# Instructions. Development of Real-Time Systems – Assignment 3

**Overview**
In this assignment we will focus a bit more on the theoretical side. We will have a look at verifying real-time system by using the cyclic structured construct handled in the course and a simulation environment to automatically schedule a full timeline. The main purpose of the assignment is to expose the student to several ways of planning and verifying a real-time system in practice.

**Theory assignment.**

The following part of assignment is a purely theoretical task that requires no additional tools. The task is to find the largest possible frame size for the cyclic structured scheduler by following requirements 1,2 and 3 for finding the largest frame size. The following three task sets should be used:

Based on week 2 : Cyclic Structured Scheduling

Ti(Pi, ei, Di) pi = period, ei = execution time, Di = Relative deadline

1. T1(15, 1, 14)
2. T2(20, 2, 26)
3. T3(22, 3)

If there is not deadline, we consider deadline the new time the task will appears. That is to say the period.

- $f \geq \max(e1,e2,...,en)$ f → f >= 3
- ∃i: mod(pi,f)= 0 f E {22, 20, 15, 11, 10, 5, 4, 3, 2, 1} → All the divisors of the period can be a feasible candidate.
- $2*f - \gcd(pi,f) \leq Di$ for i= 1, 2, ..., n

| f | T1 | T2 | T3 |
|---|---|---|---|
| 22 | 2 * 22 – gcd(15, 22) = 44 – 1 = 43 ≤ 14 No | | |
| 20 | 2 * 20 – gcd(15, 20) = 40 – 5 = 35 ≤ 14 No | | |
| 15 | 2 * 15 – gcd(15, 15) = 40 – 15 = 25 ≤ 14 No | | |
| 10 | 2 * 10 – gcd(15, 10) = 20 – 5 = 15 ≤ 14 No | | |
| 5 | 2 * 5 – gcd(15, 5) = 10 – 5 = 5 ≤ 14 Yes | 2 * 5 – gcd(20, 5) = 10 – 5 = 5 ≤ 26 Yes | 2 * 5 – gcd(22, 5) = 10 – 1 ≤ 22 Yes |

**The optimal framesize is 5.**

Ti(Pi, ei, Di) pi = period, ei = execution time,  Di = Relative deadline

1. T1(4, 1)
2. T2(5, 2, 7)
3. T3(20, 5)
   - $f \geq \max(e_1,e_2,...,e_n)$   f → f >= 5
   - $\exists i$: $\mathrm{mod}(p_i,f)= 0$ f E {20, 10, 5, 4, 2, 1} → All the divisors of the period can be a feasible candidate.
   - $2*f - \gcd(p_i,f) \leq D_i$ for i= 1, 2, ..., n

| f | T1 | T2 | T3 |
|---|---|---|---|
| 20 | 2 * 20 – gcd(4, 22) = 40 – 1 = 39 ≤ 4 No | | |
| 10 | 2 * 10 – gcd(4, 10) = 20 – 2 = 18 ≤ 4 No | | |
| 5 | 2 * 5 – gcd(4, 5) = 10 – 1 = 9 ≤ 4 No | | |

**It is not feasible to find a period that fulfil the requirements. Job T3 should be splited in smaller parts.**

Ti(Pi, ei, Di) pi = period, ei = execution time,  Di = Relative deadline

1. T1(5, 0.1)
2. T2(7, 1)
3. T3(12, 6)
4. T4(45, 9)
   - $f \geq \max(e_1,e_2,...,e_n)$   f → f >= 9
   - $\exists i$: $\mathrm{mod}(p_i,f)= 0$ f E {45, 15, 12, 9, 7, 6, 5, 3, 2, 1} → All the divisors of the period can be a feasible candidate.
   - $2*f - \gcd(p_i,f) \leq D_i$ for i= 1, 2, ..., n

| f | T1 | T2 | T3 | T4 |
|---|---|---|---|---|
| 45 | 90 - 5 ≤ 5 No | | | |
| 15 | 30 - 5 ≤ 5 No | | | |
| 12 | 24 - 1 ≤ 5 No | | | |
| 9 | 18 - 1 ≤ 5 No | | | |
| 7 | 14 - 1 ≤ 5 No | | | |
| 6 | 12 - 1 ≤ 5 No | | | |
| 5 | 10 - 5 ≤ 5 Yes | 10 – 1 ≤ 7 No | | |
| 4 | 8 – 1 ≤ 5 No | | | |
| 3 | 6 - 1 ≤ 5 Yes | 6 – 1 ≤ 7 Yes | 6 – 3 ≤ 12 Yes | 6 – 3 ≤ 45 Yes |

**It is not feasible to find a period that fulfil the requirements. Job T3 and T4 should be splited in smaller parts.**

**Simulation assignment**

The assignment is to use a real-time simulator to verify feasibility of a set of tasks. Input the tasks: Ti(Pi, ei, Di) pi = period, ei = execution time, Di = Relative deadline

1. T1(2, 0.5)
2. T2(3, 1.2),
3. T3(6, 0.5)

and the RM scheduler into the SimSo simulator. Use SimSo to schedule the task set

Provide a report answering the following questions:

- What is the utilization factor of the system and what is the value for Urm(3).

$$U=\sum_{i=1}^{n}\frac{e_i}{P_i}=\frac{0,5}{2}+\frac{1,2}{3}+\frac{6}{0,5}=0,7333 \qquad U_{rm}(n)=n(2^{1/n}-1) \qquad U_{rm}(3)=3(2^{1/3}-1)=0,7798$$

Simulator shows: 0,7410

- What is the minimum/maximum/average response time of all tasks?

Response time:

| Task | min | avg | max | std dev |
|---|---|---|---|---|
| TASK T1 | 0.600 | 0.600 | 0.600 | 0.000 |
| TASK T2 | 2.800 | 6.511 | 8.400 | 1.924 |
| TASK T3 | 0.100 | 0.100 | 0.100 | 0.000 |
| TASK T4 | 17.300 | 17.300 | 17.300 | 0.000 |

- Is any task missing the deadline? Which task? Where?
  There is no task that miss the deadline

Input the tasks:

Ti(Pi, ei, Di) pi = period, ei = execution time, Di = Relative deadline

1. T1(2, 0.5, 1.9)
2. T2(5, 2)
3. T3(1, 0.1, 0.5)
4. T4(10, 5, 20)

   and the EDF scheduler into the SimSo simulator. Use SimSo to schedule the task set.

Provide a report answering the following questions:

- What is the utilization factor of the system and what is the value for Urm(4).

The simulator says that the total utilization is 1.

| | Total load | Payload | System load |
|---|---|---|---|
| CPU 1 | 1.0000 | 1.0000 | 0.0000 |
| Average | 1.0000 | 1.0000 | 0.0000 |

$$U=\sum_{i=1}^{n}\frac{e_i}{P_i}=\frac{0,5}{2}+\frac{2}{5}+\frac{0,1}{1}+\frac{5}{10}=1,25 \qquad U_{rm}(n)=n(2^{1/n}-1) \qquad U_{rm}(4)=4(2^{1/4}-1)=0,7568$$

- What is the minimum/maximum/average response time of all tasks?

Response time:

| Task | min | avg | max | std dev |
|---|---|---|---|---|
| TASK T1 | 0.600 | 0.600 | 0.600 | 0.000 |
| TASK T2 | 2.800 | 6.511 | 8.400 | 1.924 |
| TASK T3 | 0.100 | 0.100 | 0.100 | 0.000 |
| TASK T4 | 17.300 | 17.300 | 17.300 | 0.000 |

- Is any task missing the deadline? Which task? Where?

| | | |
|---|---|---|
| 27100000 | 27.1 | TASK T3_28 Terminated. |
| 27100000 | 27.1 | TASK T4_2 Executing on CPU 1 |
| 28000000 | 28.0 | TASK T1_15 Activated. |
| 28000000 | 28.0 | TASK T3_29 Activated. |
| 28000000 | 28.0 | TASK T4_2 Preempted! ret: 1800000 |
| 28000000 | 28.0 | TASK T3_29 Executing on CPU 1 |
| 28100000 | 28.1 | TASK T3_29 Terminated. |
| 28100000 | 28.1 | TASK T1_15 Executing on CPU 1 |
| 28600000 | 28.6 | TASK T1_15 Terminated. |
| 28600000 | 28.6 | TASK T4_2 Executing on CPU 1 |
| 29000000 | 29.0 | TASK T3_30 Activated. |
| 29000000 | 29.0 | TASK T4_2 Preempted! ret: 1400000 |
| 29000000 | 29.0 | TASK T3_30 Executing on CPU 1 |
| 29100000 | 29.1 | TASK T3_30 Terminated. |
| 29100000 | 29.1 | TASK T4_2 Executing on CPU 1 |
| 30000000 | 30.0 | Job TASK T4_2 aborted! ret:0.5 |
| 30000000 | 30.0 | TASK T4_4 Activated. |
| 30000000 | 30.0 | TASK T2_7 Activated. |

Task 4 fails at time 30 it exceeds the deadline.

- If a deadline is missed, could it be avoided by changing the scheduler?

  The main problem here is the utilization is too high to allow the system work properly. Adding a new processor would fix the problem.

ANNEX

References: http://nptel.ac.in/courses/Webcourse-contents/IIT%20Kharagpur/Embedded%20systems/Pdf/Lesson-29.pdf

References : http://www.engr.iupui.edu/~dskim/Classes/ESW5004/RTSys%20Lecture%20Note%20-%20ch04%20Clock-Driven%20Scheduling.pdf

Minimum Context Switching:

- This constraint is imposed to minimize the number of context switches occurring during task execution. The simplest interpretation of this constraint is that a task instance must complete running within its assigned frame. Unless a task completes within its allocated frame, the task might have to be suspended and restarted in a later frame. This would require a context switch involving some processing overhead. To avoid unnecessary context switches, the selected frame size should be larger than the execution time of each task, so that when a task starts at a frame boundary it should be able to complete within the same frame. Formally, we can state this constraint as:
  - max({e}) < F where ei is the execution times of the of task Ti , and Fis the frame size. Note that this constraint imposes a lower-bound on frame size, i.e., frame size F must not be smaller than max({ei})
  - To avoid preemption, want jobs to start and complete execution within a single frame: $f \geq \max(e_1,e_2,...,e_n)$

- To minimize the number of entries in the cyclic schedule, the hyper-period should be an integer multiple of the frame size (⇒f divides evenly into the period of at least one task): $\exists i: \mathrm{mod}(p_i,f)= 0$

- To allow scheduler to check that jobs complete by their deadline, should be at least one frame boundary between release time of a job and its deadline:
  $2*f – \gcd(p_i,f) \leq D_i$ for i= 1, 2, ..., n

References: http://britton.disted.camosun.bc.ca/GCD_LCM.htm

# Greatest Common Divisor

The *greatest common divisor* (GCD), which is also known as the *greatest common factor* (GCF), of two or more integers is the largest integer that is a common divisor (factor) of all the given integers.

The greatest common divisor of two or more integers can be obtained in three steps:

**STEP 1: Find the prime factorization of each integer.**

$375 = 3 \times 5^3$
$525 = 3 \times 5^2 \times 7$

**STEP 2: List the common prime divisors (factors) with the least power of all the given integers.**

$375 = 3 \times 5^3 = 3 \times 5^2 \times 5$
$525 = 3 \times 5^2 \times 7 = 3 \times 5^2 \times 7$

Common Prime Divisors (Factors) with Least Power: $3$ and $5^2$

**STEP 3: Multiply the common prime divisors (factors) to find the greatest common divisor (factor).**

$3 \times 5^2 = 75$

GCD (GCF) of 375 and 525 = 75

# Least Common Multiple

The *least common multiple* (LCM), which is also known as the *least common denominator* (LCD) in fractions, of two or more integers is the smallest integer that is a common multiple (denominator) of all the given integers.

The least common multiple (denominator) of two or more integers can be obtained in three steps:

**STEP 1: Find the prime factorization of each integer.**

$4 = 2^2$
$10 = 2 \times 5$
$45 = 3^2 \times 5$

**STEP 2: List the prime divisors (factors) with the greatest power of all the given integers.**

$4 = 2^2$
$10 = 2 \times 5$
$45 = 3^2 \times 5$

Prime Divisors (Factors) with Greatest Power: $2^2$, $3^2$, and $5$

**STEP 3: Multiply the prime divisors (factors) to find the least common multiple (denominator).**

$2^2 \times 3^2 \times 5 = 180$

LCM of 4, 10 and 45 = 180