DT228-2 Micros Lab. Keeping track of time.

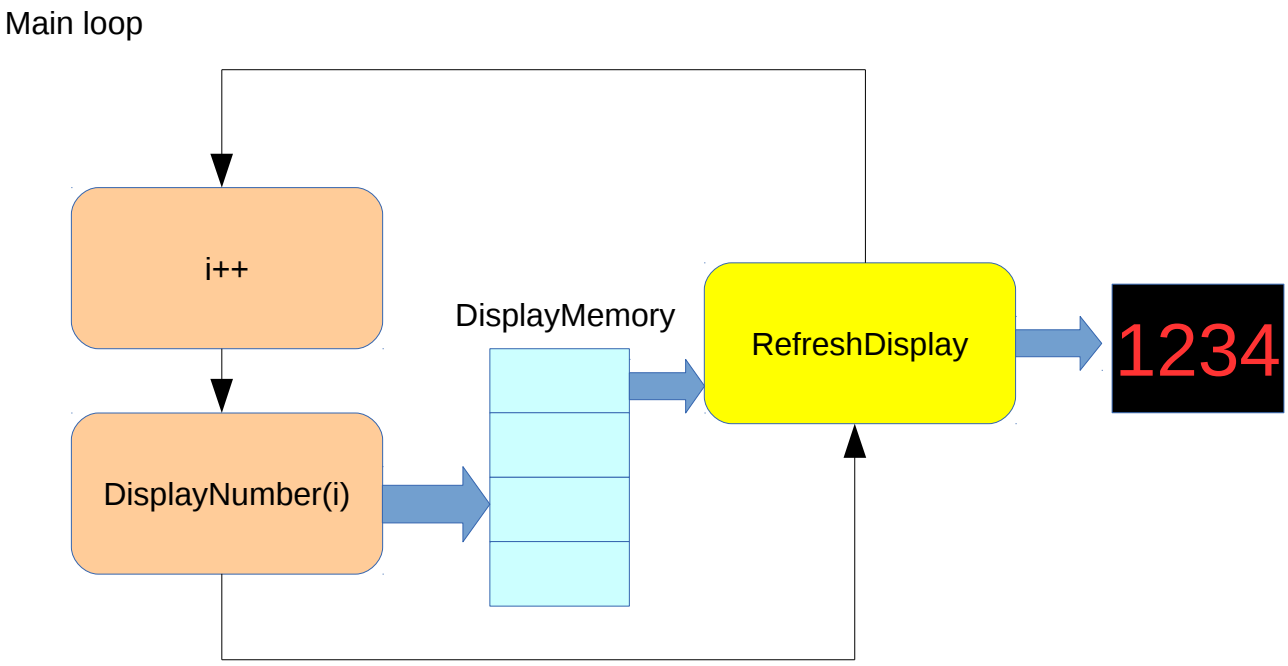
Key topics: Frequency, Period, Interrupts, Cortex SysTick, task switching.

Introduction.

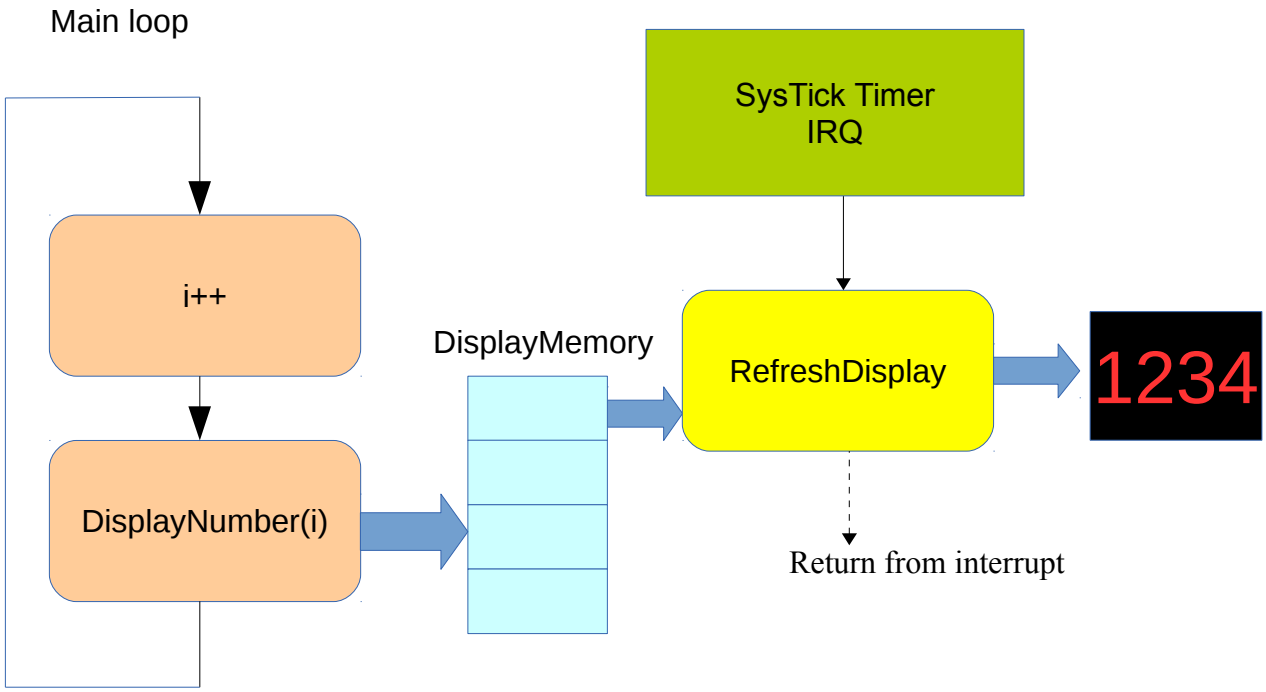
This lab introduces the concept of interrupts; in particular, the periodic ARM Cortex SysTick interrupt. The starting point of the exercise is a modified version of the previous laboratory assignment which presented an increasing count on the LED display. This new version of the display program uses the SysTick interrupt to refresh the contents of the display. This frees the main function to do other work without fear of display

Theory of operation

The previous version of this lab used a polling method to refresh the display. This involved repeated calls to the RefreshDisplay function inside the main program loop



The new version of the program uses interrupts to generate a predictable, periodic call to RefreshDisplay (without the need for an explicit call from other code). This allow the main program loop to do other things. Furthermore, the fact that we now have an accurate periodic interrupt opens up the possibility of new applications. The new structure may be represented as follows.



Initial testing

Using your assembled board from the previous lab, download the supporting code for this exercise, unzip it to your home directory. Build and download to the board. You should see the counter advance quickly on the display – much quicker than before as the software delay function is not used.

Extending the code

The TV series "24" involved a certain agent saving the world over a 24 hour period. Given that our lab sessions are only 2 hours long, it would be unrealistic to ask you to implement a 24 hour count down – we will instead go for 24 minutes. Your task is to modify the code so that it implements a 24 minute countdown. At the end of the 24 minutes, the display should show some non-numeric pattern (perhaps animated). To help testing you can adjust the speed of the count by changing the millisecond counter code (make seconds pass faster).