Priority Inversion

Priority Inversion

- Priority inversion is a situation in which a low-priority task executes while a higher priority task wait on it due to resource contentions
- Task interdependency

Priority Inversion Example

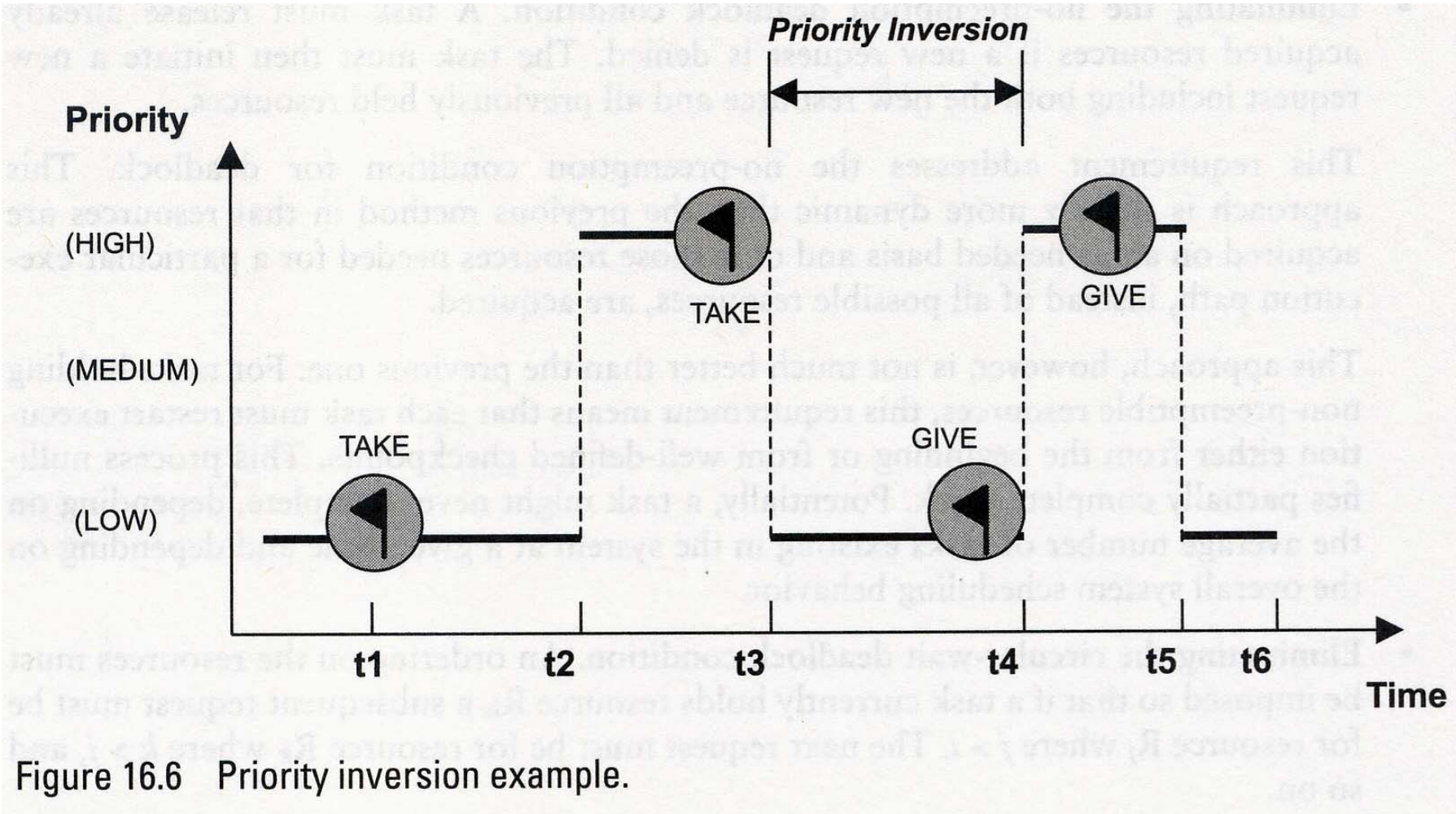

Figure 16.6 Priority inversion example.

Unbounded Priority Inversion Example

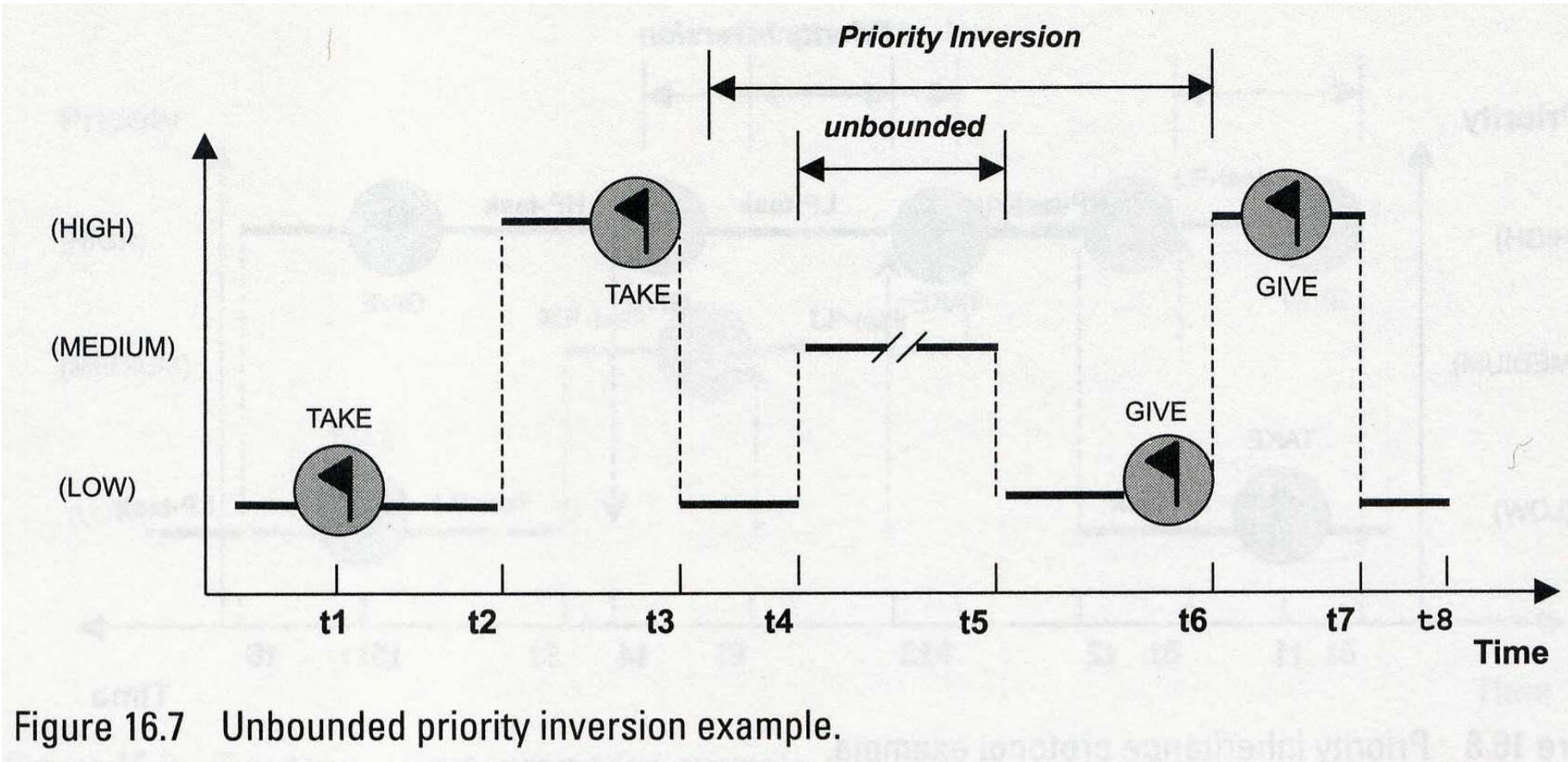

Figure 16.7 Unbounded priority inversion example.

Priority Inheritance Protocol

- R: resource, T: the Task requesting R

1. If R is in use, T is blocked
2. If R is free, R is allocated to T
3. When a task of a higher priority requests the same resource, T’s executing priority is raised to the requesting task’s priority
4. The task returns to its previous priority when it releases R

Priority Inheritance Protocol

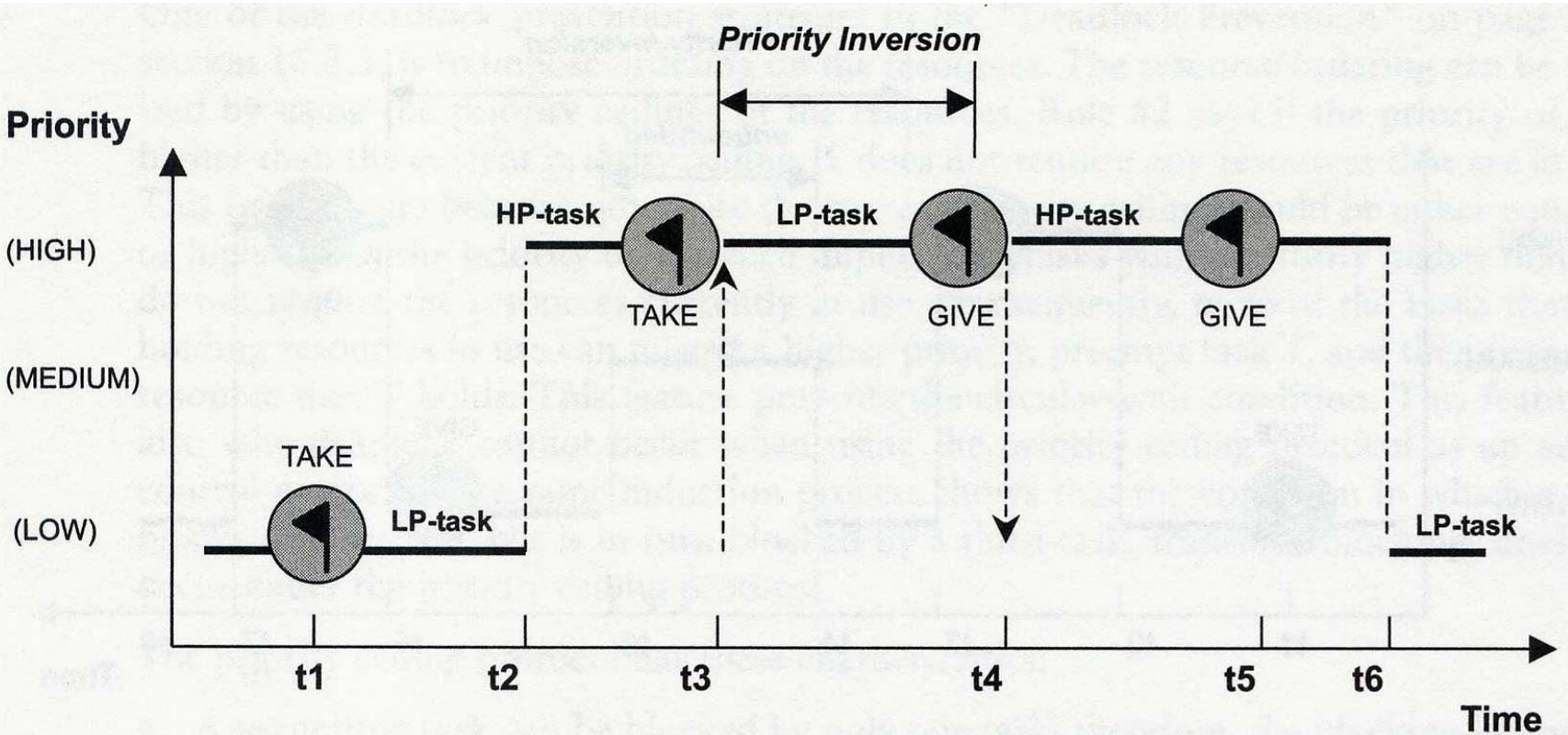

Figure 16.8 Priority inheritance protocol example.

Priority Inheritance Protocol

- Priority inheritance is dynamic

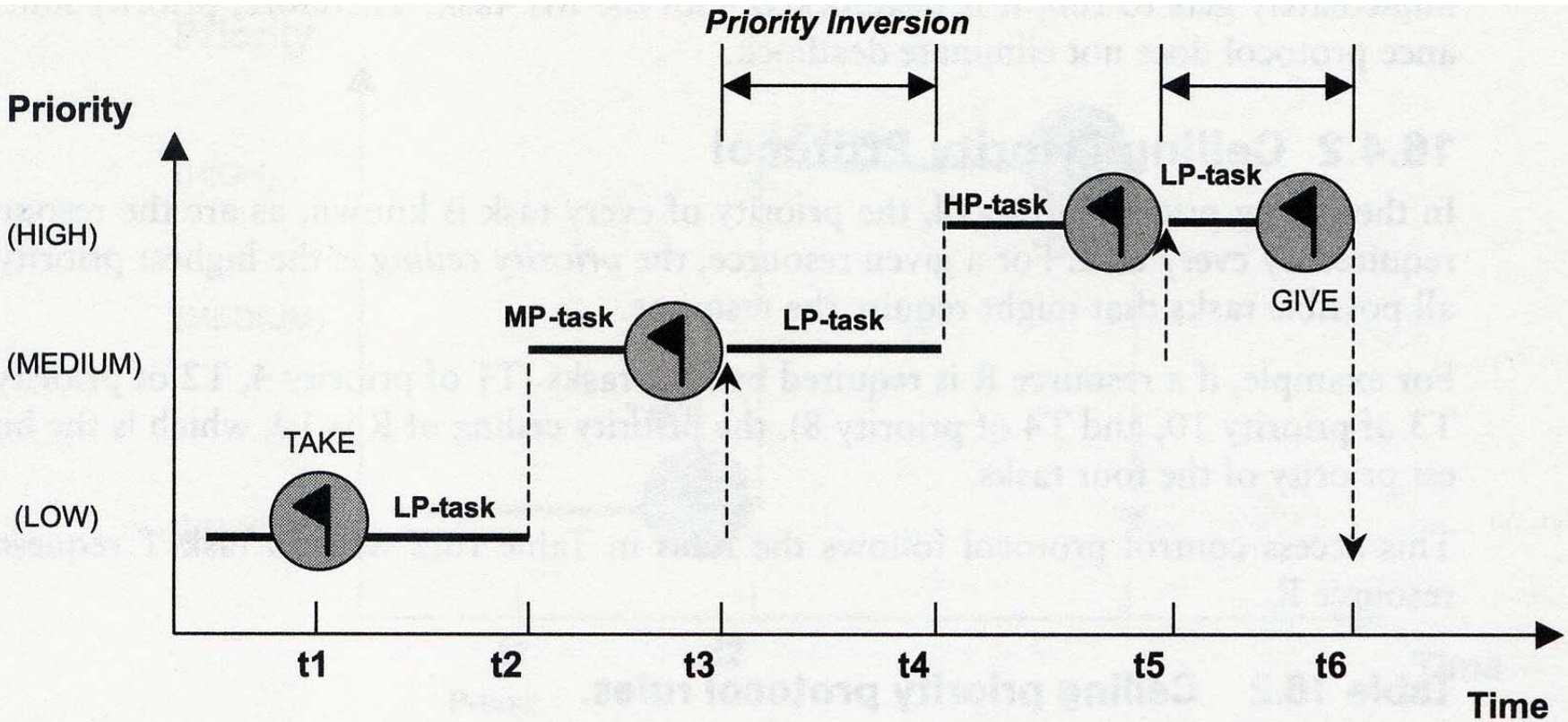

Figure 16.9 Transitive priority promotion example.

Ceiling Priority Protocol

- Priority ceiling: the highest priority of all possible tasks that might require the resource
- Example: a resource R requested by 4 tasks. T1 priority 4, T2 priority 9, T3 priority 10, T4 Priority 8 -> priority ceiling of R is 4 (highest)

Ceiling Priority Protocol Rules

- When a Task T requests a resource R

1. If R is in use, T is blocked
2. If R is free, R is allocated to T. T's executing priority is raised to the priority ceiling of R if that's higher.
3. T's priority is assigned the next highest priority ceiling of another resource when the resource with the highest priority ceiling is released
4. The task returns to its assigned priority after it has released all resources

Ceiling Priority Protocol Rules

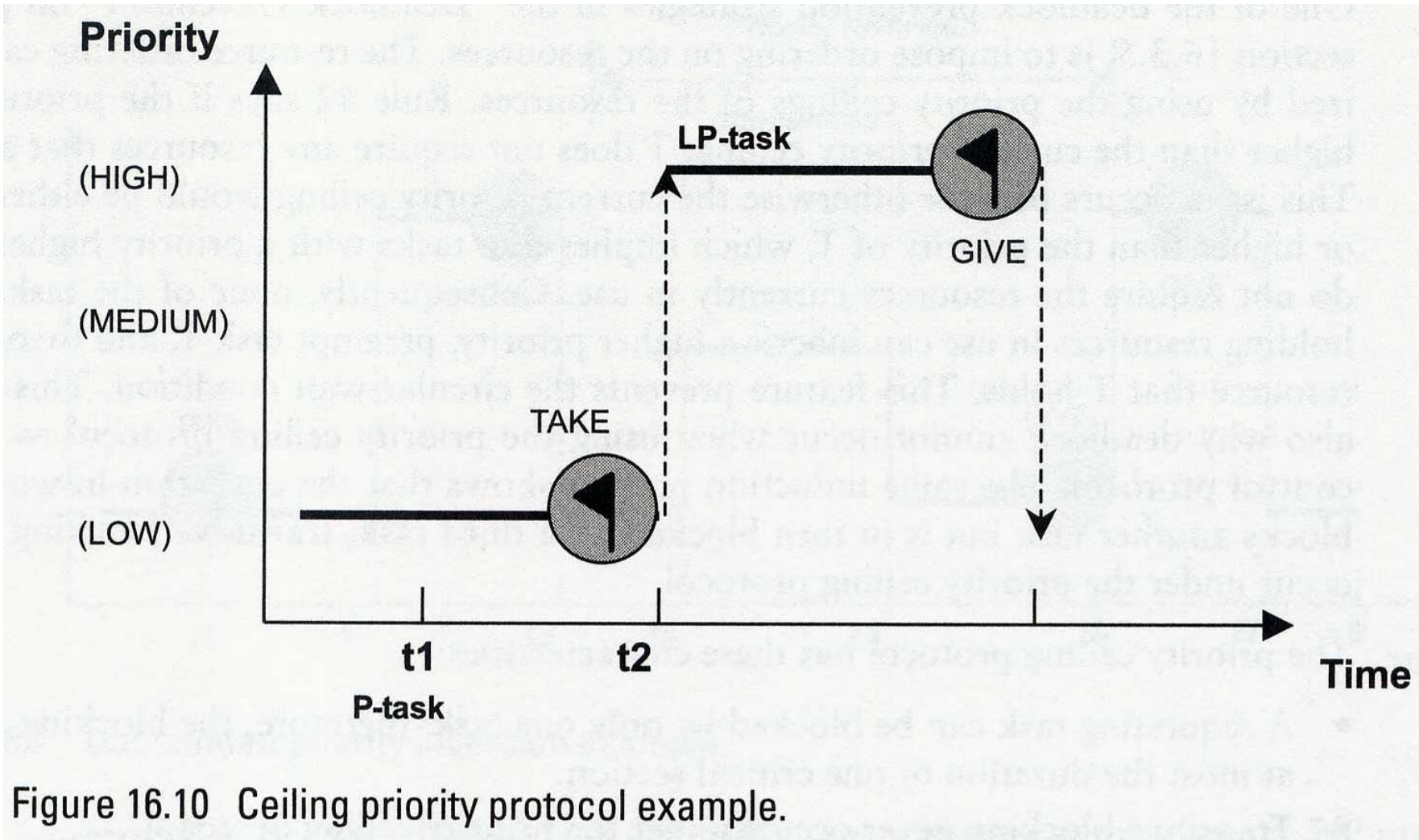

Figure 16.10 Ceiling priority protocol example.