#include <avr/io.h>

#include <util/delay.h>

#include "lcd.h"

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

26

27 Configuration Area.

28 UltraSonic (US) sensor connection.

29

30 in this example it is connected to PORTA bit 0

31

32 Adjust the following to connect is to different i/o

33

34 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define US\_PORT PORTA

#define US\_PIN PINA

#define US\_DDR DDRA

#define US\_POS PA0 //PORTA0

#define trig PA1

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

44

45 This function measusers the width of high pulse in micro second.

46

47 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define US\_ERROR 0xffff

#define US\_NO\_OBSTACLE 0xfffe

uint16\_t getPulseWidth()

{

uint32\_t i,result;

//Wait for the rising edge

for(i=0;i<600000;i++)

{

if(!(US\_PIN & (1<<US\_POS))) continue; else break;

}

if(i==600000)

return 0xffff; //Indicates time out

//High Edge Found

//Setup Timer1

TCCR1A=0X00;

TCCR1B=(1<<CS11); //Prescaler = Fcpu/8

TCNT1=0x00; //Init counter

//Now wait for the falling edge

for(i=0;i<600000;i++)

{

if(US\_PIN & (1<<US\_POS))

{

if(TCNT1 > 60000) break; else continue;

}

else

break;

}

if(i==600000)

return 0xffff; //Indicates time out

//Falling edge found

result=TCNT1;

//Stop Timer

TCCR1B=0x00;

if(result > 60000)

return 0xfffe; //No obstacle

else

return (result>>1);

}

void Wait()

{

uint8\_t i;

for(i=0;i<10;i++)

*\_delay\_loop\_2*(0);

}

void main()

{

uint16\_t r;

LCD\_SetUp(PB\_0,PB\_1,PB\_2,P\_NC,P\_NC,P\_NC,P\_NC,PC\_4,PC\_5,PC\_6,PC\_7);

LCD\_Init(2,16);

LCD\_DisplayString("Explore Embedded");

LCD\_DisplayString("Lcd 4-bit Mode");

Wait();

while(1)

{

//Set Ultra Sonic Port as out

US\_DDR|=(1<<trig);

*\_delay\_us*(10);

//Give the US pin a 15us High Pulse

US\_PORT|=(1<<trig); //High

*\_delay\_us*(15);

US\_PORT&=(~(1<<trig));//Low

*\_delay\_us*(20);

//Now make the pin input

US\_DDR&=(~(1<<US\_POS));

//Measure the width of pulse

r=getPulseWidth();

//Handle Errors

if(r==US\_ERROR)

{

LCD\_DisplayString("Error !");

}

else if(r==US\_NO\_OBSTACLE)

{

LCD\_DisplayString("Clear !");

}

else

{

int d;

d = (r/58.0); //Convert to cm

LCD\_Printf("%d",d);

// LCDWriteIntXY(0,0,d,4);

// LCDWriteString(" cm");

Wait();

}

}

}