LUNDS TEKNISKA HÖGSKOLA
Lunds universitet

Institutionen för
REGLERTEKNIK

Real-Time Systems

Exam January 10, 2012, Hours: 14:00-19:00

Points and grades

All answers must include a clear motivation and a well-formulated answer. Answers may be given in **English or Swedish**. The total number of points is 25. The maximum number of points is specified for each subproblem.

Accepted aid

Standard mathematical tables and authorized "Real-Time Systems Formula Sheet". Pocket calculator.

Results

The result of the exam will be posted on the notice-board at the Department. The result as well as solutions will be available on WWW:
http://www.control.lth.se/Education/EngineeringProgram/FRTN01.html

1. Consider the continuous-time system

$$\frac{dx}{dt} = Ax + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u$$
$$y = \begin{pmatrix} 1 & 0 \end{pmatrix} x$$

Sample this using ZOH with sampling period h for the two cases below.

a.

$$A = \begin{pmatrix} 0 & 0 \\ 3 & 0 \end{pmatrix}$$

(1 p)

b.

$$A = \begin{pmatrix} 0 & 1 \\ 0 & -4 \end{pmatrix}$$

(1 p)

2. Consider the two following systems

$$y(k+2) - 0.25y(k) = u(k+1) + 2u(k) \qquad (1)$$
$$y(k+2) - 3.5y(k+1) + 1.5y(k) = u(k+1) - u(k) \qquad (2)$$

a. What are the pulse transfer functions from u to y for the two systems? (1 p)

b. What are the poles and zeros for the two systems? (1 p)

c. What are the static gains for the two systems? (2 p)

3. The table below describes three processes with their respective periods, deadlines and worst case execution times.

Task	T_i	D_i	C_i
A	40	40	1
B	4	4	1
C	50	2	1

a. What is the total CPU utilization? (1 p)

b. Will all deadlines be met if earliest deadline first scheduling is used? (1 p)

c. Will all deadlines be met if fixed priority scheduling with deadline monotonic priority assignment is used? (1 p)

d. Will all deadlines be met if fixed priority scheduling with rate monotonic priority assignment is used? (1 p)

4. The system

$$x(k+1) = \begin{pmatrix} 8.3 & 6.0 \\ 3.2 & 14.0 \end{pmatrix} x(k) + \begin{pmatrix} 0 \\ 0.25 \end{pmatrix} u(k)$$

should be implemented using fixed-point arithmetic. All coefficients should be stored, using the same number of fractional bits, in 16-bit signed integers.

a. How many fractional bits should be used in order to maximize the accuracy without getting any overflow in the coefficients? (1 p)

b. Calculate the fixed-point representations for all coefficients. (1 p)

5. Conny has implemented a program in Java to control a process using several threads handling the control, GUI, and reference generation. The control method used is PID-control. The code of the run-method of the Regul-class and some necessary initializations of parameters are shown below. The controller makes use of a PID class in which the calculateOutput and updateState methods are synchronized

Describe at least three things that can be improved with the code in the run-method. Write the improved code. (3 p)

```
public class Regul extends Thread {
  private ReferenceGenerator referenceGenerator;
  private PID controller;
  private AnalogSource analogIn;
  private AnalogSink analogOut;
  private double uMin = -10.0;
  private double uMax = 10.0;

...

  private double limit(double u, double umin, double umax) {
    if (u < umin) {
      u = umin;
    } else if (u > umax) {
      u = umax;
    }
    return u;
  }

  public void run() {
    double u;
    double y;
    double ref;

    while (true) {
      y = analogIn.get();
      ref = referenceGenerator.getRef();
      u = limit(controller.calculateOutput(y, ref), uMin, uMax);
      controller.updateState(u);
      analogOut.set(u);
```

```
        try {
          sleep(controller.getSamplingPeriod());
          } catch (InterruptedException e) {
        }
      }
    }
  }
```

6. Consider the following discrete-time system

$$x(k+1) = \begin{pmatrix} 1 & h \\ 0 & 1 \end{pmatrix} x(k) + \begin{pmatrix} h^2/2 \\ h \end{pmatrix} u(k)$$
$$y(k) = \begin{pmatrix} 1 & 0 \end{pmatrix} x(k)$$

a. Design a state feedback controller $u(k) = -Lx(k)$. Choose the closed loop characteristic polynomial in such a way that the sampled dynamics with $h = 0.1$ s corresponds to a continuous-time double pole in $s = -1$. (1 p)

b. Design an observer on the form

$$\hat{x}(k+1 \mid k) = \Phi\hat{x}(k \mid k-1) + \Gamma u(k) + K\Big(y(k) - C\hat{x}(k \mid k-1)\Big)$$

such that the observer dynamics is twice as fast as the closed loop dynamics, i.e., corresponds to a continuous-time double pole in $s = -2$. (1 p)

c. Design a model and feedforward generator in such a way that

1. The states of the model are compatible with the process states.
2. The discrete-time model dynamics corresponds to a continuous-time double pole in $s = -2$.
3. The steady state gain from the command signal u_c to y should be 1.

(2 p)

d. Draw the block diagram for the closed loop system including the process, the observer, the state-feedback, and the model and feedforward generator. In order to get full points the internal structure of the model and feedforward generator should be clearly shown. (1 p)

e. Write pseudo-code for the calculations that should be implemented in the controller, i.e., the calculations needed to implement the state feedback, the observer, and the model and feedforward generator. Write the code so that the input-output latency is minimized, i.e. split up the code in two parts: CalculateOutput and UpdateState, where the amount of code in CalculateOutput is minimized. Do all possible pre-calculations in UpdateState. Your pseudo-code may contain matrix expressions involving both scalar and vector variables, e.g., it is OK to write $u = -Lx$ as a statement. Use $\hat{x}$ to denote the estimated state vector and x_m to denote the model state vector.

(2 p)

7. A winder is a machine used in paper mills to split the *jumbo reels* of paper produced by the paper machine, e.g. 9 m wide and 4 m in diameter, to slimmer *sets* of rolls that are possible to ship to the customers, e.g. 1 m wide and 125 cm in diameter. The paper is rolled off the *jumbo reel* and is then cut on the breadth using circular spinning knives. Finally it is rolled onto *cores*.

The winder is run semi-automatically. The *jumbo reel* is loaded manually and some other tasks and inspections are performed by the operator before the start button is pressed. To be allowed to start, the *jumbo reel* must always also contain enough paper for a *set*, i.e., the diameter of the jumbo reel is at least the target diameter of the set. New *cores* must also always be in place before starting.

When a *set* is ready, i.e when the diameter is sufficiently large, the winder shall stop (by setting `run` to false) and then the *set* shall be ejected from the winder. The winder takes 50 seconds to stop and the ejection mechanism takes 30 seconds to complete.

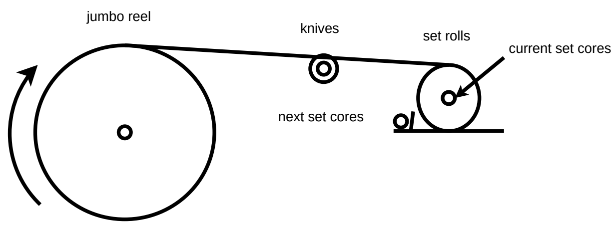
Before the ejection, new *cores* need to be loaded to be able to continue running after the ejection. This shall be done while the winder is running.

If there is enough paper left on the *jumbo reel* after the ejection, the winder shall start rolling the next *set* automatically.

Name	Type	Dir.	Description
start	Bool	In	Signal from operator to start running.
jumboReelDiameter	Real	In	The current diameter of the *jumbo reel*.
setDiameter	Real	In	The current diameter of the *set*.
targetDiameter	Real	In	Target diameter for the *set*.
run	Bool	Out	Roll the paper.
eject	Bool	Out	Ejects the *set*.
loadCore	Bool	Out	Load one *core*.
			Loading of a *core* takes 15 seconds.
coresInPlace	Bool	In	Tells if all *cores* for the next *set* are in place.

Note: The ejection mechanism and core loader both trigger on positive flanks on the signals, i.e 0 in one scan cycle followed by 1 in the following scan cycle.

Create a Grafcet application to control the winder according to the specification above. The setup is shown in Figure 1.



Figur 1 Winder process

Hint: <stepName>.s tells how many seconds a step has been active.

(3 p)