



**Design Task: lab 2**  
**Farinaz Zahiri**

	3	4	5	6	7	8	9	10	11	12	13	14	15	16
\$s2, 0	7	0	1	0	000									
\$s1, 0x0						F	D	E	M	N	B			
\$t0, \$s2, 0						F	D	E	M	N	B			

The add includes a read after write data hazard. The third instruction will not be able to read the updated value in \$3 in the decode stage because the value of \$3 will not be written back to \$3 by that time. To solve the problem, we must first stall the ALU instruction until the load word is in the memory access phase, to ensure the correct data from memory is loaded into the register. If \$3 is read before this step, its value will likely be incorrect. Now, from the memory access step, we can forward the data in \$3 directly to the execution stage of the ALU instruction.