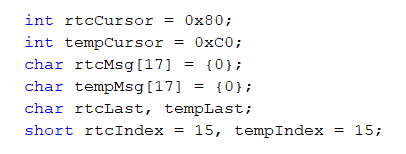
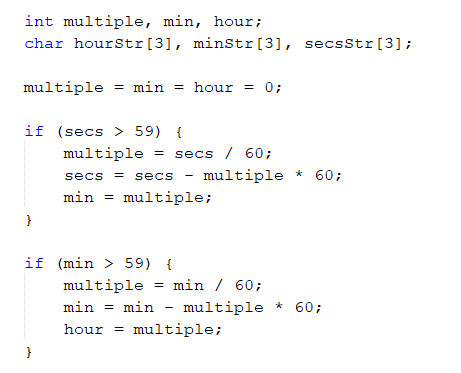
For our function for the Serial LCD, we need to display the temperature reading and real-time clock recorded by the temperature sensor and timer. However, we added in some moving and flashing animation functions into our code to make our application look more appealing.





For the declaration of the variables, we first declare the initial cursor position of the real-time clock and temperature display, with the real-time clock showing on the first line at position 0x80 on the SPI\_LCD, while the temperature reading is shown on the second line of the serial LCD at position 0xC0. Both displays are put inside an array of size 17, and we use a for-loop to display the characters on the serial LCD.

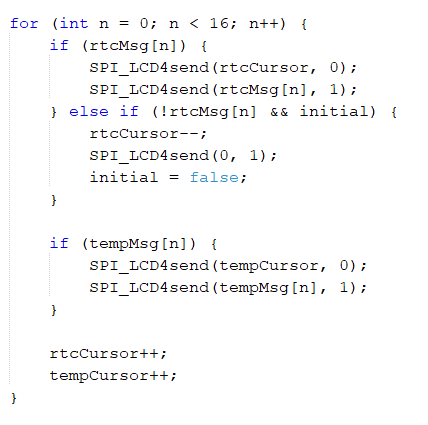


For this part, this is to create a proper conversion of the timer. In other words, if the seconds of the time display goes beyond the value of 59, it will result in the value of the seconds going back to 0, the minute value will increase. This code is also applied to the increment value of the hour when the minute value goes beyond the 59 value.

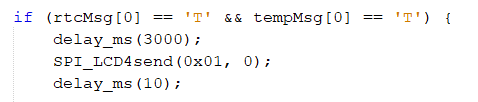
****

This part is just to display a two-digit number if the values of each part of the timer is more than 9.

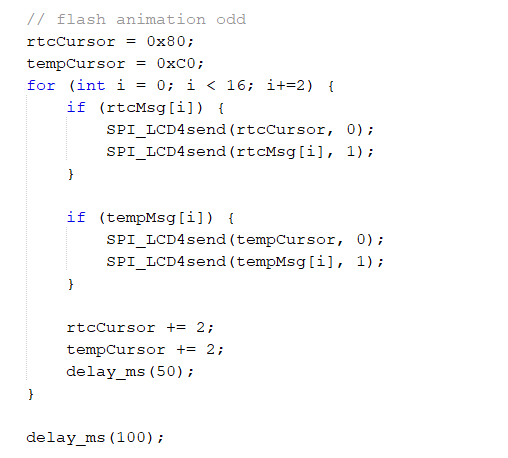




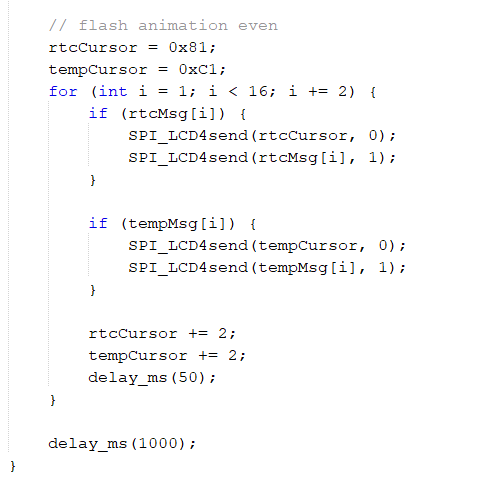
This for-loop is used to display the entire message lines of the real-time clock and temperature reading, and let every single character appear at their exact locations. By doing so, every time the for-loop loops, the **n** value representing the position of each character of the message lines will increase to match with the cursor position, which will also increase every time the for-loop loops.



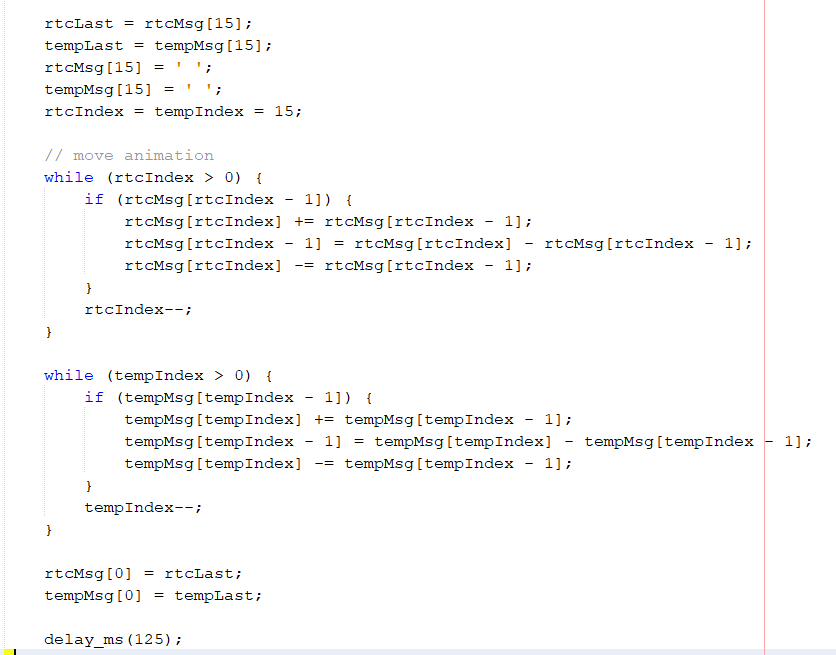
For the moving and flashing animations code, the entire message display of the real-time clock and temperature reading will pause for about 3 mins when the first character of both lines is at the exact initial position of 0x80 and 0xC0 respectively.



This part of the flash animation code is to allow the display characters located at odd positions to flash first before the even characters.



After the odd display characters have finished flashing, the characters located at even positions on the LCD will then start to flash as well.



This part is for the moving animation of the entire LCD display. After the flashing animation has completed its round, the entire LCD display will then start to move from left to right, and continue to move out of the Serial LCD and move back to its original position before it pauses.

Both the flashing and moving animations are put inside a while loop to constantly repeat the process until the program stop.