Description of Control Signals in Single Cycle Implementation of the LC4 ISA

Signal Name	# of bits	Value	Action
PCMux.CTL	3	0	Value of NZP register compared to bits I[11:9] of the current instruction if the test is satisfied then the output of TEST is 1 and NextPC = BRANCH Target, (PC+1) + SEXT(IMM9); otherwise the output of TEST is 0 and NextPC = PC + 1
		1	Next PC = PC+1
		2	Next PC = (PC+1) + SEXT(IMM11)
		3	Next PC = RS
		4	Next PC = (0x8000 | UIMM8)
		5	Next PC = (PC & 0x8000) | (IMM11 << 4)
rsMux.CTL	2	0	rs.addr = I[8:6]
		1	rs.addr = 0x07
		2	rs.addr = I[11:9]
rtMux.CTL	1	0	rt.addr = I[2:0]
		1	rt.addr = I[11:9]
rdMux.CTL	1	0	rd.addr = I[11:9]
		1	rd.addr = 0x07
regFile.WE	1	0	Register file not written
		1	Register file written: rd.addr indicates which register is updated with the value on the Write Input
regInput.Mux.CTL	2	0	Write Input = ALU output
		1	Write Input = Output of Data Memory
		2	Write Input = PC + 1
Arith.CTL	3	0	Arithmetic operation ADD
		1	Arithmetic operation MULTIPLY
		2	Arithmetic operation SUBTRACT
		3	Arithmetic operation DIVIDE
		4	Arithmetic operation MODULUS
ArithMux.CTL	2	0	Arithmetic input = RT
		1	Arithmetic input = SEXT(IMM5)
		2	Arithmetic input = SEXT(IMM6)
LOGIC.CTL	2	0	Logical operation AND
		1	Logical operation NOT (invert top input to logical unit)
		2	Logical operation OR
		3	Logical operation XOR
LogicMux.CTL	1	0	Logical input = RT
		1	Logical input = SEXT(IMM5)
SHIFT.CTL	2	0	Shift operation = SLL (Shift Left Logical)
		1	Shift operation = SRA (Shift Right Arithmetic)
		2	Shift operation = SRL (Shift Right Logical)
CONST.CTL	1	0	Constants output = SEXT(IMM9)
		1	Constants output = (RS & xFF) | (UIMM8 <<8)

CMP.CTL	2	0	Out = signed-CC(Rs-Rt) [-1, 0, +1]
		1	Out = unsigned-CC(Rs-Rt) [-1, 0, +1]
		2	Out = signed-CC(Rs – SEXT(IMM7)) [-1, 0, +1]
		3	Out = unsigned-CC(RS – UIMM7) [-1, 0, +1]
ALUMux.CTL	3	0	ALUOutput = Arithmetic output
		1	ALUOutput = Logical output
		2	ALUOutput = Shifter output
		3	ALUOutput = Constants output
		4	ALUOutput = Comparator output
NZP.WE	1	0	NZP register not updated
		1	NZP register updated from Write Input to register file
DATA.WE	1	0	Data Memory not written
		1	Data Input written into location on Data Address lines
Privilege.CTL	2	0	Set privilege bit PSR[15] to 0 – user mode
		1	Set privilege bit PSR[15] to 1 – os mode
		2	Leave privilege bit unchanged