COMP2825 Architecture Lecture 2

DEFINITIONS:

Clock anything that can be turned on and off

Clock cycle e.g. 100 MHz bus frequency: 100 million cycles per second; 100\*10^6

Period = 1/frequency; 1/1M = 1 microsecond = 10^-6

1 GHz cpu 1 billion rising and falling edges per second

1 rising edge per nanosecond

POSSIBLY we could start one instruction each nanosecond

The clock cycle of cable tv shows is 30 minutes: that means tv shows could begin every half hour. It does NOT mean that tv shows are finished in 30 minutes.

The clock cycle time of movies in multi-screen theatres (i.e. parallel) is 5 minutes, which means we can trigger a “movie starting” every five minutes…it does NOT mean the movies are 5 minutes long.

Icon

Description automatically generated

von Neumann Datapath (the inside of the CPU): (page 57)

Diagram

Description automatically generated

Page 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: complex and expensive and physics limits
3. Instructions should be easy to decode: backward compatibility,

same-sized instructions

1. Only LOAD and STORE instructions should reference main memory: cannot avoid going to MM (RAM)
2. Provide lots of registers: expensive

GOAL: 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

<https://cs.stanford.edu/people/eroberts/courses/soco/projects/risc/risccisc> :

The CISC approach attempts to minimize the number of instructions per program, sacrificing the number of cycles per instruction. RISC does the opposite, reducing the cycles per instruction at the cost of the number of instructions per program.

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

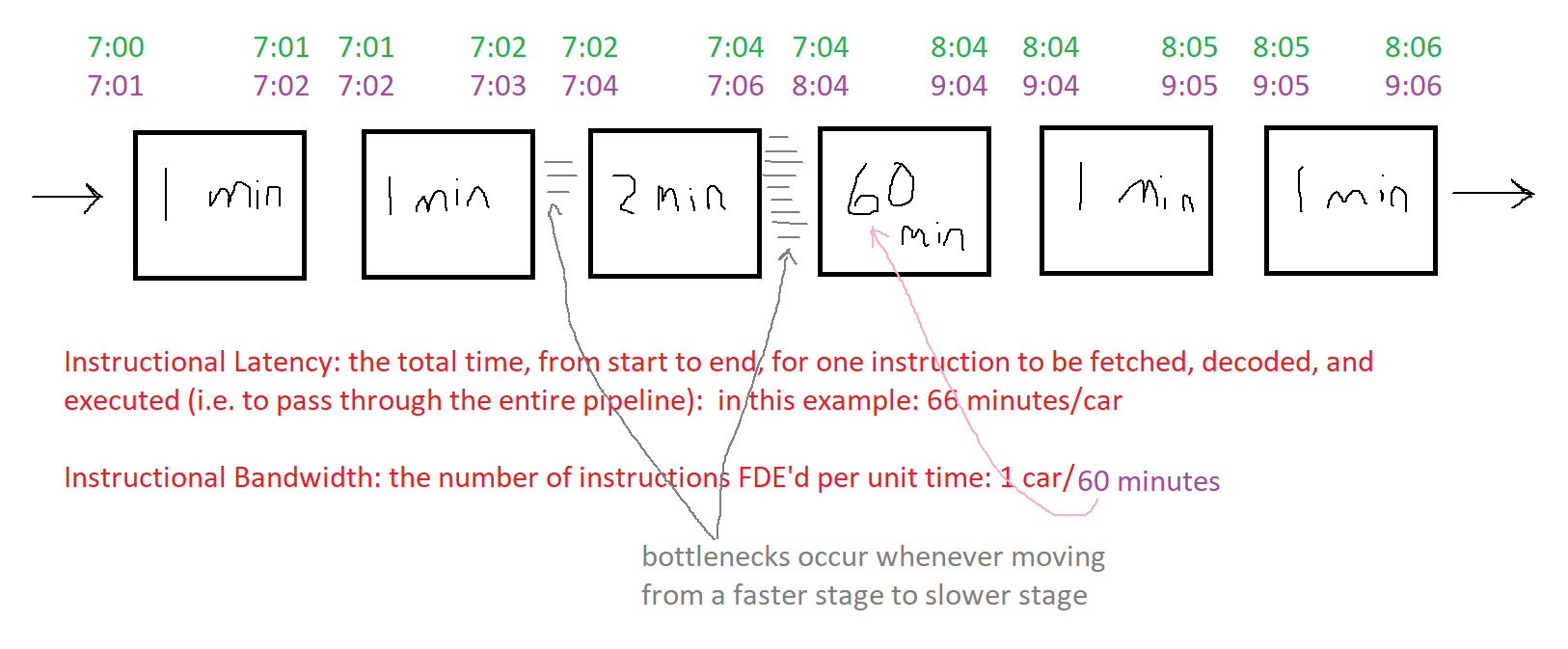
Any software program (aka a virtual machine) that FDE (fetches, decodes, and executes) the instructions of another program. The output of this is another program that is simpler to run. But, this process takes time. This is cheaper than having hardware run the original (more complex, higher-level) program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

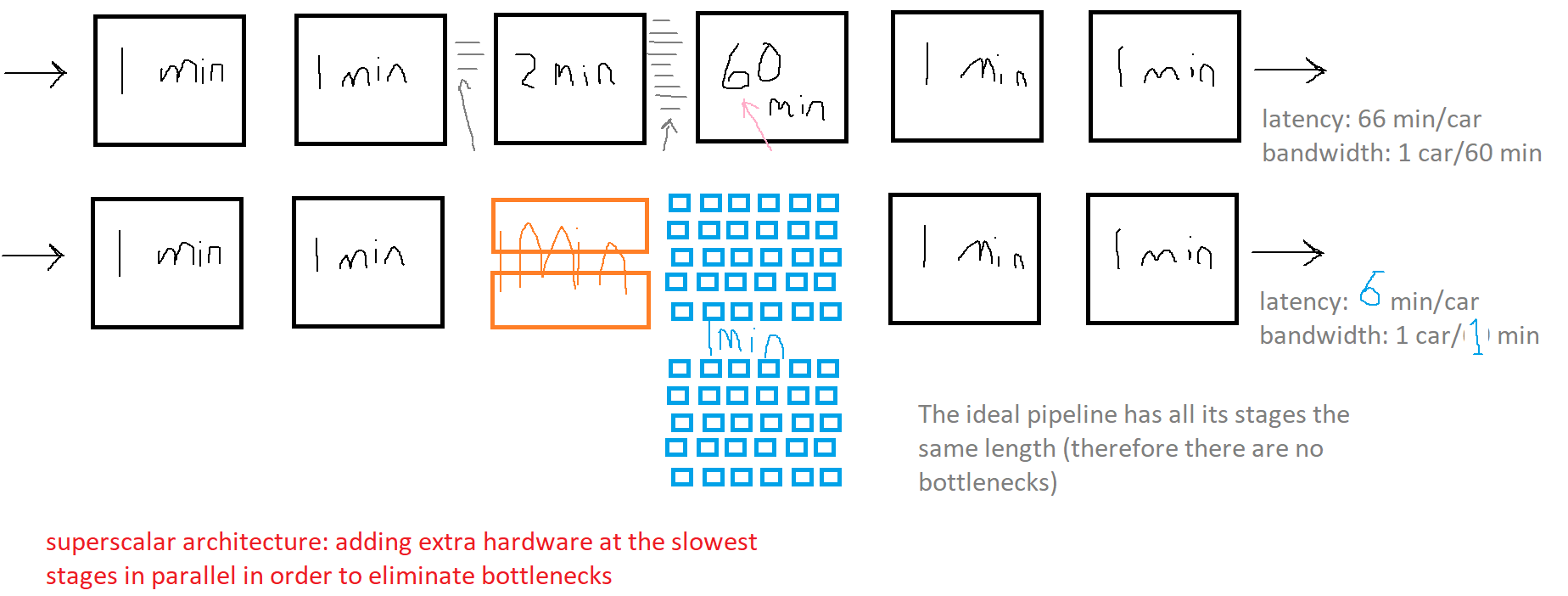
# Pipelines

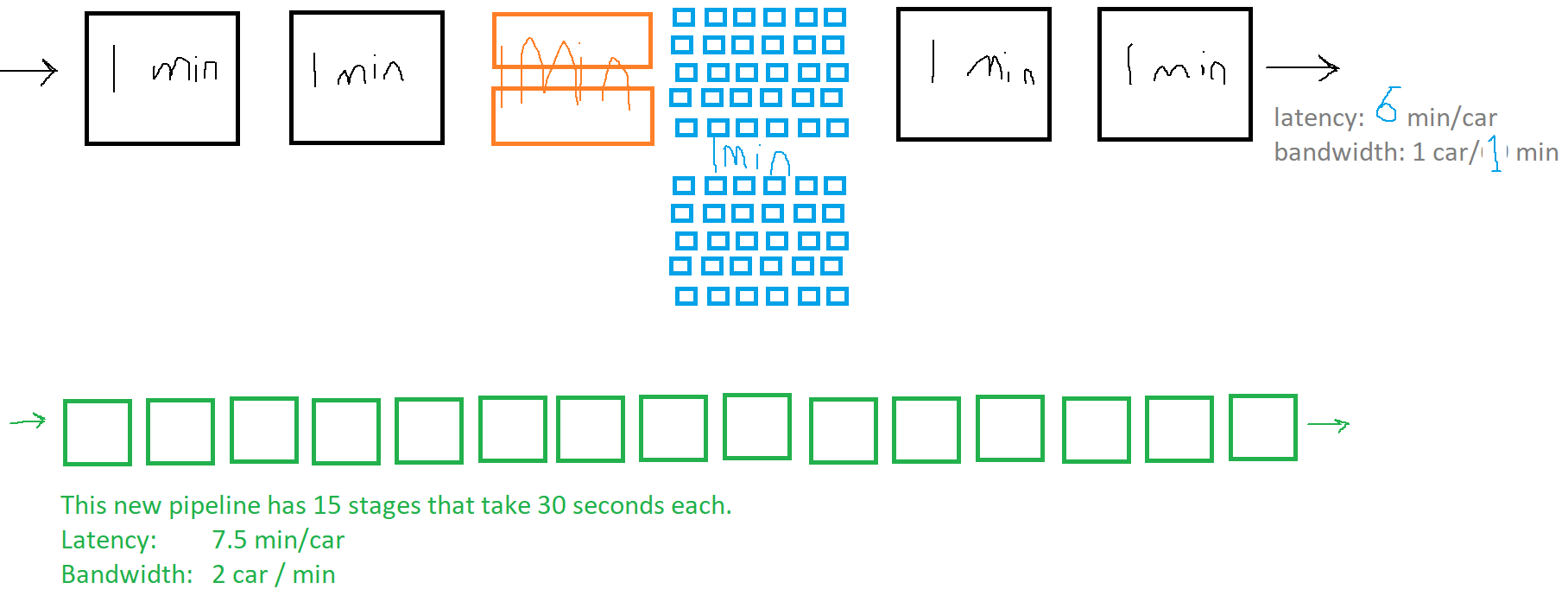
Refer to page 66 of the textbook

Analogy: a computer pipeline is a set of multiple hardware devices that FDE instructions in parallel and in series, in order to maximize instructional bandwidth.

A car assembly line is a set of multiple hardware devices that build cars in parallel and in series, in order to maximize car-creation bandwidth. Assembly lines can build cars in a way that is simpler, cheaper, and faster.







Original Netflix when first created. What was its latency (time to start a movie from when it was ordered): days! Does Netflix have high or low bandwidth? High bandwidth (can serve a lot of movies at once). Low bandwidth would look like: only a few people could watch at once, very low resolution, stalling and buffering.

Disneyland versus Playland rollercoasters:

Rollercoaster latency means how much time a rider spends from start to finish on the ride.

The lineup could be considered a bottleneck.

Bandwidth of a rollercoaster: the number of people getting off the ride per unit time

What are the latency and bandwidth of a pipeline whose stages take 20ns, 15ns, 40ns, and 5ns?

Latency: 20+15+40+5 = 80ns/instruction

Bandwidth: 1 instruction / 40 ns (1 / slowest stage)