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

This program was created by the CodeWizardAVR V3.49a

Automatic Program Generator

© Copyright 1998-2022 Pavel Haiduc, HP InfoTech S.R.L.

http://www.hpinfotech.ro

Project :

Version :

Date : 18/11/2022

Author :

Company :

Comments:

Chip type : ATmega32

Program type : Application

AVR Core Clock frequency: 12,000000 MHz

Memory model : Small

External RAM size : 0

Data Stack size : 512

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

#ifndef F\_CPU

#define F\_CPU 12000000UL

#endif

// I/O Registers definitions

#include <mega32.h>

#include <delay.h>

#include <stdbool.h>

// Alphanumeric LCD functions

#include <alcd.h>

// Declare your global variables here

// Variabel melacak timer

unsigned int ms = 0, s = 0, m = 0;

// Variabel melacak status timer

// On true jika timer sedang berjalan

// Running true jika timer masih memiliki waktu

// Prev melacak status timer sebelumnya

bool on = false, prev = true, prev2 = true, running = false;

// Alert digunakan untuk LED kedap-kedip

bool alert = 0;

// Array berisi BCD yang dipakai untuk menampilkan angka pada LED 7SEG

unsigned char angka [100]={

0x00, 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x07, 0x08, 0x09,

0x10, 0x11, 0x12, 0x13, 0x14, 0x15, 0x16, 0x17, 0x18, 0x19,

0x20, 0x21, 0x22, 0x23, 0x24, 0x25, 0x26, 0x27, 0x28, 0x29,

0x30, 0x31, 0x32, 0x33, 0x34, 0x35, 0x36, 0x37, 0x38, 0x39,

0x40, 0x41, 0x42, 0x43, 0x44, 0x45, 0x46, 0x47, 0x48, 0x49,

0x50, 0x51, 0x52, 0x53, 0x54, 0x55, 0x56, 0x57, 0x58, 0x59,

0x60, 0x61, 0x62, 0x63, 0x64, 0x65, 0x66, 0x67, 0x68, 0x69,

0x70, 0x71, 0x72, 0x73, 0x74, 0x75, 0x76, 0x77, 0x78, 0x79,

0x80, 0x81, 0x82, 0x83, 0x84, 0x85, 0x86, 0x87, 0x88, 0x89,

0x90, 0x91, 0x92, 0x93, 0x94, 0x95, 0x96, 0x97, 0x98, 0x99,

};

// Fungsi dipanggil ketika waktu habis dan timer masih berjalan

void toggleLED(){

running = false;

while (PINA.0 == 0){

alert = !alert;

PINA.5 = alert;

delay\_ms(10);

}

}

// Fungsi untuk mengupdate angka pada LED 7 Segment

void update(){

PORTB = angka[m];

PORTD = angka[s];

//lcd

}

// Fungsi utama

void main(void)

{

// Port A input dan output

PORTA = 0xFF;

DDRA = 0xFF;

// Port B,C dan D hanya berupa output

PORTB = 0x00;

DDRB = 0xFF;

PORTC = 0x00;

DDRC = 0xFF;

PORTD = 0x00;

DDRD = 0xFF;

// TCCR1B pada CS10, CS11 dan CS12 diset menjadi 101 sehingga prescalar bernilai 1024

TCCR1B = (1<<CS10) | (1<<CS12);

// Jumlah iterasi pada counter sebelum direset

OCR1A = 7;

// Nilai counter awal

TCNT1 = 0;

// 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: 16

lcd\_init(16);

while (1)

{

//Jika saklar tersambung maka on = true

if (PINA.0 == 0){

on = true;

}

else {

on = false;

}

//Jika waktu habis maka running = false

if (m <= 0 && s <= 0){

running = false;

}

else {

running = true;

}

// Jika status timer berubah (prev1 & on dan prev2 & running) maka update status di LCD

if (prev != on || prev2 != running){

lcd\_clear();

lcd\_gotoxy(0,0);

if (on && running){

lcd\_puts("TIMER BERJALAN..");

}

else if (on && !running){

lcd\_puts("WAKTU HABIS!!!");

}

else if (!on && running){

lcd\_puts("TIMER BERHENTI..");

}

else if(!on && !running){

lcd\_puts("TAMBAH WAKTU!!");

}

prev = on;

prev2 = running;

}

// Jika timer sedang berhenti maka waktu dapat diubah

if (!on){

//manipulate seconds

if ((PINA.1 == 0) & (s < 60)){//increment second, bind to under 60

s++;

}

else if ((PINA.2 == 0) & (s > 0)){//decrement seconds

s--;

}

//manipulate minutes

if ((PINA.3 == 0) & (m < 60)){

m++;

}

else if ((PINA.4 == 0) & (m > 0)){

m--;

}

//Delay antar input

delay\_ms(15);

}

// Selagi timer berjalan

else{

// Menunggu hingga overflow flag dalam timer set

while((TIFR && (1<<OCF1A)) == 0);

// Jika timer habis maka LED berkedap kedip

if (m == 0 & s == 0) {

toggleLED();

goto skip;

}

// Jika second habis maka menit berkurang dan second direset

if (s == 0) {

s = 59;

m--;

}

// Jika ms habis maka second berkurang dan ms direset

if (ms == 0){

ms = 100;

s--;

}

// Dekremen ms

else {

ms--;

}

// Setelah variabel pelacak timer diupdate, reset counter dalam timer

TCNT1 = 0;

// Clear timer overflow flag

TIFR |= (1<<OCF1A);

}

skip:

// Update angka dalam LED

update();

}

}