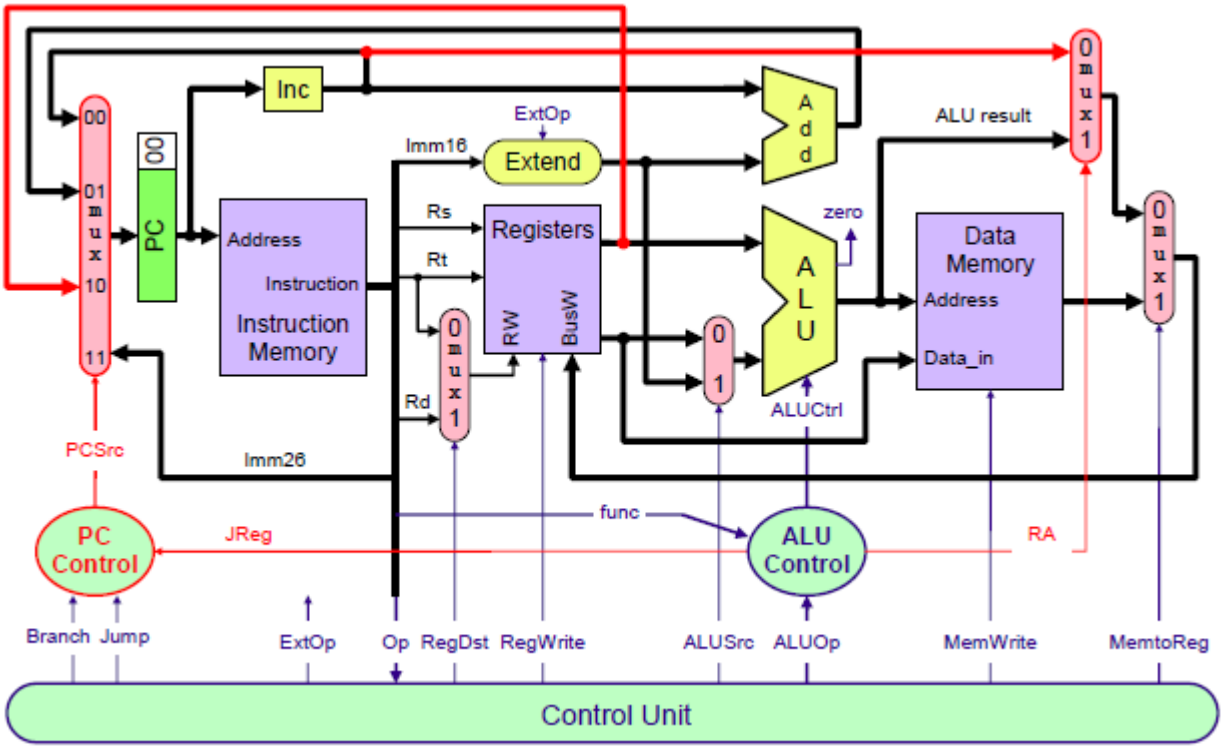
Q1. Single-Cycle MIPS Processor

We wish to add the instruction **jalr** (jump and link register) to the single-cycle datapath. The jump and link register instruction is described below:

```
jalr rd, rs # rd = pc + 4 , pc = rs
```

$op^6 = 0$	rs^5	0	rd^5	0	$Funct^6 = 0x9$

a) Add any necessary datapath and control signals and draw the result datapath. You should only add wires, gates, muxes to the datapath; do not modify the main functional units (the memory, register file, and ALU) themselves. Try to keep your diagram neat!



The necessary changes to the datapath and control:

For the datapath, we need a bigger 4-input multiplexer at the input of the PC. The first input is used to increment the PC. The second input is used for taken branches, where the branch target is PC-relative. The third input is used to jump register, where the input to the PC comes from a general-purpose register, and the fourth input is used for jump instructions.

For the implementation of the JALR instruction: to jump to register 'Rs', we need to add a path from the output of register Rs (first ALU input) back to the PC multiplexer input. PC control unit needs to be updated by adding an input control signal JReg (Jump Register) to select PC according to the value of register Rs. JReg is generated by the ALU control unit, since JALR is a R-type instruction and JReg depends on the function field only. When JReg is equal to '1', PCSrc (PC control unit output control signal) will be '10' to select the value of register Rs as input to PC.