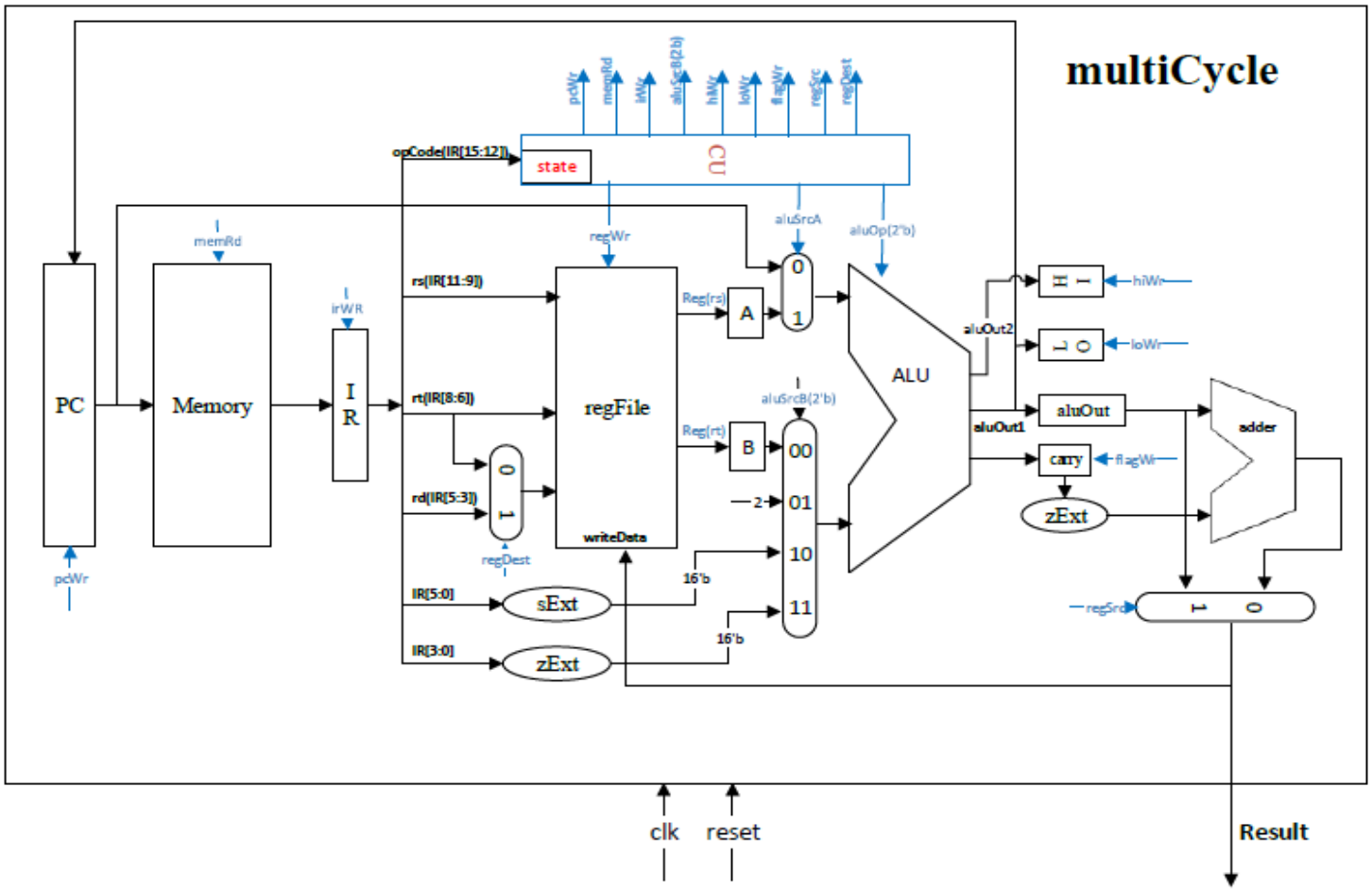
Multi-cycle Architecture

The following design was implemented using verilog HDL in Xilinx 14.7 software

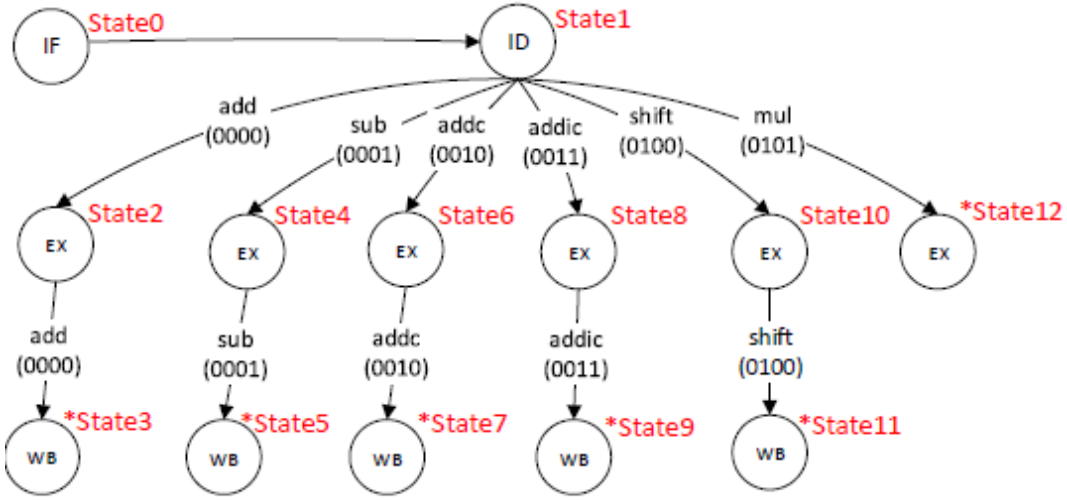


Instruction Format

Instruction	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
add	0	0	0	0	rs			rt			rd			X	X	X
sub	0	0	0	1	rs			rt			rd			X	X	X
addc	0	0	1	0	rs			rt			rd			X	X	X
addic	0	0	1	1	rs			rt			imm(6bits)					
shift	0	1	0	0	rs			rt			X	X	shiftAmt(4bits)			
mul	0	1	0	1	rs			rt			X	X	X	X	X	X

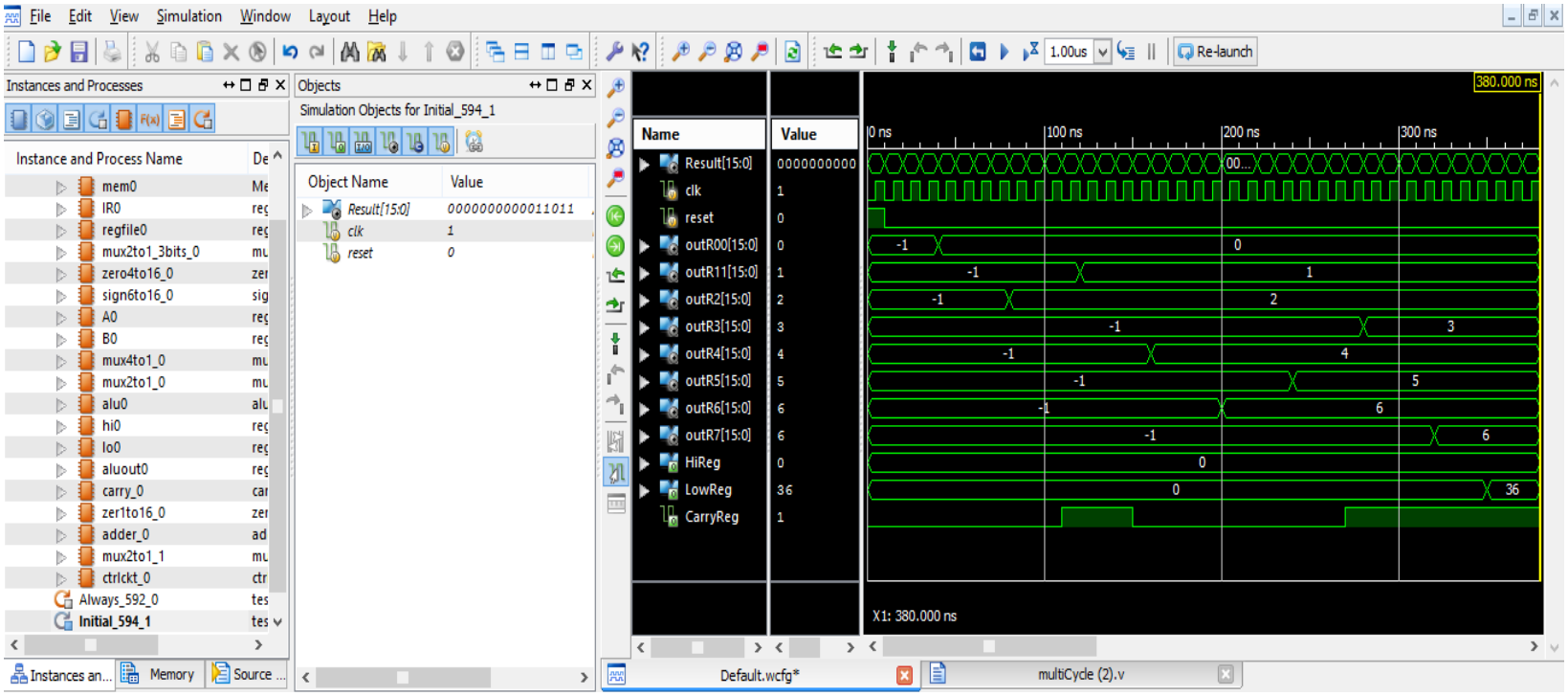
Control circuit

State Diagram :



State#	Stage	Control Signals											
		pcWr	irWr	memRd	regDest	regWr	regSrc	aluSrcA	aluSrcB	aluOp	hiWr	loWr	flagWr
State0	IF	1	1	1	0	0	0	0	01	00	0	0	0
State1	ID	0	0	0	0	0	0	0	00	00	0	0	0
State2	EX(add)	0	0	0	0	0	0	1	00	00	0	0	1
State3	WB(add)	0	0	0	1	1	1	0	00	00	0	0	0
State4	EX(sub)	0	0	0	0	0	0	1	00	01	0	0	1
State5	WB(sub)	0	0	0	1	1	1	0	00	00	0	0	0
State6	EX(addc)	0	0	0	0	0	0	1	00	00	0	0	1
State7	WB(addc)	0	0	0	1	1	0	0	00	00	0	0	0
State8	EX(addic)	0	0	0	0	0	0	1	10	00	0	0	1
State9	WB(addic)	0	0	0	0	1	0	0	00	00	0	0	0
State10	EX(shift)	0	0	0	0	0	0	1	11	10	0	0	1
State11	WB(shift)	0	0	0	0	1	1	0	00	00	0	0	0
State12	EX(mul)	0	0	0	0	0	0	1	00	11	1	1	0

WaveForm:


File Edit View Simulation Window Layout Help
Re-launch
Instances and Processes
Instance and Process Name
mem0
IR0
regfile0
mux2to1_3bits_0
zero4to16_0
sign6to16_0
A0
B0
mux4to1_0
mux2to1_0
alu0
hi0
lo0
aluout0
carry_0
zer1to16_0
adder_0
mux2to1_1
ctrlckt_0
Always_592_0
Initial_594_1
Objects
Simulation Objects for Initial_594_1
Object Name
Value
Result[15:0]
0000000000011011
clk
reset
Name
Value
Result[15:0]
clk
reset
outR00[15:0]
outR11[15:0]
outR2[15:0]
outR3[15:0]
outR4[15:0]
outR5[15:0]
outR6[15:0]
outR7[15:0]
HiReg
LowReg
CarryReg
380.000 ns
0 ns
100 ns
200 ns
300 ns
X1: 380.000 ns
Instances an...
Memory
Source ...
Default.wcfg*
multiCycle (2).v