# Device Drivers Part 1

MCIT 595

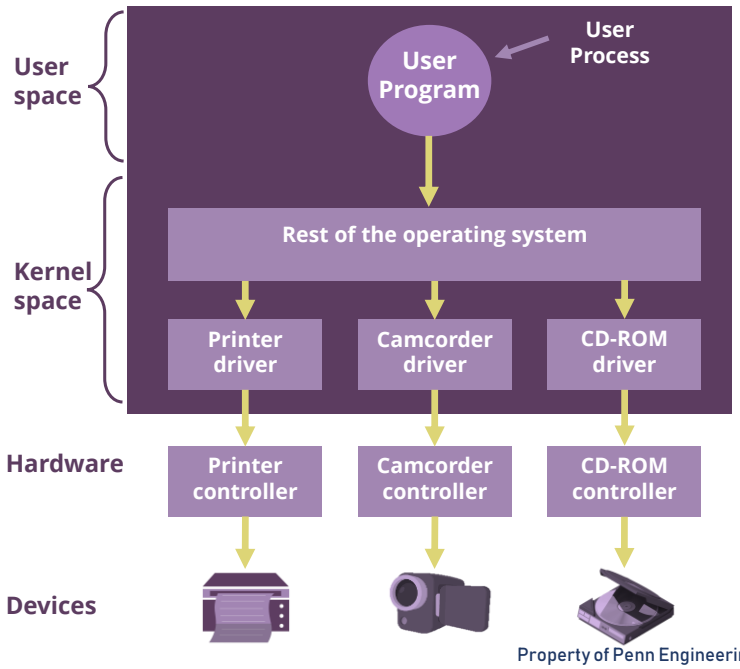



## Device Drivers

- Device drivers are the interface between the OS and a device
  - Whenever a new device is plugged into a computer, the OS installs new device drivers
  - Device Driver includes device-specific interrupt handlers
    - Device-specific I/O routines
    - Device-specific control routines (ioctl)
- Device drivers may be hard-coded part of OS kernel, dynamically loaded, or (sometimes) part of a user process
  - Depends on OS and kind of device

# Device Drivers

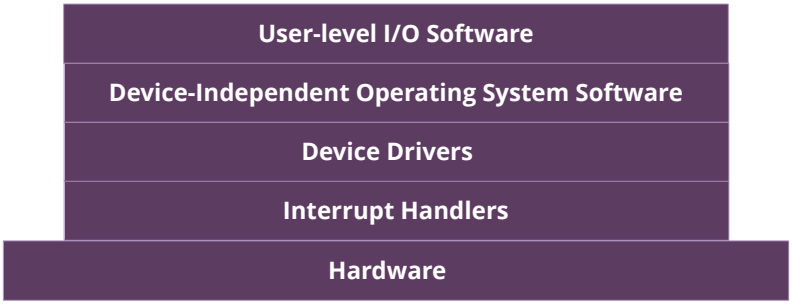
Logical positioning of device drivers. In reality all communication between drivers and device controllers goes over the bus.







# Layers of the I/O Software System



# I/O Software Layers

- **Device drivers:** Starts I/O operation and blocks until I/O completes
  - Typically structured as kernel processes, with own states, stacks, and program counters
- **Interrupt handler:** unblocks driver typically using semaphores or condition variables
  - Setting up a context for interrupt service procedure
  - Running the interrupt service procedure which extracts information from device controller register
  - Unblocking device driver to run
  - Upon completion, context switch to new process (which may be high priority process which blocked in I/O previously)