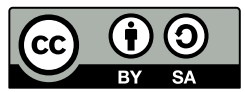
CPU implementation

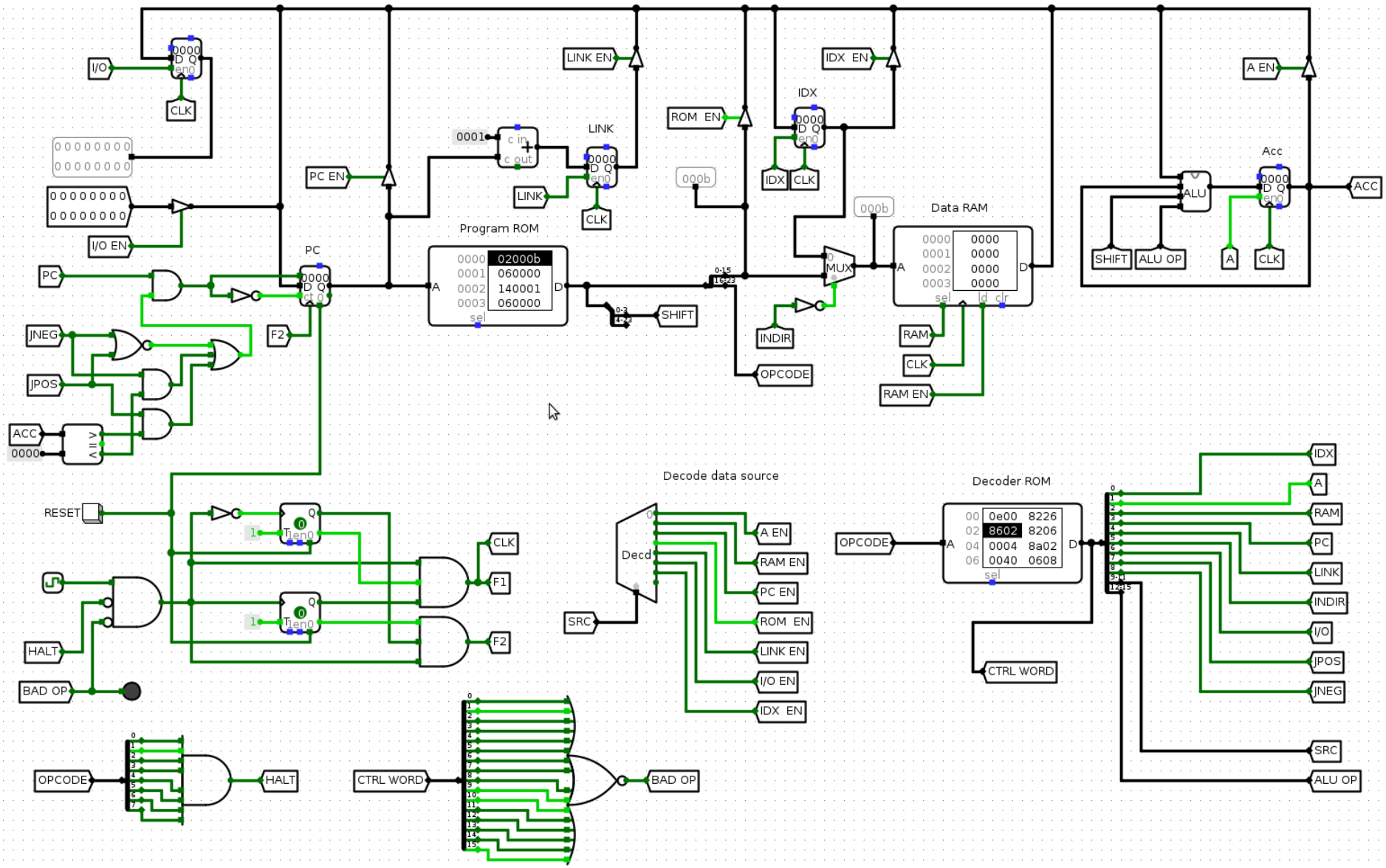
Saulius Gražulis

Vilnius, 2020

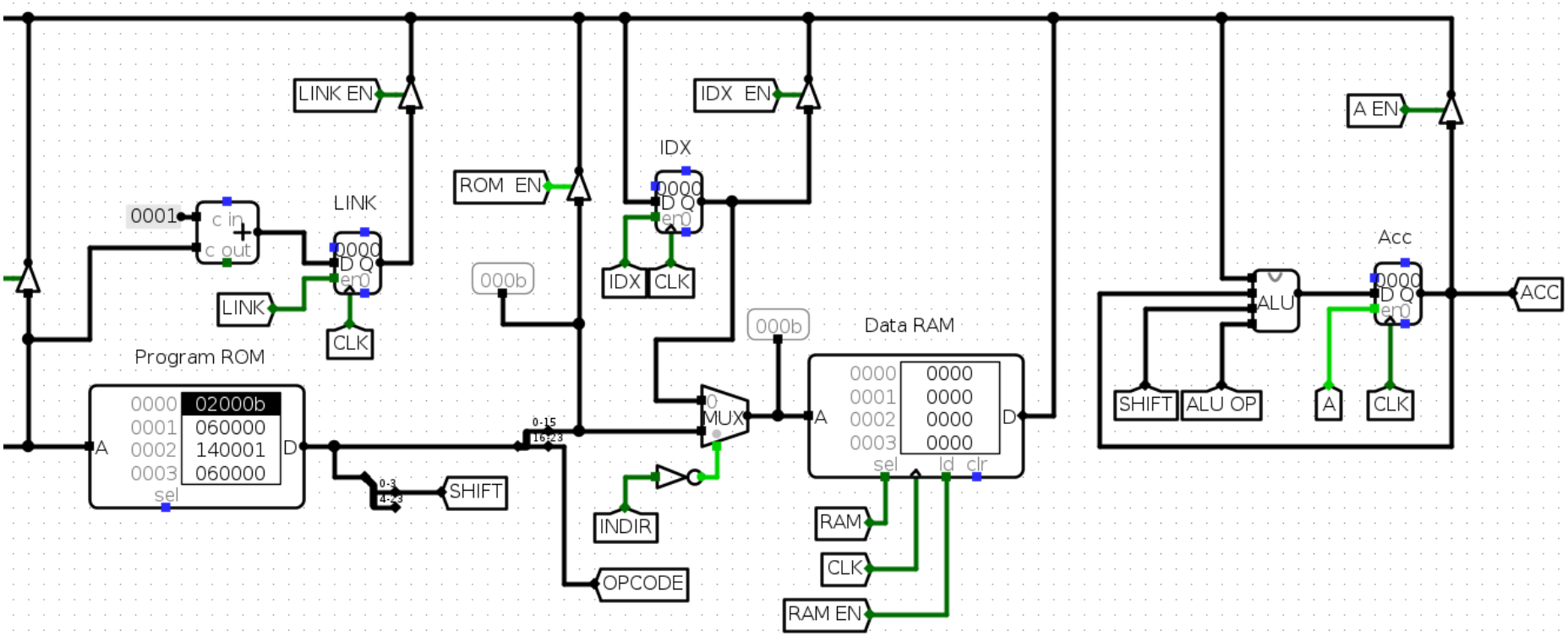
Vilnius University, Faculty of Mathematics and Informatics
Institute of Informatics



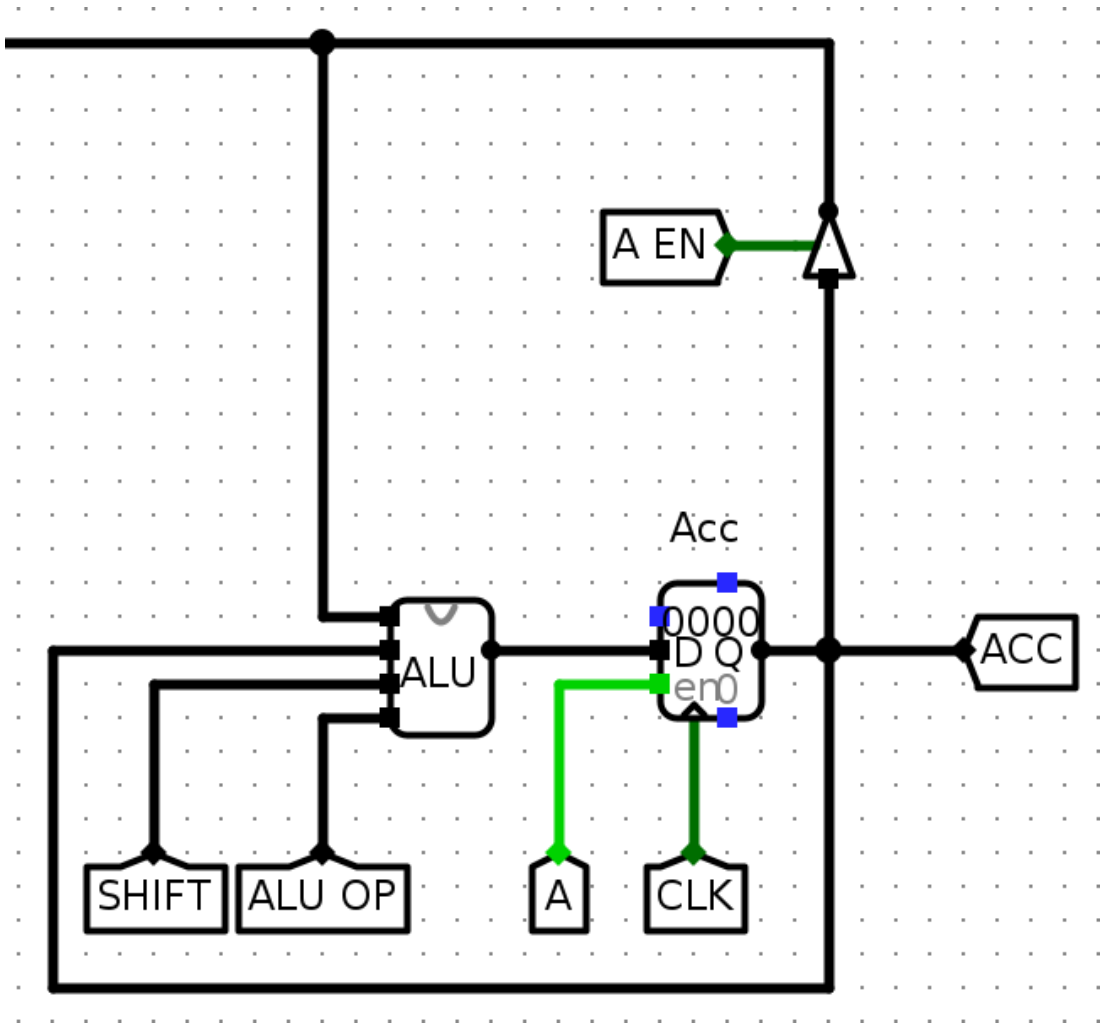
Simple Harvard architecture processor



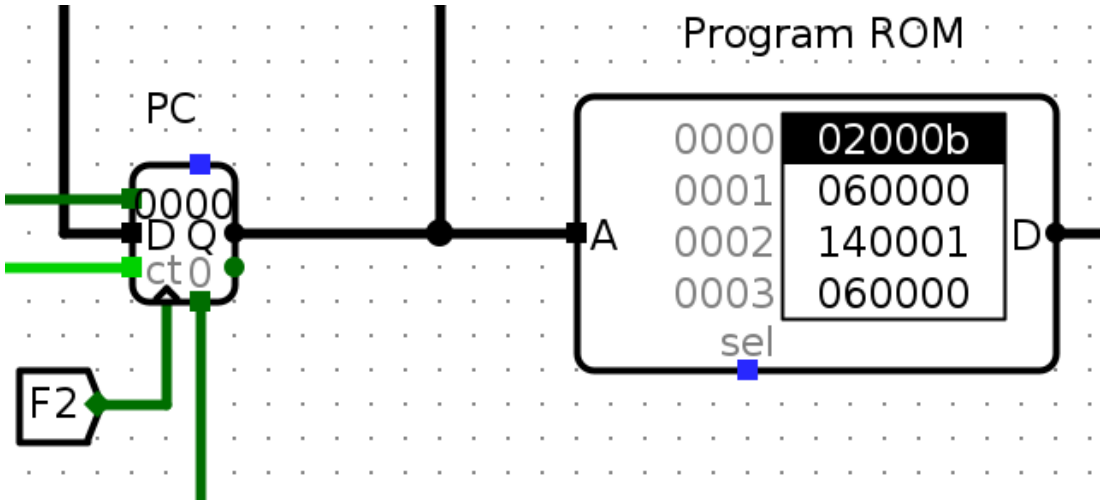
Data tract


LINK EN
IDX EN
A EN
IDX
LINK
ROM EN
0001
c in
c out
0000
D Q
en
LINK
CLK
000b
IDX
CLK
Acc
ALU
ACC
SHIFT
ALU OP
A
CLK
Program ROM
0000 02000b
0001 060000
0002 140001
0003 060000
A
D
sel
0-15
16-23
0-3
4-23
SHIFT
MUX
INDIR
OPCODE
Data RAM
0000 0000
0001 0000
0002 0000
0003 0000
A
D
sel
ld
clr
RAM
CLK
RAM EN

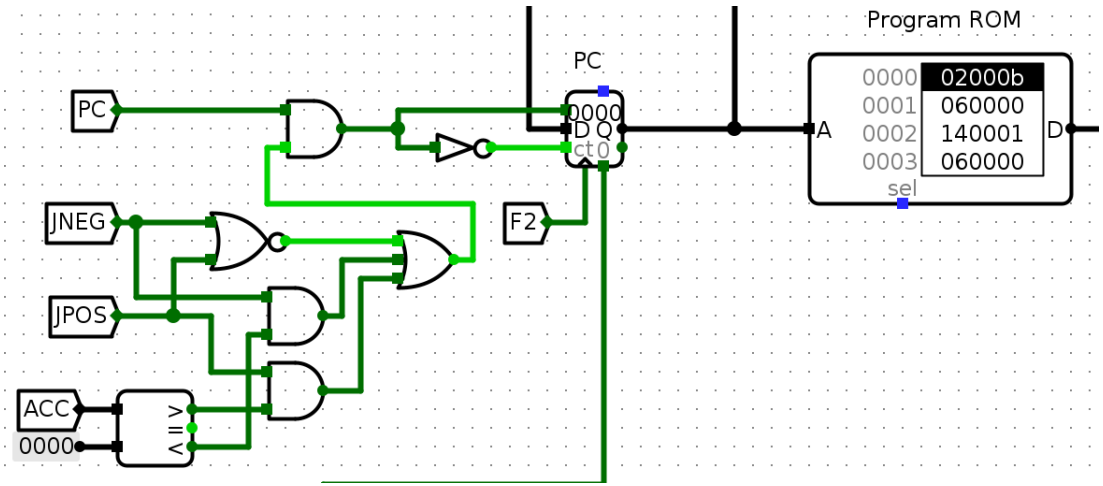
ALU and Accumulator



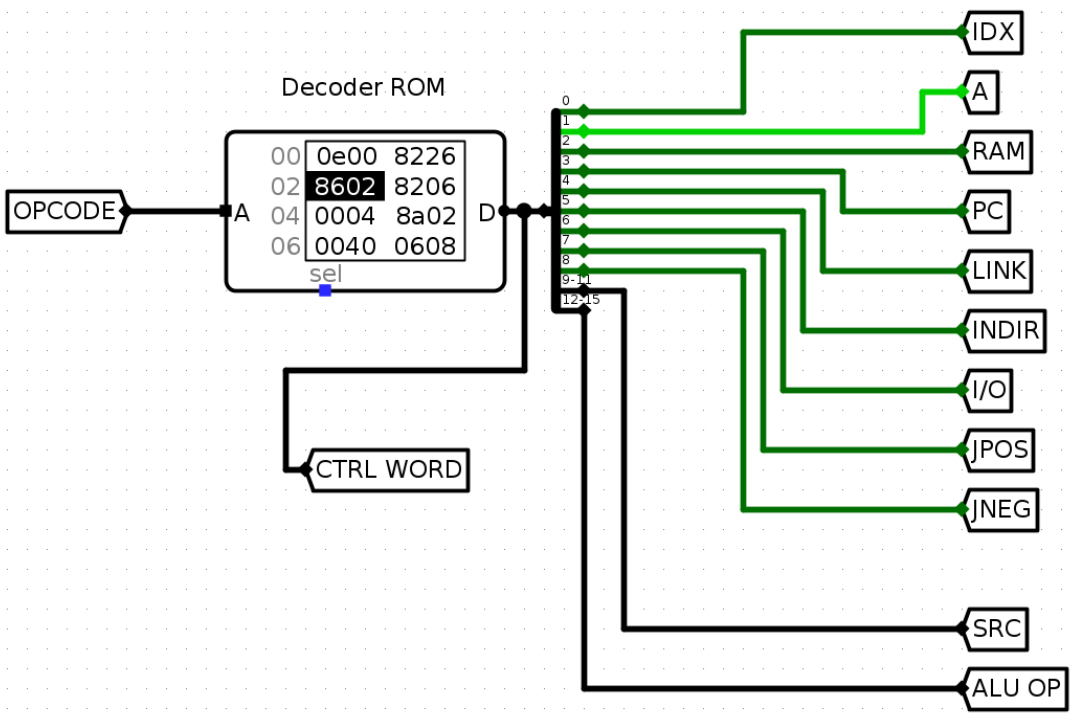
Program counter



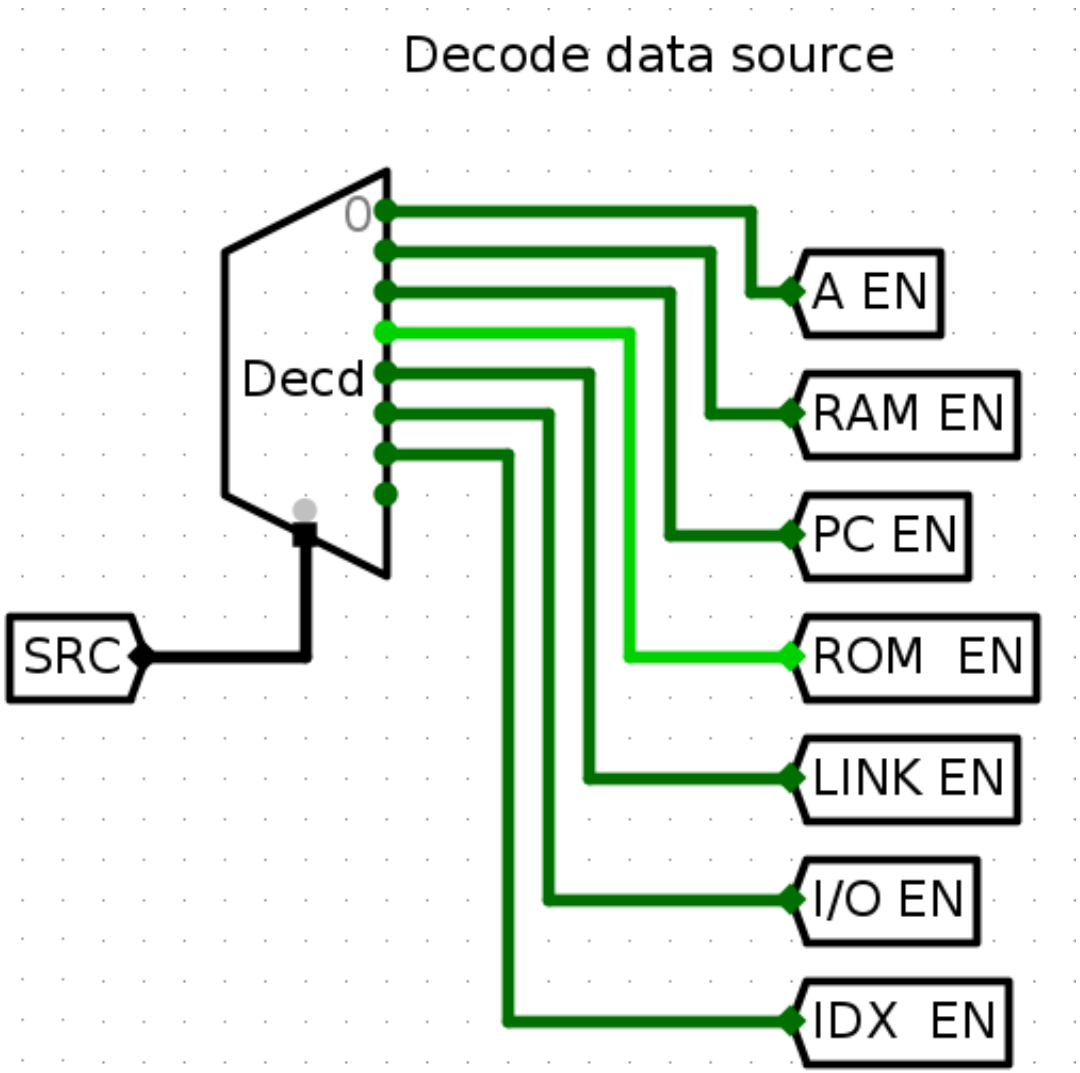
Conditional jump logic



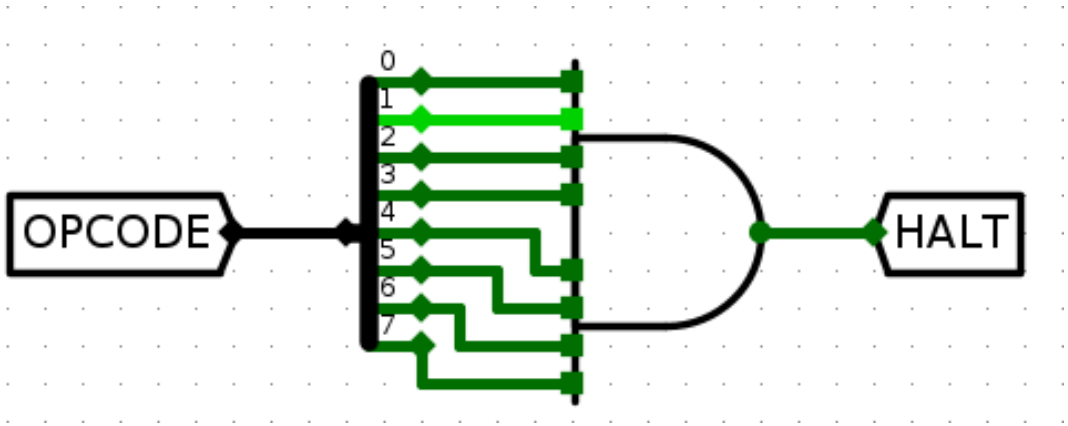
Opcode decoder



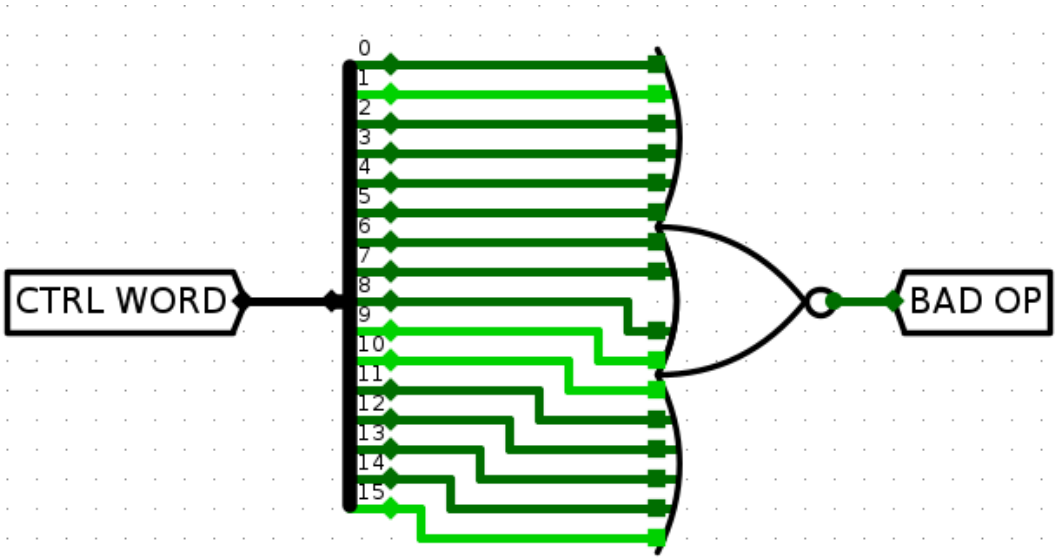
Data source decoder



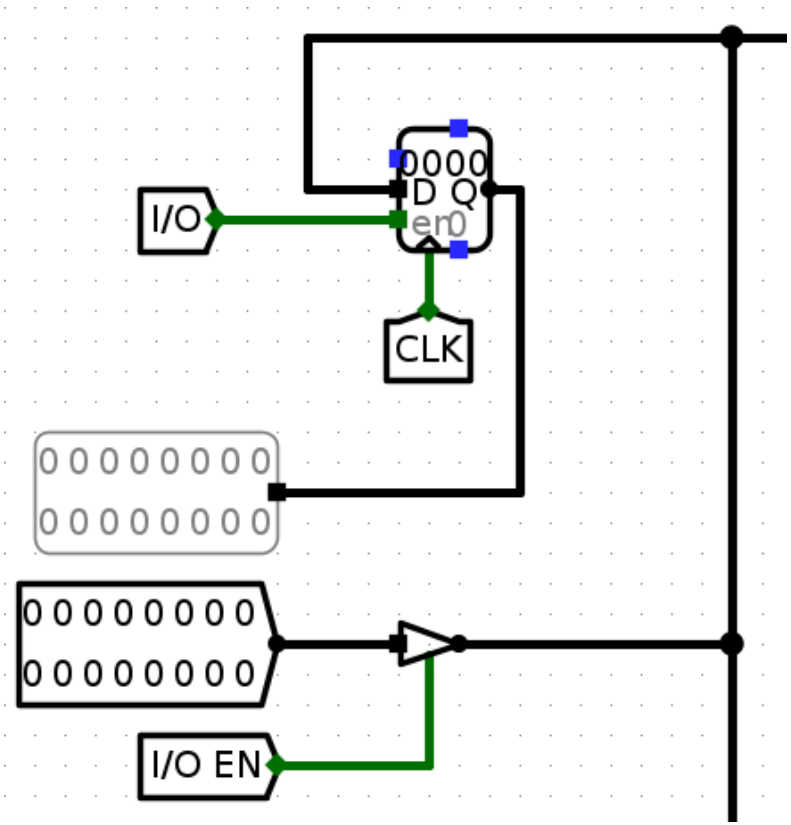
Implementation of the HALT instruction



Unimplemented/bad instruction



I/O ports



Clock oscillator