

1A: It perform an addi instruction. It gets its operands from the registers. The immediate is stored in one register and the other operand comes from the zero register. The result is stored in Data Memory

1B: It perform an addi instruction. It gets its operands from the registers. The immediate is stored in one register and the other operand comes from the zero register. The result is stored in Data Memory

1C: The result written into the register file is the loaded word from data memory. That is 300 sign extended (32 bit). According to the instruction it is doing a write back into \$s3

1D: Clock cycles 4,5,6 because that is where the DM reads from the EX/MEM register.

1E: It is preserved as soon as the instruction is decoding and holds till the end of the execution of the instruction. It is forwarded to that step.

2A: The alternative source would be the immediate value stored in the instruction for an I-type instructions in the last 16 bits, which is held in the ID/EX registers.

2B: What is needed is ALUSrc. It comes from the decoding of the instruction. If it is an R-Type it's a 0 and if it's I-type it's a 1.

2C:

ID/EX	9
EX/MEM	5
MEM/WB	2