**Problem Statement**

Designing of a 32-bit RISC processor that will support the following assembly instruction:

* MOVE Ri, Rj: The content of Rj is transferred to Ri.
* MOVE Ri, Immediate (16 – bit): The immediate value (32 – bit unsigned extended) will be transferred to Ri.
* LOAD Ri, X(Rj): The content of memory location [ [ Rj ] + X ] is loaded into Ri, where X is a 16-bit unsigned immediate value.
* STORE Ri, X(Rj): The content of register Ri is stored in memory [ [ Rj ] + X ], where X is a 16 – bit unsigned immediate value.
* ADD Ri, Rj, Rk: Ri = Rj + Rk
* ADI Ri, Rj, Immediate (16 – bit): Ri = Rj + Immediate value (32 – bit unsigned extended)
* SUB Ri, Rj, Rk: Ri = Rj – Rk
* SUI Ri, Rj, Immediate (16 – bit): Ri = Rj – Immediate value (32 – bit unsigned extended)
* AND Ri, Rj, Rk: Ri = Rj AND Rk
* ANI Ri, Rj, Immediate (16 – bit): Ri = Rj AND Immediate value (32 – bit unsigned extended)
* OR Ri, Rj, Rk: Ri = Rj OR Rk
* ORI Ri, Rj, Immediate (16 – bit): Ri = Rj OR Immediate value (32 – bit unsigned extended)
* HLT: Stops the execution

Instruction Encoding: