Final Project ECE36200

Smart Plantation

*By: Zhen Hong Tan (tanzh@iu.edu)*

**Objective:**

Write an assembly program to simulate a smart water irrigation system for a local plantation company. Its functionalities should include: (1) the ability to allow authorized users to manually control the temperature, (2) photosynthesis lights, (3) sprinkler activation, and (4) displaying the time and date for the wall clock. Safety is very important in the plantation site, so a fire alarm must be built in that can disable the power and securely lock the area. The overall objective is to create a program that replicates the interface control center in a user-friendly and glitch-free way.

* Sprinkler. The stepper motor is meant to replicate the sprinkler. When the switch is turned on, water will be pumped to water the plants.
  + Uses: Switches, stepper motor, speaker, LCD, LEDs
  + If switch 7 [8~1] is flipped from high to low, the sprinkler is activated.
  + The speaker plays a song (or tone) to alert the people that water is sprinkling.
  + The LCD should display the message “Sprinkling in Progress” throughout the whole process.
  + LED 7 [8~1] should light up.
  + The sprinkler (stepper motor) should ***gradually*** speed up from a stationary position to a constant speed in the clockwise direction. The stepper motor should constantly spin for approximately 15 seconds.

\* Think: you can simulate acceleration by further reducing the time between each step. You will need an algorithm that does this.

* + After 15 seconds, the speed of the sprinkler (stepper motor) should ***gradually*** decrease from a constant speed back to a stationary position.
  + The system should return to its normal state: the speaker should stop sounding, LED 7 will turn off and the LCD should be displaying the home screen menu.
  + When switch 7 is turned from high to low again, it should repeat the whole process.
* Agriculture industrial fans. The DC motor is meant to replicate the agriculture industrial fans used to improve the air quality inside the warehouse. The DC motor should spin at a controllable duty cycle.
  + Uses: DC motor, potentiometer
  + The value read from the potentiometer should create duty cycles proportional to the potentiometer value. Do this by scaling between values [0-T] (where T will be the period you decide to use). Refer to past labs on scaling and duty cycles.
  + The DC motor should not spin when the pot value is 0, and it should gradually speed up when the user turns it clockwise.
* Digital Clock and Settings Menu. The wall clock in the warehouse shows the date and time. The keypad is the primary way that the user provides input to change the date and time. In the system settings menu, using the keypad, the user can change the current date and time as well as the password.
  + Used: Keypad, LCD
  + When the LCD is on the home screen, if the F key on the keypad is pressed, the user should be prompted on the LCD to enter their Admin password. Upon the correct password, the LCD should go to the system settings menu.
  + Keys A and B should be used to navigate next/back through the options, and Key F should be used to select. You have the freedom to decide how to implement “scrolling/selecting” of the sub-menus as long as it makes sense.
  + The user must be able to access a sub-menu to edit the date, time, and password.
    - For the date, create multiple sub-menus to edit the year, month, and day.
    - For the time, create a sub-menu to edit the hours. (*Editing the minutes and seconds can be an extra feature*)
    - For passwords, a 2-digit password is sufficient.
  + Appropriate input validation is MANDATORY for each step. In the main menu, only keys A, B, and F should affect the system. You should not be able to set the month as 13, the day as 45, or the year as 2AB3, etc.
  + When the control center is turned on for the first time, it should already have preloaded time and date values. The preloaded date and time should be 11/25/2022 at 12.00 a.m. You are free to set any initial password that you like.
* Reset. The push button is meant to reset the system. The user will have to enter the pin before doing so.
  + Used: LCD, Keypad, PB
  + Push button push should prompt user pin entry on LCD. An incorrect password should return the system to the digital wall clock.
  + Once the correct pin is inputted, the date and time should be reset to 11/25/2022 at 12 am.
  + The password should also be reset to the initial password that you preloaded.
* Home screen. should display date and time.
  + Uses: LCD (Keypad by association)
  + Display should *automatically* update as time passes.
  + You MUST have a real-time clock that counts from the seconds up.
  + HINT: It is highly recommended you use a 24-hour hh:mm:ss mm/dd/yy format. You ***may*** assume all months have 30 days. *(Though further accurate input validation would be a good bonus feature)*
* Fire Alarm. The IRQ is meant to simulate a fire alarm for the building. When the IRQ is activated, the system should shut down completely and ***remain*** disabled until the correct admin password is inputted.
  + Uses: IRQ, LEDs, Speaker, LCD, Keypad, DC motor
  + IRQ press puts the system in shutoff mode
  + Speaker should start playing an alarm song that is different from the one during the sprinkling process.
  + The LEDs should blink on and off
  + The fan should be forced to stop, light switches and pushbuttons should not work in this state.
  + The LCD should prompt for pin entry. Incorrect pins will keep the system in emergency shutdown mode.
  + When correct, the system should return to its normal state: The alarm should stop sounding, the HVAC fan should resume normal operation controlled by the potentiometer, and rows of lights should return to how they were before.
* Photosynthesis lights. The switches are meant to replicate the on/off switch for the photosynthesis lights. Switches 1, 2, and 3 [8~1] should turn row 2, row 3, and row 4 lights (LEDs 2-4) [8~1] on or off, respectively.
* EXTRA FEATURES
  + You are required to create an additional feature(s) as part of the project. Part of the grade will be based on extras that you will create on your own. *You are encouraged to be creative*–the more complex, the more points are earned.
  + One feature is acceptable if it is advanced enough. (A bit simpler than one of the main bullet points above; involving multiple steps/peripherals, etc. About the same difficulty as the door access is good.)
  + Otherwise, multiple smaller things would suffice. Many things on par with this are listed in italics throughout the instructions.
  + Refer to some suggested ideas from the “Note” Section.

**Universal Requirements**

1. The user must be able to turn on/off any row of lights at any time not in an emergency (i.e. even when in a menu or animation state). Any changes to the switches must be indicated by the appropriate LED.
2. Each song should be unique, repeating on loop for the duration of the given event. “Songs” are truly just patterns of a handful of different notes. The speaker should be able to play two distinct songs. One should play when the sprinkler is sprinkling, and one during the fire alarm.
3. The user must be able to adjust the fan speed at any time not in an emergency, and the DC motor should automatically adjust accordingly.
4. The fire alarm should have the ability to be activated at any time (i.e. even when in a menu).
5. The overall layout of your system should be easy to understand and make sense. The user should be able to operate the system with little to no training or explanation. If you are unsure if the layout of your system makes sense, ask one of your TAs or fellow students to try to move through your system.
6. No delay loops are allowed; you must utilize the Real Time Interrupt (RTI). DELAY LOOPS ARE ONLY ALLOWED FOR HEX KEYPAD DE-BOUNCE. HINT: If you are having trouble with the switches bouncing, maybe try only scanning them every X many RTIs.

**Note:**

You are encouraged to be creative and make this project your own. You can make reasonable assumptions in the development of this project, but keep in mind that the assumptions must make sense to the user (and to the Lab TAs) so be ready to explain your choices. This is not an excuse to cut corners.

For your project, 10% of the grade will be based on extras that you will create on your own. You are encouraged to be creative. You can either include one Gold Tier extra feature, two Silver Tier extra features, **OR** three Bronze Tier extra features to be eligible for full credit.

Ideas for Gold Tier Extra Features (just one required):

1. Create a timer for the sprinkler. During the sprinkling process, your LCD will display something that says, “Sprinkling Ending in (10,9, 8, 7)”.
2. Create an auto-lock system. If no key is pressed within 10 seconds when the LCD is in the settings menu, it reverts to the home screen.

Ideas for Silver Tier Extra Features (two required):

1. Create a Fahrenheit mode for temperature. A user can select either to display the temperature in Celsius or Fahrenheit mode on their digital clock.
2. Stepper motor turns during the fire state (this represents the carbon dioxide pump)
3. Create a 12-hour mode for time. A user can select either to display the time in 12-hour or 24-hour mode on their digital clock.

Ideas for Bronze Tier Extra Features (three required):

1. Password encryption. Whenever a key is pressed, the LCD displays “\*”.
2. Prompt the user to answer a security question if they forgot their password.
3. Allow the user to change the minute and hours.
4. Create a 4-digit password.

\*\*Any other ideas are welcomed. Please refer to your TA to check what tier those extra features are.

If you have any questions pertaining to this project, please discuss them with your Lab TAs as early as possible. You may use any C code provided through the lab, but all other code must be written in assembly.