Design Name: AVS

Instruction	**Functionality**	**Opcode**
init *Rx, imm*	Rx = imm	000 xx ii
base *Rx, [Mem]*	Rx = [Mem]?	001 xx yy
load *imm*	$R1 = imm	100 11 ii
store *Rx, imm*	Mem[imm] = Rx	011 xx ii
shl *Rx*	Rx shift left one bit, 0 shifted into LSB	100 00 xx
sll *Rx, Ry*	Rx = Rx * (2^Ry)	100 xx yy
slt *Rx, Ry*	$R0 = 1 if Rx < Ry	101 xx yy
BezDec *imm*	If $R0 == 0, then PC = PC + imm, else $R0 = $R0 – 1, PC = PC + 1	100 01 ii
BnezDec *imm*	If $R0 != 0, then PC = PC + imm, else $R0 = $R0 – 1, PC = PC + 1	100 10 ii
xori *Rx, imm*	$R0 = Rx (EXCL) with imm	110 imm
andi *Rx, imm*	$R0 = Rx (AND) with imm	111 xx ii
jump *‘branch’*	PC = PC + imm	010 iiii
addi *Rx, imm*	Rx = Rx + imm	001 xx yy
halt	Stop	000 00 00

Machine Code for Program 1: