

Q6-2: MP3 播放器, 带触摸屏与显示输出的系统

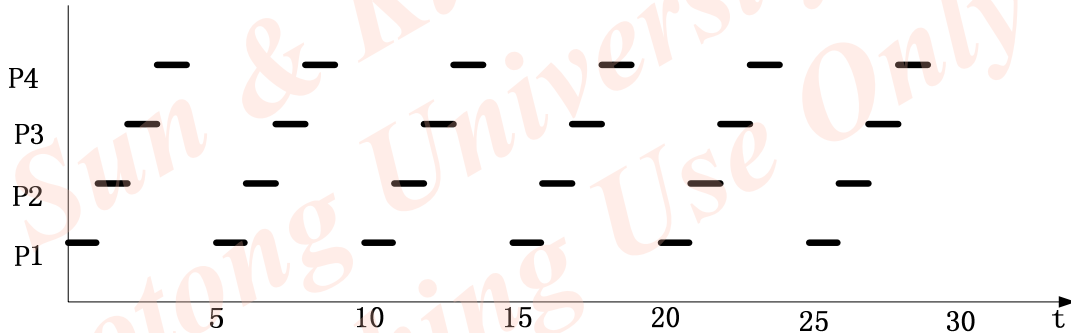
Q6-3 前面板的采样频率可以选择:

44.1K/100=441Hz 以简化系统调度

Q6-7: CPU 利用率= $(2\text{ms}/5\text{ms}) + (3\text{ms}/10\text{ms}) + (3\text{ms}/10\text{ms}) + (6\text{ms}/15\text{ms}) = 0.4+0.3+0.3+0.4=1.4$ 这组任务是不可调度的。

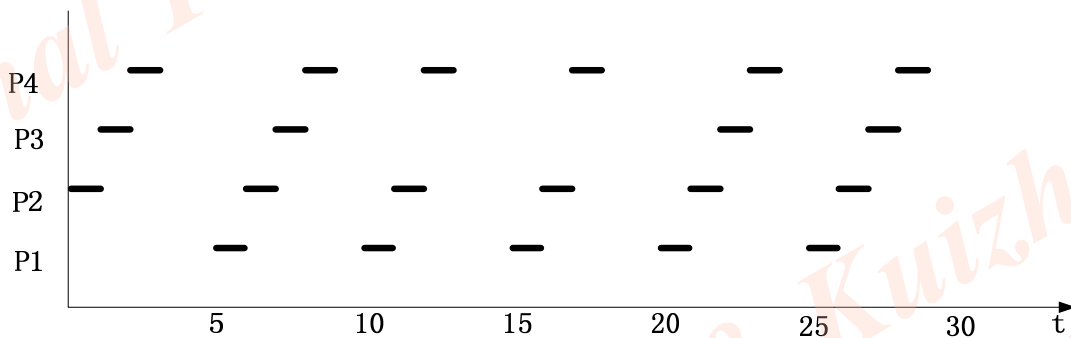
Q6-8: 超周期为 30ms

假设以 5ms 为调度单位, 则在 5ms 内 P1 P2 P3 P4 各执行 1ms



CPU 利用率= $1/5+2/10+2/10+3/15=0.2+0.2+0.2+0.2=0.8$

Q6-9: 轮询调度



Q6-10:

就绪态表示该进程已经可以执行, 等待 CPU 调度

等待态表示该进程的执行条件还不满足, 还在等待 I/O 或其他进程的数据, 或触发其运行的定时器尚未期满等

Q6-12

信号量 (semaphore) 的数据结构为一个值和一个指针, 指针指向等待该信号量的下一个进程。信号量的值仅能由 PV 操作来改变, 信号量 $S \geq 0$ 时, S 表示可用资源的数量。执行一次 P 操作意味着请求分配一个单位资源, 因此 S 的值减 1; 当 $S < 0$ 时, 表示已经没有可用资源, 请求者必须等待别的进程释放该类资源, 它才能运行下去。而执行一个 V 操作意味着释放一个单位资源, 因此 S 的值加 1; 若 $S < 0$, 表示有某些进程正在等待该资源, 因此要唤醒一个等待状态的进程, 使之运行下去。

在本题中, 示意的信号量结构为 1 个值, 没有考虑进程指针, 重点是 swap 的应用。

Semaphore 为信号量结构体定义, value 为 S 中的信号量的值且为结构体的第一个成员。

```
procedure p(var *s:semaphore);
```

```
{int *temp;
```

```
*temp= (*s).value-1;
```

```

swap (temp, s) ;
}
procedure v(var *s:samephore);
{ int *temp;
*temp= (*s) .value+1;
swap (temp, s) ;
}

```

Q6-14

- a: 最小时间间隔为进程超周期: 15
- b: 30
- c: 60

Q6-15: 进程超周期为 200

增加一个 P1 示例后的 CPU 利用率= $4/200+4/200+1/10+2/40+6/50=0.31$

远没有达到 RMS 的 CPU 利用率的最小上界 0.69, 可以增加 P1 的另一个示例。

Q6-16:

进程超周期为 100,

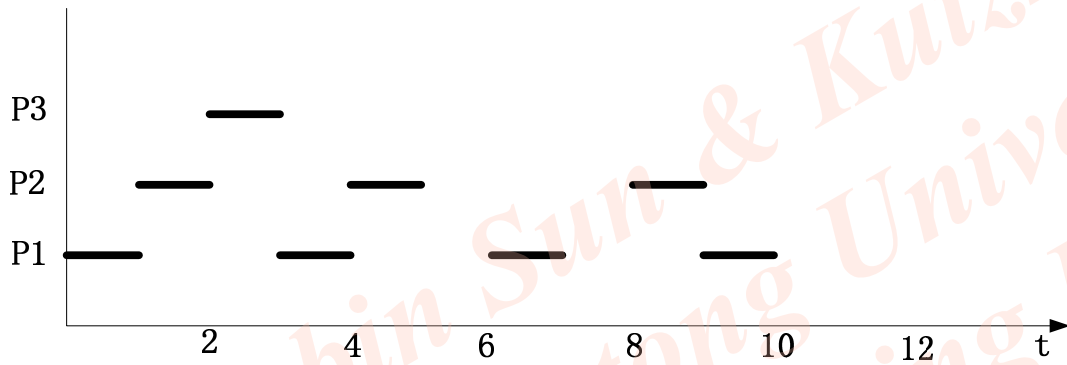
则 CPU 利用率= $0.1+0.18+0.1+0.1+x/25$,如按 5 个任务的 RMS 的利用率的最小上界~74.3% 计算

$x < (0.69 - 0.48) * 25 = 6.5$, P5 的最大执行时间可选为 6。

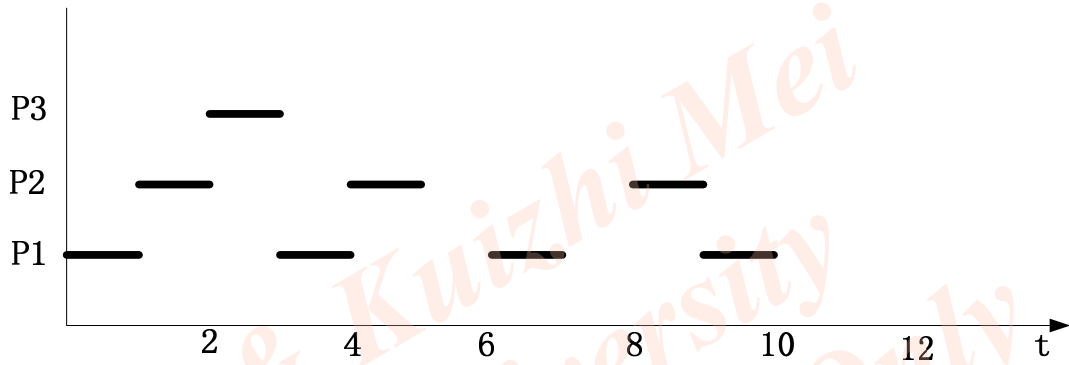
Q6-19

应用 RMS 调度的时序图

根据 RMS 的优先级定义, 优先级 $P1 > P2 > P3$, 超周期为 12



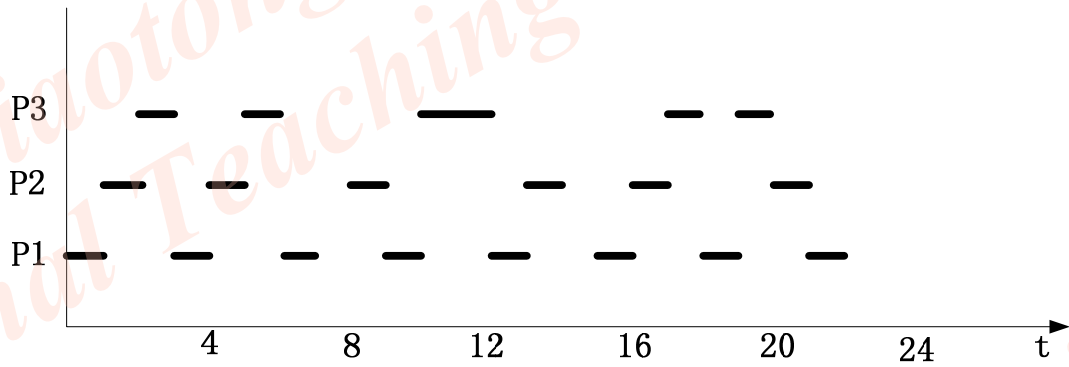
EDF 调度(该题的 EDF 和 RMS 的调度一样)



Q6-20

进程超周期为 24

采用 RMS 调度，优先级 $P1 > P2 > P3$ ，具体调度如下图



如果采用 EDF 调度，如进程距离截止时限一样近，则优先级按 $P1 > P2 > P3$ 顺序，具体调度如下表所示：

时间	运行中的进程	截止时限
0	P1	
1	P2	
2	P3	P1
3	P1	P2
4	P2	
5	P3	P1
6	P1	
7	idle	P2, P3
8	P2	P1
9	P1	
10	P3	
11	P3	P1, P2
12	P1	
13	P2	
14	idle	P1
15	P1	P2, P3

16	P2	
17	P3	P1
18	P1	
19	P3	P2
20	P2	P1
21	P1	
22	idle	
23	idle	P1,P2,P3

Q6-22

使用 RMS 调度，优先级顺序为 P5 P1 P2 P4 P3，进程超周期为 120

时间	运行中的进程
0	P5
1	P1
2	P1
3	P2
4	P2
5	P2
6	P2
7	P4
8	P4
9	P4
10	P4
11	P4
12	P3
13	P3
14	P3
15	P5
16	P3
17	P3
18	P3
19	P3
30	P5
31	P1
32	P1
40	P2
41	P2
42	P2
43	P2
45	P5
60	P5
61	P1

62	P1
63	P4
64	P4
65	P4
66	P4
67	P4
75	P5
80	P2
81	P2
82	P2
83	P2
90	P5
91	P1
92	P1
105	P5

若增加上下文切换时间为 1，则

时间	运行中的进程
0	P5
1	P5
2	P1
3	P1
4	P1
5	P2
6	P2
7	P2
8	P2
9	P2
10	P4
11	P4
12	P4
13	P4
14	P4
15	P5
16	P5
17	P4
18	P4
19	P3
20	P3
21	P3
22	P3
23	P3
24	P3
25	P3

26	P3
30	P5
31	P5
32	P1
33	P1
34	P1
40	P2
41	P2
42	P2
43	P2
44	P2
45	P5
46	P5
60	P5
61	P5
62	P1
63	P1
64	P1
65	P4
66	P4
67	P4
68	P4
69	P4
70	P4
75	P5
76	P5
81	P2
82	P2
83	P2
84	P2
85	P2
90	P5
91	P5
92	P1
93	P1
94	P1
105	P5
106	P5

Q6-23

使用 RMS 调度，超周期为 100。

时间	运行中的进程
0	P1
1	P2

2	P3
3	P3
4	P4
5	P1
6	P4
7	P4
8	P4
9	P4
10	P1
11	P2
12	P4
13	P4
14	P4
15	P1
16	P4
17	P5
18	P5
19	P5
20	P1
21	P2
22	P3
23	P3
24	P5
25	P1
26	P5
27	P5
28	P5
30	P1
31	P2
35	P1
40	P1
41	P2
42	P3
43	P3
45	P1
50	P1
51	P2
52	P4
53	P4
54	P4
55	P1
56	P4
57	P4
58	P4

59	P4
60	P1
61	P2
62	P3
63	P3
64	P4
65	P1
66	P4
70	P1
71	P2
75	P1
80	P1
81	P2
82	P3
83	P3
85	P1
90	P1
91	P2
95	P1
100	P1

EDF 调度:

时间	运行的进程	截止时限	时间	运行的进程	截止时限
0	P1		50	P1	
1	P2		51	P2	
2	P3		52	P4	
3	P3		53	P4	
4	P4	P1	54	P4	P1
5	P1		55	P1	
6	P4		56	P4	
7	P4		57	P4	
8	P4		58	P4	
9	P4	P1 P2	59	P4	P1 P2 P3
10	P1		60	P1	
11	P2		61	P2	
12	P4		62	P3	
13	P4		63	P3	
14	P4	P1	64	P4	P1
15	P1		65	P1	
16	P4		66	P4	
17	P5		67	idle	
18	P5		68	idle	
19	P5	P1 P2 P3	69	idle	P1 P2
20	P1		70	P1	

21	P2					71	P2				
22	P3					72	idle				
23	P3					73	idle				
24	P5		P1			74	idle			P1	
25	P1					75	P1				
26	P5					76	idle				
27	P5					77	idle				
28	P5					78	idle				
29	idle		P1 P2			79	idle			P1 P2 P3	
30	P1					80	P1				
31	P2					81	P2				
32	idle					82	P3				
33	idle					83	P3				
34	idle		P1			84	idle			P1	
35	P1					85	P1				
36	idle					86	idle				
37	idle					87	idle				
38	idle					88	idle				
39	idle		P1 P2 P3			89	idle			P1 P2	
40	P1					90	P1				
41	P2					91	P2				
42	P3					92	idle				
43	P3					93	idle				
44	idle		P1			94	idle			P1	
45	P1					95	P1				
46	idle					96	idle				
47	idle					97	idle				
48	idle					98	idle				
49	idle		P1 P2 P3 P4			99	idle			P1 P2 P3 P4 P5	