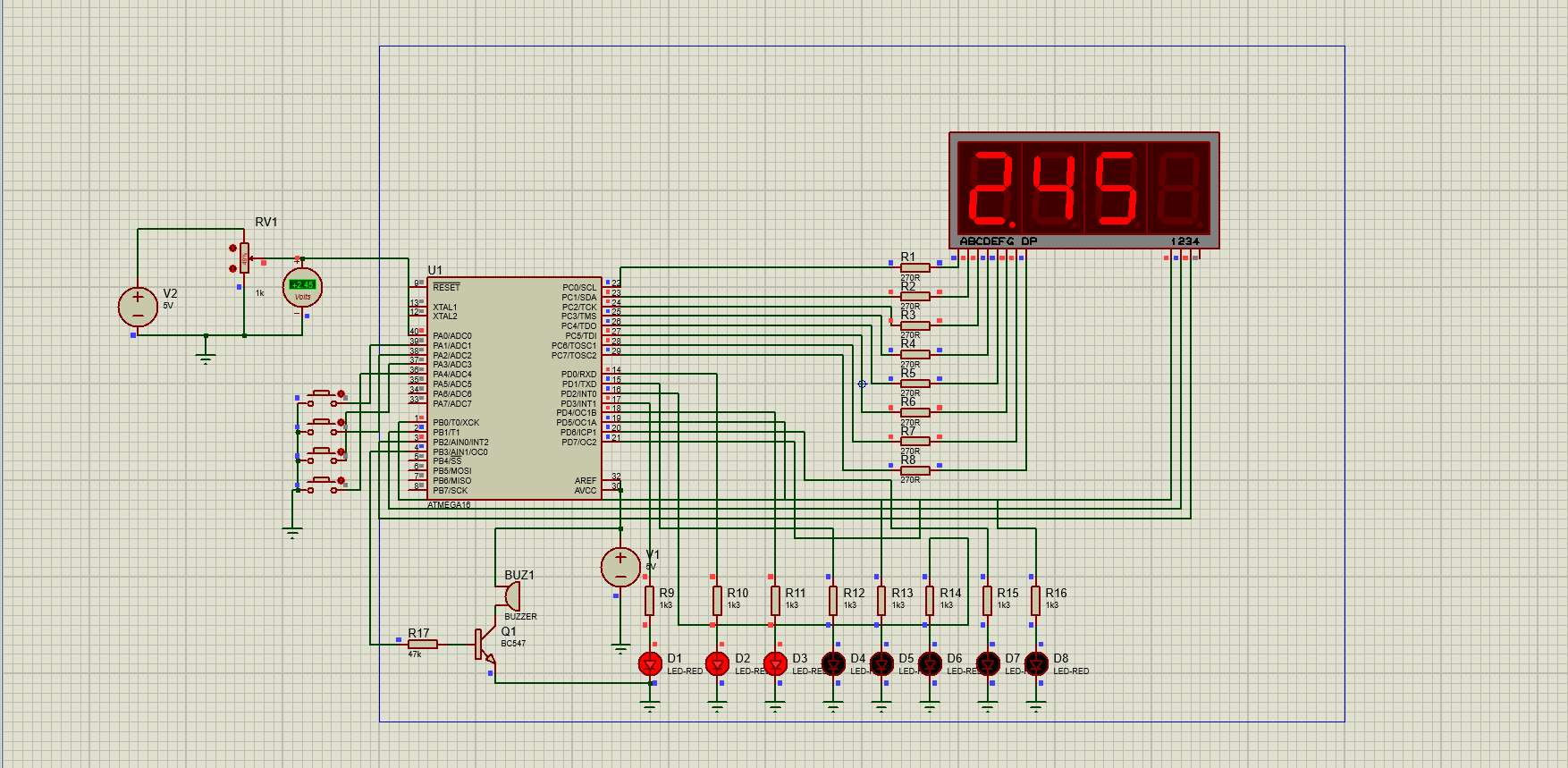
Задание 1.

С помощью АЦП измерить напряжение на переменном резисторе, вывести на семисегментный индикатор, продублировать изменение напряжения на светодиодах.



|  |
| --- |
| /\*  \* task\_1.c  \*  \*/  #define *F\_CPU* 8000000UL  #include <avr/io.h>  #include <util/delay.h>  void numberIndicator(unsigned int ); // Конфигурация пинов для каждой цифры  void deshifrator(unsigned int); // Дешифратор  unsigned int adc(); // Получение бита АЦП  void LedIndicator(unsigned int); // Индикация напряжения на светодиодах  int main(void)  {    // Подключение индикатора  DDRC = 0b11111111;  DDRB = 0b00000111;  // Подключение переменника  DDRA &= ~(1<<0);  // Подключение светодиодов  DDRD = 0b11111111;    ADCSRA |= (1<<ADEN); // Подключение АЦП  ADCSRA |= (1<<ADATE); // Включаем непрерывное выполнение измерения  ADCSRA |= (1<<ADPS2)|(1<<ADPS1);// Установка частоты дискретизации 125кГц  ADCSRA &= ~(1<<ADPS0);  //ИОН - 5В  ADMUX &= ~(1<<REFS1);  ADMUX |= (1<<REFS0);  ADMUX &= ~(1<<ADLAR);//Выбор порядка заполнения регистра ADC  ADMUX &= ~((1<<MUX4)|(1<<MUX3)|(1<<MUX2)|(1<<MUX1)|(1<<MUX0));//Вывод ADC0    while (1)  {  LedIndicator(adc());  deshifrator(adc()\*0.489); // 5\*100/1023 =0.489  }  }  void numberIndicator(unsigned int n)  {  switch(n)  {  case 0:  PORTC = 0b00111111;  break;  case 1:  PORTC = 0b00000110;  break;  case 2:  PORTC = 0b01011011;  break;  case 3:  PORTC = 0b01001111;  break;  case 4:  PORTC = 0b01100110;  break;  case 5:  PORTC = 0b01101101;  break;  case 6:  PORTC = 0b01111101;  break;  case 7:  PORTC = 0b00000111;  break;  case 8:  PORTC = 0b01111111;  break;  case 9:  PORTC = 0b01101111;  break;  }    }  void deshifrator(unsigned int n)  {  int a,b,c;  a =(n)/100;  b = ((n) % 100)/10;  c = (n) % 10;    PORTB = 0b00000110;  numberIndicator(a);  *\_delay\_ms*(1);  PORTB = 0b00000101;  numberIndicator(b);  *\_delay\_ms*(1);  PORTB = 0b00000011;  numberIndicator(c);  *\_delay\_ms*(1);    //Подключение точки  PORTB = 0b00000110;  PORTC = 0b10000000;  *\_delay\_ms*(1);  }  unsigned int adc()  {    ADCSRA |= (1 << ADSC); // Начинаем преобразование  while ((ADCSRA&(1 << ADIF))== 0); // Ждем флага окончания преобразования  return (unsigned int)ADC;  }  void LedIndicator(unsigned int n)  {    //Значения, с которыми происходит сравнение, получены из ADC = (Uвх/Uион)\*1024 (см. datasheet)    if(n>=128)  PORTD |= (1<<3);  else  PORTD &= ~(1<<3);  if(n>=256)  PORTD |= (1<<0);  else  PORTD &= ~(1<<0);  if (n>=384)  PORTD |= (1<<4);  else  PORTD &= ~(1<<4);  if(n>=512)  PORTD |= (1<<1);  else  PORTD &= ~(1<<1);  if(n>=640)  PORTD |= (1<<5);  else  PORTD &= ~(1<<5);  if(n>=768)  PORTD |= (1<<2);  else  PORTD &= ~(1<<2);  if(n>=896)  PORTD |= (1<<6);  else  PORTD &= ~(1<<6);  if(n == 1023)  PORTD |= (1<<7);  else  PORTD &= ~(1<<7);      } |