Objective:

Understand the basic handling of a real-time operating system.

Preparation:

2.1. Prepare the code of the tasks below.

2.2. Set up a graphic representation of tasks, semaphores, variables.

Task:

3.1. Complete the previous program as follows:

3.2. The first task shall also do the following:
- It also creates and starts tasks four and five.
- It also creates additional semaphores (if needed).

3.3. The fourth task shall do the following:
- It reads the keyboard approx. each 100 ms.
- If a new key has been pressed, it is written into a global variable (Key).

3.4. The third task shall additionally do the following:
- It checks if there is any new key pressed – global variable (Key).
- It changes the way the global variables (ADIn) are shown onto the display as follows:
  After key “2” has been pressed, the value shall be shown as Volt.
  After key “3” has been pressed, the value shall be shown in hex representation.
  After key “1” has been pressed, the value shall be shown in decimal representation again.
  If any other key has been pressed an error message shall be output.

3.5. The fifth task shall do the following:
- It reads the system time approx. each 3 seconds.
- It writes the system time in a readable format onto the display.
- Remark: since both printf() and writing onto the display are not reentrant
  task three and five have to synchronize themselves.

Report:

Not requested.

Notes:

5.1. Use the hardware access functions supplied (See HwFunc.pdf).

5.2. You have to include the following header files:
- #include <stdlib.h>
  #include <stdio.h>
  #include <intLib.h>
  #include <taskLib.h>
  #include <sysLib.h> (already known)
- #include <time.h> (new)