Superscalar Problem

Consider the computation \(((M_1 \times M_2) + (M_3 \times M_4))/(M_5 + M_6)\), where \(M_1, \ldots\) denote the content of memory locations.

1. Write a RISC assembly program for this computation. Use a minimal number of registers.

2. Identify the (true and false) dependencies in your code.

3. Show the execution of your program on a superscalar processor. There is an instruction window where all the fetched and decoded instructions can be stored. The remaining pipeline stages are RR, EX and WB. Assume that there are four fetch units, two decode units, two units for reading the registers, two ALUs and two units writing to the registers. Data movement between the CPU and memory happens in the EX stage, and there are MBRs where data can be stored temporarily before moving them into the destination registers.

4. Remove the false dependencies by using register renaming.

5. Draw a diagram showing the execution of the modified code on the above superscalar processor.