



ORGANIZACIJA RAČUNALNIKOV

MiMo – Izvajanje strojnega ukaza
JNEZ Rs,immed

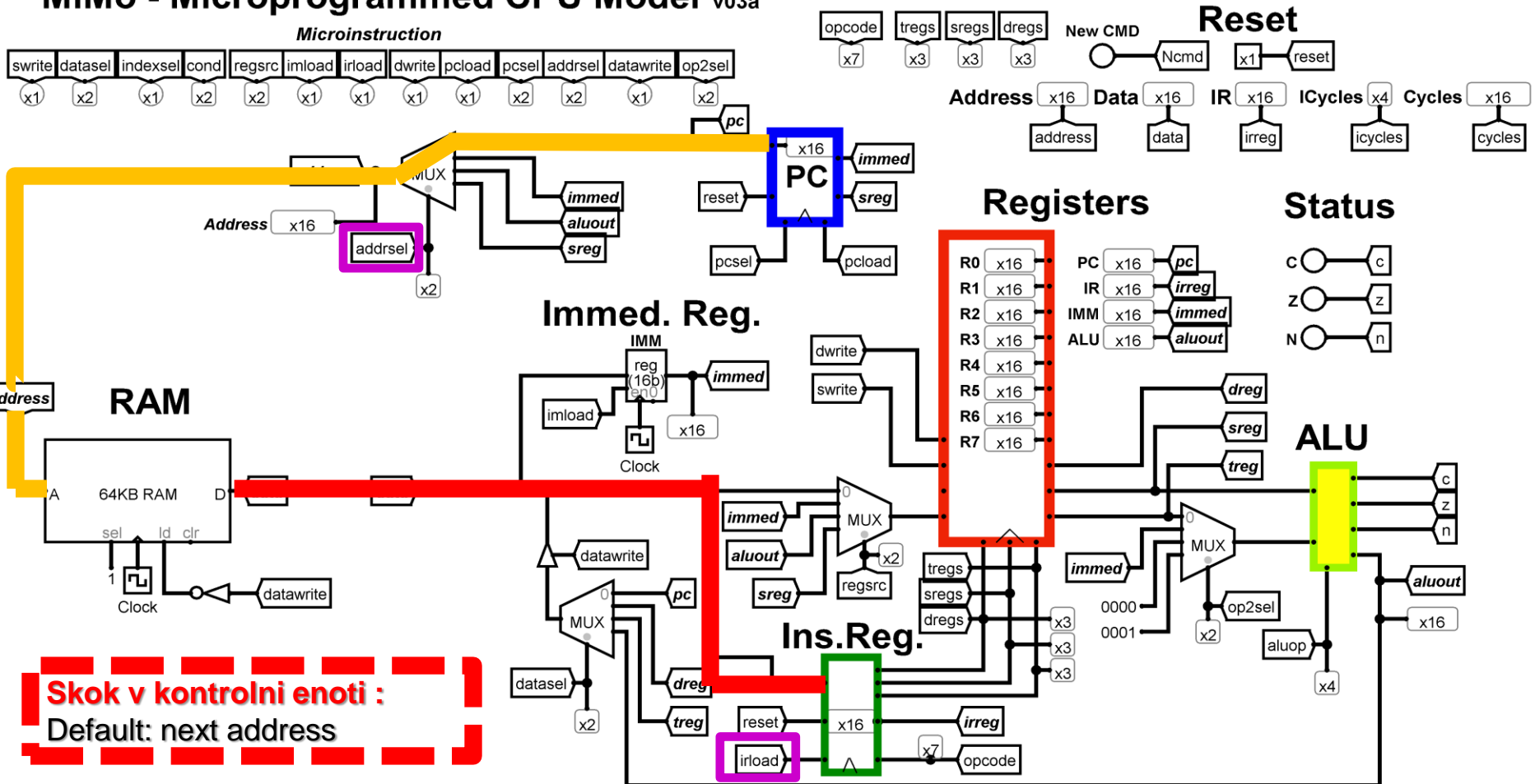
Primer programa v zbirniku :

```
main: li    r0, 0           # r0 is the running sum
      li    r1, 100        # r1 is the counter
      li    r2, -1        # Used to decrement r1
loop: add   r0, r0, r1     # r0= r0 + r1
      add   r1, r1, r2     # r1--
      jnez  r1, loop      # loop if r1 != 0
      sw    r0, 256       # Save the result
```

JNEZ Rs,immed:

fetch:	addrsel=pc irload=1	# Address=PC, Load IR register
40:	pload=1 ptsel=pc, opcode_jump	# PC=PC+1, jump to 2+OPC
	addrsel=pc imload=1	# Read Immediate operand -> IMRegister
	aluop=sub op2sel=const0, if z then pcincr else jump	# ALU: Rs-0, If z then pcincr else jump
pcincr:	pload=1 ptsel=pc, goto fetch	# Increment PC and goto new command;
jump:	pload=1 ptsel=immed, goto fetch	# Set address to immed and goto new command

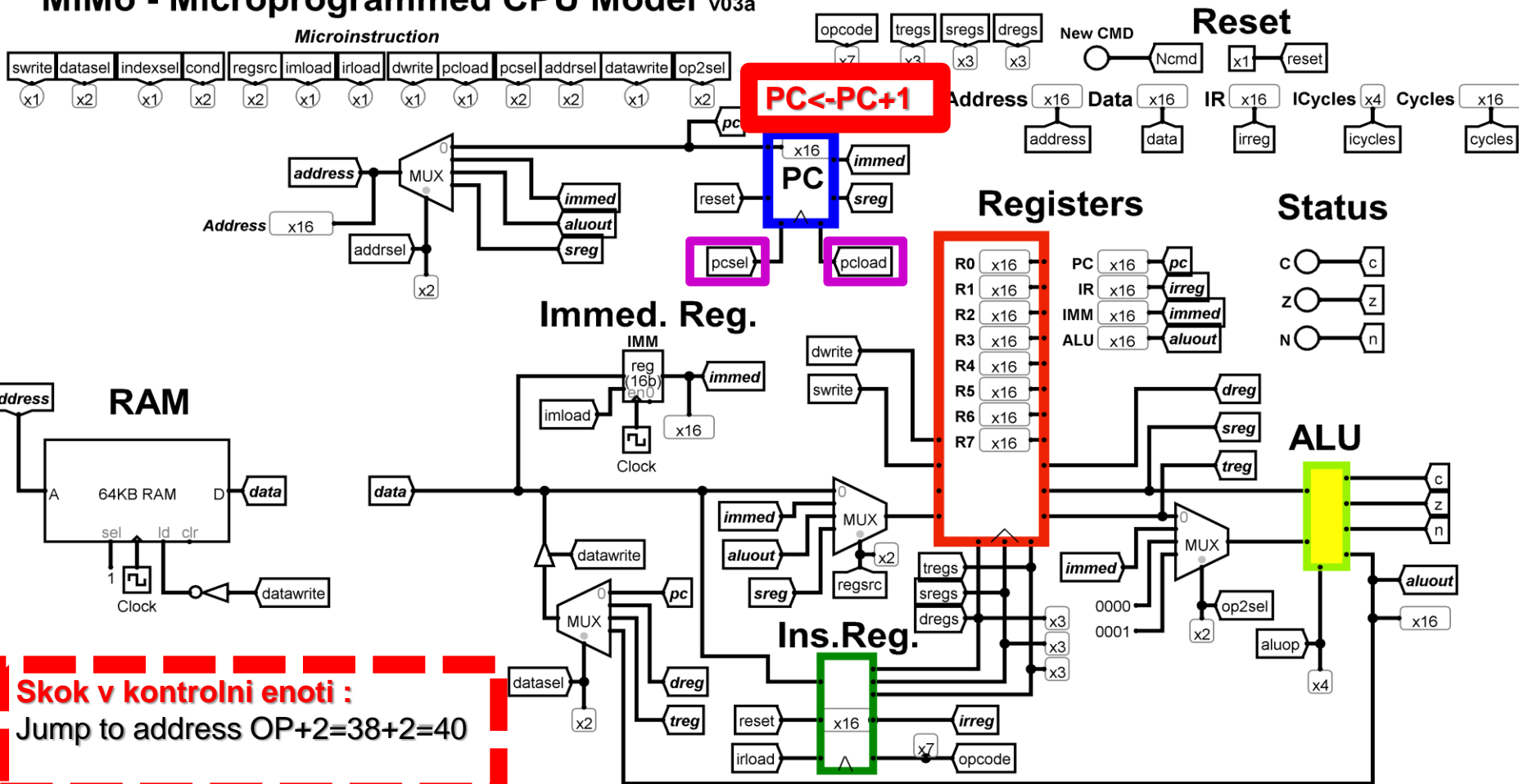
MiMo - Microprogrammed CPU Model v03a



JNEZ Rs,immed:

fetch:	addrsel=pc irload=1	# Address=PC, Load IR register
	pload=1 ptsel=pc, opcode_jump	# PC=PC+1, jump to 2+OPC
40:	addrsel=pc imload=1	# Read Immediate operand -> IMRegister
	aluop=sub op2sel=const0, if z then pcincr else jump	# ALU: Rs-0, If z then pcincr else jump
pcincr:	pload=1 ptsel=pc, goto fetch	# Increment PC and goto new command;
jump:	pload=1 ptsel=immed, goto fetch	# Set address to immed and goto new command

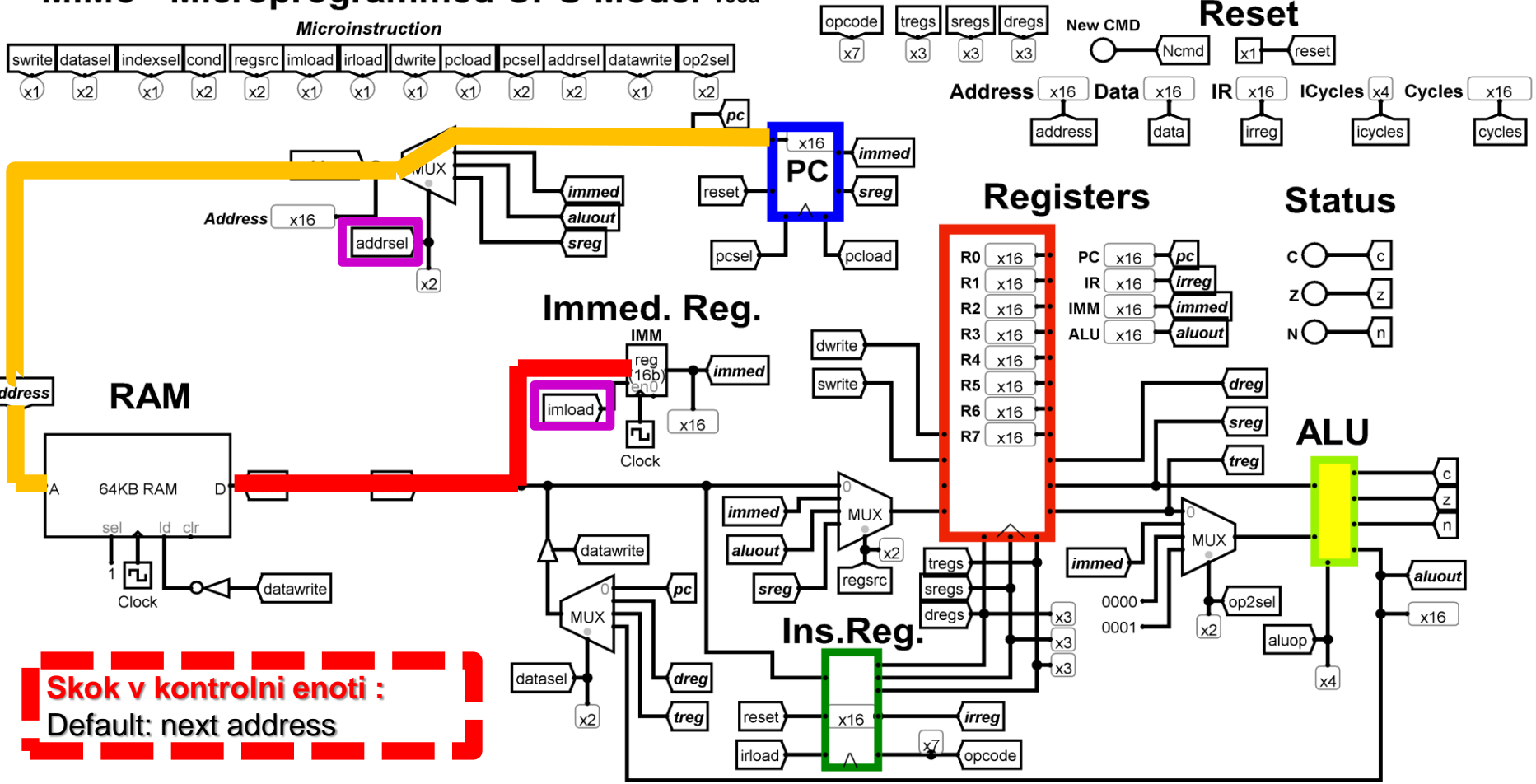
MiMo - Microprogrammed CPU Model v03a



JNEZ Rs,immed:

fetch:	addrsel=pc irload=1	# Address=PC, Load IR register
	pload=1 ptsel=pc, opcode_jump	# PC=PC+1, jump to 2+OPC
40:	addrsel=pc imload=1	# Read Immediate operand -> IMRegister
	aluop=sub op2sel=const0, if z then pcincr else jump	# ALU: Rs-0, If z then pcincr else jump
pcincr:	pload=1 ptsel=pc, goto fetch	# Increment PC and goto new command;
jump:	pload=1 ptsel=immed, goto fetch	# Set address to immed and goto new command

MiMo - Microprogrammed CPU Model v03a

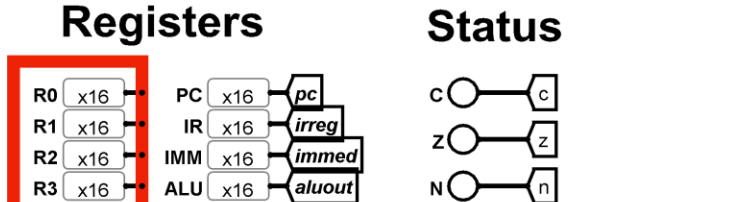
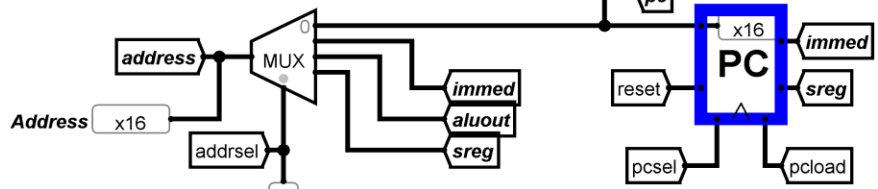
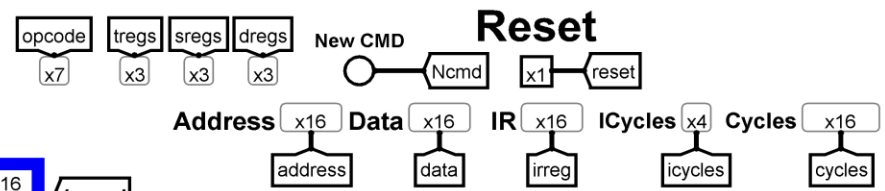
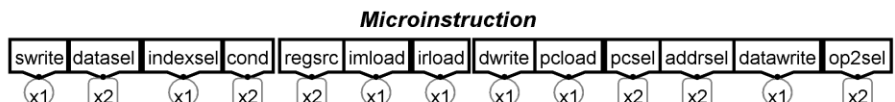


JNEZ Rs,immed:

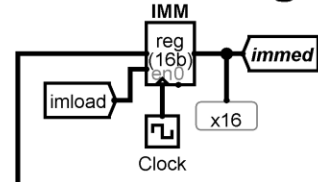
fetch: addrsel=pc irload=1
 pload=1 ptsel=pc, opcode_jump
 40: addrsel=pc imload=1
 aluop=sub op2sel=const0, if z then pcincr else jump
 pcincr: pload=1 ptsel=pc, goto fetch
 jump: pload=1 ptsel=immed, goto fetch

Address=PC, Load IR register
 # PC=PC+1, jump to 2+OPC
 # Read Immediate operand -> IMRegister
 # ALU: Rs-0, If z then pcincr else jump
 # Increment PC and goto new command;
 # Set address to immed and goto new command

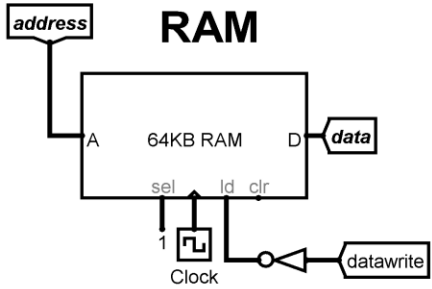
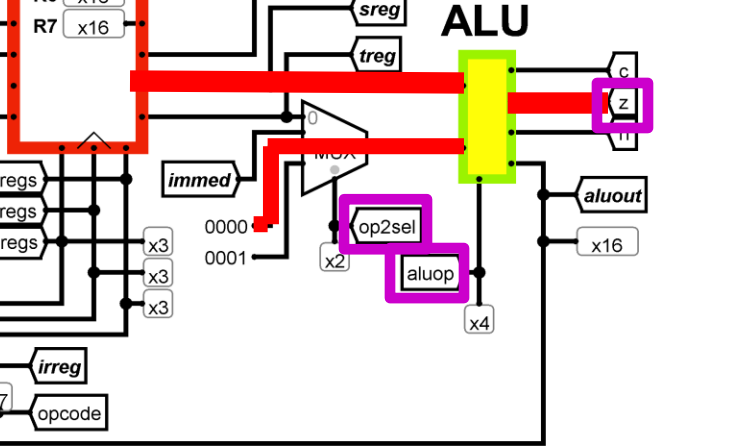
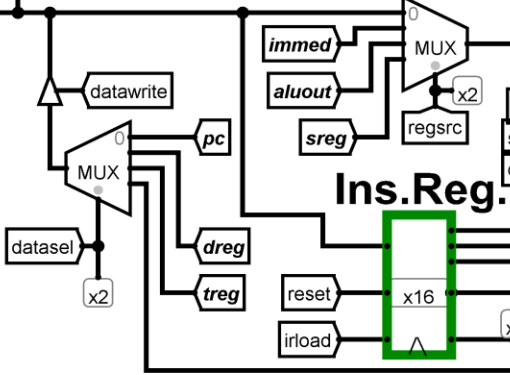
MiMo - Microprogrammed CPU Model v03a



Immed. Reg.



Ins.Reg.

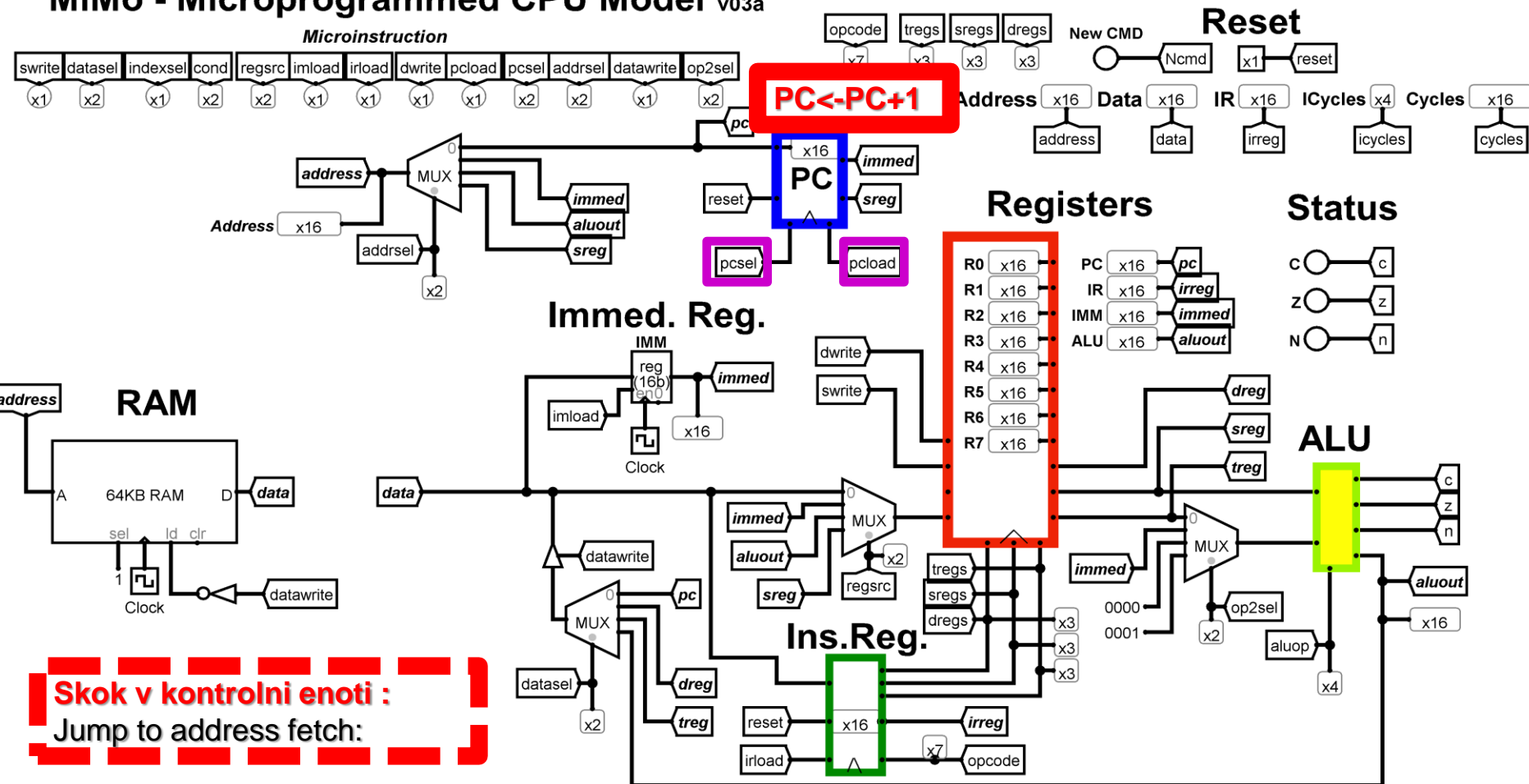


Skok v kontrolni enoti :
 if z
 then pcincr
 else jump

JNEZ Rs,immed: velja Rs=0

fetch:	addrsel=pc irload=1	# Address=PC, Load IR register
	pload=1 ptsel=pc, opcode_jump	# PC=PC+1, jump to 2+OPC
40:	addrsel=pc imload=1	# Read Immediate operand -> IMRegister
	aluop=sub op2sel=const0, if z then pcincr else jump	# ALU: Rs-0, If z then pcincr else jump
pcincr:	pload=1 ptsel=pc, goto fetch	# Increment PC and goto new command;
jump:	pload=1 ptsel=immed, goto fetch	# Set address to immed and goto new command

MiMo - Microprogrammed CPU Model v03a



Skok v kontrolni enoti :
Jump to address fetch:

JNEZ Rs,immed: velja Rs≠0

fetch:	addrsel=pc irload=1	# Address=PC, Load IR register
	pload=1 ptsel=pc, opcode_jump	# PC=PC+1, jump to 2+OPC
40:	addrsel=pc imload=1	# Read Immediate operand -> IMRegister
	aluop=sub op2sel=const0, if z then pcincr else jump	# ALU: Rs-0, If z then pcincr else jump
pcincr:	pload=1 ptsel=pc, goto fetch	# Increment PC and goto new command;
jump:	pload=1 ptsel=immed, goto fetch	# Set address to immed and goto new command

MiMo - Microprogrammed CPU Model v03a

