# Appendix C Instruction Set

The following table is from Appendix C of the text. It is included here for your reference. **The questions in this lab refer to this table as the *“language description table.”***

Op-

code Operand Description

1 RXY LOAD the register R with the bit pattern found in the memory cell whose address is XY. *Example:* 14A3 would cause the contents of the memory cell located at address A3 to be placed

in register 4.

2 RXY LOAD the register R with the bit pattern XY.

*Example:* 20A3 would cause the value A3 to be placed in register 0.

3 RXY STORE the bit pattern found in register R in the memory cell whose address is XY.

*Example:* 35B1 would cause the contents of register 5 to be placed in the memory cell whose

address is B1.

4 0RS MOVE the bit pattern found in register R to register S.

*Example:* 40A4 would cause the contents of register A to be copied into register 4.

5 RST ADD the bit patterns in registers S and T as though they were two’s complement representations

and leave the result in register R.

*Example:* 5726 would cause the binary values in registers 2 and 6 to be added and the sum placed

in register 7.

6 RST ADD the bit patterns in registers S and T as though they represented values in floating-point

notation and leave the floating-point result in register R.

*Example:* 634E would cause the values in registers 4 and E to be added as floating-point values

and the result to be placed in register 3.

7 RST OR the bit patterns in registers S and T and place the result in register R.

*Example:* 7CB4 would cause the result of ORing the contents of registers B and 4 to be placed in

register C.

8 RST AND the bit patterns in register S and T and place the result in register R.

*Example:* 8045 would cause the result of ANDing the contents of registers 4 and 5 to be placed in

register 0.

9 RST EXCLUSIVE OR the bit patterns in registers S and T and place the result in register R.

*Example:* 95F3 would cause the result of EXCLUSIVE ORing the contents of registers F and 3 to

be placed in register 5.

A R0X ROTATE the bit pattern in register R one bit to the right X times. Each time place the bit that

started at the low-order end at the high-order end.

*Example:* A403 would cause the contents of register 4 to be rotated 3 bits to the right in a circular

fashion.

B RXY JUMP to the instruction located in the memory cell at address XY if the bit pattern in register R

is equal to the bit pattern in register number 0. Otherwise, continue with the normal sequence of

execution. (The jump is implemented by copying XY into the program counter during the execute

phase.)

*Example:* B43C would first compare the contents of register 4 with the contents of register 0. If

the two were equal, the pattern 3C would be placed in the program counter so that the next

instruction executed would be the one located at that memory address. Otherwise, nothing would

be done and program execution would continue in its normal sequence.

C 000 HALT execution.

*Example:* C000 would cause program execution to stop.

**NOTE: the first number is the op-code. An example 6xxx would be an ADD operation (see table above)**

***Each instruction requires two bytes***