Limited Direct Execution Idea: To run as fast as possible, USER code runs derectly on the CPU. Some instructions are too - powerful - alter machine state beyond one process There for...
Some instructions, et attempted by user process, kill the

So how do these special instructions get executed?

HND... how to enter/exit the kanel state - for example - septem calls.

Vser processes cannot access kernel memory. Solution: processe states highest often reserved for a "hypervisor" moltiple levels or rings When a user process needs the Services of the kernel, it TRAPS to Cheese a state transition. What it looks like: // blah blah blah vor = read (_____) // blah blah blah

what it is: Kernel Save state // blah interptoup do syscoll Von = read (-) Set up book keepen restore state return from trap >//blah the open () sejstem call put prevameters in right Open (.,,) ->

pot "I am asystem coll" in right place. put "I am Open" in right place trap.

ARM64 Syscall number > X8

parameters > x0 > x7

trap > suc Ø

return value in XØ More détails on RiseV lookat trampoline. S uservec Methods of multitasbeing

— illusion of ocerning the CPU

premptive / 1 created by making interval

Faster & Gaster Hway Stop Segn amiga in 1985 screw you apple Cornot take Connot take down system

cooperative yield () or sleeped etc make a segitem call in general.
mae OS 1-> 9 +: (1 2001
1: b 16 or inother nords
lable: branch to babbe.
Aug kind of infenite bop in User code courses tight lock ap!!!

Constept switch %21 us expervelent to about 2000 instruction