CPE 301 Microprocessor System Design Lab

LAB #09   Fall 2013   **(TWO WEEK LAB)**

Objective:

Develop an Arduino C language Command Processor (aka Monitor Program, aka mini operating system) for the Arduino Mega SBC.

Procedure:

Write a command processor program which will take over control of the Arduino Mega SBC, with the following properties. The main menu of the command processor should be written in loop() and appear as below and contain the following options:

Welcome to Sarah's Arduino Mega Command Processor… (Message with your name).
Please enter the desired function:

        1. Keyboard echo
        2. ASCII character echo
        3. Change the rotating cursor speed
        4. Blink LED
        5. Sound Tone

Select Command>

All five menu selections should be implemented in functions and they should print a sign-on message explaining what the function does, requesting the user to press ESC to exit back to the menu, and then request input data 1) press key, 2) press key, 3) input cursor speed change + (faster) or – (slower), 4) input blink time change + (faster) or – (slower), 5) input tone frequency change + (higher) or – (lower). For functions 3 through 5 you will need to decide on a proper (and noticeable) interval of change for each +/- input. Also, you may need to fix minimum and maximum values for each function.

1.   Write a menu loop in the Arduino loop() function which will: a) write out the main menu and start the rotating cursor by calling crson() then continually loop and: b) call kbhit() to see if an input key is ready, c) call a function called syswrk() each time after it calls kbhit() and there is no input, and d) when kbhit() returns TRUE indicating that there is a keyboard input call cmdproc() which will read the key and act on its value to accomplish one of the menu items.

2.   The function syswrk() will maintain the rotating cursor by calling crsrot() (until there is input), and the speed of the cursor will be determined by a software delay loop. For example, define a static variable counter and each time syswrk() is called decrement the counter. When the counter reaches zero go rotate the cursor once. Displaying the following sequence, will give the appearance that the cursor is rotating:" |", BS, "/ ", BS, "–", BS, "\", BS (where BS =ASCII backspace).

3.   The function cmdproc() will turn off the rotating cursor by calling crsoff(), then read in a key from the serial port and examine its value. If the value is less than 30H or greater than 35H it will simply report an illegal input and then return to the beginning of menu loop. Otherwise, a subroutine will be called which will accomplish the specified menu function. After the subroutine has ended the cmdproc() function should return back to the start of the menu loop and print out the main menu again.

4.   The rest of the program will consist entirely of functions which will either be support functions such as "print string" etc. or will be functions which implement specific menu functions and then return to cmdproc().

5.   One set of functions should control the rotating cursor. When the Main Menu is written the prompt should be "Select Command>". When crson() is called it should start the rotating cursor by sending BS and "|" to the terminal and set a cursor Flag to NOT zero. When crsrot() is called it should send BS followed by the "next" element in the rotating cursor. When crsoff() is called it should send a BS followed by ">" and set a cursor Flag to zero. Thus, when the menu is in the default position the prompt always looks like Select Command> and when it is in wait mode the prompt continuously changes: Select Command| Select Command/ Select Command- Select Command\ and the four symbols will look as if they are a single symbol which is rotating when the speed is right.

Notes:

1. You should have already written similar programs for most of the menu options. You will need to modify that code so that it is modular and fits into subroutines. **Be sure to use Global and Static variable where appropriate.**

2. For each option, hitting ESC should exit the routine and take you back to the main menu.

3. **ALL programs written for this course need to be documented in a complete manner which will make sense to you if you need to go back and review the program a year or two from now. You need to include your NAME and a revision number on ALL programs as well.**