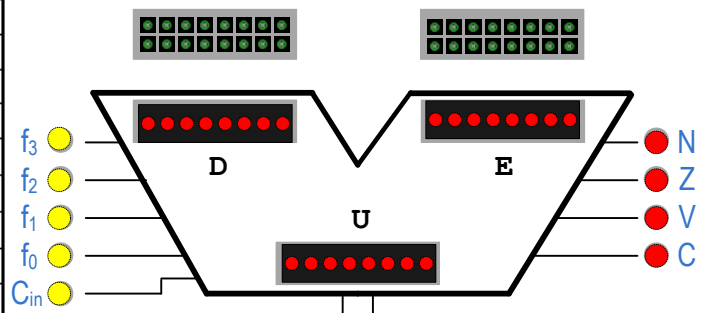
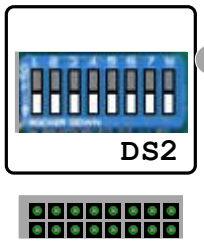
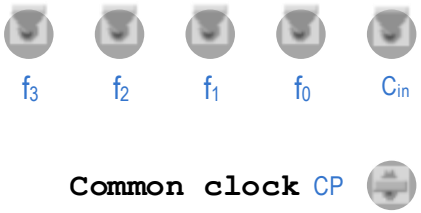
# Arithmetic Logic Unit

| function | | | | operation |
|---|---|---|---|---|
| $f_3$ | $f_2$ | $f_1$ | $f_0$ | RTN |
| 0 | 0 | 0 | 0 | U=0 |
| 0 | 0 | 0 | 1 | U=$FD_{16}$ |
| 0 | 0 | 1 | 0 | U=$FE_{16}$ |
| 0 | 0 | 1 | 1 | U=$FF_{16}$ |
| 0 | 1 | 0 | 0 | U=E |
| 0 | 1 | 0 | 1 | U=$D_{1k}$ + $C_{in}$ |
| 0 | 1 | 1 | 0 | U=D∨E |
| 0 | 1 | 1 | 1 | U=D∧E |
| 1 | 0 | 0 | 0 | U=D⊕E |
| 1 | 0 | 0 | 1 | U=D + $C_{in}$ |
| 1 | 0 | 1 | 0 | U=D + $FF_{16}$ |
| 1 | 0 | 1 | 1 | U=D + E + $C_{in}$ |
| 1 | 1 | 0 | 0 | U=D + $E_{1k}$ + $C_{in}$ |
| 1 | 1 | 0 | 1 | U=D<<1 ($C_{in}$) |
| 1 | 1 | 1 | 0 | U=($C_{in}$) D >> 1 |
| 1 | 1 | 1 | 1 | U=($d_7$) D >> 1 |

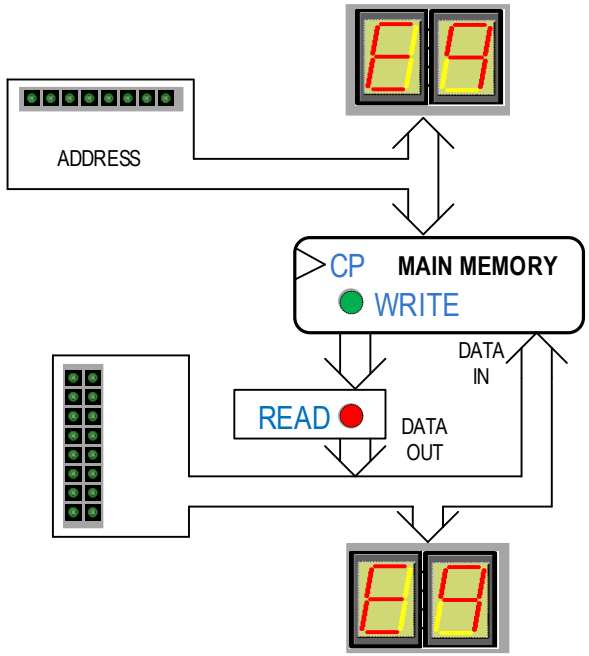


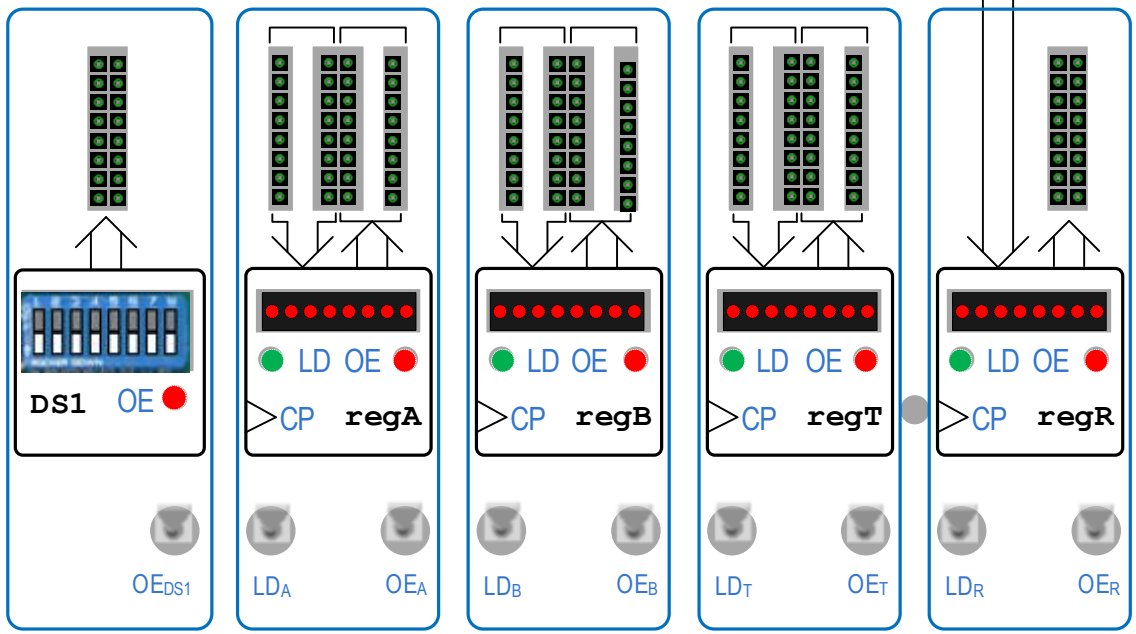



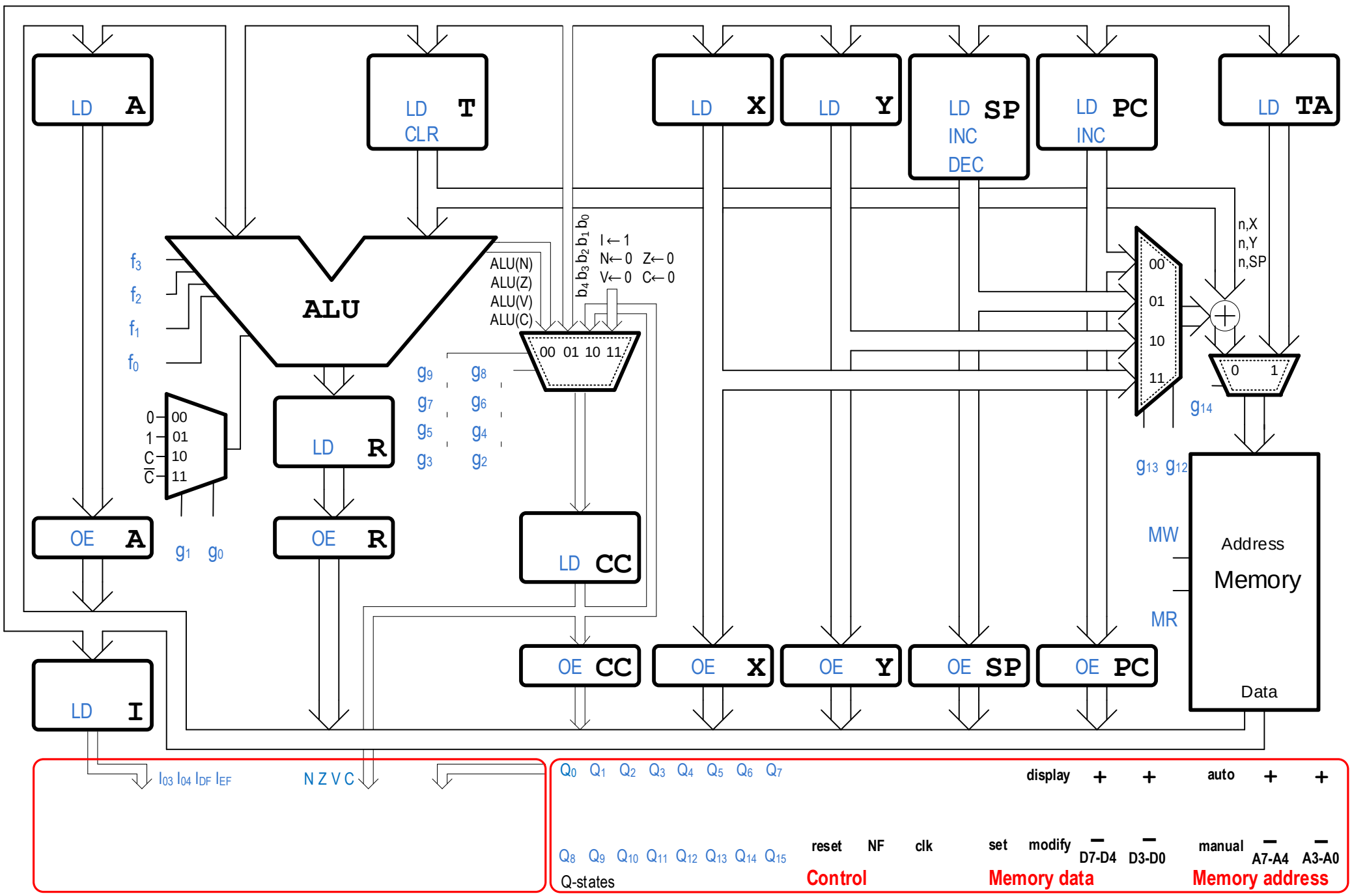

## ALU control

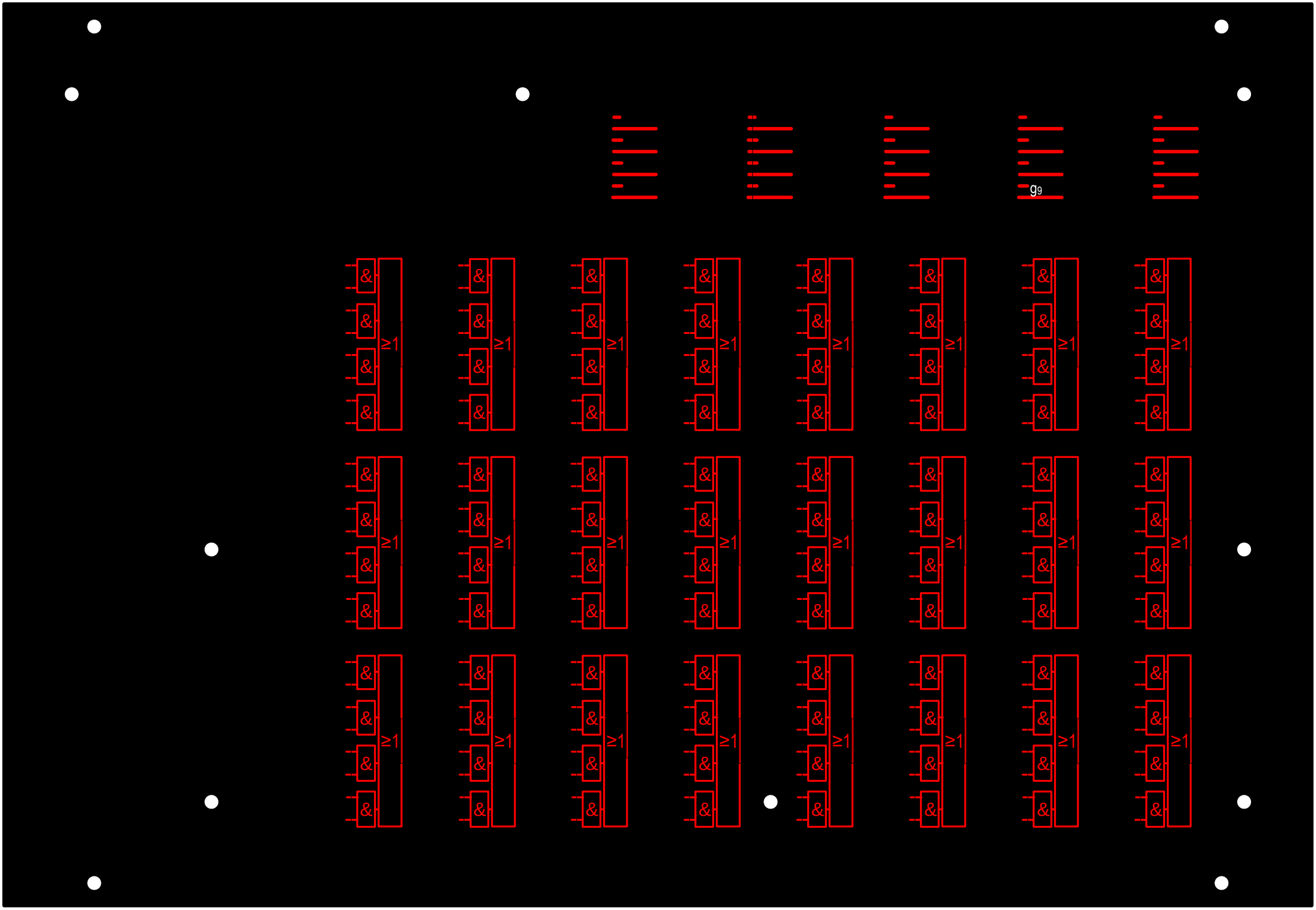


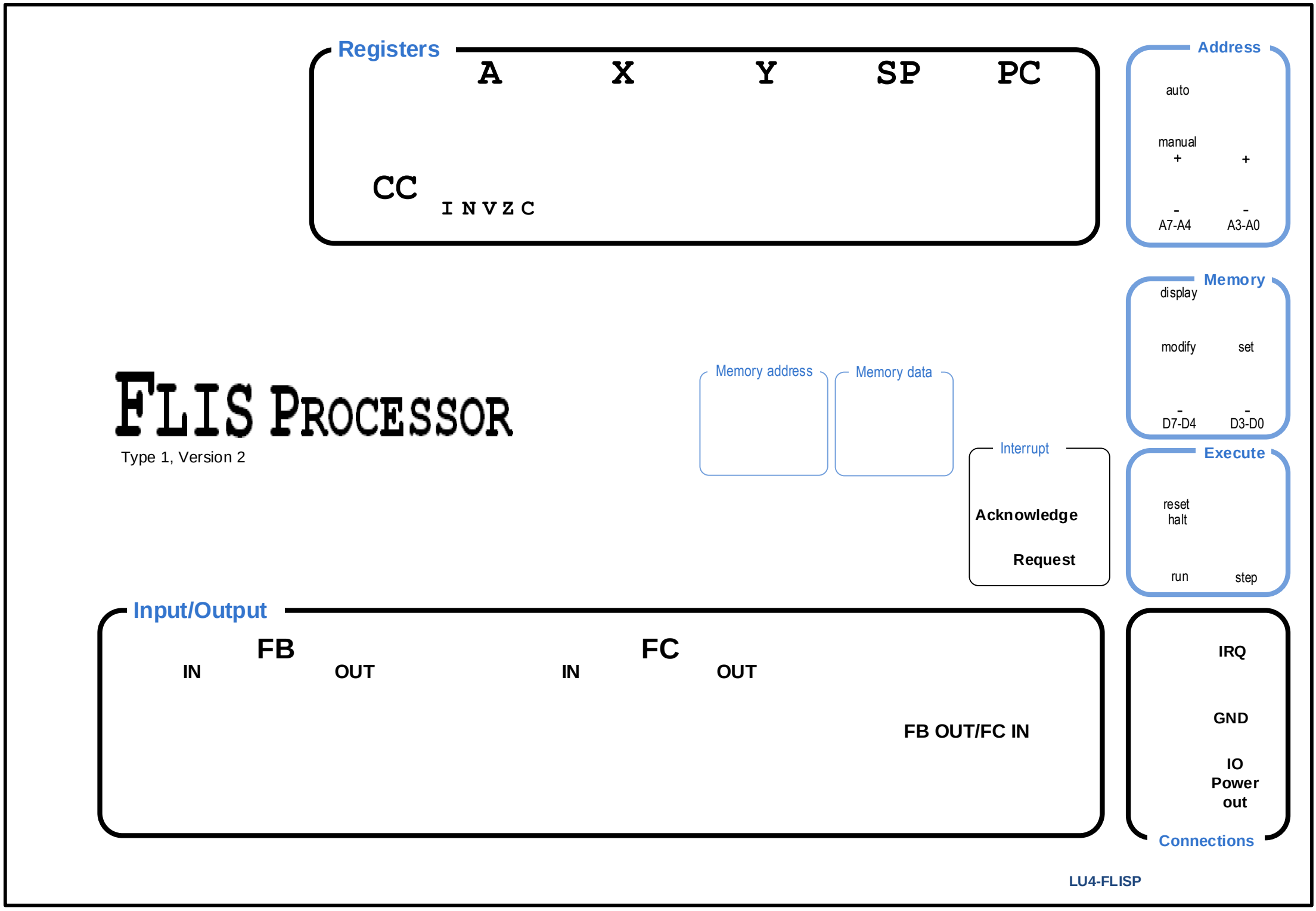

## Memory




LD A
LD T
CLR
LD X
LD Y
LD SP
INC
DEC
LD PC
INC
LD TA
ALU
f3
f2
f1
f0
ALU(N)
ALU(Z)
ALU(V)
ALU(C)
b4 b3 b2 b1 b0
I ← 1
N ← 0 Z ← 0
V ← 0 C ← 0
00 01 10 11
n,X
n,Y
n,SP
0 1
g14
g9 g8
g7 g6
g5 g4
g3 g2
0
1
C
C̄
00
01
10
11
LD R
g13 g12
OE A
g1 g0
OE R
LD CC
MW
Address
Memory
MR
Data
LD I
OE CC
OE X
OE Y
OE SP
OE PC
I03 I04 IDF IEF
N Z V C
Q0 Q1 Q2 Q3 Q4 Q5 Q6 Q7
Q8 Q9 Q10 Q11 Q12 Q13 Q14 Q15
Q-states
reset
NF
clk
set
modify
display
auto
manual
D7-D4
D3-D0
A7-A4
A3-A0
Control
Memory data
Memory address
LU3-FLISP-DV


g9
&
≥1


Registers
A
X
Y
SP
PC
CC
I N V Z C
Address
auto
manual
+
-
A7-A4
+
-
A3-A0
Memory
display
modify
set
-
D7-D4
-
D3-D0
FLIS PROCESSOR
Type 1, Version 2
Memory address
Memory data
Interrupt
Acknowledge
Request
Execute
reset
halt
run
step
Input/Output
FB
IN
OUT
FC
IN
OUT
FB OUT/FC IN
IRQ
GND
IO
Power
out
Connections
LU4-FLISP