**GROUP 2: MASTER CONTROL**

This folder is going to be our milestone/final demo code.

This is where you will obtain the most up to date version of the code and be able to upload any changes that you make to it.

The folder is organized into subfolders that are named as follows:

<date> <version> <editor initials>

Anytime someone makes ANY changes to the code, please create a new subfolder and name it with the above information.

For the version number, v1.xx will be for Milestone 4 updates and v2.xx will be for the final demo updates. Please follow the version numbering systems so that we can keep our code as up to date and conflict/error free. When you want to make a change to the code, download the most recent version, edit it, then add 0.01 to the version you downloaded. (For example: I want to edit version 1.13. Once I edit it and upload it, that version I created is now 1.14)

Below we will have version logs. There will be an “Ongoing changes” and a “Completed Versions” section. BEFORE YOU BEGIN TO EDIT THE CODE, ADD THE VERSION NUMBERS WITH A BRIEF DESCRIPTION OF WHAT YOU ARE GOING TO ADD/CHANGE. That way someone else won’t do the same task that you are currently working on. Use the following format for the “Ongoing changes” section:

**<version before edit> - <version after edit> <Your Initials>**

* Description:

Anytime you complete a change please fill out the changes you made to the code in the following format under “Completed Versions”:

**<version before edit> - <version after edit> <Your Initials>**

* (EXAMPLE) Fixed format on file <file name>
* (EXAMPLE) Added function <function name> to file <file name>
* (EXAMPLE) Created new file <file name>
* (REQUIRED) Compile command: gcc –Wall <file names> -o <output name> <flags>

ANYTIME YOU WANT TO EDIT CODE, PLEAS REFER BACK TO THIS DOCUMENT TO VIEW NEW UPDATES SO THAT YOU KNOW WHAT HAS ALREADY BEEN DONE OR WHAT IS CURRENTLY BEING WORKED ON!!

\*REMEMBER TO SAVE THIS DOCUMENT ONTO THE DRIVE TO UPDATE IT.

\*MAKE SURE YOUR CHANGES COMPILE CORRECTLY ON YOUR PI BEFORE UPLOADING IT AS A NEW COMPLETED VERSION.

**INITIAL BELOW IF YOU ARE ABLE TO OPEN AND EDIT THIS FILE CORRECTLY THROUGH THE ONLINE WORD DOC EDITOR AND YOU HAVE READ THE ABOVE!!**

|  |  |
| --- | --- |
| Name | Initials |
| Raul | RR |
| Christa | CM |
| Alex | AF |

**TO DO:**

* Filling in comment header blocks for each function in menufunctions.c

**ONGOING CHANGES:**

**COMPLETED VERSIONS:**

**V0.00 - v1.00 RR**

**Description:** Files in v1.00 are now up to date and are working/compiled versions of the code.

**Known issues:** Program not correctly looping from the main function call. Program will not terminate when power switch is turned off. Possible issue source is from the "updateDisplay()" function in the "menufunctions.c" file.

* Organized all functions in appropriate files. (All functions that have to do with the lcd/menu are in menufunctions.c etc)
* Changed the LCD handle from an integer variable to a global integer variable and updated all functions appropriately.
* Added in function: **void lcdBoot(void);**
  + On program start, program will wait until power switch is turned on. After, the lcd will display group names and course and begin running main code when done.

**v1.00 - v1.01 RR**

**Description:** Fixed the incorrect looping error in the "updateDisplay()" function in the "menufunctions.c" file.

**Known issues:** No known issues at this time.

* In "updateDisplay()": removed "while(1)" loops and removed redundant "smartDelay()" calls. Created a variable "int buttonPressed" to store the button which was pressed from only ONE call to "smartDelay()" in the beginning of "updateDisplay()" function. This makes it so "smartDelay()" isnt called multiple times in the "if/else if" checks. These changes make the code more efficient, responsive, and still works as intended.

**V1.01 - v1.02 RR**

**Description:** Created functions which will handle the syncing of devices and organize their addresses.

**Known issues:** no known issues at this time.

* Created new file,"syncing.c", to contain all syncing related functions.
* The following functions were added to "syncing.c"
  + Void initArray(void);
    - Initializes the arrays to contain 0's
  + Void populateArrays(void);
    - Takes the address that were populated into "addresses[]" array by protocol group and organizes the addresses into a two dimensional array called "devices[][]".
  + Void addTherm(void);
    - Adds a new thermostat address and organizes it into the two dimensional array.
  + Void addReg(void);
    - Adds a new register address and organizes it into the two dimensional array.

Compile line: gcc –Wall finalMain.c menufunctions.c setups.c syncing.c -o finalMain –lwiringPi –lwiringPiDev

**V1.02 - v1.03 AF**

**Description:** Created functions which will handle the display when sync button is pressed.

**Known issues:** Page numbers need to be assigned to the new displays created; the function display and updateDisplay need to take into account these new pages in order to make them responsive to up, down, and enter buttons press; the function Success Message also needs to be updated when the page numbers shall be assigned to the new displays.

This functions just do the display. It doesn't perform any actions. So, action functions still need to be build for the add, remove, and reinitialize options.

* I encountered some errors when compiling V1.02. I had to do the following updates in other to get rid of the errors

- /tmp/ccpOGJTj.o: In function `main':

finalMain.c:(.text+0x20): undefined reference to `setups'

finalMain.c:(.text+0x58): undefined reference to `initArray'

finalMain.c:(.text+0x5c): undefined reference to `populateArrays'

finalMain.c:(.text+0x74): undefined reference to `addTherm'

finalMain.c:(.text+0xa0): undefined reference to `addReg'

finalMain.c:(.text+0xa8): undefined reference to `updateDisplay'

finalMain.c:(.text+0x120): undefined reference to `newTherm'

finalMain.c:(.text+0x128): undefined reference to `newReg'

finalMain.c:(.text+0x130): undefined reference to `addToRoom'

In other to solve this, I had to include setup.c, syncing.c, and menu.c in the finalMain.c

- syncing.c:7:14: error: expected ‘;’, ‘,’ or ‘)’ before numeric constant #define rows 128

In other to solve this one I had to change the variable name rows to rws. My thought is that rows is a reserved word; that is why I changed it to rws. However, it might be something else.

* File updated: "menufunctions.c", to contain new display functions.
* The following functions were added to "menufunctions.c"
  + Void syncMenu(int);
    - Displays sync menu when sync button is pressed.
    - Menu contains Manage Device and Re-initialize options.
  + Void manageDevice(int);
    - When the option manage device is selected, this function displays the options add or remove device
  + Void addDevice(int);
    - This function displays the option add thermostat or add register for the user to select
  + Void removeDevice(int);
    - This function displays the options remove thermostat or remove register for the user to select
  + Void SuccessMessage(int, int);
    - This function was updated in other to take a new parameter page, and also to take into account the success message for add devices.

After doing all these modifications, the code compiled properly.

**v1.02 - v1.04 RR**

**Description:** Modified functions within syncing.c file so that they properly work alongside with the protocol group's functions. At this version, initialization with real device addresses, acquired through successful rf communication, has been successful. Also, in this version, changes made in version 1.03 have been merged in to include the new menu functions. However, changes made to fix errors listed above in log v1.03 have been reverted as the issues were not caused by code but by incorrect compile commands.

**Known Issues:** The new sync menu functions have yet to be implemented into the code so that they are used in runtime.

* Changed variable type of address arrays in syncing.c from integer arrays to unsigned char arrays. This is done so that these functions can properly work with the protocol group addressing method.
* Added in test code in main() function in finalMain.c file to test the syncing process with the protocol group. This test code is currently commented out for testing purposes.

This version of the code correctly compiles and works as is.

**Compile command used:**

gcc –Wall finalMain.c menufunctions.c setups.c syncing.c messaging.c -o finalMain –lwiringPi –lwiringPiDe

**v1.04 - v1.05 AF**

**Description:** This version assigns page numbers to the new pages created. It also updates the successMessage function, updateDisplay function, and Display function in order to take into account the new pages created

**Known Issues:** Appropriate comments still needs to be done on the function headers. The success message of the reinitialization option still needs to be updated

* The updateDisplay was completely modified in other to take into account all the 9 pages. The previous version had an if else statement for each page; this made the function very long and repetitive. It was made in a more global way in other to avid repetition and made the code shorter.

This version of the code correctly compiles and works as is.

**Compile command used:**

gcc –Wall finalMain.c menufunctions.c setups.c syncing.c messaging.c -o finalMain –lwiringPi –lwiringPiDev

**V1.05 - 1.06 RR**

**Description:** This update has a new file "time.c" which contains the functions necessary to have a real time clock running in background. This clock will be used to display the current time on the LCD. This clock is also used to provide a timer/stopwatch feature for our program so that we may record time intervals between temperature changes. This file also contains a function which parses the hour and minute integers into their tenths and ones (if hour is 12, separate the 1 and 2 so that they may be displayed into the lcd properly as single digit integers).

* New file "time.c": contains all timing related functions.
  + Created: "getSec()" returns the current seconds from the internal clock on the pi.
  + Created: "initTimeArr()" initializes the mcClock[] array to fill it with predefined known values.
  + Created: "updateTime()" calls the getSec() function and updates the time stored in mcClock() when necessary.
  + Created: "parseTime()" takes the time in mcClock[] and separates the hour and minute values into single digit numbers.
  + Created: "rtTimeStart()" This is the first part to the timer/stopwatch function. This function records the current time into a separate array which will be used to find the difference between the start and end time.
  + Created: "rtTimerEnd() this function is currently unfinished. However, it will take the current time and find the difference between the start time and current time and return the difference in seconds.
* Modified the "addTherm()" and "addReg()" functions so that they actually call the protocol function to establish a connection with a device and obtain a new address for it.
* Added in comment header blocks for the functions in timer.c and syncing.c.

**Known issues**: no known issues at this time, the code compiles correctly with the compile line below.

**Compile command used:**

gcc –Wall finalMain.c menufunctions.c setups.c syncing.c time.c messaging.c -o finalMain –lwiringPi –lwiringPiDev