

# CO-OP 999

## Mini CPU

เข้าสู่ระบบ

ลงทะเบียน

Mini CPU created by using Digital.exe to run following op-code.

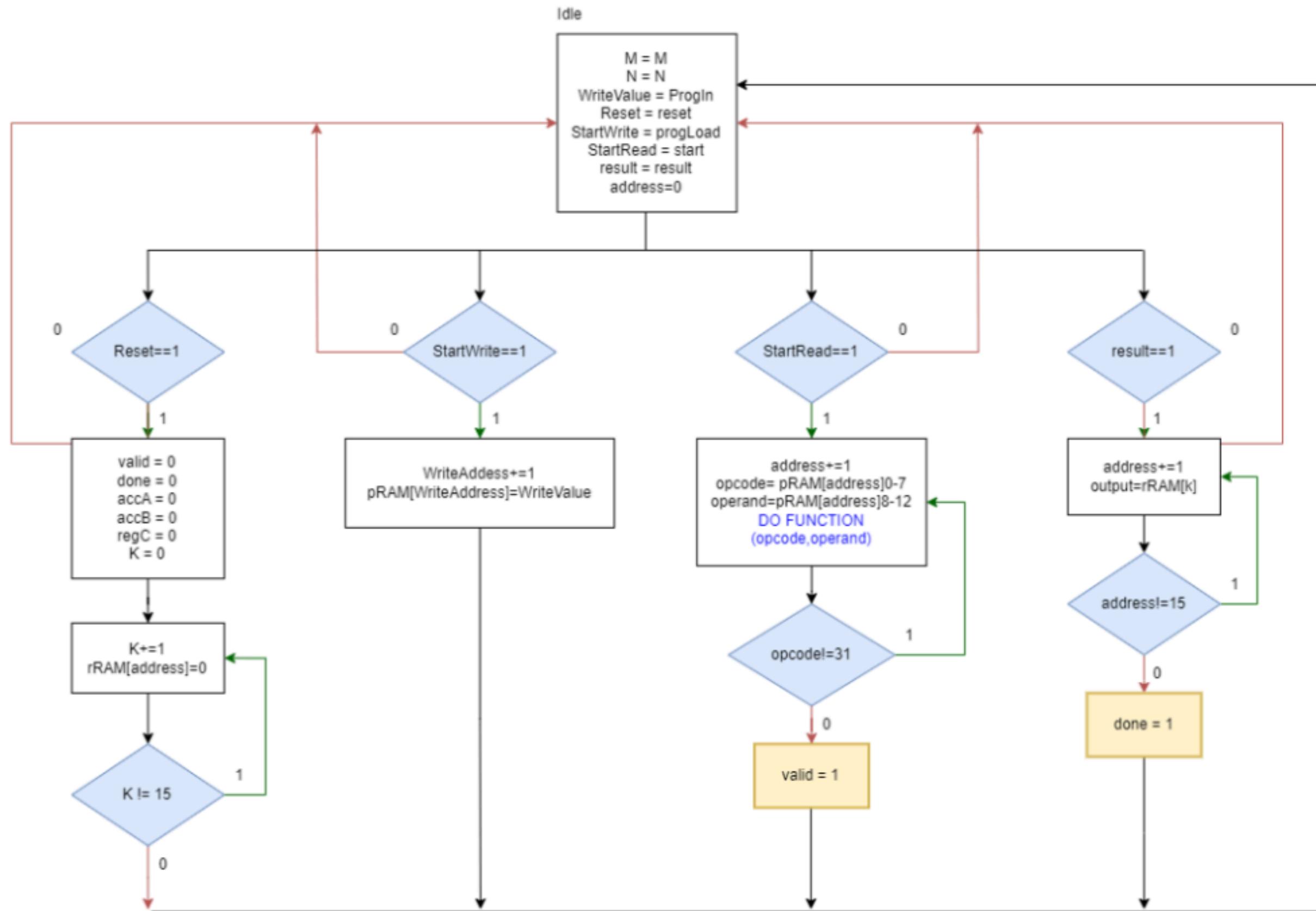
The components M, N, prog IN, reset, progLoad, start, result, and clk are inputs. Done, valid, output, 7-segment, and rRam are outputs.

So, we designed to divide the functions into 13 parts, and write an ASM chart to check the steps we should follow according to the operations of the CPU

# ASM CHART CO-OP999

เข้าสู่ระบบ

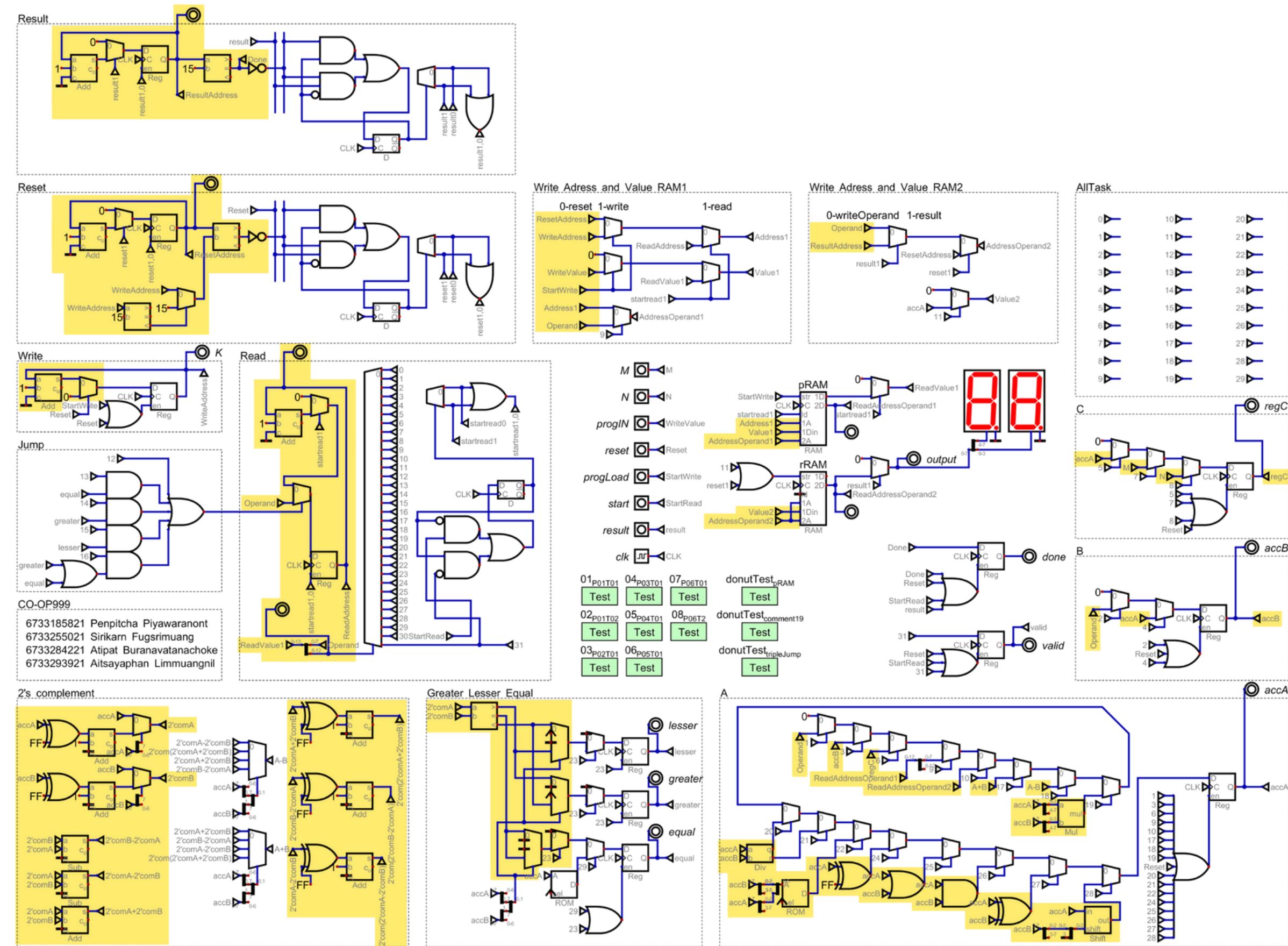
ลงทะเบียน



# DATA PATH CO-OP999

ເຂົ້າສູ່ຮະບບ

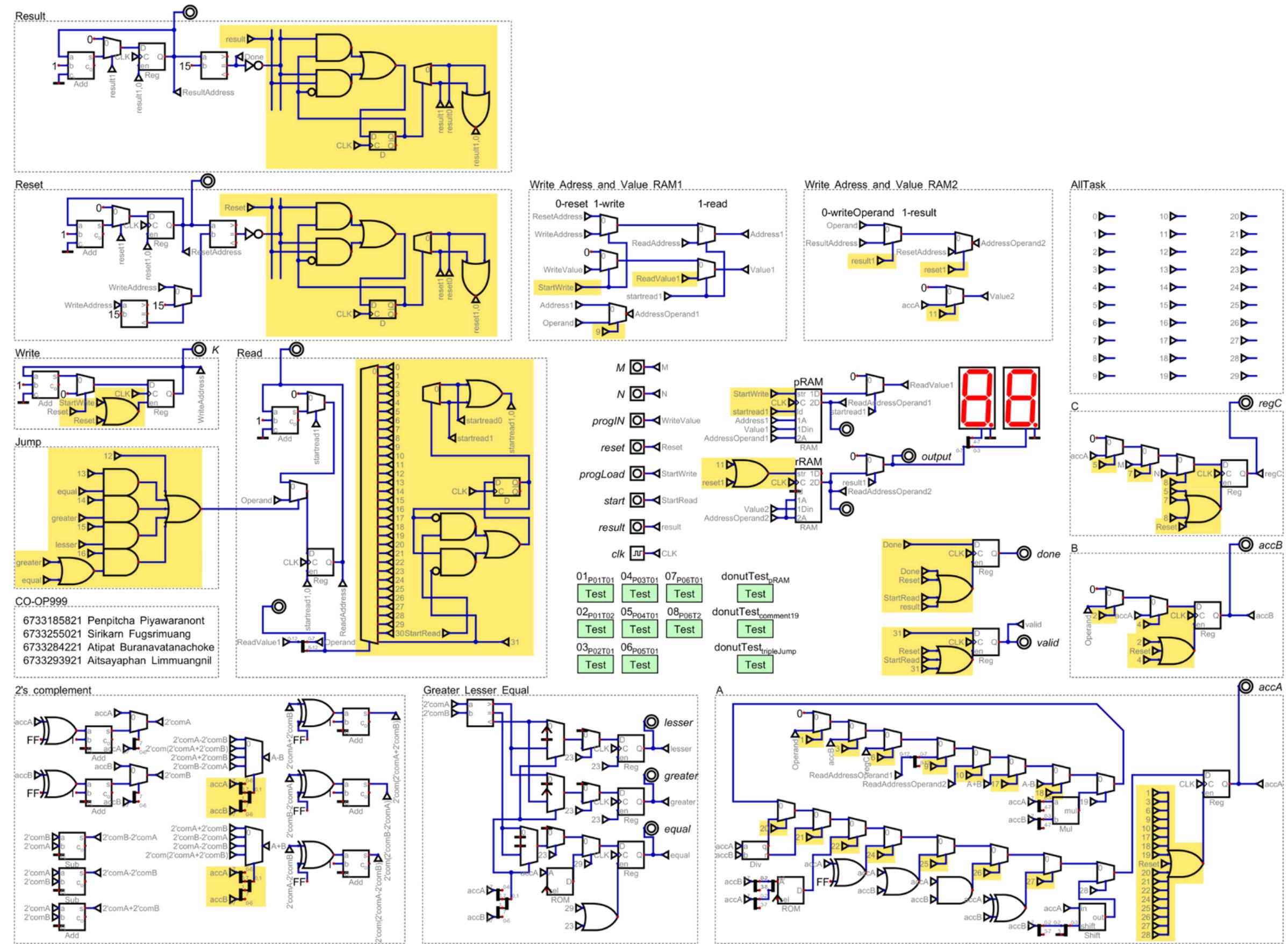
ລາຍທະບຽນ



# CONTROL UNIT CO-OP999

ເຂົ້າສູ່ຮະບບ

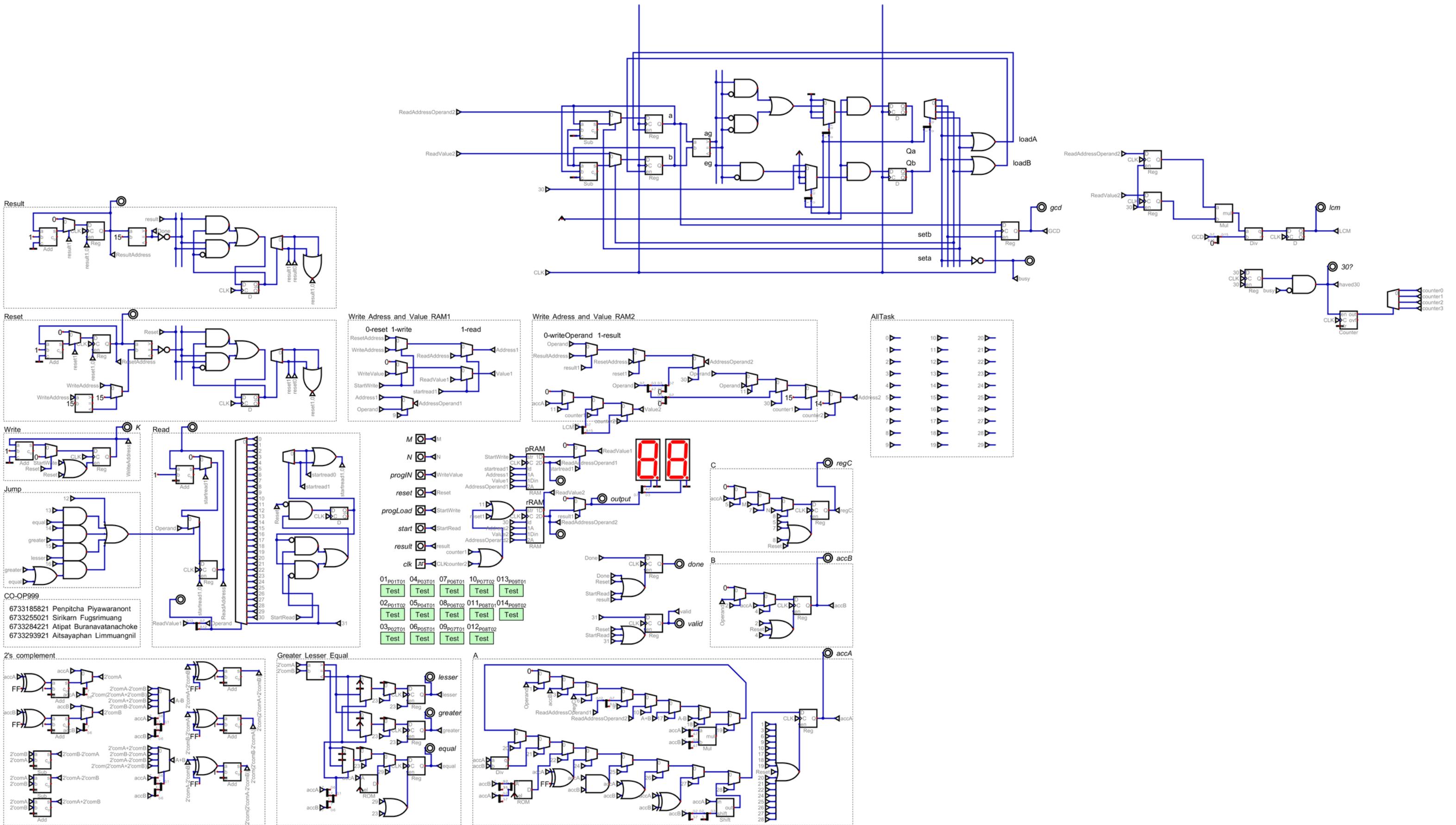
ລາຍທະບຽນ



# MINI-CPU CO-OP999

เข้าสู่ระบบ

ลงทะเบียน



# EXAMPLE CO-OP999

เข้าสู่ระบบ

ลงทะเบียน

**OPCODE 10001 accA  $\leftarrow$  accA + accB**

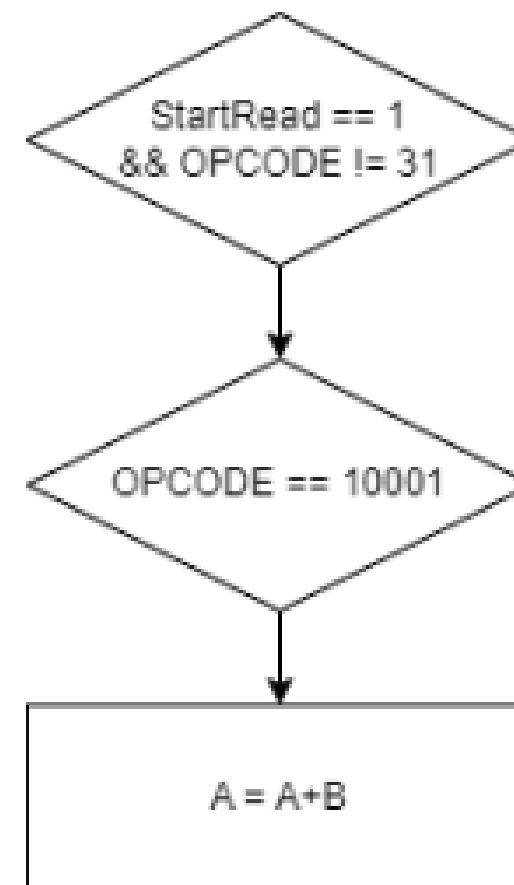
## User

- 1.start
- 2.load OPCODE & OPERAND into ProgIN
- 3.press ProgLoad 1 clock
- 4.If load not complete do 2. Again
- 5.press Start
- 6.wait until valid show 1
- 7.press Result

OPCODE & OPERAND	FUNCTION
0000100001111	accA $\leftarrow$ Operand
0001000110101	accB $\leftarrow$ Operand
<b>1000100000000</b>	<b>accA <math>\leftarrow</math> accA + accB</b>
0101100000011	RAM_add[Operand] $\leftarrow$ accA
1111100000000	STOP

## CPU (Do Function)

- OPCODE == 00001
  - 1. read OPERAND
  - 2. accA  $\leftarrow$  OPERAND
- OPCODE == 00010
  - 1. read OPERAND
  - 2. accB  $\leftarrow$  OPERAND
- OPCODE == 10001
  - 1. accA  $\leftarrow$  accA + accB
- OPCODE == 01011
  - 1. read OPERAND
  - 2. rRam[OPERAND]  $\leftarrow$  accA
- OPCODE == 11111
  - 1. Valid = 1
  - 2. STOP



# THANK YOU

