University of South-Eastern Norway

Faculty of Technology, Natural Sciences and Maritime Sciences
Campus Porsgrunn

Industrial IT and Automation (IIA)

Course IIA2017
Industrial Information Technology

Multitasking and real-time assignment (GuiSw)
Version 1.0

Nils-Olav Skeie

April 11, 2023

Multitasking and real-time assignment tasks descriptions

1.1 Introduction

This assignment is a combination of theory, practical exercises using both an application and developing your own application, and calculation regarding time requirements in a multitasking system. The tasks in this assignment is based on an application 1) giving a set of theory exercises, 2) running a set of threads (tasks) and 3) giving a set of parameters that are specific for each candidate. See Figure A.1 for Scheduler Setup tab page of the application.

Use the knowledge from the multitasking and real-time, instrumentation and Ex sections in this course to evaluate and solve these assignment tasks. The goals of this assignment is to use the theory to understand 1) the running times of tasks in an application when sharing one or several resources, 2) how to estimate any time requirements for an industrial process and 3) how to develop an application simulating a Data Acquisition System (DAQ) with focus on analog and digital input devices.

The results of the tasks must be documented in a technical report, delivered as a pdf file in *the Learning Management System (LMS),* within the deadline (due date), and must be approved to pass this course. This includes writing a technical report with the introduction, theory, results and discussion/conclusion parts and include references. You should use the IMRaD (Introduction, Methods, Results and Discussion) method when structure and writing the report. Remember to deliver within the due time. If an extension is needed, ask **BEFORE** the due time. This assignment is a soft real-time task!

The version history of this document:

Version	Description	
0.1	First version, include console application and RTOS specification page.	NOS−08
0.2	Extend document and application with GUI, runtime window and real-time evaluation.	NOS−10
0.3	Extend document and application configuration and analysis part.	NOS−12
0.4	Extend document and application with theory section.	NOS−14
0.5	Extend document with more analysis, pseudo coding and development of C# application.	NOS−17
0.6	Minor adjustment for some of the task descriptions.	NOS−17
0.7	Simplifying the task sections.	NOS−18
0.8	Add more hints for the evaluation of the real-time system.	NOS−19
0.9	Minor adjustments and updates for the assignment report.	NOS−20
1.0	Extend document and application with instrumentation and Ex parts.	NOS−23

1.2 Assignment Introduction (2%)

Use either the MS-word report template available in the LMS or use Latex and the book template with the first page similar to the MS-word template. Do NOT make your own templates. **Make a screen dump of the Scheduler Setup tab page for the introduction section of your report**. This screen dump is needed to be able to evaluate your answers. The screen dump must be made of the main screen, after the input fields are approved. Use figure text, references to the figures in your text, and add references to support your statements. A technical report must contain several references, may be to this task description and the course lecture document (Skeie n.d.).

An introduction informs the reader what the report is about and sets the project in a wider context. The introduction should also provide some background information that the reader needs to understand the topic in the report. The introduction should 1) introduce short the context of the topic, 2) explain the problem and/or motivation for the topic, 3) aims, purpose and contribution of the project, and 4) briefly outlines the report structure (not mandatory for this report).

The requirements for getting this assignment approved is that 1) **minimum** 65% of the tasks must be completed and approved, and 2) all tasks must be started with some type of analysis.

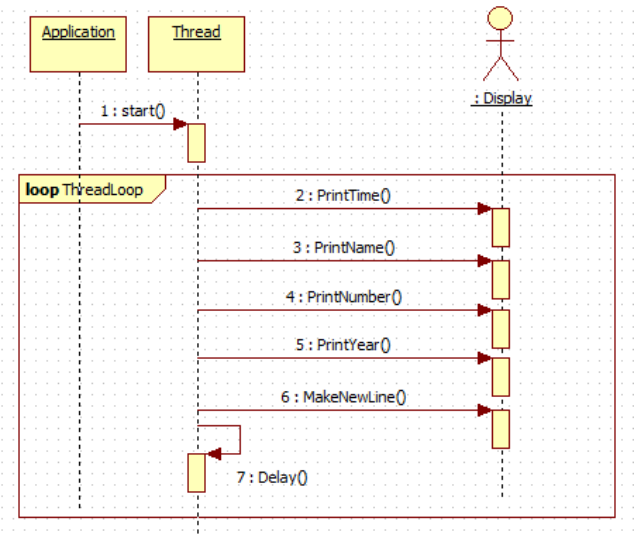

Figure 1.1: A System Sequence Diagram (SSD) for the application.

1.3 Theory (26%)

Use the Theory Exercise tab for the theory part. **Start this section with a screen dump of the application with the Theory Exercise tab activated.** Please use your own words in answering, a good training for the real-life as well as the final test and the master thesis. When copy information, use references. This section cover both multitasking, real-time, Ex and instrumentation. Limit your answers to six sentences for each exercise. This report section should contain (as a minimum):

A) a screen dump of the application with the "Theory Exercise" tab activated and include in this chapter.

B) your own answers for all the theory exercises from the application (Some of the questions may be repeated).

C) This section SHOULD contain references to support your statements.

1.4 Evaluation of a multitasking system (25%)

Use the Running Tasks tab page for these exercises. Base your evaluation(s) and assumption(s) that the application should be used in a multitasking system. The application is running in the Windows environment and using the scheduler in your Windows system. Each thread (task) performs a fixed number of loops and in each loop the thread (task) will print some information starting with $[Tx]$ where x is the thread (task) ID. The goal in this part is to evaluate the real-time performance of the system. The real-time performance for a system, or a task, is always about the total time, the maximum time, for execution of the system or the task, and this time depends on the type and number of resources used by the system or the task. The report should contain (as a minimum):

A) Analyze the system based on the multitasking and real-time requirements, with focus on this application only. A System Sequence Diagram (SSD) of the application is shown in Figure 1.1.The real-time requirements will be any factor that may affect the running time of the tasks (threads). The real-time requirements involve tasks running in parallel, any common resources used by the application that will affect the timing and the time used by the application. **Hint:** the application consists of several tasks sharing one or several resources. Evaluate these common resources disregarding cpu and computer memory. Is the running time affected by sharing common resources? How to share common resources in a multitasking system?

B) Select the running code 1 in the Scheduler Setup tab page and switch to the Running Tasks tab page. Use the Run button to start the tasks. Use the *Clear* button if the *Run* button is not activated. All tasks should stop after approximately 5 to 50 seconds. Include two screen dumps of

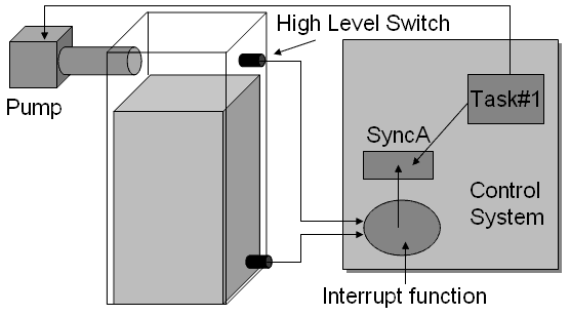

Figure 1.2: A sketch of the control system for the liquid vessel with the pump, two switches and the control system.

the output from your application in the report, one screen dump for the start of the threads and a screen dump of the end. Include a discussion of the number of loops, the delays in the loop and running time for each thread.

C) Select the running code 2 in the Scheduler Setup tab page and switch to the Running Tasks tab page. Use the Run button to start the tasks. Use the *Clear* button if the *Run* button is not activated. All tasks should stop after approximately 5 to 50 seconds. Include two screen dumps of the output from your application in the report, one screen dump for the start of the threads and a screen dump of the end. Include a discussion of the number of loops, the delay in the loops and running time for each thread. Comment the display output for running code 1 and 2.

D) Select the running code 3 in the Scheduler Setup tab page and switch to the Running Tasks tab page. Use the Run button to start the tasks. Use the *Clear* button if the *Run* button is not activated. All tasks should stop after approximately 5 to 50 seconds. Include two screen dumps of the output from your application in the report, one screen dump for the start of the threads and a screen dump of the end. Include a discussion of the number of loops, the delay in the loops and running time for each thread. Comment the running pattern after all treads has been running more than 5 loops.

E) The application is using mutex to protect the common resource(s). Discuss short the difference of a mutex and a semaphore regarding this application, and explain how running code 3 should work if using a semaphore.

F) Make a short conclusion of the results of this system based on 1) the analysis done in the first exercise in this evaluation part, 2) the running time for each threads in code 1, 2, and 3, and 3) why the running patterns for the threads in code 3 are changed after five loops . **Hint:** Focus on sharing resources and time requirements.

1.5 Time requirements for a real-time system (20%)

You are going to select a real-time operating system for your control system running an one core cpu system. The control system is shown in Figure 1.2.

The system consists of a tank with liquid and a pump for filling the tank. The control system has no information about the input or output flow of the vessel. The vessel should **never** be overfilled nor empty, giving that a hard real-time system should be used. Two switches are used, one high level switch and one low level switch. The switches are connected to an external interrupt signal with the highest interrupt priority level in your system, except the timer controlling the scheduler. The system needs to perform maximum 20 instructions in the interrupt function. A synchronization mechanism is used to inform the control task, task#1, to control the pump. The task #1 needs to perform between 150 and 200 instructions before the state of the pump can be changed. The state change delay of the pump is maximum 200 ms. Task #1 will be the only task at the highest priority running level. The pump must be turned off within 250 ms after an active signal from any of the level switches. You can choose between 4 different real-time system with the specification as listed in the *RTOS Specification* tab page in the application. The system will have several other tasks (task#2 - task#N) running on lower priority levels. The report should contain (as a minimum):

A) Include a screen dump of the *RTOS Specification* tab page in your report.

B) Which operating system (OS) can/will you choose for your control system? **Discuss your answer including a timing diagram of the system!**

1.6 Development of a DAQ simulator (25%)

Develop a DAQ simulator according to the parameters giving in the Instrumentation tab in the application. This DAQ simulator should only simulate input devices and can be very useful for more automatic testing of process control systems (PCS). The DAQ simulator should simulate the number of sensors defined, using the input voltage and the ADC bits to define the accuracy of the digital representation of the sensor values. The sensor values should be logged on a comma separated values (csv) file. The type of application can be either a console application or a GUI application. The digital representation of the sensor values should be presented on the screen when the system is running in addition to saving to the csv text file. The report should contain (as a minimum):

A) A flow chart of the application,

B) A discussion of developing this application as a single- or multitasking application,

C) A screen dump of the user interface,

D) An extract of the csv file,

E) The source code included as an appendix,

F) A discussion of integrating a configuration module in this application.

1.7 Report conclusion (2%)

Make a conclusion where you summarize the main points or goals of the tasks with focus on your most important findings and/or results. Do not add any new information in the conclusion. Do not focus on the details, focus more on the overview "picture" and examine the greater significance of what you have done. Leave your readers with something to think about. You need to fulfill minimum 65% of the tasks to get this assignment approved. Make a self evaluation of your effort for each sections and include the following table with your evaluation in the report.

Section	Requirements (%)	Your evaluation (%)	Comments
Introduction	2		
Theory	26		
MT system	25		
Time Req.	20		
DAQ sim.	25		
Conclusion	2		
Sum	100		

Appendix A

Usage of the application

Download and start the application, fill in your name, student number and semester year (four numbers), and press the "Get System Information" button (startup section). The application contains a configuration part, a simple scheduler, and sections for theory exercises, real-time operating system specifications and DAQ simulator specifications. The configuration part is using your name, your student number and semester year for loading a set of data for the scheduler and specifications. A screen dump of the main page, after pressing the "Get System Information" MUST be included in the introduction of your report.

Use the tab page selection to select the different part of the application. These tabs contains among others information about:

- application setup and configuration,
- theory section with a set of questions to be answered/explained, with references to support your statements,
- showing information from a set of running tasks,
- specification of the different RTOS that can be used in a control system,
- specification for the DAQ simulator.

Figure A.1 shows the startup page of the application after filling inn the requested information. Figure A.2 shows the *Running Tasks* tab page running the tasks specified in the *Scheduler Setup* tab page.

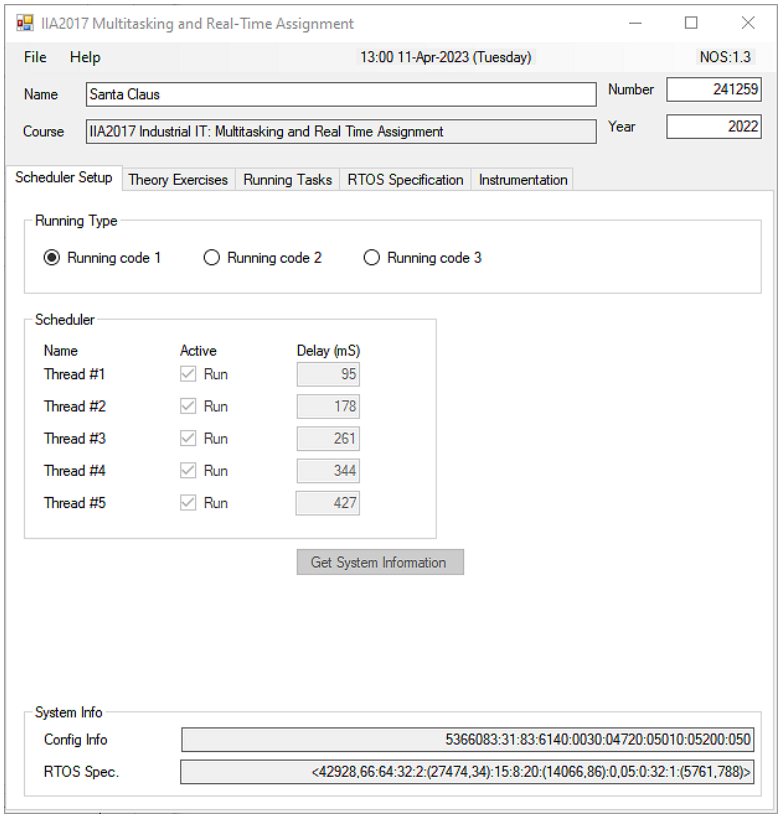

Figure A.1: The startup page of the application with the requested information filled in, and button pressed.

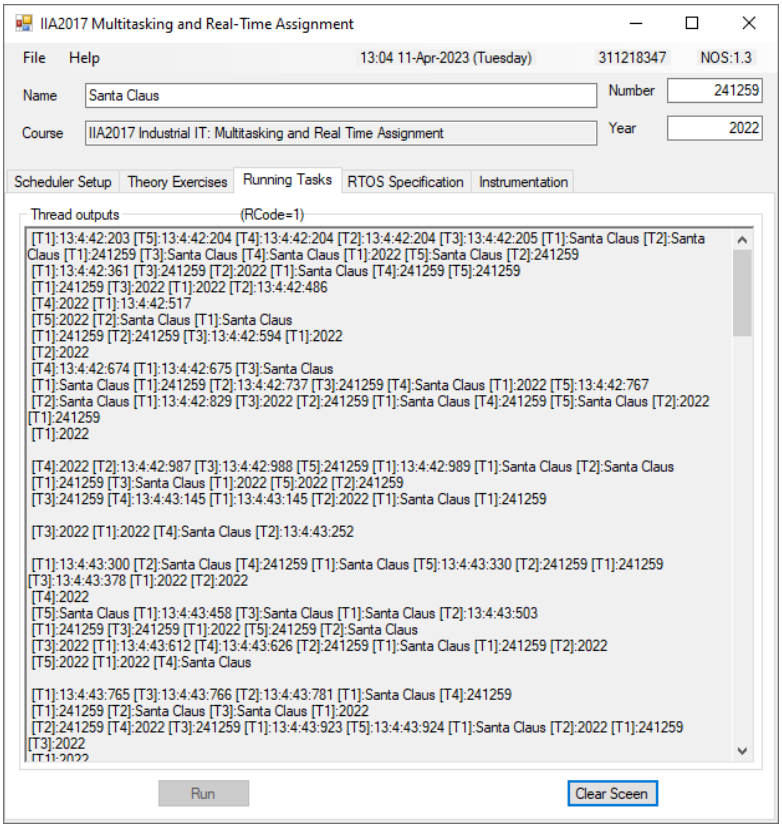

Figure A.2: The *Running Tasks* tab page showing the running of the tasks specified in the *Scheduler Setup* tab page.

Bibliography

Skeie, N.-O. (n.d.), 'IIA2017: Industrial information technology', Lecture notes for the master course IIA2017 at the University College of Southeast Norwat (USN) (and SCE2006 at the Telemark University College (TUC)). A master course for data communications, field buses, OPC, process databases,embedded systems, mechatronics, and real-time systems for industrial systems.