# CS305
# Computer Architecture

## Introduction to Pipelining

Bhaskaran Raman
Room 406, KR Building
Department of CSE, IIT Bombay

http://www.cse.iitb.ac.in/~br

# Some Fun Videos to Watch

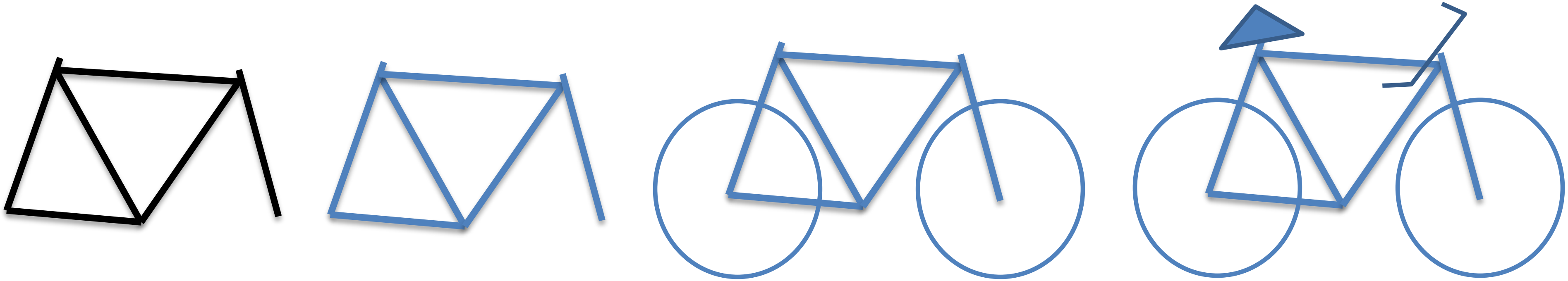
- Charlie Chaplin’s critique of industrialization
  - https://www.youtube.com/watch?v=a-FbVF1x1_U
- Modern Times, Assembly Line with Charlie Chaplin
  - https://www.youtube.com/watch?v=QdwH84AT5fU
- I Love Lucy – The Candy Wrapping Job
  - https://www.youtube.com/watch?v=Clr6Zyjj7iI

# Lessons Learnt

- Pipelining is natural (at least for machines)
- Overall speed only as good as the slowest unit
- Non-uniformity is bad for pipeline
- Exceptions are really bad

# Example Task: Bicycle Manufacture

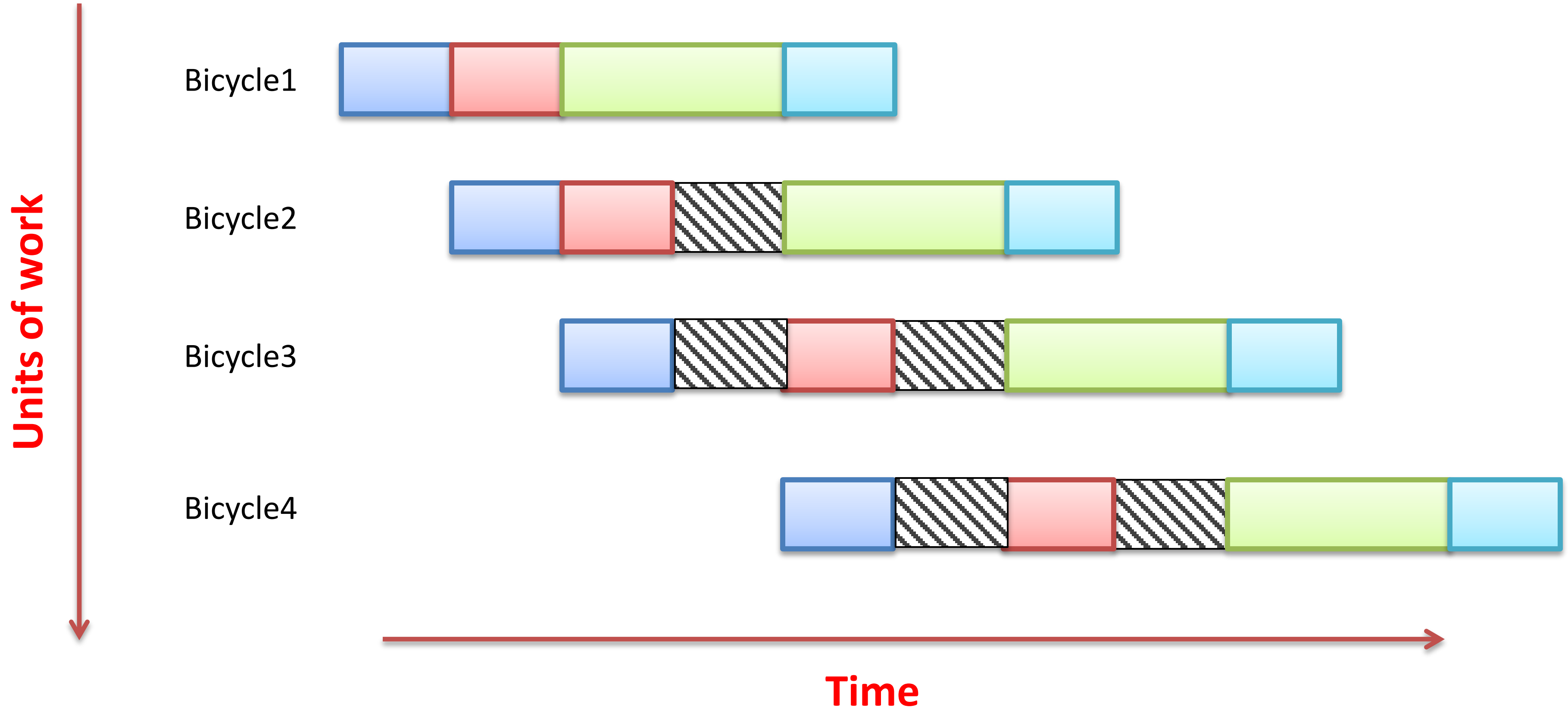
Make frame → Paint frame → Fit wheels → Fit accessories



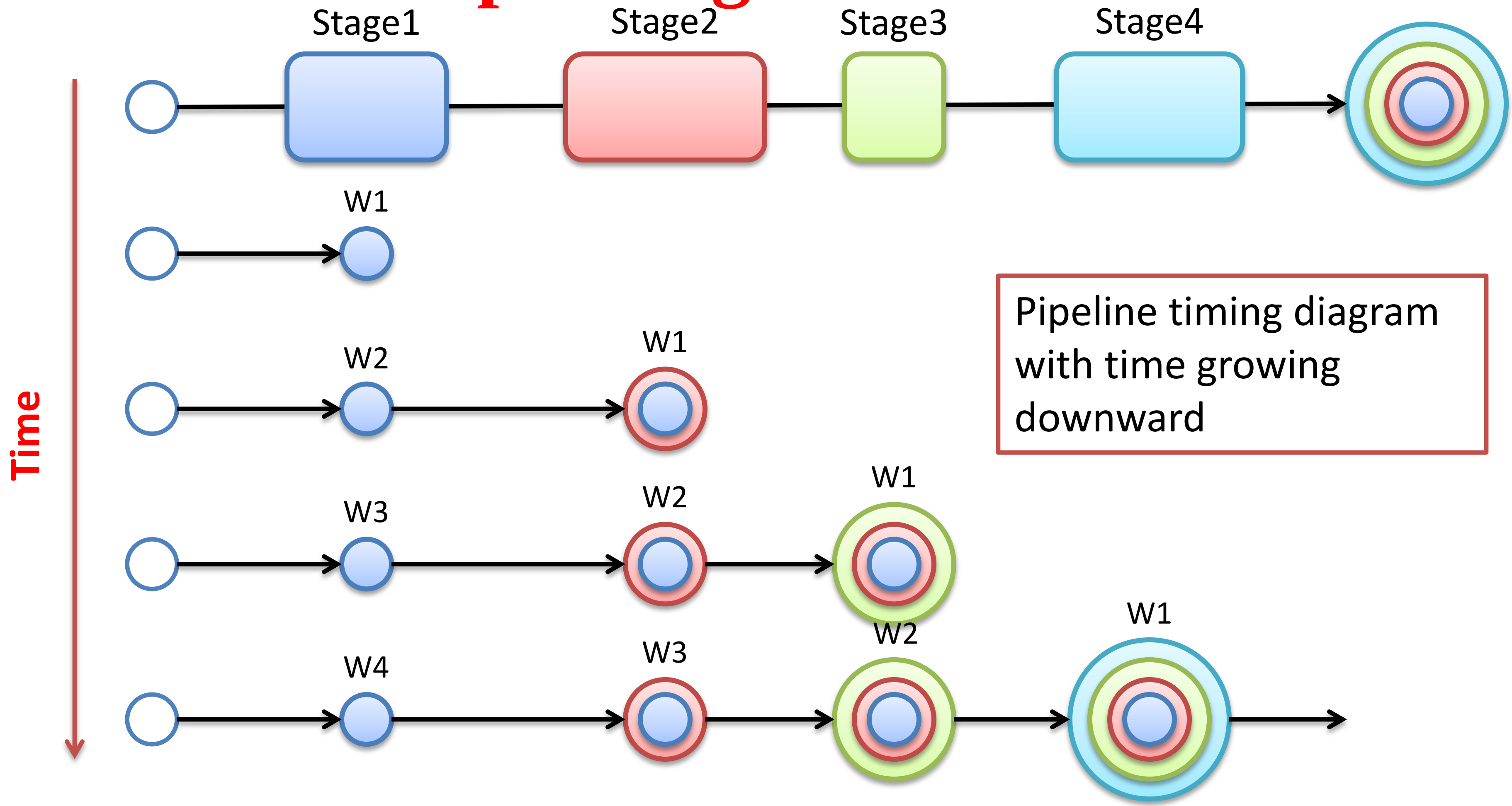
1 hour | 1 hour | 2 hours | 1 hour

- Time to make 1 bicycle?
- Time to make 10 bicycles?
  - With pipelining?  Without pipelining?
  - Pipelining:
    - "Make frame" for $2^{nd}$ bicycle, when "Paint frame" for $1^{st}$ bicycle
    - "Make frame" for $3^{rd}$ bicycle, "Paint frame" for $2^{nd}$ bicycle, when "Fit wheels" for $1^{st}$ cycle
    - And so on…

# Pipeline Timing Diagram



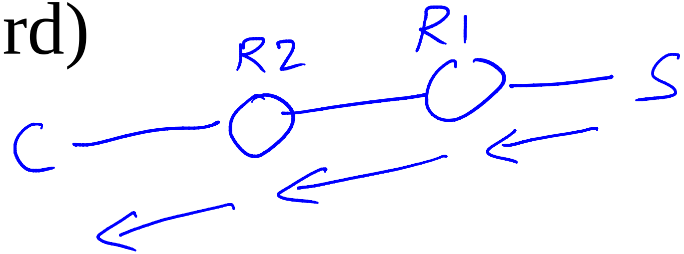
# Pipelining in Abstract

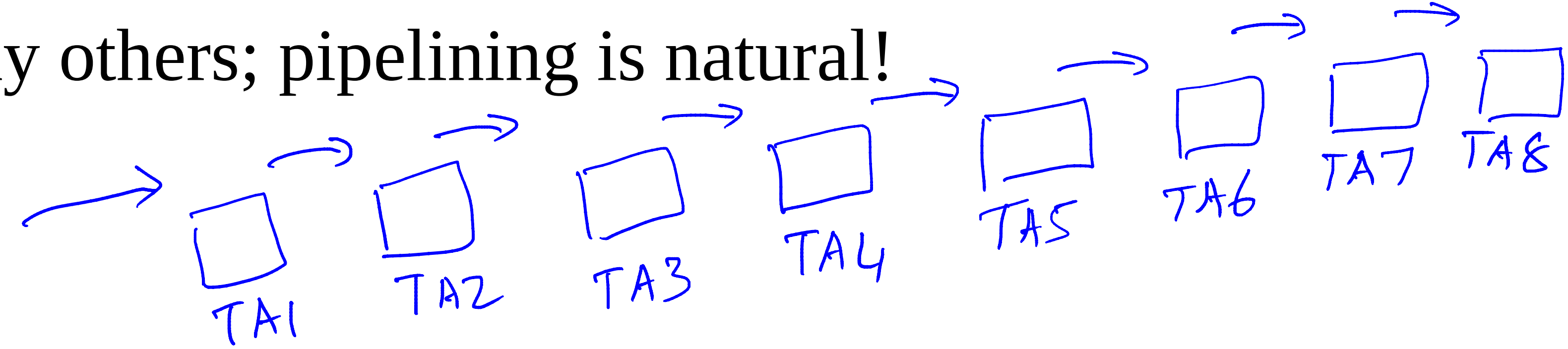



Pipeline timing diagram with time growing downward

# Other Examples of Pipelining

- Water pipeline (origin of word)
- Network packets
- Course project evaluation
- Many others; pipelining is natural!



# Pipeline Speedup

- Time taken to manufacture *N* bicycles
  - Without pipelining: ***5N***
  - With pipelining: ***5 + (N-1) x 2***
  - Speedup = ***5/2*** for large ***N***
- Suppose "Fit wheels" also takes 1 hour?
  - Speedup = ***4N/(4+N-1)*** = ***4*** for large ***N***
- Suppose "Fit wheels" takes 2 hours, but is broken into "Fit front wheel", "Fit rear wheel" each of 1 hour?
  - Speedup = ***5N/(5+N-1)*** = ***5*** for large ***N***

# Pipeline Speedup (continued)

- In general, ***k*** stages, each of time ***t***
- Time for ***N*** units:
  - Without pipeline: ***N x kt***
  - With pipelining: ***kt + (N-1)t***
    - Pipeline setup time = time to fill pipeline = time to first output
    - Is a significant component, for small ***N***
  - Speedup = ***k*** for large ***N***
  - This is the ***ideal*** speedup

# Ideal Pipeline Speedup

- Ideal pipeline speedup = number of pipeline stages
- Necessary conditions:
  - All stages are of equal length in time
  - Enough hardware/hands: e.g. same spanner/person/machine cannot be used for “fit wheels” and “fit accessories”
  - Many, many more conditions (stated later)

# Latency versus Throughput

- Latency: from perspective of single unit of work
- Throughput: rate at which work completes

# Pipelining in MIPS

| | | |
|---|---|---|
| Stage-1: | Instruction fetch, PC=PC+4 (all instructions) | **IF** |
| Stage-2: | Reg read, branch target computation (all instructions) | **ID** |
| Stage-3: | ALU operation (reg-reg)<br>Memory address computation (lw, sw)<br>Branch condition computation (beq) | **EX** |
| Stage-4: | Memory access (lw, sw)<br>Reg write back (reg-reg) | **MEM** |
| Stage-5: | Reg write back (lw) | **WB** |

# Pipelining for Sequence of `lw`

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| `lw $t0, 0($sp)` | IF | ID | EX | MEM | WB | | |
| `lw $t1, 4($sp)` | | IF | ID | EX | MEM | WB | |
| `lw $t2, 8($sp)` | | | IF | ID | EX | MEM | WB |

Such pipelined execution of MIPS instructions is ***potentially*** possible

# Pipelining for Sequence with Other Instructions

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| `add $t0, $t1, $t2` | IF | ID | EX | MEM | WB | | |
| `ori $t1, $s0, 15` | | IF | ID | EX | MEM | WB | |
| `lw $t2, 8($sp)` | | | IF | ID | EX | MEM | WB |

Effect of moving WB to stage-5 of reg-reg ?
Latency of those instructions increases
Instruction throughput not affected!

# Summary

- Pipelining a natural idea for speed-up
- Latency versus throughput
- MIPS: uniformity of instructions allows pipelining
- Issues with pipelining: hazards
  - Charlie Chaplin wants to scratch under his arm, take a break
  - Gets slowed down by buzzing bee
  - Pipeline stops ➔ startup delays