Birzeit University - Faculty of Engineering and Technology
Electrical & Computer Engineering Department - ENCS4330
Real-Time Applications & Embedded Systems - 2nd semester - Final exam - 2020/21

Real-Time Applications And Embedded Systems

Instructor: Dr. Hanna Bullata

Student Name/ID: ____

Question 1 (30 points)

Please create a folder called Q_1 and put all files of question 1 under that folder.

We would like to build a multi-processing application that simulates a catching game that kids often play at home. We'll assume we have 4 brothers/sisters that will participate in that game. The game is described as follows:

1. All kids are assigned as players. They are usually located on rectangle edges.
2. The ball is with one of the players and the game initially consists of throwing the ball from one player to a neighbor player in a clockwise mode.
3. While trying to throw the ball to a neighbor player, the thrower player might accidentally drop the ball. Assume there is 10% chance for a thrower player to drop the ball.
4. Similarly, while trying to catch the ball, the catcher player might accidentally miss the ball. Assume there is 40% chance for a catcher player to miss the ball.
5. Once the ball is dropped or missed by a player, the game resumes starting from the next neighbor player.
5. The above-described game continues until any of the players has dropped or missed the ball for 5 times.

What you should do

- Write the code for the above system.
- Compile and test your program.
- Check that your program is bug-free. Use the `gdb` debugger in case you are having problems during writing the code (and most probably you will :-). In such a case, compile your code using the `-g` option of the `gcc`.

Question 2 (70 points)

Please create a folder called Q_2 and put all files of question 2 under that folder.

We would like to build a 16F877A-based controller for a vending machine that sells 10 products. The controller is composed of the following components:

- A 3 × 4 keypad containing the numbers 0 - 9 in addition to `Cancel` and `OK` buttons.
- A 16 × 2 LCD screen to issue informative or warning messages.
- A buzzer that generates a 15KHZ sound to issue warning sounds.
- 6 analog temperature sensors distributed inside the vending machine to check on the temperature inside the vending machine.
- A device to insert coins. The allowed coins are:
  - 1-cent Palestinian coins
  - 1-dime Palestinian coins (value = 10 cents)
  - 1-pound Palestinian coins (value = 10 dimes or 100 cents)
  - 5-pound Palestinian coins

The following functions are *provided* so you can use them without writing the code:

- `get_coin()`: A non-blocking function that returns a positive number when a coin is inserted. Returns -1 (255) if the inserted coin is not recognized. Otherwise, it returns 1 for every inserted cent, 10 for every inserted dime, 100 for every inserted pound and 500 for every inserted 5-pound coins.
- `get_keypad()`: A non-blocking function that returns the keypad key pressed by the user.
- `display_screen(char line, char position, char character)`: Prints a `character` on the LCD screen on the specified `line` and `position`.
- `get_change_5_pound()`: Returns the number of 5-pound coins that should be returned to the user.
- `get_change_1_pound()`: Returns the number of 1-pound coins that should be returned to the user.
- `get_change_dime()`: Returns the number of 1-dime coins that should be returned to the user.
- `get_change_cent()`: Returns the number of 1-cent coins that should be returned to the user.
- `get_shake()`: A non-blocking function that returns 1 if the vending machine is being vandalized, 0 otherwise. If it returns 1, servicing the user is denied.

Assume also the following:

- The vending machine knows the number of remaining items of each product and will issue a warning message on the LCD screen if a user requests an item that is out of stock (message = `Out of Stock`). The buzzer is set ON and the warning message is displayed for 5 seconds.
- The vending machine maintains the products at a temperature lower than 10° and issues a warning on the LCD screen if the average temperature is higher than 10° (message = `Error temp`). The buzzer is set ON and the warning message is displayed for 5 seconds.
- The vending machine keeps track of how many 1-cent coins, 1-dime coins, 1-pound coins and 5-pound coins it has in order to return the change to the user after delivering the item. If it doesn't have enough change, the controller issues a warning message on the LCD screen and will not service the user (message = `Error Change`). The buzzer is set ON and the warning message is displayed for 5 seconds.

To do

A. Draw a schematic on paper for the above vending machine controller. Connect the rows and columns of the keypad to `Port D`. Connect the LCD pins to `Port B` (use the 4-bit mode). Connect the remaining pins as you see fit. Take a clear picture of your design using a suitable camera.

B. Write the PIC assembly code for a PICMicro 16F877A that implements the behavior described above using the pin selection done in **A**. Put comments in the code as necessary.