CMPSCI 377 Operating Systems **Fall 2005**

Lecture 17: November 22

Lecturer: Emery Berger *Scribe: Kowal, Asztalos*

Today: I/O

- Ports
- DMA
- Polling/Interrupts
- Devices
- Buffering
- Caching
- A Worked Example: Read
- Storage

17.1 Ports

Device ports act like a door, allow the sending or receiving of data. Due to its complications, ports are used for small amounts of data.

- Each device port has a status indicator, or "light". When the status "light" goes out, the device is available for sending or receiving of data.
- The status indicators for devices are busy, data, and error.
- Ports consist of three main parts:
  1. Control - Command to perform. Read by the controller.
  2. Data In - Read by the host to get input.
  3. Data Out - Written to the device

17.2 DMA

DMA is used for devices using large amounts of data. DMA involves shared memory between the processor and device.

17.3 Interrupts and Polling

- Polling causes the processor to continuously check for completed tasks. This results in busy work, which wastes the processor's time.
- Interrupts avoid the busy work of polling, allowing the processor to give its attention to other processes, and make better use of its time.

17.4 Devices

The abstraction of system devices at the API level can be broken down into three groups:

- Block Device - These devices can read, write, and seek Examples include:
  - Unix /dev/ files
  - Memory Mapping
- Character Devices - These devices can put and get characters
- Network Devices - These devices use a socket connection

17.4.1 Blocking and Non-Blocking Calls

- Blocking - Wait until the call is complete
- Non-Blocking - Return immediatly with any available data
- Asyncronous - Returns immediatly and signals completion via interrupts

17.5 Buffering

Buffering uses a memory area to store data prior to transfer to/from the CPU Why buffering?

- CPUs are fast, Devices are slow
- Devices have different transfer sizes and speeds. Buffering allows a uniform way for the processor to send and receive data.

17.6 Caching

Caching keeps recently used disk blocks in main memory after I/O call completes. Cache write policies are a trade off between speed and reliability:

- Write Through - Writes to all levels of memory. Reliable
- Write Back - Writes only to memory. Faster

Note: AMD processors use exclusive caching, which insures no duplicates are found between L1 and L2

17.7 A Worked Example: Read

Read Operation:

1. Request read from device
2. Check if the data requested is in the buffer, if not:
   - OS tells the device driver to perform input
   - Device driver tells DMA what to do
   - DMA controller transfers data to kernel buffer
   - DMA controller interrupts CPU when transfer is complete
3. OS transfers data to process, process is placed in ready queue
4. Process continues at point after system call

17.8 Storage

Goal: Improve performance Disks vs. I/O:

- Disks contain moving parts, unreliable
- Disks are slow
- Disks are **cheap**

Disks are slow: 1 seek = 40,000,000 cycles

Disk Transfer Time = Seek Time + Rotational Delay + Transfer(n)