#include <avr/io.h>  
#define F\_CPU 16000000UL//  
#include <util/delay.h>  
#include <stdlib.h>  
#define BAUDRATE 19200  
#define BAUD\_PRESCALLER (((F\_CPU / (BAUDRATE \* 16UL))) - 1)  
  
uint16\_t adc\_value;            //Variable used to store the value read from the ADC  
void adc\_init(void);            //Function to initialize/configure the ADC  
uint16\_t read\_adc(uint8\_t channel);    //Function to read an arbitrary analogic channel/pin  
void USART\_init(void);            //Function to initialize and configure the USART/serial  
void USART\_send( unsigned char data);    //Function that sends a char over the serial port  
void USART\_putstring(char\* StringPtr);    //Function that sends a string over the serial port  
  
int main(void){  
    adc\_init();        //Setup the ADC  
    USART\_init();        //Setup the USART  
    DDRB=0xff;  
    while(1){  
        uint16\_t x= read\_adc(0);  
          
        char buffer[4];  
        utoa(x,buffer,10);  
        USART\_putstring("value is :");  
        USART\_putstring(buffer);  
        if (x >= 512){  
            PORTB|=(1<<PB0);  
        }  
            else{  
                PORTB &=~ (1<<PB0);  
            }  
        USART\_send(13);  
        \_delay\_ms(100);  
          
    }  
      
    return 0;  
}  
  
void adc\_init(void){  
    ADCSRA |= ((1<<ADPS2)|(1<<ADPS1)|(1<<ADPS0));    //16Mhz/128 = 125Khz the ADC reference clock  
    ADMUX |= (1<<REFS0);                //Voltage reference from Avcc (5v)  
    ADCSRA |= (1<<ADEN);                //Turn on ADC  
    //ADMUX = (1<<ADLAR);  
    ADCSRA |= (1<<ADSC);                //Do an initial conversion because this one is the slowest and to ensure that everything is up and running  
}  
  
uint16\_t read\_adc(uint8\_t channel){  
    ADMUX &= 0xF0;                    //Clear the older channel that was read  
    ADMUX |= channel;                //Defines the new ADC channel to be read  
    ADCSRA |= (1<<ADSC);                //Starts a new conversion  
    while(ADCSRA & (1<<ADSC));            //Wait until the conversion is done  
    return ADC;                    //Returns the ADC value of the chosen channel  
}  
  
void USART\_init(void){  
      
    UBRR0H = (uint8\_t)(BAUD\_PRESCALLER>>8);  
    UBRR0L = (uint8\_t)(BAUD\_PRESCALLER);  
    UCSR0B = (1<<RXEN0)|(1<<TXEN0);  
    UCSR0C = (3<<UCSZ00);  
}  
  
void USART\_send( unsigned char data){      
    while(!(UCSR0A & (1<<UDRE0)));  
    //printf("value is %u ",data);  
    //UDR0 = '1';  
    UDR0 = data;      
}  
  
void USART\_putstring(char\* StringPtr){      
    while(\*StringPtr != 0x00){          
        USART\_send(\*StringPtr);  
        StringPtr++;  
    }  
      
}