#define F\_CPU 1000000UL

#include <avr/io.h>

#include <util/delay.h>

void InitPWM()

{

/\*

TCCR0 - Timer Counter Control Register (TIMER0)

-----------------------------------------------

BITS DESCRIPTION

NO: NAME DESCRIPTION

--------------------------

BIT 7 : FOC0 Force Output Compare [Not used in this example]

BIT 6 : WGM00 Wave form generartion mode [SET to 1]

BIT 5 : COM01 Compare Output Mode [SET to 1]

BIT 4 : COM00 Compare Output Mode [SET to 0]

BIT 3 : WGM01 Wave form generation mode [SET to 1]

BIT 2 : CS02 Clock Select [SET to 0]

BIT 1 : CS01 Clock Select [SET to 0]

BIT 0 : CS00 Clock Select [SET to 1]

The above settings are for

--------------------------

Timer Clock = CPU Clock (No Prescalling)

Mode = Fast PWM

PWM Output = Non Inverted

\*/

TCCR0|=(1<<WGM00)|(1<<WGM01)|(1<<COM01)|(1<<CS00);

//Set OC0 PIN as output. It is PB3 on ATmega16 ATmega32

DDRB|=(1<<PB3);

}

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

Sets the duty cycle of output.

Arguments

---------

duty: Between 0 - 255

0= 0%

255= 100%

The Function sets the duty cycle of pwm output generated on OC0 PIN

The average voltage on this output pin will be

duty

Vout= ------ x 5v

255

This can be used to control the brightness of LED or Speed of Motor.

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

void SetPWMOutput(uint8\_t duty)

{

OCR0=duty;

}

void initADC()

{

ADCSRA |= (1 << ADPS2) | (1 << ADPS1) | (1 << ADPS0); // Set ADC prescalar to 128 - 125KHz sample rate @ 16MHz

ADMUX |= (1 << REFS0); // Set ADC reference to AVCC

ADMUX |= (1 << ADLAR); // Left adjust ADC result to allow easy 8 bit reading

// No MUX values needed to be changed to use ADC0

// ADCSRA |= (1 << ADATE); // Set ADC auto trigger enable

ADCSRA |= (1 << ADEN); // Enable ADC

/\* The ADSC bit is 1 when the conversion is going on and

cleared by harware when done \*/

ADCSRA |= (1 << ADSC); // Start A2D Conversions

}

int main (void)

{

DDRD=0xFF; //used as output pins

uint8\_t brightness=0;

initADC();

InitPWM();

for(;;) // Loop Forever

{

//start conversion...

ADCSRA |= (1 << ADSC); // Start A2D Conversions

while((ADCSRA&(1<<ADSC))==1);

if(ADCH < 86)

{

PORTD = 0b00000001; // Turn on LED1

}

else if(ADCH < 172)

{

PORTD = 0b00000010; // Turn on LED1

}

else

{

PORTD = 0b00001000; // Turn on LED1

}

brightness = ADCH;

SetPWMOutput(brightness);

}

}

/\*

if(ADCH <32)

{

PORTD = 0b00000001; // Turn on LED1

}

else if(ADCH <64)

{

PORTD = 0b00000010; // Turn on LED1

}

else if(ADCH <96)

{

PORTD = 0b00000100; // Turn on LED1

}

else if(ADCH <128)

{

PORTD = 0b00001000; // Turn on LED1

}

else if(ADCH <160)

{

PORTD = 0b00010000; // Turn on LED1

}

else if(ADCH <192)

{

PORTD = 0b00100000; // Turn on LED1

}

else if(ADCH <224)

{

PORTD = 0b01000000; // Turn on LED1

}

else

{

PORTD = 0b10000000; // Turn on LED1

} \*/