Carlo Provenzani
CPE325
Kumar Sharshembiev
due: 04/27/2018

Final Project: MSP430 LCD based Alarm Clock

This project is a digital alarm clock based on the MSP430 microcontroller, the program uses the LCD screen to display the current time and the alarm time set by the user. Synchronous serial communication was applied so the user will use the hyperterminal to enter the current time and the time alarm to go off. The buzzer will play a tune at the time at the alarm time. When the alarm goes off the user can either press the SW2 on the MSP430 for the snooze button this will add 10 minutes to the alarm time so the buzzer will replay the tune 10 minutes later, or the user can press the SW1 to stop the alarm. The LCD will show the current time if SW1 and SW2 are not pressed and the alarm time while SW2 pressed.

SPI_Setup
- sync, mode, 3-pin SPI, Master mode, 8-bit data
- SMCLK and software reset
- Data rate = SMCLK/2 ~=500kHz
- Initialize USCI state machine
- Back to main

playSong
- while SW1 or SW2 is not pressed
- change frequencies to play song
- if SW2 is pressed add 10 min to alarm time
- change frequencies to play song
- if SW1 is pressed cancel alarm

main
- Stop WDT
- enable global interrupts
- set P2.1 and P2.2 to output
- Set P5.2/3/4 = LCD COM lines
- turn LED 1and LED2 off
- Initialize timer A in up mode and use ACLK
- set TACCR0 to 32768 (1 sec)
- set up P3.5 for buzzer
- Initialize timer B in up mode and use SMCLK
- TB0 output is in toggle mode
- Serial_Initialize()
- SPI_Setup()
- Print start message
- cleat LCD
- while(1)
- if char '0' to '9' is pressed put it int an array and print it to Hyperterminal
- if ENTER is pressed check if time entered is correct
- put time entered into integers hr, min, sec for time, and Ahr,Amin, and Asec for alarm time
- if ESC is pressed re-enter alarm

TA ISR
- SetTime()
- if itme and alarm time have been entered send time and changeLCD

SetTime()
- increment seconds
- if sec = 60 set sec =0 min++
- if min = 60 set min =0 hr++
- if hr ==24 hr=0
- if min = alarm minutes
- if hr = alarm hr
- playSong()

SPI_State
- verifies busy flag
- write new state
- check USCI_B0 TX buffer ready?
- Back to main

UART_putchar
- wait for other character to transmit
- Print character
- Back to main

Serial_Initialize
- Set UC0TXD and UCRXD to transmit and receive data
- Software reset
- USCI_A0 control register
- Clock source SMCLK 1048576Hz/19200
- baund rate - 104576 Hz / 19200
- Modulation
- Enable USCI_A0 RX interrupt
- Back to main

SendTime()
- change hr, min, and sec to ASCII characters
- and send using UART0_putchar() function

ChangeLCD()
- change lcd screen according to function marameters

USCIBORX_ISR
- character received is moved to a variable
- signal main function receiving a char
- back to main