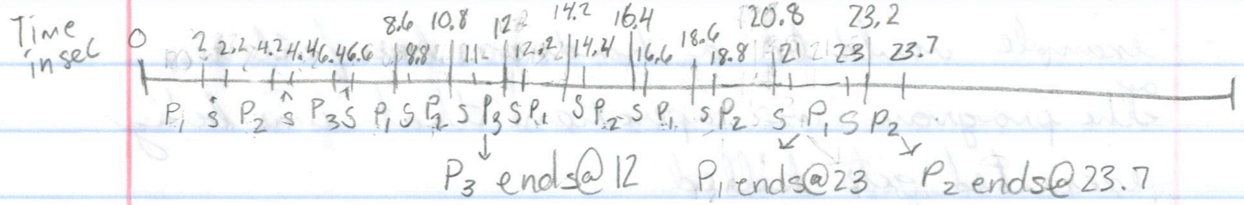
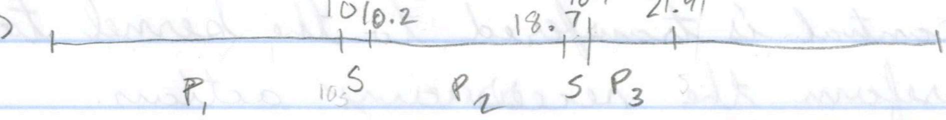
Andrew Candelaresi

1. S = switch



.2×11=2.2 2.2/23.7 = 9.283% of time spent on context switches

2) 

Batch multiprogrammed finished execution fastest if a CPU had multiple cores it could actually execute multiple processes simultaneous or in the case where one of the processes has an I/O call causing the* process to wait and sit idle in a batch run in a time slice it would switch to another process

3 The 4 exceptions are trap, fault, (Hardware) Interrupt, Abort. Trap is an intentional interruption form something like a software interrupt an example is a syscall. the return behavior returns to the next instruction where the call was made. A fault is a potentially recoverable error, for example division by zero, invalid op code or a segmentation fault, If the error is recoverable ~~the~~ it will return control to the user. The interrupt is a signal from an I/O device an example of would be input from a keyboard, or disk read finished. It always returns to the next instruction, asynchronous

An abort is a non recoverable error an example would be a hardware bus failure. The program or operation that was being executed gets killed

A software interrupt is when an application or a program sends an interrupt such as a syscall where control is transfered to the kernel to preform the necessary action. This differs from a hardware interrupt where the interrupt is issued by a hardware device like a disk, or keyboard. Eeach device has a unique Interrupt ReQuest line. This line. Based on the IRQ the CPU will dispatch the appropriate hardware driver.

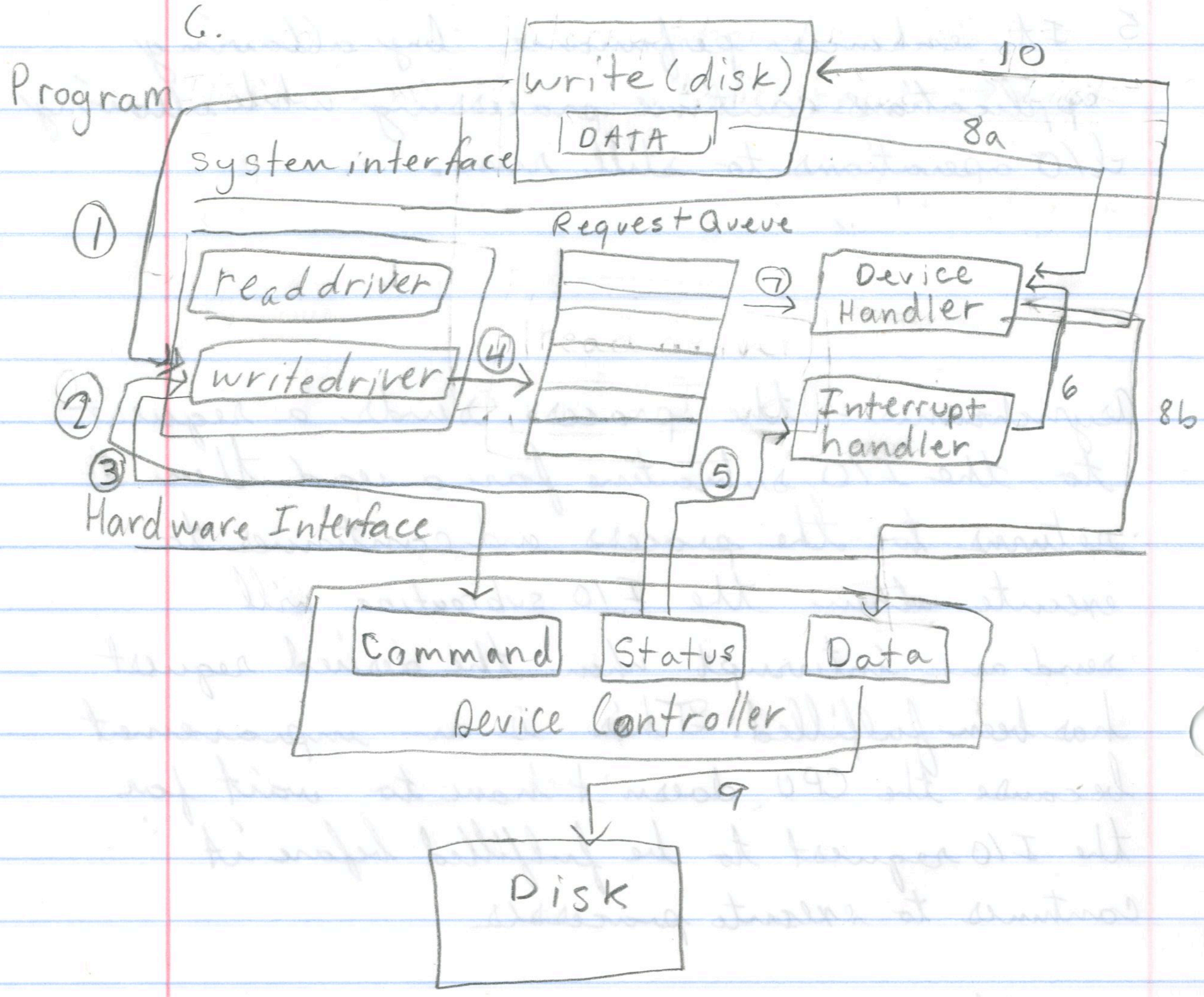
4. The jump table is designed to specify by number how each syscall should be handled. Fore each numbered syscall the table has associated functions and program files. The jump table is also called the Trap table

5 It enhances perfomance by allowing applications continue processing while allowing I/O operations to still run.

Asynchronus - the process sends a request to the I/O subroutine for a read then returns to the process and continues to execute then the I/O subroutine will send an interrupt when the desired request has been fulfilled. This is an improvement because the CPU doesn't have to wait for the I/O request to be fulfilled before it contunes to execute processes

Non blocking - the process sends an request to the I/O and returns imediatly with a value indicating how many bytes were tranfered the process then loop back to gather all data requred for the process. This is an impiovement because it fees up the Processor to continue execute processes while it rechecks if the fullan I/O request

6.


Program
write(disk)
DATA
10
8a
system interface
1
Request Queue
read driver
write driver
2
3
4
7
Device Handler
Interrupt handler
6
8b
5
Hardware Interface
Command
Status
Data
Device Controller
9
Disk