The MAL Instruction Set

Format		Effect	Notes
`la`	R, label	R ← label	M[i] is the contents of the (aligned) word of memory beginning at location i. m[i] is the contents of the byte of memory at location i. address can take several forms: label—absolute address (R_b)—base address I (R_b)—base displacement
`li`	R, constant	R ← constant	
`lw`	R, address	R ← M[address]	
`lb`	R, address	R ← $(m[address]_7)^{24}$ ‖ m[address]	
`lbu`	R, address	R ← 0^{24} ‖ m[address]	
`sw`	R, address	R → M[address]	
`sb`	R, address	$[R]_{7..0}$ → m[address]	
`add`	D, S_1, S_2	D ← S_1 + S_2	D specifies a general register where the result is placed. S_1 is the contents of a general register. S_2 can be either the contents of a general register or a constant. If S_1 is not present, then S_1 is the same as D.
`sub`	D, S_1, S_2	D ← S_1 − S_2	
`mul`	D, S_1, S_2	D ← S_1 * S_2	
`div`	D, S_1, S_2	D ← S_1 div S_2	
`rem`	D, S_1, S_2	D ← S_1 rem S_2	
`and`	D, S_1, S_2	D ← S_1 **AND** S_2	
`or`	D, S_1, S_2	D ← S_1 **OR** S_2	
`xor`	D, S_1, S_2	D ← S_1 **XOR** S_2	
`nor`	D, S_1, S_2	D ← S_1 **NOR** S_2	
`not`	D, S_1	D ← **NOT** S_1	
`move`	D, S_2	D ← S_2	
`sll`	R_d, R_t, AMT	R_d ← $[R_t]_{31-AMT..0}$ ‖ 0^{AMT}	AMT may be either a general register or a constant. $0 \leq AMT < 32$.
`srl`	R_d, R_t, AMT	R_d ← 0^{AMT} ‖ $[R_t]_{31..AMT}$	
`sra`	R_d, R_t, AMT	R_d ← $([R_t]_{31})^{AMT}$ ‖ $[R_t]_{31..AMT}$	
`l.s`	F, address	F ← M[address]	F specifies a floating point register where the result is placed. W specifies a floating point register whose content is to be interpreted as a two's complement integer. F_1, F_2, and G each specify a floating point register whose content is to be interpreted as a single-precision floating point number.
`s.s`	F, address	F → M[address]	
`li.s`	F, constant	F ← constant	
`mov.s`	F, F_1	F ← F_1	
`add.s`	F, F_1, F_2	D ← F_1 + F_2	
`sub.s`	F, F_1, F_2	D ← F_1 − F_2	
`mul.s`	F, F_1, F_2	D ← F_1 * F_2	
`div.s`	F, F_1, F_2	D ← F_1 / F_2	
`cvt.s.w`	G, W	G ← W	
`cvt.w.s`	W, G	W ← G	