### Testing

### MOVX opcodes work as expected

**Purpose:** Checks for successful execution of MOVX

**Configuration:** Load any file and check for first printed instruction.

**Expected Results:** The first instruction should be MOV, with SRC of R0, and DST of R0.

**Actual results:** The instructions print as expected.

A black background with white text

Description automatically generated

### Prints first real instruction at the correct memory start address

**Purpose:** Checks for successful recognition of program memory address and prints first instruction after “MOV R0, R0” at that address.

**Configuration:** Load any file and check for second printed instruction.

**Expected Results:** The first instruction should be MOV, with SRC of R0, and DST of R0.

**Actual results:** The instructions print as expected.

A screen shot of a computer

Description automatically generated

### Recognizes end of program and prints instruction hexadecimal 0000

**Purpose:** Checks for successful recognition of end of program by printing 0000 after completion of reading the program.

**Configuration:** Load any file and check that the instruction hexadecimal 0000 is printed after the last command.

A screenshot of a computer screen

Description automatically generated

**Expected Results:** The 0000 instruction bytes will be printed at memory address 101C, after instruction 3FFF.

**Actual results:** The instructions print as expected,

A black screen with a black background

Description automatically generated

### Recognizes all commands from MOVL to MOVH

**Purpose:** Checks for if the program recognizes MOVL to MOVH instructions’ mnemonics and operands.

**Configuration:** Load a .xme file with MOVL to MOVH instructions.

A computer screen shot of numbers

Description automatically generated

**Expected Results:** The memory will print all expected instruction mnemonics and operands.

**Actual results:** The instructions print as expected.

A screenshot of a computer program

Description automatically generated

### Recognizes all commands from MOV to SWAP

**Purpose:** Checks for if the loader recognizes MOV to SWAP’s instruction mnemonics and operands.

**Configuration:** Load a .xme file with MOV to SWAP instructions.

A black text on a white background

Description automatically generated

**Expected Results:** The memory will print all expected instruction mnemonics and operands.

**Actual results:** The instructions print as expected.

A screen shot of a computer

Description automatically generated

### Recognizes all commands from SRA to RRC

**Purpose:** Checks for if the loader recognizes SRA to RRC’s instruction mnemonics and operands.

**Configuration:** Load a .xme file with SRA to RRC instructions.

A close up of a white background

Description automatically generated

**Expected Results:** The memory will print all expected instruction mnemonics and operands.

**Actual results:** The instructions print as expected.

A screen shot of a computer

Description automatically generated

### Recognizes all commands from SWPB to SXT

**Purpose:** Checks for if the loader recognizes SWPB to SXT’s instruction mnemonics and operands.

**Configuration:** Load a .xme file with SWPB to SXT instructions.

A black text on a white background

Description automatically generated

**Expected Results:** The memory will print all expected instruction mnemonics and operands.

**Actual results:** The instructions print as expected.

A screen shot of a computer

Description automatically generated

### Recognizes all commands from ADD to BIS

**Purpose:** Checks for if the loader recognizes ADD to BIS’s instruction mnemonics and operands.

**Configuration:** Load a .xme file with ADD to BIS instructions.

A computer screen shot of a computer code

Description automatically generated

**Expected Results:** The memory will print all expected instruction mnemonics and operands.

**Actual results:** The instructions print as expected.

A screenshot of a computer

Description automatically generated

### Recognizes invalid commands

**Purpose:** Checks for if the loader recognizes LD to ST as invalid commands

**Configuration:** Load a .xme file with LD to ST instructions.

A computer screen shot of a number

Description automatically generated

**Expected Results:** The memory will print the instruction opcode instead of the instruction mnemonic and operands.

**Actual results:** The instruction opcodes print as expected.

A screen shot of a computer

Description automatically generated

### Can recognize all 4 registers

**Purpose:** Checks for if the loader can recognize R0, R1, R2, and R3.

**Configuration:** Load a .xme file that uses all four registers.

A close-up of a white background

Description automatically generated

**Expected Results:** The loader will recognize all four registers.

**Actual results:** The loader behaved as expected.

A screen shot of a computer

Description automatically generated

### Can recognize all constants in constant array

**Purpose:** Checks for if the loader can recognize valid constant values of 0, 1, 2, 4, 8, 16, 32, -1.

**Configuration:** Load a .xme file that uses all valid constant values.

A close up of a white background

Description automatically generated with medium confidence

**Expected Results:** The loader will print a success statement after xme file is loaded.

**Actual results:** The loader behaves as expected.

A screenshot of a computer

Description automatically generated

### Can print more than one file’s opcode and operands

**Purpose:** Checks for the loader successfully printing more than one file’s instruction mnemonic and operands, one after the other.

**Configuration:** Two files are loaded in.

**Expected Results:** The loader will be able to successfully print both file’s instruction mnemonic and operands, one after the other.

**Actual results:** The loader behaves as expected.

A screen shot of a computer

Description automatically generated