Introduction

- Assignment #3
- Pipelining concept
- Hazards

Please put your cellphone on vibrate
1. Optimize your CPU using Pipelining (design only)
   a. Do you need to change your ISA?
   b. How many stages are in your design?
   c. Rewrite your assembler if necessary?
   d. Are you implementing forwarding?
   e. How do you handle branch-hazards? No-ops?
   f. Draw out a pipelining CPU circuit block
      i. Registers
      ii. Control unit
      iii. ALU
      iv. Memory
   g. Write the Verilog for each of the blocks
   h. Must implement hazards detection & forwarding (or installing) in hardware.

2. Due Next Friday 6pm (8/28)
lw $t0, 16($s0)

1. Read from instruction
2. Read the register $s0 data (i.e $s0=52)
3. Calculate the memory address (52 + 16)
4. Read the data from memory address 68
5. Store the value of the memory address 68 to $t0

MIPS Example:

- **LW $1, 100($4)**
- **LW $2, 104($5)**
- **LW $3, 108($4)**
- **LW $5, 4($9)**
- **LW $7, 8($1)**
Hazards

ADD T1, T2, T3
SUB S1, S2, T1

Data Hazard
stall?
forward?

Detect hazard
Compare regs of
all instructions the
pipelined CPU has loaded