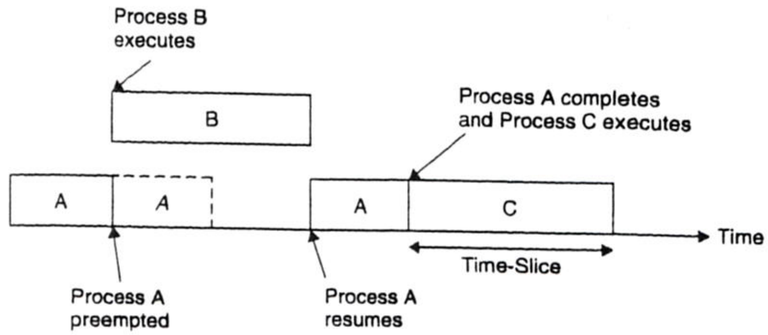

Figure 3.6 Mixed scheduling of three tasks.

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

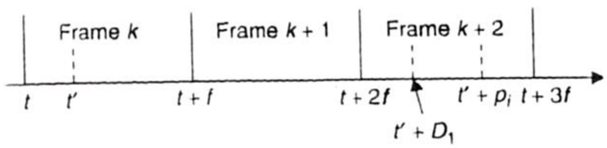

Figure 3.7 Constraints on the value of frame size.

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.

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

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:

$$C_1 : \forall i f \ge e_i \Rightarrow f \ge 3$$
$$C_2 : \lfloor p_i/f \rfloor - p_i/f = 0 \Rightarrow f = 2, 3, 4, 5, 10, \ldots$$
$$C_3 : 2f - gcd(p_i, f) \le D_i \Rightarrow f = 2, 3, 4, 5$$

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