Lab 3 requirements and specs

Project II Requrements

All of the ones from Project 1, plus the following additions/modifications:

* Utilize a hardware-based time reference
* System will support dynamic task creation and deletion
* System will support a user input device
* System will support data logging capabilities
* System will support remote communication capability
* Overall system performance will be improved
* Overall system safety will be improved

Specifications:

The specifications for the system are the same except for the following as noted:

* Individual task runtime will be less than 15ms or show a 25% decrease (performance increase)
* Tasks will perform their own initialization routines when called by the scheduler (safety improvement)
* Measurement variables will be stored in circular buffers, details of which will be handled by a CircularBuffer type (safety improvement)
* Utilize a keypad for user input
  + The keypad must have be capable of performing measurement selection, scrolling though menu options, selecting an option, and acknowledging a system alarm
* Task Control Blocks will have forward and backward pointers to allow references to the next task
* Data memory will be expanded to allow storage of 8 numerical values per vital measurement
* Data memory will be expanded to include new global data from the keypad
* Remote communication will be over a RS-232 compliant protocol
* Depressing the alarm silence button will create an interrupt that will silence the audible alarm (our addition)
* Changes to Schedule:
  + System will maintain a list of activated and deactivated tasks. The list must be updatable during runtime based on the system state
  + The hardware timer will provide a system interrupt every 250ms or equal to the minor cycle, whichever is shorter
  + At runtime, upon a timer interrupt event, all tasks will be added or removed according to the task activation list. The tasks will then be run
  + Task Control Blocks will have forward and backward pointers to allow references to the next task
  + The scheduler cannot block for five seconds
* Changes to the MeasureTask:
  + Once a complete set of measurements has been taken, the compute task is added to the task queue
  + Pointers to the variables used in the measure task will be relocated to accommodate the new data architecture
  + The pulse measurement will monitor and count the frequency of a pulse rate event interrupt
    - A new value will be stored to memory if the present reading is greater than ±15% of the previous measurement
    - The measurement limits will correspond to 200bpm and 10bpm, determined empirically.
* Changes to ComputeTask:
  + All measurements will be recomputed
  + After computing the corrected values for all measurements, the ComputeTask will remove itself from the task queue
* Changes to DisplayTask:
  + Display will now support multiple display options
    - Menu mode will allow selection of each of the individual measurements. Upon selection of a measurement, the current value of the measurement will be displayed onscreen
    - Annunciation mode will display the current status of each measurement as in project 1, and provide the same functionality as the display in project 1.
* Changes to Warn/AlarmTask:
  + The warnings will be activated and indicated as before in project 1
  + The alarm state is changed to activate only when the systolic pressure is 20% above the normal range.
  + The alarm will sound in 1 second tones (1 second on, 1 second off)
  + When an alarm or warning state occurs, the serial communication task will be added to the task queue
  + The deactivation period of the alarm sound is defined as 5 measurement periods
* New task: Serial Communication:
  + The task is enabled by the warn/alarm task
  + When run, the task will open an RS-232 connection at XXXX baud, XXinsertdetails hereXXX
  + The present corrected measurement will be displayed on the terminal in the same fashion as the display task annunciation mode
  + After sending data to the terminal, the serial communication task will remove itself from the task queue
* New task: KeypadTask
  + The keypad task will scan the keypad and decode any keypresses
  + The task will have support the following user inputs:
    - Mode selection between 2 modes (1 button)
    - Menu selection between 3 options (1 button)
    - Alarm acknowledgement (1 button)
    - Up and down scroll functionality (1-2 buttons)
  + A new set of global variable will be created to store the state of the keypad and key presses
* New Task: Initialize (Startup):
  + This task is to run once each time the system is started
  + It will not be part of the task queue
  + It will perform any necessary system initialization, configure and begin the system timebase.
  + After completing these tasks, the task will exit and normal operation will commence.