

of that code, and the routine is terminated at 00F0 when RA.1 goes to 0A, past the last byte of text.

In the meantime, comment lines are again ignored (@ 009D is the character ";" 3B). Labels are ignored during the second pass -- their usefulness only extended to the creation of the symbol table during pass 1. Therefore the first character of the instruction field is immediately examined @ 00A0 provided the line is not a comment line.

If the instruction field contains an ASCII space (20) the next line is examined. If it is an "R", reserve memory is called to accomodate the pseudo instruction RMKK. (The M is completely inconsequential -- change it if you like.) At 00B4, the four-character instruction is placed in the object code. At this point the instruction is only an embryo. If it is of the addressing type, it will still be incomplete -- the address has not been inserted yet.

Following any instruction, data, or pseudo instruction, the character in position #11 is examined @ 00B7. If a semicolon exists there, the rest of the line is ignored. This will occur regardless of the instruction type. If a semicolon does not exist after an instruction the program checks if an argument is entered starting at position #12. If a space is in