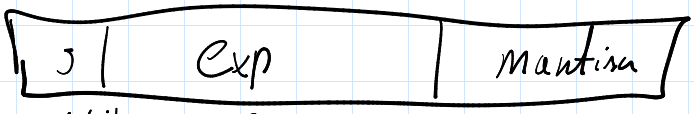
Rounding hardware



M 1. . . . . .

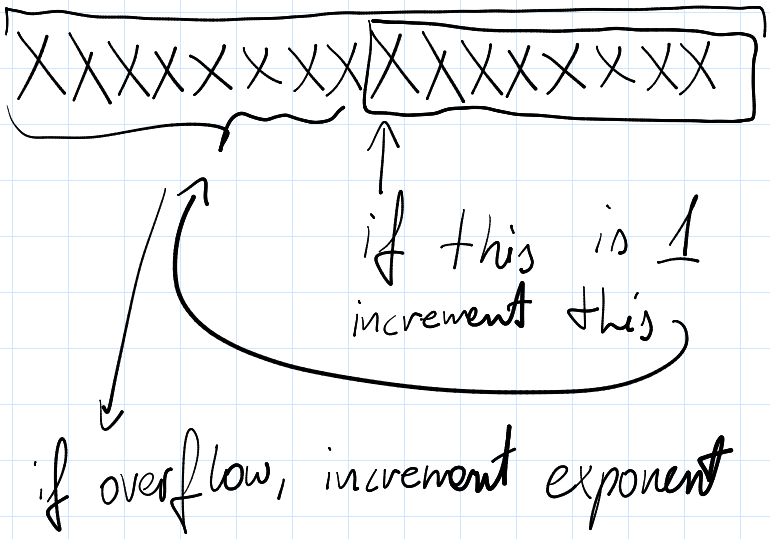
$(-1)^S \cdot M_S \cdot 2^{E-63}$

Normalize step

first bit of mantisa must be a 1 or fpn = 0

After "Big ALU"

M_B - 16 bits



$-1 \cdot 0.5 \approx -1 \cdot 1_2$

$-0.5 \approx 0$?

Overflow

small ALU overflow **OR** rounding

Steps

1. Normalize the exponent to the bigger one, shift the smaller mantissa to the right to reflect the exponent change.

Note: Once you shift the mantisa 10, the result will be rounded to the bigger number

2. Add the Mantissas together and round
3. Determine the sign bit
4. Determine overflow

Actual steps

1. Calculate exponent difference and determine the smaller or equal

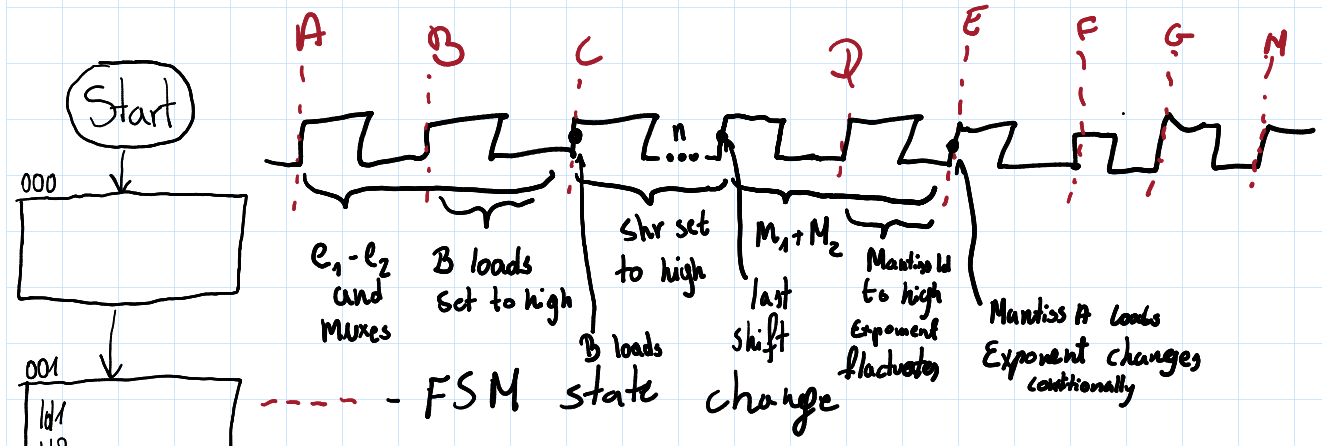
If the exponent difference is bigger or equal to 10 return the bigger-exponent number

1. Convert 8 bit unsigned mantissa to a 9 bit signed mantisa
2. Place the left operand in the left 18-bit register and equivalently do the same to the right, unless the left exponent is smaller, then allocate the inversely
3. Shift the left operand exponent-difference times
4. Add the 2 operands to ether
5. If overflow occured (not floating point representation), normalize the result

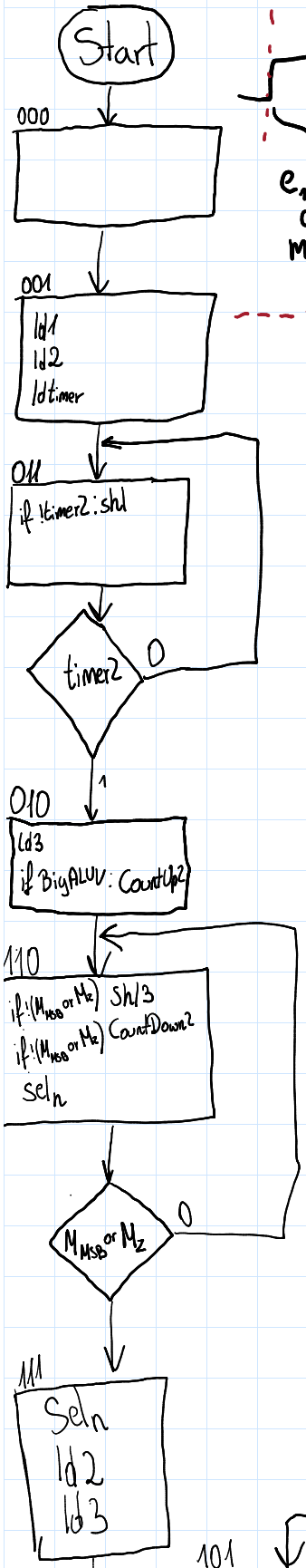
Increment exponent
shifts right BIG ALU result

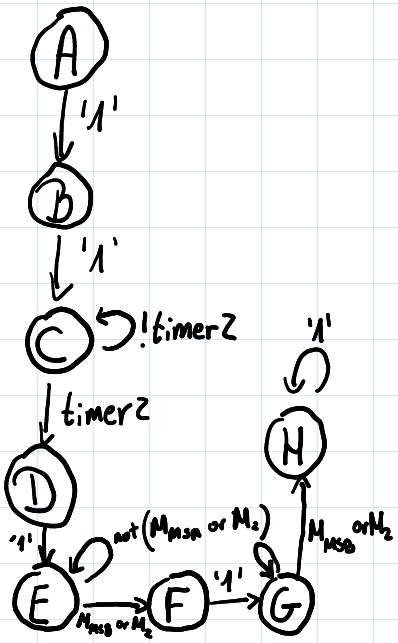
decrement exponent
shift left BIG ALU result

6. Round the result according to above instructions
7. Return number

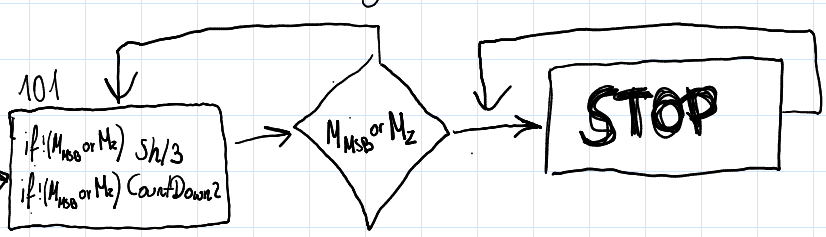


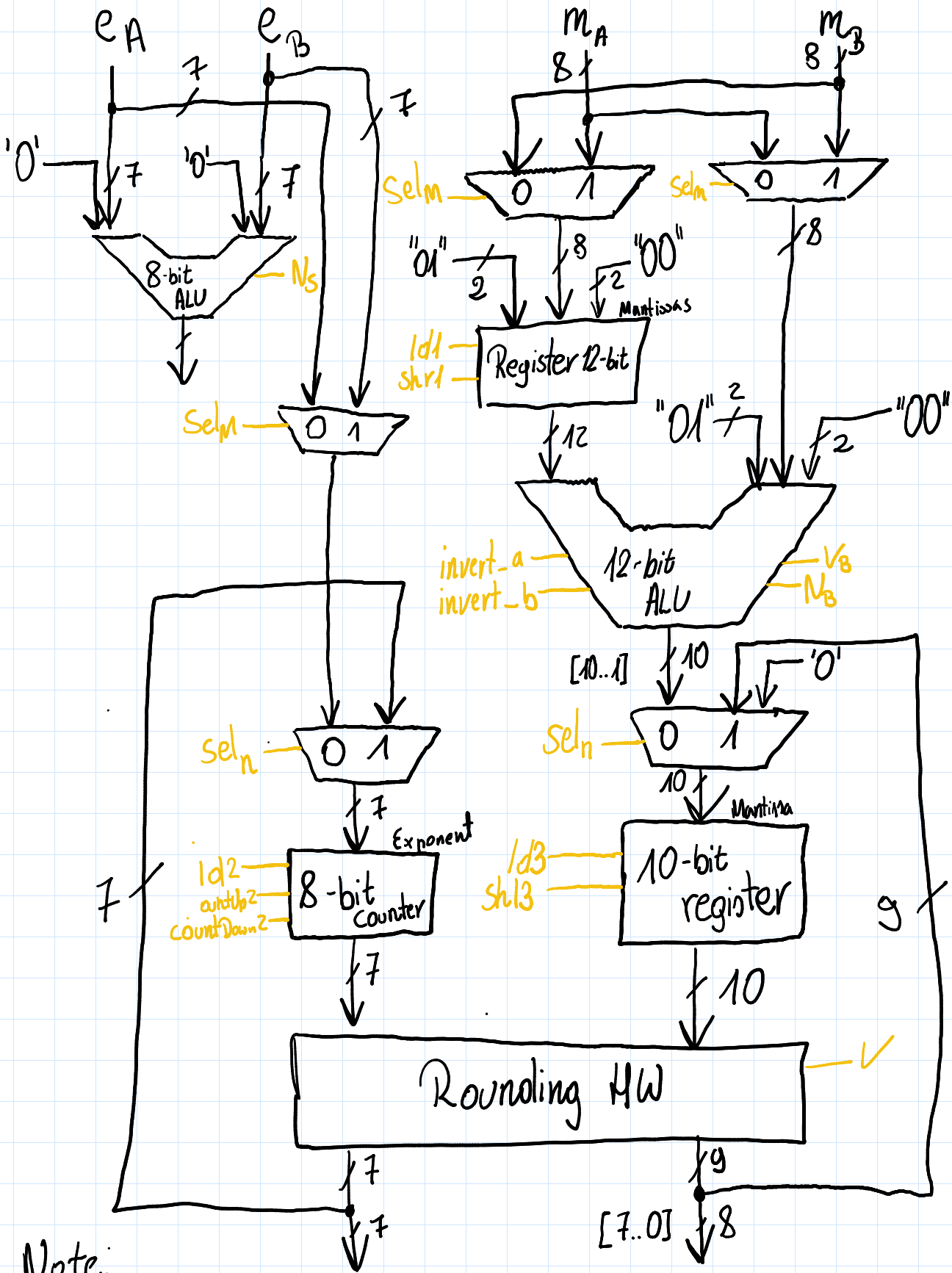
0 000 A - $e_1 - e_2$, set the input muxes
1 001 B - ld MantissaS, ld CounterE, ld timer
3 011 C - MantissaS shr, $M_1 + M_2$
2 010 D - ld Mantissa, increment/decrement exponent if Mantissa V
6 110 E - Mantissa shl, exponent decrement, Round, normalize mux
7 111 F - normalize mux, ld Mantissa, ld E
5 101 G - Mantissa shl, exponent decrement
4 100 H - Stop





FP ADDER




eA
eB
mA
mB
'0'
'0'
8-bit ALU
Ns
Selm
Selm
"01"
"00"
Mantissas
ld1
shr1
Register 12-bit
Selm
"01"
"00"
12
invert_a
invert_b
12-bit ALU
VB
NB
[10..1]
'0'
Seln
Seln
Exponent
Mantisa
ld2
addUp2
countDown2
8-bit Counter
ld3
shl3
10-bit register
Rounding HW
V
[7..0]
Note:
- if Ns: eA < eB

A. $1.m_A \cdot 1.m_B$, e_1+e_2, ldE, ld mantissa

B. $(e_1+e_2)+63$ (or +65 or +67 or +69 depending on BiG ALU Out[14..13]), ldE

C. Normalize, round

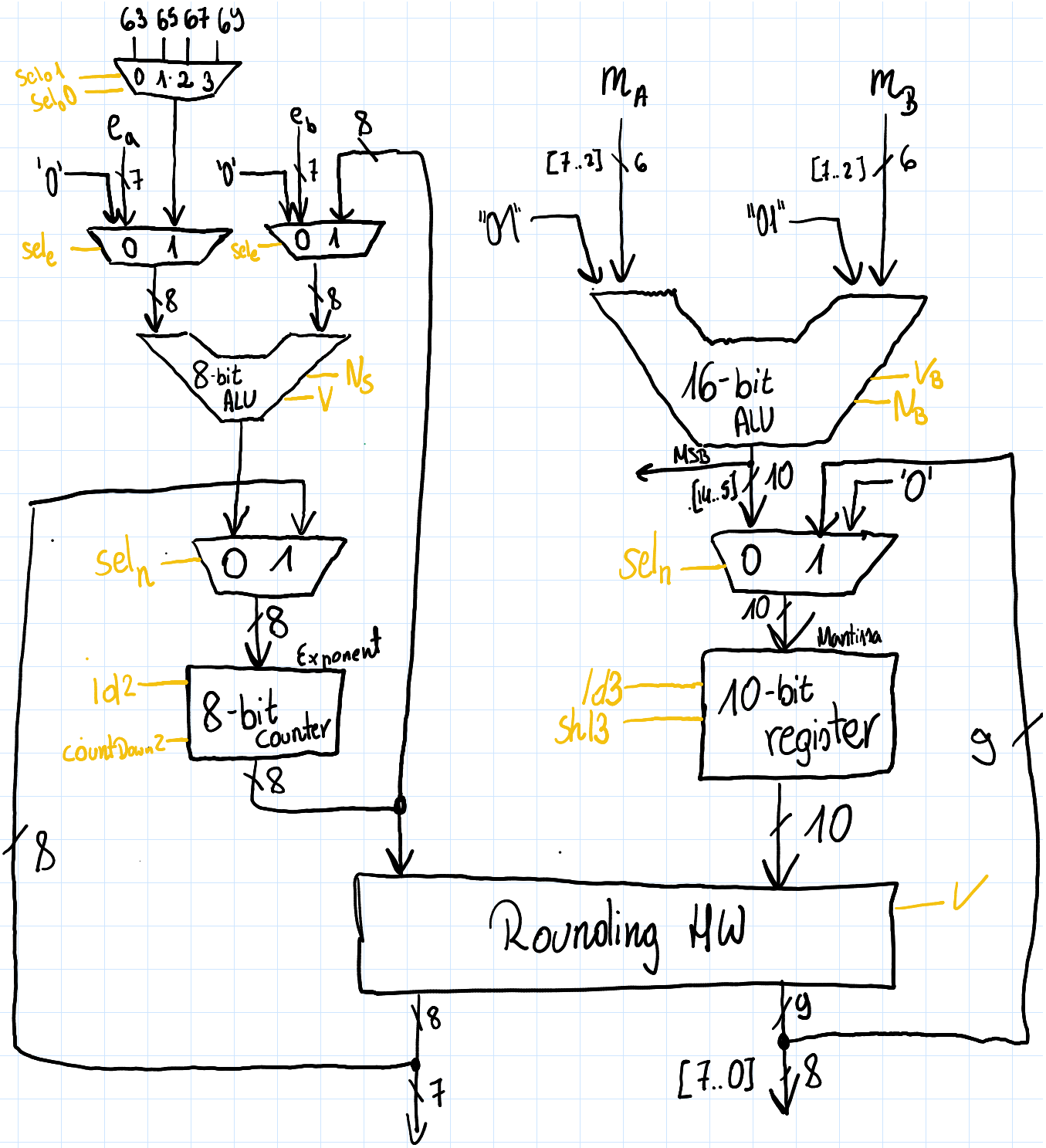
D. Reload

E. Renormalize, round

F STOP

G dc

H dc


63 65 67 69
sel1
sel0
0 1 2 3
ea
eb
8
'0'
7
sele
0 1
8
8-bit ALU
V
NS
mA
mB
[7..2]
6
[1..2]
"01"
16-bit ALU
VB
NB
MSB
[14..5]
10
selm
Exponent
ld2
8-bit counter
countDown2
Mantissa
ld3
sh13
10-bit register
9
Rounding HW
[7..0]

$$S = S_A \oplus S_B$$