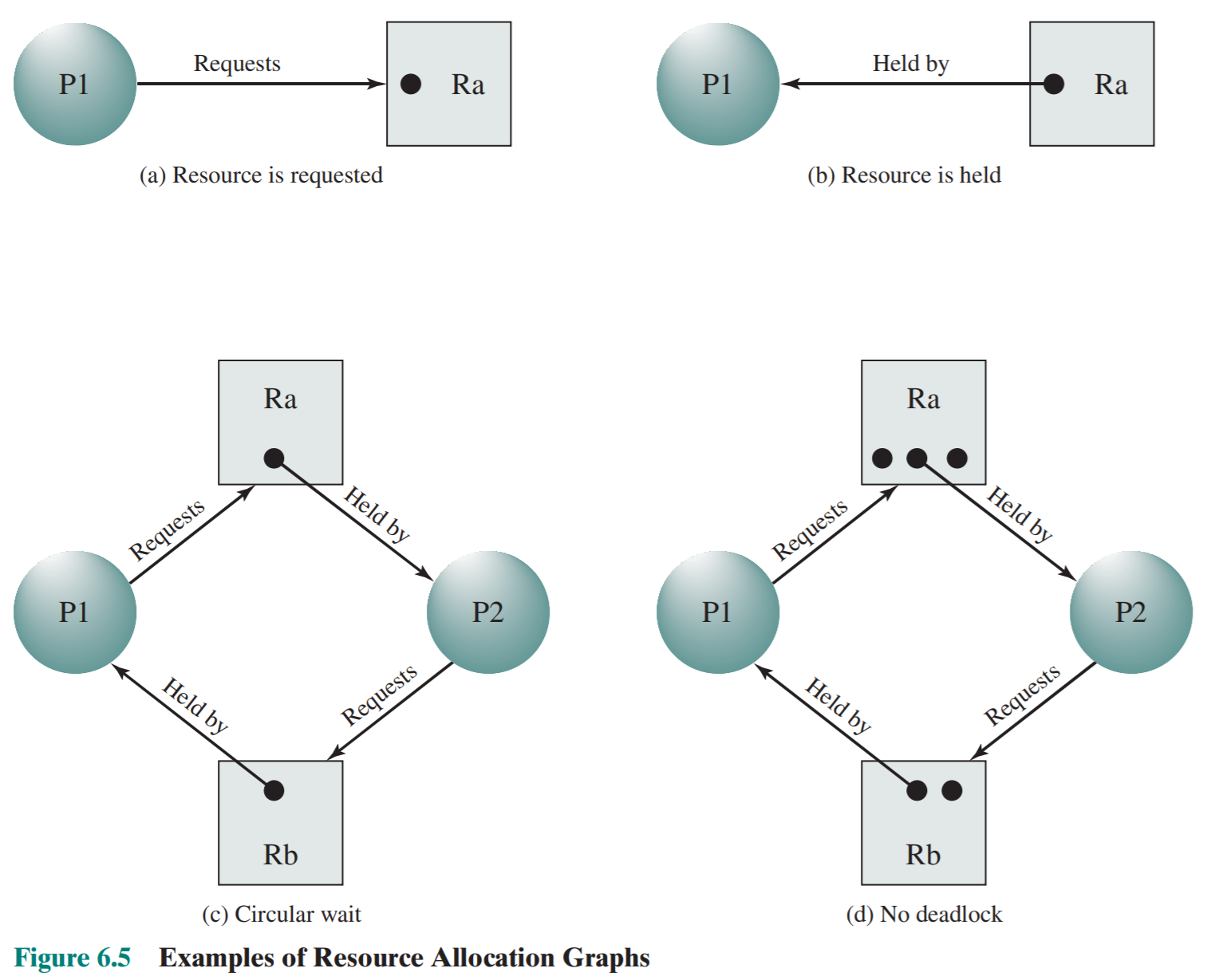
Deadlock Handout

Deadlock occurs if two processes need the same two resources to continue and each has ownership of one.

A joint progress diagram illustrates the progress of two processes competing for two resources.

A reusable resource can be safely used by one process at a time and is not depleted by that use.

A consumable resource is one that can be created (produced) and destroyed (consumed). When a resource is acquired by a consuming process, the resource ceases to exist.

A resource allocation graph is a directed graph that depicts a state of the system of resources and processes; each represented by a node.

For deadlock to occur, you must have:

1. Mutual Exclusion: only one process may use a resource at a time
2. Hold and Wait: a process may hold allocated resources while awaiting assignment of other resources
3. No preemption: no resource can be forcibly removed from a process holding it
4. Circular wait: a closed chain of processes exist, each process holds at least one resource needed by another process

Three general approaches exist to deal with deadlock:

1. Deadlock Prevention:guarantees that a deadlock will not occur. Prevention is achieved by assuring that one of the necessary conditions for deadlock is not met.
2. Deadlock Avoidance:examines each new resource request for deadlock. If the new request could lead to a deadlock, then the request is denied.
   1. Banker’s algorithm: strategy of resource allocation denial
3. Deadlock Detection:requested resources are always granted when available. Periodically, the operating system tests for deadlock.

UNIX provides a variety of mechanisms for interprocessor communication and synchronization. Pipes, messages, and shared memory can be used to communicate data between processes, whereas semaphores and signals are used to trigger actions by other processes.

A pipe is a circular buffer allowing two processes to communicate on the producer–consumer model. FIFO queue, written by one process and read by another.

Message: A block of information that may be exchanged between processes as a means of communication.

A spinlock is a [lock](https://en.wikipedia.org/wiki/Lock_(computer_science)) which causes a [thread](https://en.wikipedia.org/wiki/Thread_(computer_science)) trying to acquire it to simply wait in a loop ("spin") while repeatedly checking if the lock is available.

Starvation: A situation in which a runnable process is overlooked indefinitely by the scheduler; although it is able to proceed, it is never chosen.