

的地址、它的目的端口号或者其他标准。每个优先级类通常都有自己的队列。当选择一个分组传输时，优先级排队规则将从队列为非空（也就是有分组等待传输）的最高优先级类中传输一个分组。在同一优先级类的分组之间的选择通常以 FIFO 方式完成。

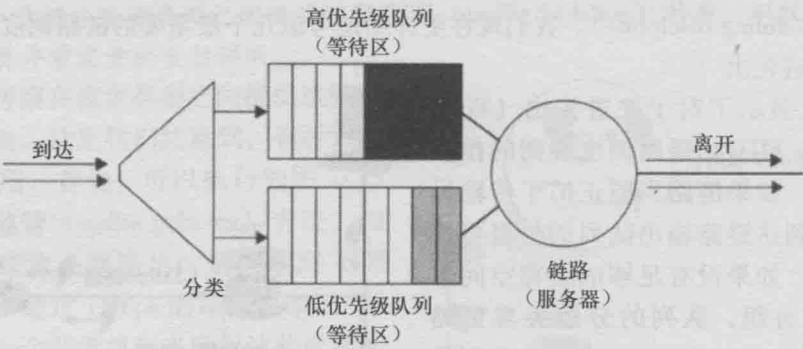


图 7-19 优先级排队模型

图 7-20 描述了有两个优先级类的一个优先级队列的操作。分组 1、3 和 4 属于高优先级类，分组 2 和 5 属于低优先级类。分组 1 到达并发现链路是空闲的，就开始传输。在分组 1 的传输过程中，分组 2 和 3 到达，并分别在低优先级和高优先级队列中排队。在传输完分组 1 后，分组 3（一个高优先级的分组）被选择在分组 2（尽管它到达得较早，但它是一个低优先级分组）之前传输。在分组 3 的传输结束后，分组 2 开始传输。分组 4（一个高优先级分组）在分组 2（一个低优先级分组）的传输过程中到达。在所谓的非抢占式优先级排队规则下，一旦分组开始传输，传输就不能打断。在这种情况下，分组 4 排队等待传输，并在分组 2 传输完成之后开始传输。

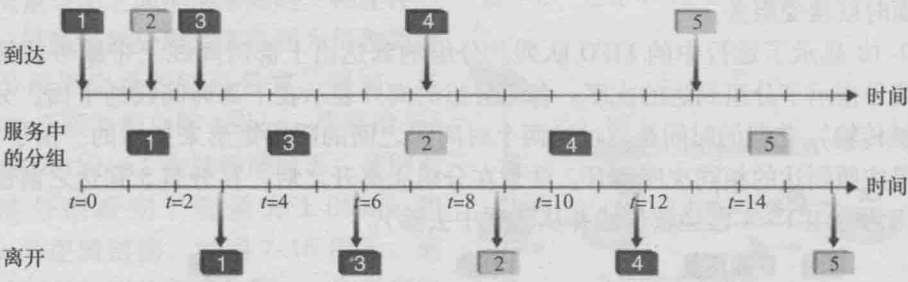


图 7-20 优先级队列的操作

(3) 循环和加权公平排队

使用循环排队规则（round robin queuing discipline），分组像使用优先级排队一样被分类。然而，在类之间不存在严格的服务优先级，循环调度器在这些类之间轮流提供服务。在最简单形式的循环调度中，类 1 的分组被传输，接着是类 2 的分组，接着又是类 1 的分组，再接着又是类 2 的分组等等。一个所谓的保持工作的排队规则，即在有（任何类的）分组排队等待传输时不允许链路保持空闲。当寻找给定类的分组但是没有找到时，保持工作的循环规则（work-conserving round robin discipline）将立即检查循环序列中的下一个类。

图 7-21 描述了一个两类循环队列的操作。在这个例子中，分组 1、2 和 4 属于第一类，分组 3 和 5 属于第二类。分组 1 一到达输出队列就立即开始传输。分组 2 和 3 在分组 1 的