

Analysis

Lab 4 is related to 7 segment display. Several objectives are identified in this lab, including 1) scrolling my name on the display; 2) using the on-board button to instantaneously toggle between my name and my Z number scrolling on the display; 3) using the potentiometer to control the speed of scrolling; and 4) improved user experience with no LED bleeding and smooth scrolling. The given code provides the basic framework in displaying characters, including the `matrix[]` array, the `digitShow()` function, and codes for multiplexing 7 segment display. This lab extends the given code to achieve the objectives stated above.

Design

Objective 1 is achieved by using a char array `d[]` with a size of 4 storing the characters displayed on each of the 4 digits, with its content updated from another char array `name[]` used to store my name. `d[]` is updated after each digit is displayed for a certain duration, thereby achieving a scrolling effect.

Objective 2 is achieved by using hardware button implemented with Timer2 and CLC1 similar to Homework 2, and expanding `d[]` to a size of 8, with `d[0]` to `d[3]` used for my name and `d[4]` to `d[7]` for my Z number stored in another char array `zNum[]`. When CLC1 output is LOW, the 4 elements starting at `d[0]` is displayed and updated with my name; otherwise, the 4 elements starting at `d[4]` is displayed and updated with my Z number. Hardware button allows simple implementation of instantaneous toggle.

Each digit is displayed for a duration proportional to the number of iterations that the multiplexing codes run in a `for` loop. This also determines the scrolling speed since the content of `d[]` is updated at the end of the `for` loop. Therefore, objective 3 is achieved by updating the number of iterations of the `for` loop proportional to the analog read value from the potentiometer.

In objective 4, LED bleeding is caused by the residual current on the 7 segment pins from the previous digit. This is resolved by displaying a SPACE (all segment pins are LOW) first before displaying the current digit. Smooth scrolling can be achieved by using a wrap-around calculation with modulo (%) of array size of `name[]` or `zNum[]` when locating the elements to update `d[]`. In addition, constant current driver is enabled on the 7 segment pins instead of the common cathodes so the brightness of each segment is even across characters.

Implementation

The `main` function performs 2 main tasks within the `while(1)` loop: 1) depending on the value of `CLC1CONbits.LC1OUT`, update `d[]` and the next position to read in either `name[]` or `zNum[]`; 2) execute the `for` loop which updates the number of iterations based on the potentiometer reading and displays 4 elements starting at `&d[4*CLC1CONbits.LC1OUT]` to the 7 segment display.

In addition to the function provided in the given code, several additional functions, discussed below, were created for the implementation. Detailed descriptions are included in the Javadoc in the code.

- `LEDDisplay(char*)` contains modified multiplexing codes, and displays the first 4 elements of the char array argument on the 7 segment display. This is run inside the `for` loop in the `main` function.
- `updateMessage(char*, char*, uint8_t, uint8_t)` updates the content of `d[]`.
- `charToIndex(char)` is a utility function which takes a char as argument and returns the corresponding index in the `matrix[]` array.
- `charArrayLength(char*)` is a utility function which determines the number of characters in a char array, excluding the NUL ('\\0') character.

User Interface

There is no user interface on the Evaluation Board to update the displayed message. The message can be updated by updating the initialized values of `name[]` or `zNum[]` in the `main` function, recompiling the code, and loading the updated hex file to the MCU. A SPACE should be added at the end of the message. Other modifiable parameters include the multiplexing delay in the constant `MULTIPLEX_DELAY` which controls the refresh rate, and the constant current driver current limit in the `CCDCON` register which controls the brightness of the 7 segment display.