Lecture 2

Clock: anything that can be turned on and off; used for timing and synchronization

Clock cycles: e.g. 100 MHz bus frequency: 100 Million cycles each second: 100 \* 10^6

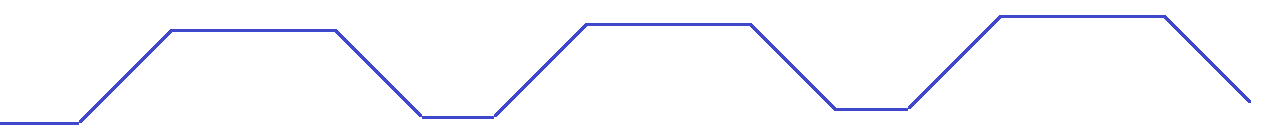
1 GHz cpu: 1 billion rising and falling edges per second

1 rising edge per nanosecond

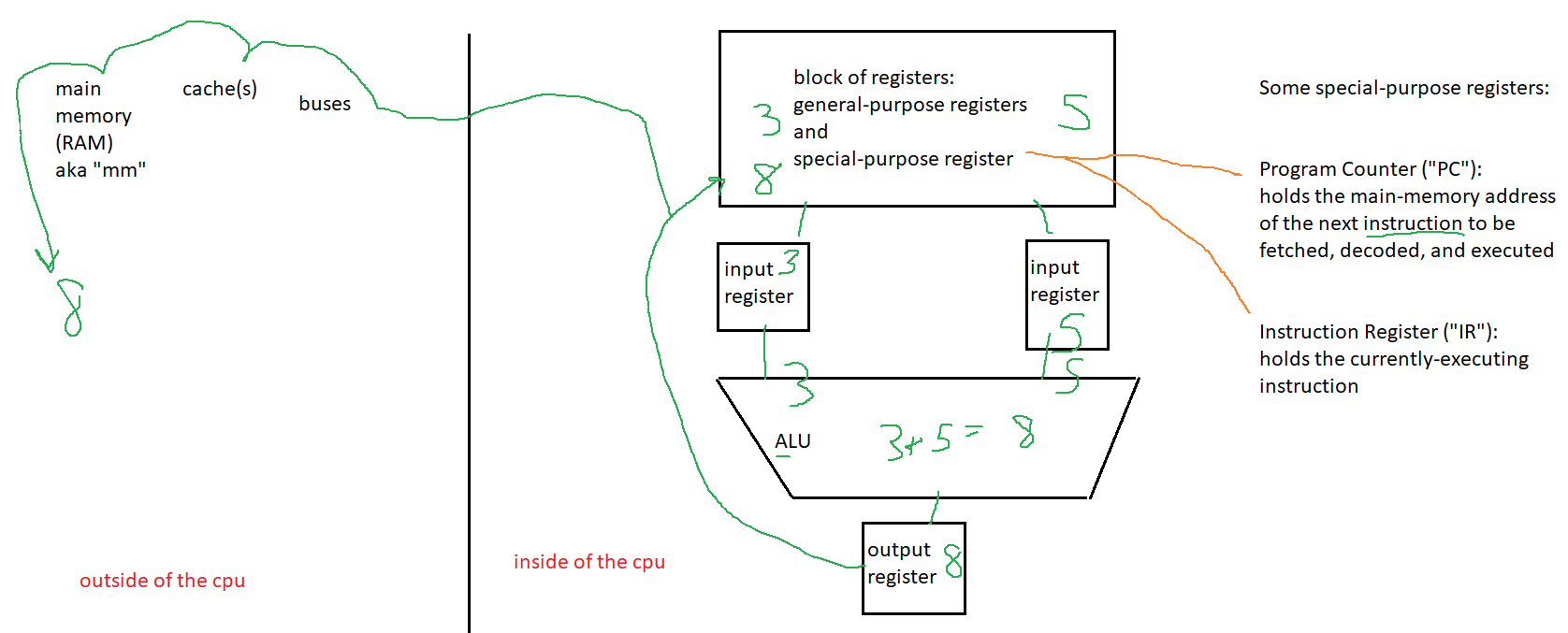
POSSIBLY we could start one instruction per nanosecond

The clock cycle of regular cable television shows is 30 minutes…which means shows could BEGIN every half hour. It does not mean shows are half an hour long.

The clock cycle of movies in theatres is 5 minutes…which means shows could BEGIN on the five minutes (e.g. 7:05, 7:10, etc). It does not mean shows are five minutes long.



von Neumann datapath: the inside of the cpu



Pages 63-65: a wish list of how to speed up instruction execution:

1. All instructions should be directly executed by hardware complex and expensive

2. Issue instructions as fast / often as possible physics limits, complex, expensive

3. Instructions should be easy to decode same-sized instructions, backward compatibility

4. Only LOAD and STORE instructions should reference mm cannot avoid going to main memory

5. Provide lots of registers expensive

Overall goal: to prevent the cpu from starving

RISC vs CISC architectures:

Reduced instruction set computer: fewer, and simpler instructions

Complex instruction set computer: more, and more complex instructions: e.g. implement the most common instructions directly in hardware (no interpretation required), even the complex ones

OUR computer is a mix: mostly RISC with some CISC

Chart

Description automatically generated with low confidence

In computer programming, programs tend to run in linear order. If the CPU, for example, requested word 6000, it is likely that soon, words around 6000 will be requested. This is called the **principle of spatial locality**. Because of the principle of spatial locality, neighboring words are fetched into the cache whenever memory words are, because cache serves words faster to the cpu. Analogy: if you hear people talking about Tiger Woods, you can guess words they will probably say in the near future (2 minutes?): golf, cheater, titles, sports, nike. Another analogy: if you see someone at the store with taco shells in their basket, you can guess other items that will be in there too: beef, cilantro, cheese, sour cream

There is also a **principle of temporal locality**, which means the same word being requested now, will likely be re-requested again soon. Because of the principle of temporal locality, the cache tries to KEEP recent words there (don’t let them get evicted). Analogy: you hear someone in the hallway talking about Tiger Woods. You can guess some words they’ll say (probably) in the near future: Tiger Woods.

DEFINITION: Interpreter: Any software program (aka a virtual machine) that FDE the instructions of another program. The output of this is another program that is simpler to run. But, this process takes time. This is cheaper than having hardware run the original (more complex, higher-level) program and it is *cheaper* than having hardware do the conversion (interpretation).

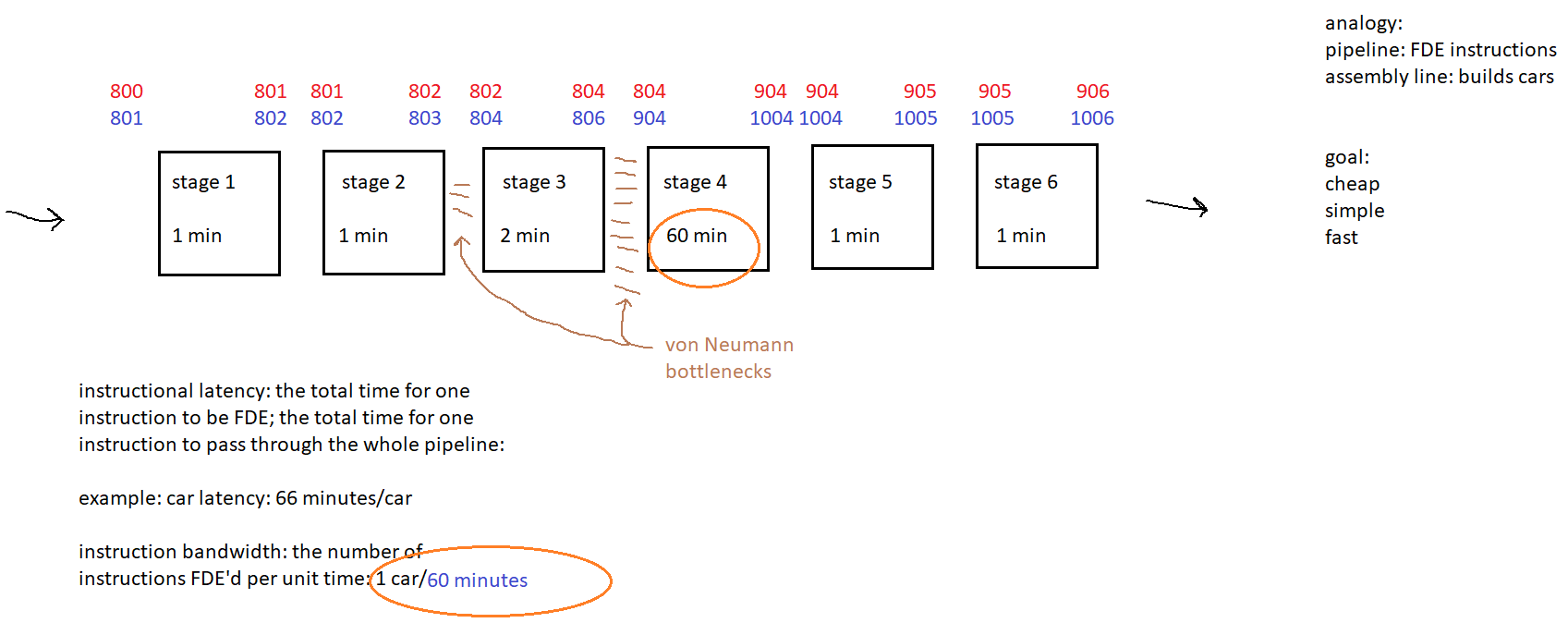
**Before** next class, read pages 65-69 of the textbook. Also read pages 73-78

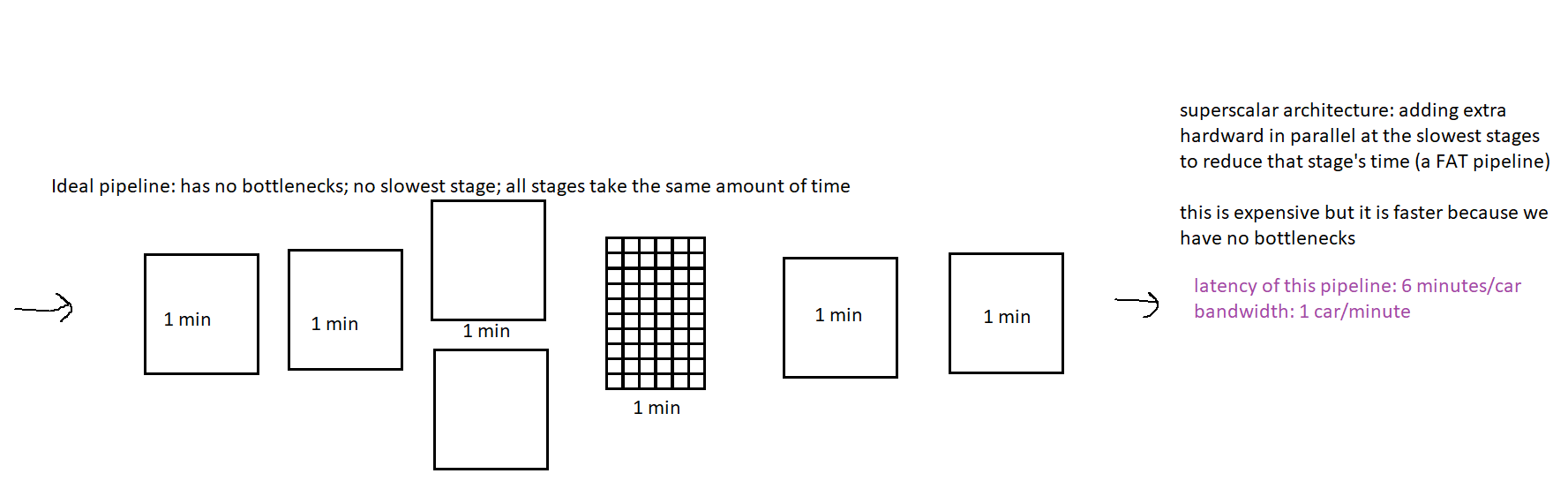
# Pipelining

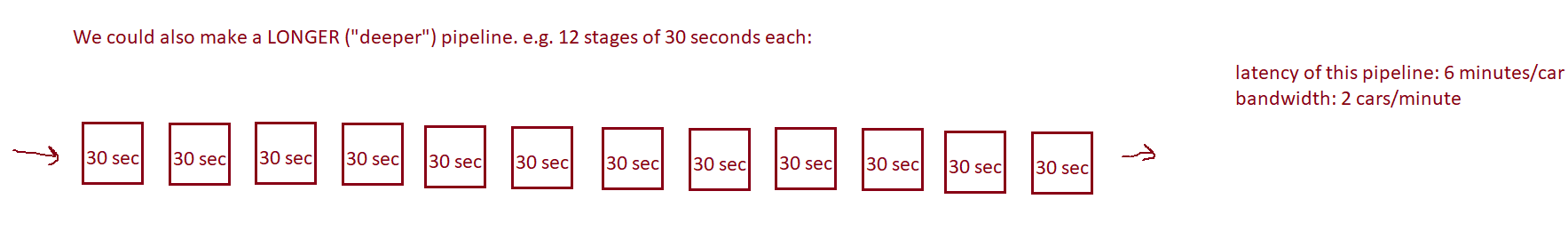
Definition: a pipeline is multiple hardware units working in parallel and series to fetch, decode, and execute instructions in order to maximize instructional bandwidth (which is measured in instructions FDE/second; e.g. MIPS).

Instructional Latency: the total time for one instruction to be fetched, decoded, and executed from start to finish (the total time for one instruction to pass through a pipeline).

We want high bandwidth and low latency.







Latency and bandwidth of Netflix: high or low bandwidth? High.

Low bandwidth could look like: freezing/buffering, lagging, low resolution

High or low latency? High latency could look like: LONG time to start a movie

Roller coaster: Do you want high or low latency? You want to spend some time on the ride…so you want higher latency. Latency is how long it takes to go through one entire circuit on the roller coaster. Do you want high or low bandwidth? Higher bandwidth means more people finishing the ride per unit time. You get high bandwidth and high latency by having lots of cars ($$$$).

Lineup=bottleneck

Playland: many rides have long lines, short rides

Disneyland: there is one very long ride with very short lines: it’s a small world. 12-15 minutes. Possible only with lots of money.

QUESTION:

What are the bandwidth and latency for a pipeline with four stages which take .01millisecond, 250 000 nanoseconds, 30 microseconds, and 2 milliseconds?

ANSWER: Latency: 0.01 ms = 10 us (us = microseconds) +  
 250000 ns = 250 us +  
 30 us = 30 us +  
 2 ms = 2000 us  
 = 2290 us  
 = **2.29 ms / instruction**

Bandwidth: 1 instruction / slowest stage…**1 instruction/2 ms**