Drew Pulian DTP180003
CS 3340.004

#1a

$\frac{4229}{4 \times 1024} = 1.03$, tag = 1 TLB: miss, Page table: valid bit = 0, page fault, page num = 28

$\frac{2447}{4096} = 0.597$, tag = 0 TLB: miss, Page Table: valid bit = 0, page fault, page # = 56

$\frac{10756}{4096} = 2.63$, tag = 2 TLB = Hit,

$\frac{35557}{4096} = 8.68$, tag = 8 TLB = miss, Page table: valid bit = 0, page fault, page # = 112

$\frac{44530}{4096} = 10.87$, tag = 10 TLB = miss, Page table: valid bit = 1, page num = 3.

$\frac{12068}{4096} = 2.94$, tag = 2 TLB = Hit

$\frac{42925}{4096} = 10.47$, tag = 10 TLB = Hit

$\frac{10756}{4096} = 2.62$, tag = 2 TLB = Hit

$\frac{35557}{4096} = 8.68$, tag = 8 TLB = Hit

Page Table

Valid	Page Num
1	56
1	28
1	5
1	6
1	9
1	11
0	Disk
1	4
1	112
0	Disk
1	3
1	12
1	7
1	8
0	Disk
0	Disk
0	Disk
0	Disk
1	2
0	Disk
1	10
1	14

TLB

Valid	Tag	Page Num	LRU
1	8	112	0
1	2	5	1
1	0	56	2
1	10	3	3

Drew Fullam DTF180003
CS 3340.004 (#16)

$\frac{4229}{2\times1024} = 2.06$, tag = 2 TLB = HIT

$\frac{2447}{2048} = 1.19$, tag = 1 TLB = miss, Page Table, valid bit = 0, page fault, page # 28

$\frac{10756}{2048} = 5.25$, tag = 5 TLB = miss, Page Table = HIT

$\frac{35557}{2048} = 17.36$, tag = 17 TLB = miss, Page Table = valid bit = 0, page fault, page # = 56

$\frac{44530}{2048} = 21.74$, tag = 21 TLB = miss, Page Table = HIT

$\frac{12068}{2048} = 5.89$, tag = 5 TLB = HIT

$\frac{40295}{2048} = 20.95$, tag = 20 TLB = miss, Page Table = HIT

$\frac{10765}{2048} = 5.25$, tag = 5 TLB = HIT

$\frac{35557}{2048} = 17.36$, tag = 17 TLB = HIT

Page Table

Valid	Page #
0	Disk
1	28
1	5
1	6
1	9
1	11
0	Disk
1	4
0	Disk
0	Disk
1	3
1	12
1	7
1	8
0	Disk
0	Disk
0	Disk
1	56
1	2
0	Disk
1	10
1	14

TLB

Valid	Tag	Page #	LRU
1	17	56	0
1	21	14	3
1	5	11	1
1	20	10	2

Drew Pallian DTP180003
CS3340.004

#2

(a) 420 ps

(b) load - store word = 945 ps
ALU = 200 ps
branches = 840 ps

#3 (a) 0.45 (b) 0.65 (c) nothing

Drew Pulliam: DTP180003

CS 3340.004

Assignment #6

Question #4

Figure 4.2 in the textbook shows all of the basic components of the MIPS architecture. It shows a PC counter that is connected to an adder to always add 4 to the program counter. This is then connected to a multiplexor that will decide whether there was a branch/jump command or whether it just needs to add 4 to the program counter.

There are also the basic instruction memory, register, and data memory blocks. The PC determines which instruction to run in instruction memory. This instruction then goes to the control unit which sets all of the required control lines. A simple example is whether we need to read or write to data memory. There are two different control lines for either option, and the control unit will determine which one to set.

The last important piece is the ALU. The ALU inputs are connected to the register block. One of the inputs is actually connected to a MUX which will allow for commands such as add immediate. The immediate number is not actually stored in a register and must be used instead of a register in that case, so the control unit determines which input to use. The ALU output is then sent to either data memory or back into the register block through use of another MUX. This is determined by the control unit again.

Question #5

A structural hazard is when instructions cannot execute because the hardware does not support that specific combination of instructions. An example would be if two separate commands try to use the ALU at the same time. This also works with other parts of hardware such as memory or registers. MIPS solves this problem by having separate data and instruction memories, which would otherwise conflict with later commands.

A data hazard occurs when a command does not have the correct data to execute. This is easily seen by two consecutive add commands. The second add command uses the result of the first as one of the arguments. The second command must wait until the first completely finishes before it can use the result. This ends up wasting multiple pipeline clock cycles.