ECE 483 Project 1 (Stack Implementation)

ECP Educational Computer Products is planning on releasing a computer system based on the SEP to the educational market for use in introductory software and hardware classes. However, one of the central concepts taught in second level classes is the concept of the stack. As it stands the SEP lacks the hardware to properly implement a stack.

In order to properly implement a stack, a stack pointer (SP) register must be added to the SEP. This register will be used by two new instructions, PSH and POP. The stack should grow down in memory (e.g. pushing data on the stack will decrement the stack pointer, and popping data off will increment the stack pointer). These instructions will have the effect as follows

PSH: \( M[SP] = AC \), Decrement SP
POP: \( AC = M[SP] \), Increment SP

Also, in order to manage the stack properly, another two instructions must be added: Save Stack Pointer(SSP), and Load Stack Pointer(LSP). These instructions need not deal with memory indirections (e.g. operand = \( M[M[address]] \)). These two instructions have the following effects

SSP: \( M[accumulator] = SP \)
LSP: \( SP = M[accumulator] \)

In addition, any memory references to the address 0xFFF should use the value of the stack pointer in its place. For example

ADD $FFF which usually performs
\( AC = AC + M[$FFF] \)
Should actually perform
\( AC = AC + M[SP] \)

Your job, should you choose to accept the challenge is to implement the above scheme in the SEP. You should design your modified SEP in Verilog. You should also write a report and give a presentation that gives specific details of your changes.