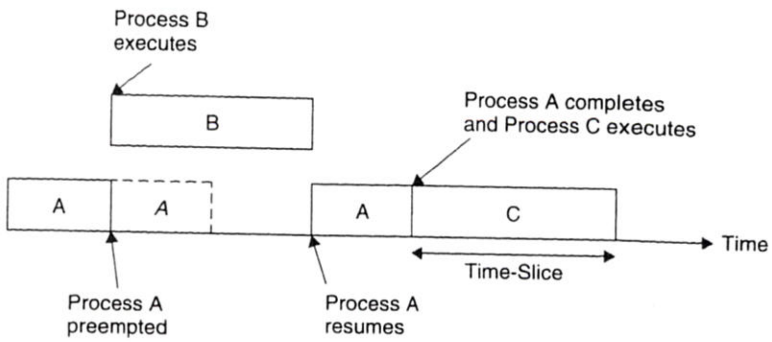


Figure 3.6 Mixed scheduling of three tasks.

Round-robin systems can be combined with preemptive priority systems, yielding a kind of mixed system. Figure 3.6 illustrates the process. Here processes A and C are of the same priority, whereas process B is of higher priority. Process A is executing for some time when it is preempted by task B, which executes until completion. When process A resumes, it continues until its time slice expires, at which time context is switched to process C, which begins executing.

3.2.3 Cyclic Executives

The cyclic-executive (CE) approach is very popular, as it is simple and generates a complete and highly predictable schedule. The CE refers to a scheduler that deterministically interleaves and sequentializes the execution of periodic tasks on a processor according to a pre-run-time schedule. In general terms, the CE is a table of procedure calls, where each task is a procedure, within a single do loop.

In the CE approach, scheduling decisions are made periodically, rather than at arbitrary times. Time intervals during scheduling decision points are referred to as frames or minor cycles, and every frame has a length, f, called the frame size. The major cycle is the minimum time required to execute tasks allocated to the processor, ensuring that the deadlines and periods of all processes are met. The major cycle or the hyperperiod is equal to the least common multiple (lcm) of the periods, that is, $\text{lcm}(p_1, \ldots p_n)$.

As scheduling decisions are made only at the beginning of every frame, there is no preemption within each frame. The phase of each periodic task is a non-negative integer multiple of the frame size. Furthermore, it is assumed that the scheduler carries out monitoring and enforcement actions at the beginning of each frame (see Figure 3.7).

Frames must be sufficiently long so that every task can start and complete with a single frame. This implies that the frame size, f, is to be larger than the execution time, e_i, of every task, T_i, that is,

$$C_1 : f \geq \max_{1 \leq i \leq n} (e_i) \tag{3.4}$$

Realtime Sys Design & Analysis by P.A. Laplan

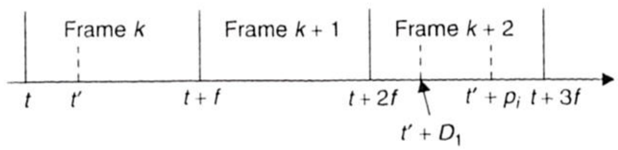


Figure 3.7 Constraints on the value of frame size.

In order to keep the length of the cyclic schedule as short as possible, the frame size, f, should be chosen so that the hyperperiod has an integer number of frames:

$$C_2 : \lfloor p_i/f \rfloor - p_i/f = 0 \tag{3.5}$$

In order to ensure that every task completes by its deadline, frames must be small so that between the release time and deadline of every task, there is at least one frame. The following relation is derived for a worst-case scenario, which occurs when the period of a process starts just after the beginning of a frame and, consequently, the process cannot be released until the next frame.

$$C_3 : 2f - gcd(p_i, f) \leq D_i \tag{3.6}$$

where gcd is the greatest common divisor and D_i is the relative deadline of task i.

To illustrate the calculation of the framesize, consider the set of tasks shown in Table 3.1. The hyperperiod is equal to 660, since the least common multiple of 15, 20, and 22 is 660. The three conditions, C_1, C_2 and C_3 are evaluated as follows:

$$C_1 : \forall i f \geq e_i \Rightarrow f \geq 3$$

$$C_2 : \lfloor p_i/f \rfloor - p_i/f = 0 \Rightarrow f = 2, 3, 4, 5, 10, \ldots$$

$$C_3 : 2f - gcd(p_i, f) \leq D_i \Rightarrow f = 2, 3, 4, 5$$

From these three conditions, it can be inferred that a possible value for f could be any one of the values of 3, 4, or 5.

Table 3.1 Example task set for framesize calculation

τ_i	p_i	e_i	D_i
τ_2	15	1	14
τ_3	20	2	26
τ_4	22	3	22

too large and exceeds a predefined threshold. If it does, a corresponding message will be sent on the bus to make the reaction.

The workflow of handling 'Obstacle Distance Message' is described by Figure 5-8.

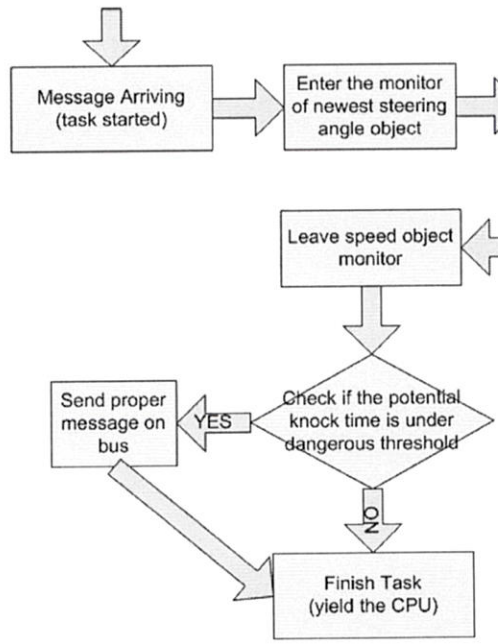


Figure 5-8 Workflow of handling Obstacle Distance Message

The handler of Obstacle Distance Message first needs to retrieve the steering angle value from the memory through the monitor of the steering angle object. And then, it goes into the speed monitor to get the newest value of speed. After enough information is gathered, the handler calculates the potential knock time of the vehicle to the obstacle according to three parameters: vehicle speed, steering angle and obstacle distance. If the time result indicates a dangerous state, the handler will send out a proper message onto the bus, for example a control message to stop the vehicle.

5.2.3 Time Constraints in the Sample Application

The following table shows the deadlines of each message-handling task:

Message Handling Type	Period (ms)	Deadline (ms)
Speed Message Handling	100	100
Tire Pressure Message Handling	500	200
Steering Wheel Angle Message Handling	200	150
Obstacle Distance Message Handling	100	70

Table 5-2 Messages handling deadlines

Rules for designing cyclic schedule

4

0. if Utilization >1, the tasks cannot be scheduled in the same processor.

If U is okay,

Hyperperiod H is lcm (pi) + these constraints

1. Frame f $\geq$ max(ei)
2. Frame f should evenly divide H.
3. There should be at least 1 frame between release time of a task and its deadline:

   $2f - gcd(pi,f) \leq Di$

Very often Di and Pi are same for periodic task. For simplicity in discussion we will assume this default setting.

Example

5

ti	ri	ei	pi	Di
t1	0	1	4	4
t2	0	1.8	5	5
t3	0	1.0	20	20
t4	0	2.0	20	20

Given the task set above design the cyclic executive schedule or cl static schedule.

(1)

CSE 321

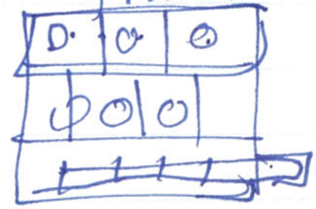
Sept 21, 2016

1. $U \le 1$ $\qquad \sum \frac{ei}{pi} \le 1$

sleep / work / misc

$\frac{8}{24} + \frac{10}{24} + \frac{6}{24} + \cancel{\frac{2}{24}}$

frames



$\frac{1}{4} + \frac{1.8}{5} + \frac{1}{20} + \frac{2}{20} \le 1$

$0.25 + 0.36 + 0.05 + 0.1 \le$

$0.76 < 1.0$ $\qquad$ 76% ✓

2. Hyper-period lcm(pi)

lcm(4, 5, 20, 20) = 20 cycles

3. Frame size : ≥ max(ei)

≥ (1, 1.8, 1, 2)

≥ 2

4 will

$2f - \gcd(pi, f) \le Di$ for all i

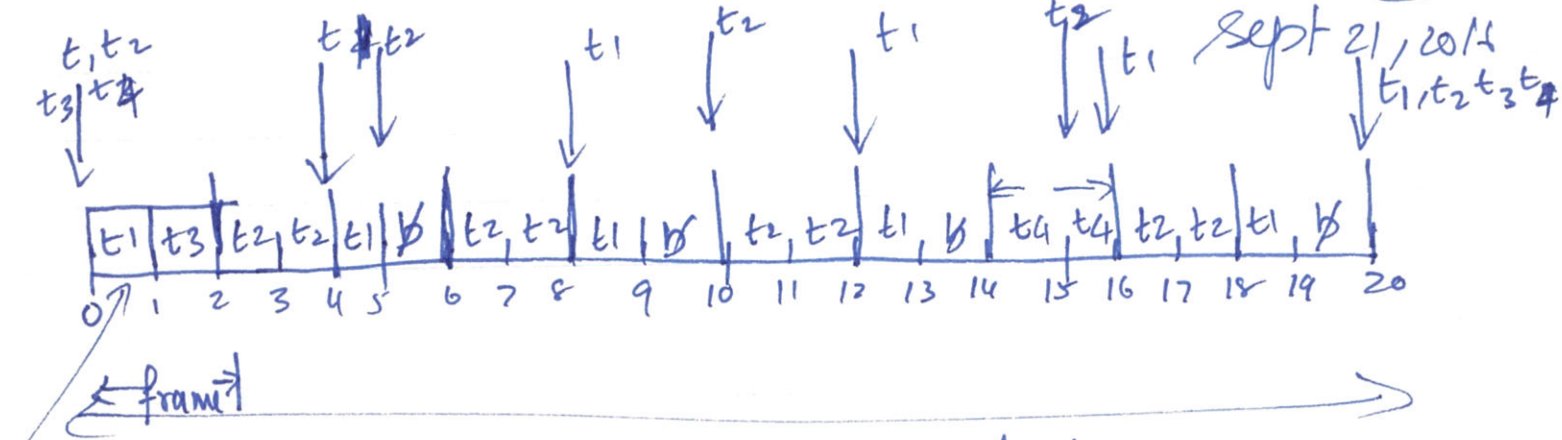
4. draw the schedule: static-schedule

clock-driven schedule

→ cyclic executive $\quad$ repeats

table or data-driven ← schedule is in a table

Sept 21, 2011



slot

	e_i
t1	1
t2	1.8
t3	[illegible]
t4	2

hyper-period
↓
frames
↓
slots

is measured in cycles

fit the tasks in the slots

multiple / correct schedules possible.

EDS

5. Write the cyclic executive.

{ {t_1, t_3}
{t_2, t_2}
{t_1, b}
{t_2, t_2}
{t_1, b}
{t_2, t_2}
{t_1, b}
{t_4, t_4}
{t_2, t_2}
{t_1, b}
}

See the code linked to lecture notes.

⟹ Implement this.

Translate this into a table and write/code the cyclic executive