

File: adc2.c, Date: 4/23/2016, Time: 8:13:25 AM

This program was produced by the

CodeWizardAVR V2.05.3 Standard

Automatic Program Generator

© Copyright 1998-2011 Pavel Haiduc, HP InfoTech s.r.l.

<http://www.hpinfotech.com>

Project :

Version :

Date : 4/16/2016

Author : Reza

Company :

Comments:

Chip type : ATmega64

Program type : Application

AVR Core Clock frequency: 8.000000 MHz

Memory model : Small

External RAM size : 0

Data Stack size : 1024

*****/

#include <mega64.h>

#include <delay.h>

// Alphanumeric LCD functions

#include <alcd.h>

#include <string.h>

#define ADC_VREF_TYPE 0x00

// Read the AD conversion result

unsigned int read_adc(**unsigned char** adc_input)

{

ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);

// Delay needed for the stabilization of the ADC input voltage

delay_us(10);

// Start the AD conversion

ADCSRA|=0x40;

// Wait for the AD conversion to complete

while ((ADCSRA & 0x10)==0);

ADCSRA|=0x10;

return ADCW;

}

// Declare your global variables here

void main(**void**)

{

// Declare your local variables here

int buf;

char charBuff[5];

// Input/Output Ports initialization

// Port A initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

```

File: adc2.c, Date: 4/23/2016, Time: 8:13:25 AM
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTA=0x00;
DDRA=0x00;

// Port B initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTB=0x00;
DDRB=0x00;

// Port C initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTC=0x00;
DDRC=0x00;

// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTD=0x00;
DDRD=0x00;

// Port E initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTE=0x00;
DDRE=0x00;

// Port F initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTF=0x00;
DDRF=0x00;

// Port G initialization
// Func4=In Func3=In Func2=In Func1=In Func0=In
// State4=T State3=T State2=T State1=T State0=T
PORTG=0x00;
DDRG=0x00;

// ADC initialization
// ADC Clock frequency: 1000.000 kHz
// ADC Voltage Reference: AREF pin
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0x83;

// Alphanumeric LCD initialization
// Connections are specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
// RS - PORTC Bit 0
// RD - PORTC Bit 1
// EN - PORTC Bit 2
// D4 - PORTC Bit 4
// D5 - PORTC Bit 5
// D6 - PORTC Bit 6
// D7 - PORTC Bit 7
// Characters/line: 20
lcd_init(20);
//char* adcString = "%000";

```

```
while (1)
{
    //      // Place your code here
    //      x=read_adc(0)      ;
    //      x =( x / 1023) * 100;
    //      //sprintf(adcString,"%d",x);
    //      lcd_gotoxy(0,0);
    //      lcd_puts((string) x);
    //

    buf = read_adc(0);
    if(buf >= 1000) buf = 1000;
    buf/=10;
    charBuff[0]=buf/100;
    buf%=100;
    charBuff[1]=buf/10;
    charBuff[2]=buf%10;

    //      buf = (buf/1023)*100;
    //
    //      charBuff[2]=buf % 10;
    //      charBuff[0]=buf/10 % 10;
    //      charBuff[1]=buf/100;

    lcd_gotoxy(0,0);
    lcd_putchar('%');
    lcd_gotoxy(1,0);
    lcd_putchar(charBuff[0]+'0');
    lcd_gotoxy(2,0);
    lcd_putchar(charBuff[1]+'0');
    lcd_gotoxy(3,0);
    lcd_putchar(charBuff[2]+'0');
}
}
```