Real-time Systems

Homework 1,
(Total of 100 points)

Note: On all homeworks, please interpret "periodic" and "sporadic" to have their "real-world" meanings unless specified otherwise.

1. (5 points) Because hard aperiodic jobs (what Liu calls "sporadic") may have varying release times and execution times, the sporadic task model may be too inaccurate and can lead to undue under-utilization of the processor even when the inter-release times of jobs are bounded from below and their executions are bounded from above. As an example, suppose that we have a stream of hard aperiodic jobs whose inter-release times are uniformly distributed from 9 to 11. Their execution times are uniformly distributed from 1 to 3.

   (a) What are the parameters of the sporadic task if we were to use such a task to model the stream?

   (b) Compare the utilization of the sporadic task in part (a) with the average utilization of the aperiodic job stream.

2. (60 points) Consider the synchronous task set shown in Table 1 (on the next page). This task set comes from an embedded signal processing application for an anti-submarine warfare (ASW) system. More specifically, the task set is a subset of the tasks that implement the Directed Low Frequency Analysis and Recording (DIFAR) acoustic signal processing application from the Airborne Low Frequency Sonar (ALFS) system of the U.S. Navy's SH-60B LAMPS MK III anti-submarine helicopter. The ALFS system processes low frequency signals received by sonobuoys in the water. Its primary function is to detect and track submarines and to calculate range and bearing estimates to each target.

   The task set shown in Table 1 represents an implementation of a portion of the DIFAR application on a Mercury PowerPC 6U VME board with a 200MHz 603e processor. This portion of the DIFAR application processes five bands of Constant Resolution (CR) data from each of five sonobuoys. In this assignment, we will assume all tasks are independent.

   a. (5 pts) For all $1 \leq i \leq 39$, what is the utilization of task T_i?

   b. (5 pts) What is the system utilization?

   c. (5 pts) What is the hyperperiod of the task set?

   d. (10 pts) **(a)** What are the possible frame sizes that could be used to create a cyclic schedule for this task set if all three frame size constraints are enforced? **(b)** What if we drop the first constraint, as done when the Iterative Network Flow algorithm is applied?

   e. (30 pts) Create a cyclic schedule for the task set. You may want to create a program to do this using the Iterative Network Flow algorithm. (Feel free to download max-flow code

from the web; http://www.nist.gov/dads/terms.html looks promising.) The file **HW1table1.txt** on the homeworks webpage contains the task parameters in 4-tuple form. In your solution, you should list each frame (by number) and indicate which jobs are scheduled in each frame and how much time each job executes within a frame. If a job is sliced across multiple frames, be sure to indicate this. If you want, you can group tasks with similar parameters that are always scheduled in a similar way and use a short-hand notation to show how they are scheduled.

f. (5 pts) What is the frame size *f* of your schedule?

g. (5 pts) How many minor cycles are there in your major cycle?

Task ID	Phase in ms	Period in ms	Time/Exec in ms	Relative Deadline	Processing Primitive
1	0	250	6.4545	250	FLW
2	0	250	30.1303	250	BDFC
3	0	250	0.3437	250	MASTERMCS
4	0	250	0.1022	250	SLAVEMCS
5	0	250	5.7349	250	DIFARDAD
6	0	250	5.7557	250	DIFARDAD
7	0	250	5.7974	250	DIFARDAD
8	0	250	5.8807	250	DIFARDAD
9	0	250	6.0472	250	DIFARDAD
10	0	250	4.3071	250	DIFARDAD
11	0	250	7.7672	250	DIFARDAD
12	0	250	14.6875	250	DIFARDAD
13	0	250	7.183	250	CRFIL
14	0	250	7.3999	250	CRFIL
15	0	250	7.8337	250	CRFIL
16	0	250	8.7012	250	CRFIL
17	0	250	8.7012	250	CRFIL
18	0	250	8.1264	250	CRSPECANAL
19	0	250	8.1264	250	CRSPECANAL
20	0	250	8.4815	250	CRSPECANAL
21	0	250	9.1918	250	CRSPECANAL
22	0	250	9.1918	250	CRSPECANAL
23	0	250	3.217	250	ALLBANDMERGE
24	0	250	3.5179	250	SAD
25	0	250	3.6363	250	GRM
26	0	250	5.1914	250	BBC
27	0	250	0.1496	250	GRAMMERGE
28	0	500	3.3671	500	CRDETECT
29	0	500	3.3671	500	CRDETECT
30	0	500	3.3671	500	CRDETECT
31	0	500	3.3671	500	CRDETECT
32	0	500	3.3671	500	CRDETECT
33	0	2000	3.1913	2000	ALI
34	0	2000	5.1122	2000	BRG
35	0	2000	0.5047	2000	ALIMERGE
36	0	2000	0.5906	2000	BEARMERGE
37	0	6000	2.4799	6000	AUTODETECT
38	0	6000	0.199	6000	BINMERGE
39	0	6000	0.6898	6000	AUTODETECTMERGE

Table 1: Synchronous signal processing task set for a 200MHz 603e PowerPC.

h. (30 points) Create a static schedule for the asynchronous task set shown in Table 2. (Only the task phases and ordering are different from the task set in Table 1). The file **HW1table2.txt** on the

homeworks webpage contains the task parameters in 4-tuple form. **Note:** You will have to do some thinking here to determine how to "massage" the approach covered in class to apply to asynchronous systems. There's probably more than one way to do it. Include in your answer an explanation of the strategy you are using.

Task ID	Phase in ms	Period in ms	Time/Exec in ms	Relative Deadline	Processing Primitive
1	0	250	6.4545	250	FLW
2	0.001	250	30.1303	250	BDFC
3	0.002	250	0.3437	250	MASTERMCS
4	0.003	250	0.1022	250	SLAVEMCS
5	0.004	250	5.7349	250	DIFARDAD
6	0.004	250	5.7557	250	DIFARDAD
7	0.004	250	5.7974	250	DIFARDAD
8	0.004	250	5.8807	250	DIFARDAD
9	0.004	250	6.0472	250	DIFARDAD
10	0.004	250	4.3071	250	DIFARDAD
11	0.004	250	7.7672	250	DIFARDAD
12	0.004	250	14.6875	250	DIFARDAD
13	0.005	250	7.183	250	CRFIL
14	0.005	250	7.3999	250	CRFIL
15	0.005	250	7.8337	250	CRFIL
16	0.005	250	8.7012	250	CRFIL
17	0.005	250	8.7012	250	CRFIL
18	0.006	250	8.1264	250	CRSPECANAL
19	0.006	250	8.1264	250	CRSPECANAL
20	0.006	250	8.4815	250	CRSPECANAL
21	0.006	250	9.1918	250	CRSPECANAL
22	0.006	250	9.1918	250	CRSPECANAL
23	250.007	500	3.3671	500	CRDETECT
24	250.007	500	3.3671	500	CRDETECT
25	250.007	500	3.3671	500	CRDETECT
26	250.007	500	3.3671	500	CRDETECT
27	250.007	500	3.3671	500	CRDETECT
28	250.008	250	3.217	250	ALLBANDMERGE
29	250.009	250	3.5179	250	SAD
30	250.010	250	3.6363	250	GRM
31	250.010	250	5.1914	250	BBC
32	250.011	250	0.1496	250	GRAMMERGE
33	1750.011	2000	3.1913	2000	ALI
34	1750.011	2000	5.1122	2000	BRG
35	1750.012	2000	0.5047	2000	ALIMERGE
36	1750.012	2000	0.5906	2000	BEARMERGE
37	5750.012	6000	2.4799	6000	AUTODETECT
38	5750.013	6000	0.199	6000	BINMERGE
39	5750.013	6000	0.6898	6000	AUTODETECTMERGE

Table 2: Asynchronous signal processing task set for a 200MHz 603e PowerPC.