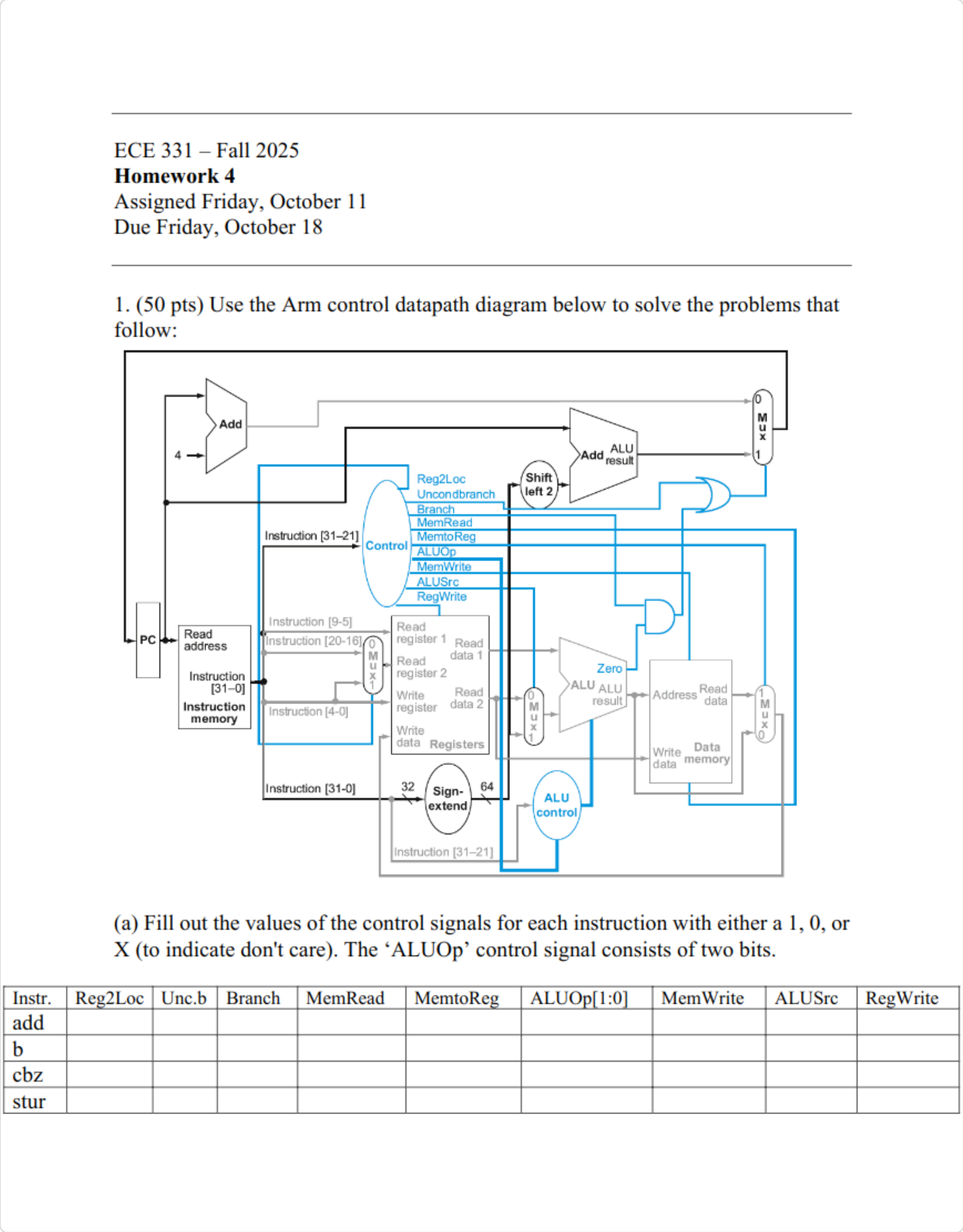
ECE331 Homework 4

PDF



ECE 331 – Fall 2025
Homework 4
Assigned Friday, October 11
Due Friday, October 18

1. (50 pts) Use the Arm control datapath diagram below to solve the problems that follow:

(a) Fill out the values of the control signals for each instruction with either a 1, 0, or X (to indicate don't care). The 'ALUOp' control signal consists of two bits.

Instr.	Reg2Loc	Unc.b	Branch	MemRead	MemtoReg	ALUOp[1:0]	MemWrite	ALUSrc	RegWrite
add									
b									
cbz									
stur									

(a) Fill out the values of the control signals for each instruction with either a 1, 0, or X (to indicate don't care). The ‘ALUOp’ control signal consists of two bits.

Instruction	Reg2Loc	Unc.b	Branch	MemRead	MemToReg	ALUOp[0:1]
add	0	0	0	0	0	10
b	x	1	0	0	x	x
cbz	1	0	1	0	x	01
stur	1	0	0	0	0	00

(b) Using X's to indicate in the table below which blocks in the datapath diagram are used for each instruction

Instruction Registers Sign-Extend Branch ALUControl LowerALU UpperALU Shift Left Data Memory

instruction	Register	Sign-Extend	Branch	ALUcontrol	Lower ALU	Upper ALU	Shift Left
Add	x	x		x	x		
b			x			x	
cbz	x	x	x	x	x		x
stur	x	x		x	x		

(c) Given the execution times for each of the blocks listed below, determine the total execution times for the instructions: add, b, cbz and stur. Note, this problem is challenging. You must determine the different components of the data path that each instruction uses, then find the time that signals require to complete the path, and then

use the path that takes the longest to determine the total execution time for that instruction.

Component	Time (ps)
InstructionMemory	280 ps
Add	230 ps
Control	70 ps
Mux	40 ps
Registers	170 ps
Sign-Extend	40 ps
ALU	240 ps
DataMemory	280 ps
Shift Left 2	12 ps
AND Gate	3 ps

Add Instruction

The **Add** instruction involves the following components:

- Instruction Memory: 280ps
- Control: 70ps
- MUX: 40ps
- Registers: 170ps
- ALU: 240ps
- Registers (again): 170ps

Total Execution Time:
280ps + 70ps + 40ps + 170ps + 240ps + 170ps = **970ps**

B Instruction

The **B** instruction uses the following components:

- Instruction Memory: 280ps
- Control: 70ps
- Shift Left 2: 12ps
- Add: 230ps

- MUX: 40ps

Total Execution Time:
280ps + 70ps + 12ps + 230ps + 40ps = **632ps**

CBZ Instruction

The **CBZ** instruction uses the following components:

- Instruction Memory: 280ps
- Control: 70ps
- Registers: 170ps
- ALU: 240ps
- AND Gate: 3 ps
- Shift Left:12 ps
- Add :230 ps

Total Execution Time:
280 ps+70 ps+170 ps+240 ps+3 ps+12 ps+230 ps= **1005 ps**

STUR Instruction

The **STUR** instruction involves the following components:
-Instruction Memory :280 ps
-Control :70 ps
-Sign Extend :40 ps
-Registers :170 Ps
-ALU :240 Ps
-Data Memory :280 Ps

Total Execution Time:
280 Ps+70 Ps+40 Ps+170 Ps+240 Ps+280 Ps= **1080 Ps**

2

IF	ID	EXE	MEM	WB
310 ps	340 ps	490 ps	325 ps	260 ps

A.) What is the minimum clock period for a pipelined and a non-pipelined processor using these parameters?

Pipelined - each stage runs idependently so the clock needs to be the length of the longest state which is EXE, 490 ps
non-piplined - minimum time to run everything once, so add it all up
310ps + 340ps + 490ps + 325ps + 260ps = 1725ps.

B.) What is the total latency of an Arm LDUR instruction in a pipelined processor? What is the throughput of a large series of LDUR instructions with no stalls or hazards? Express your answer in millions of instructions per second (MIPs).

pipelined - same, 490 ps
clock throughput: $f = \frac{1}{T}$ or $f = \frac{1}{490*10^{-12}} = 1.04 * 10^9 Hz$
in mips: $\frac{2.04*10^9}{20^6} = 2.04 * 10^3 = 2040MIPs$

3

(30 pts) Do Zybooks 4.17.7: [20] <COD §4.4> in chapter 4.17, but with the following latencies:

Instruction Memory	Add	Mux	Alu	Registers	Data Memory
250 ps	90 ps	45 ps	110 ps	130 ps	250 ps

Sign Extend	Shift Left2	single gate	control	register file
26 ps	25 ps	5 ps	50 ps	150 ps

"Register read" is the time needed after the rising clock edge for the new register value to appear on the output. This value applies to the PC only. "Register setup" is the amount of time a register's data input must be stable before the rising edge of the clock. This value applies to both the PC and Register File.

1. Although the control unit as a whole requires 50 ps, it so happens that we can extract the correct value of the `Reg2Loc` control wire directly from the

instruction. Thus, the value of this control wire is available at the same time as the instruction. Explain how we can extract this value directly from the instruction. Hints: Carefully examine the opcodes shown in COD Figure 2.20 (LEGv8 instruction encoding). Also, remember that `LSR` and `LSL` do not use the Rm field. Finally, ignore `STXR`.

    1. to extract the value of the Reg2Loc wire we can look in the instruction, the way we do this is to look in the type of the instruction, in R type instructions: the second source register is in the rm field of the command, meanwhile commands like LSR dont need to use the Reg2Loc wire allowing us to extract the command directly without consulting the control

2. What is the latency of an R-type instruction (i.e., how long must the clock period be to ensure that this instruction works correctly)?
    1. intruction memory : 250 ps
    2. register read: 130 ps
    3. ALU: 110 ps
    4. register write: 130 ps
    5. summing all of these we get 620 ps
3. What is the latency of `LDUR`? (Check your answer carefully. Many students place extra muxes on the critical path.)
    1. instruction memory: 250 ps
    2. register read: 130 ps
    3. Sign Extend: 26 ps
    4. ALU: 110 ps
    5. Data Memory: 250 ps
    6. register write: 130 ps
    7. summing all of these we get 896 ps
4. What is the latency of `STUR`? (Check your answer carefully. Many students place extra muxes on the critical path.)
    1. instruction memory: 250 ps
    2. register read: 130 ps
    3. Sign extend: 26 ps
    4. ALU: 110 ps
    5. Data Memory: 250
    6. summing all of these we get 516 ps
5. What is the latency of `CBZ`?

   1. Instruction Memory: 250 ps
   2. Register read: 130 ps
   3. ALU: 110 ps
   4. Adder: 90 ps
   5. summing all of these we get 580 ps
6. What is the latency of `B`?
   1. instruction memory: 250 ps
   2. adder: 90 ps
   3. adding these up we get 340 ps
7. What is the latency of an I-type instruction?
   1. instruction memory: 250 ps
   2. register read: 130 ps
   3. sign extend: 26 ps
   4. ALU: 110 ps
   5. register: 130 ps
   6. adding these up we get 646 ps
8. What is the minimum clock period for this CPU?
   1. highest ps is 896 with LDUR so that is the minimum clock period