

Lara Terzi CRTAP sem 1

by Lara Terzi

FILE	3643_LARA_TERZI_LARA_TERZI_CRTAP_SEM_1_168540_1478503304.D OCX (19.46K)		
TIME SUBMITTED	22-MAR-2019 09:36PM (UTC+0000)	WORD COUNT	807
SUBMISSION ID	103049119	CHARACTER COUNT	5536

Lara Terzi

P148014

BSc Software Engineering

Critical and Real-Time
Application Programming –
Semester 1 Assignment

Mike Dacey

Contents

1. Knowledge and Understanding Skills.....	3
2. Practical Skills.....	3
2.1. Combining 2 bytes	3
2.2. Combining 2 arrays (loop for each byte)	4
2.3. Pass pixel image as array (loop for each byte).....	4
2.4. Combine pixel images	4
2.5. Return a pointer to a displayable image	4
3. Cognitive and Intellectual Skills	4
3.1. Timing Constraints and Delays.....	4
3.2. Safety Constraints	4
4. Citations	5
5. References	5

1. Knowledge and Understanding Skills

A hard real-time application refers to an application whose deadlines must be met punctually, thus the project must be developed in an accurately cyclic manner and deadlines must be guaranteed. This system is typically used in life or death situations such as in nuclear power plants or medical equipment, where one small unaddressed bug can cause a large amount of complications. The word 'hard' in hard real-time applications refers to the need for the deadline to be met in the millisecond region. "A real-time task is said to be hard if producing the results after its deadline may cause catastrophic consequences on the system under control" [1].

Unlike a hard real-time application, a soft real-time application may miss some deadlines, however the more deadlines are missed, the more the application's performance will decline. An example of this is a movie's pixilation. At times, part of a scene's pixels will become corrupted. If the scene is missing only a few pixels, it can still be viewed. However, if too many pixels become corrupted, half of the screen ends up missing pixels and thus the scene mal-performs. "A real-time task is said to be soft if producing the results after its deadline has still some utility for the system, although causing a performance degradation" [1].

"A car satnav is used to guide a driver, and upon identifying a waypoint updates the screen to remind the driver with a latency of 1.5s." This application scenario uses soft real-time application requirements. The latency of 1.5 seconds is far greater than what a hard real-time application's latency should be. The latency will not cause a fatal error, thus a hard real-time application is not needed.



2. Practical Skills

2.1. Combining 2 bytes

.global main

main:

MOV R1, #3

ADD R0, R1, #2

MOV R2, #1

SWI 0

2.2. Combining 2 arrays (loop for each byte)

2.3. Pass pixel image as array (loop for each byte)

2.4. Combine pixel images

2.5. Return a pointer to a displayable image

3. Cognitive and Intellectual Skills

3.1. Timing Constraints and Delays

With a simple wash and dry cycle, the time constraints are as such time delays occur after each operation. When the machine is turned on, there is a 3 second delay before the system is operational. Once the program is selected, there is a 5 second delay before the water is pumped into the machine for 15 seconds. The machine then spins for 1 hour, changing direction every 20 seconds, with a 5 second delay before each direction change. Water is pumped out of the machine every 20 minutes for 15 seconds with what once can assume is a 5 second delay before pumping in more water for 15 seconds, again with a 5 second delay. Once the washing cycle has completed, there is a 5 second delay before the machine continues to spin for another 20 minutes pumping out water continuously. Every 20 seconds the spin direction is changed, with a 5 second delay before the spinning recommences. Once the spin cycle has completed, the machine continues to spin for 40 minutes, changing spin direction every 20 seconds with a 5 second delay whilst the heater is turned on. Once the drying cycle has finished, the machine cools down for 5 minutes before the speaker signals the cycle has finished, with a possible minute delay.



3.2. Safety Constraints

When addressing the safety of washing machine users, one must ensure that the system does not allow the user to open the door whilst a program has not completed. Similarly, if the machine is turned off mid-program, the system must stop users from being able to open the door whilst there is still water inside. The system must also guarantee that the door will not open if the machine's temperature is too high, to ensure users do not burn themselves on the metal interior. It is also very important for the machine to turn itself off in the case of critical hardware malfunction.

4. Citations

[1] G. C. Buttazzo, *Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications*, 3rd ed., New York: Springer, 2001, pp.9



5. References

Stack Overflow, Differences between hard real-time, soft real-time, and firm real-time? [Online]. Available: <https://stackoverflow.com/questions/17308956/differences-between-hard-real-time-soft-real-time-and-firm-real-time>

MeroSpark, Differences between hard and soft real time system [Online]. Available: <https://csit.merospark.com/sixth-semester/difference-hard-soft-real-time-system-real-time-system/>

FINAL GRADE

40/100

GENERAL COMMENTS

Instructor

Knowledge (20): 15

Cognitive (40): 23

Practical (40): 2

Nice description of soft/hard RT systems but the sat nav scenario required a more consideration.

Requirements analysis is a bit too terse, although the initial description was interesting.

No useful code provided.

PAGE 1

PAGE 2

PAGE 3



Comment 1

I would like to have seen a bit more debate and justification for your conclusion.

PAGE 4



Comment 2

This analysis good but more detail was required, it is a bit brief.

PAGE 5



Comment 3

Not enough references