Ahmed Adil (18L-1289)

Mahad Naveed Qureshi (18L-1039)

Limitations and suggested improvements

# Usage of timer 1 only:

* Timer 0 was also used previously which after completion started timer 1 that subsequently used to stop the clock for some time. Now the alarm checks were also implemented in timer 1 that resolved the time issue. This was the major limitation which was resolved.

# Display message (strings):

* As soon as the clock starts, a displaying message appears on the LCD screen. Which prompts the user to enter the input according to the given format; \*/#HHMM\*/#

# Input checks (clock):

* Check for scanning whether the 1st input is either # or \*

If not then prompting the user to enter the input again.

* Checks for entering hours in the clock were added. As we had to design a 12Hr clock certain checks were necessary to apply to meet the requirements.

H1 (1st numeric input) can either be 0 or 1; (H1 <=1)

If (H1=0)

H2 (2nd numeric input) should be less than or equal to nine 9; (H2 <= 9)

If (H1 =1)

H2 should be less than or equal to 2; (H2 <=2)

Else

Enter the input again

* Check for entering minutes in clock. Since there is a certain rule in time clock that after 59 the minutes changes back to 00 with an increment of 1 in hours, so changes were made accordingly.

M1 (3rd numeric input) should be less than or equal to 5; (M1 <=5)

Else

((x-9) + H2); where x is the number greater than 9

M2 (4th numeric input) should be less than or equal to 9; (M1 <=5)

If greater than 9 then increment M1

Else

Enter the input again

* The last input should also be # or \*.

# Alarm check:

* When the alarm time is equal to the clock time a check is implemented that checks the AM/PM of both the times.

Drawback:

when a wrong input is given at first and then updated with the correct format, the last value of the input does not show up on the screen for 1 minute. After 1 minutes as it refreshes the problem gets resolved