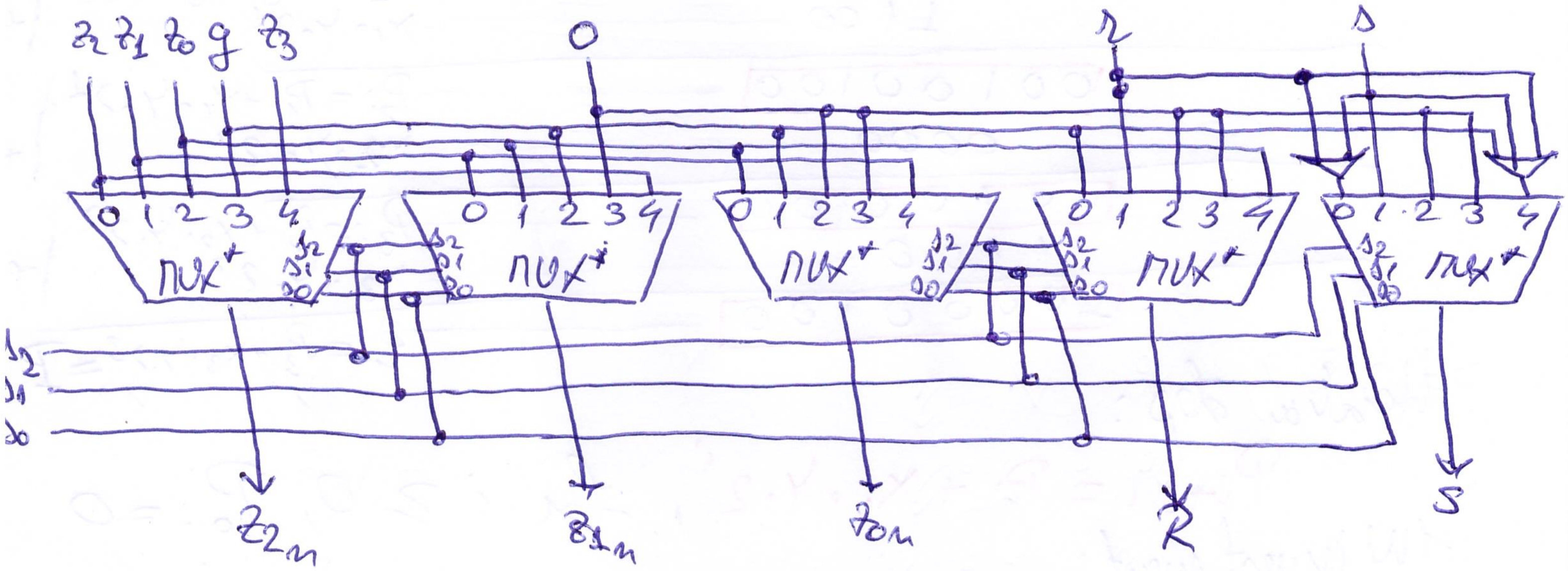
ACR 3.5. ①

Only 1 of the 5 boolean variables can be active for a given f.p. operands pair X and Y
⇒ encode the 5 variables using only 3 signals: s_2, s_1, s_0

Inputs					Outputs		
z_1	z_3	z_2	z_1	g/z_0	s_2	s_1	s_0
0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	0
0	1	0	0	0	0	1	1
1	0	0	0	0	1	0	0



MUX* ≡ degenerate MUX.

Chapter IV Functional analysis & synthesis of Binary Multiplication Devices.

4.1. Multiplication methods

a) paper and pencil. X = multiplier Y = multiplicand.

consider X = 11 Y = 12, unsigned.

```
   1100 ———————————— Y
   1011 = x3x2x1x0 —— X
 ----------------
   1100 — x0·Y·2^0
  1100 —— x1·Y·2^1
 0000 ——— x2·Y·2^2
1100 ———— x3·Y·2^3
 ----------------
10000100
```

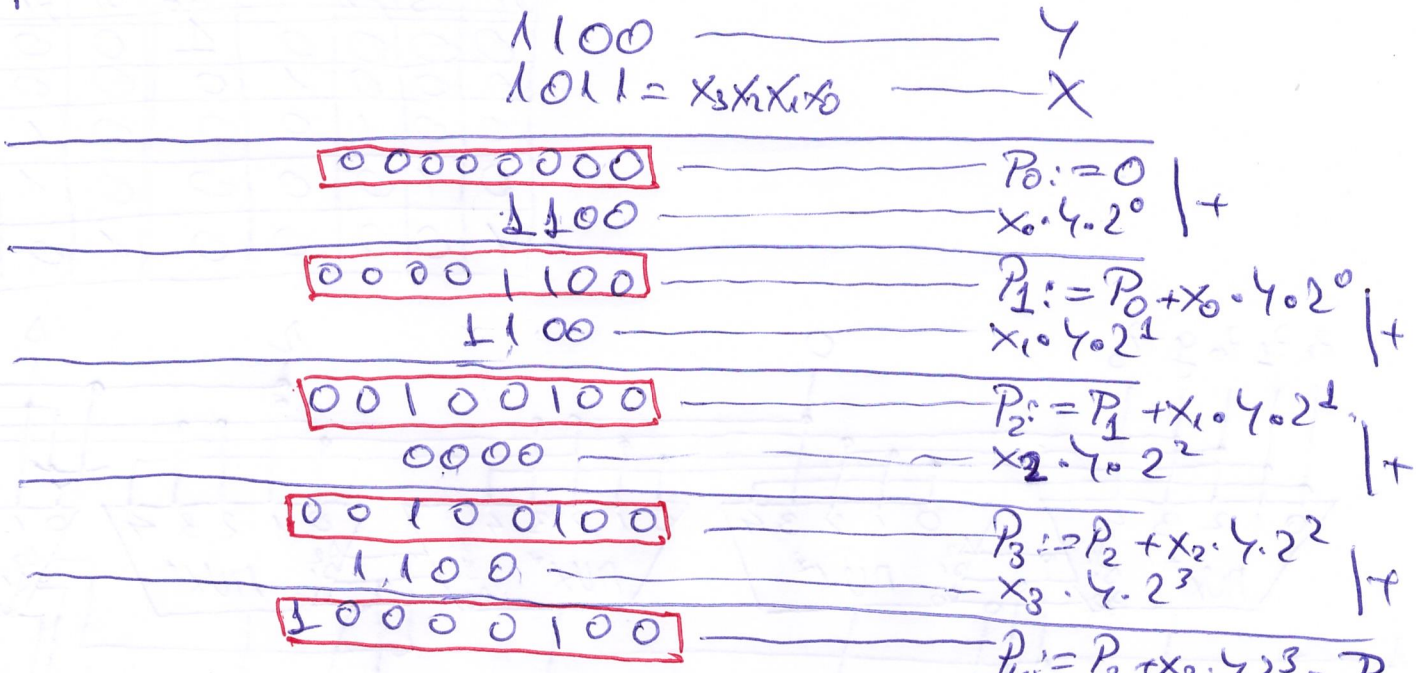
$10000100 - P = \sum_{i=0}^{3} x_i \cdot Y \cdot 2^i = 4 + 128 = 132$

12 · 11: 12 + 12 = 132

HW investment

- 2 4-bit registers for X, Y
- multi-operand adder (CSA)
- multiplexed gating

b) Keep the partial products fixed.



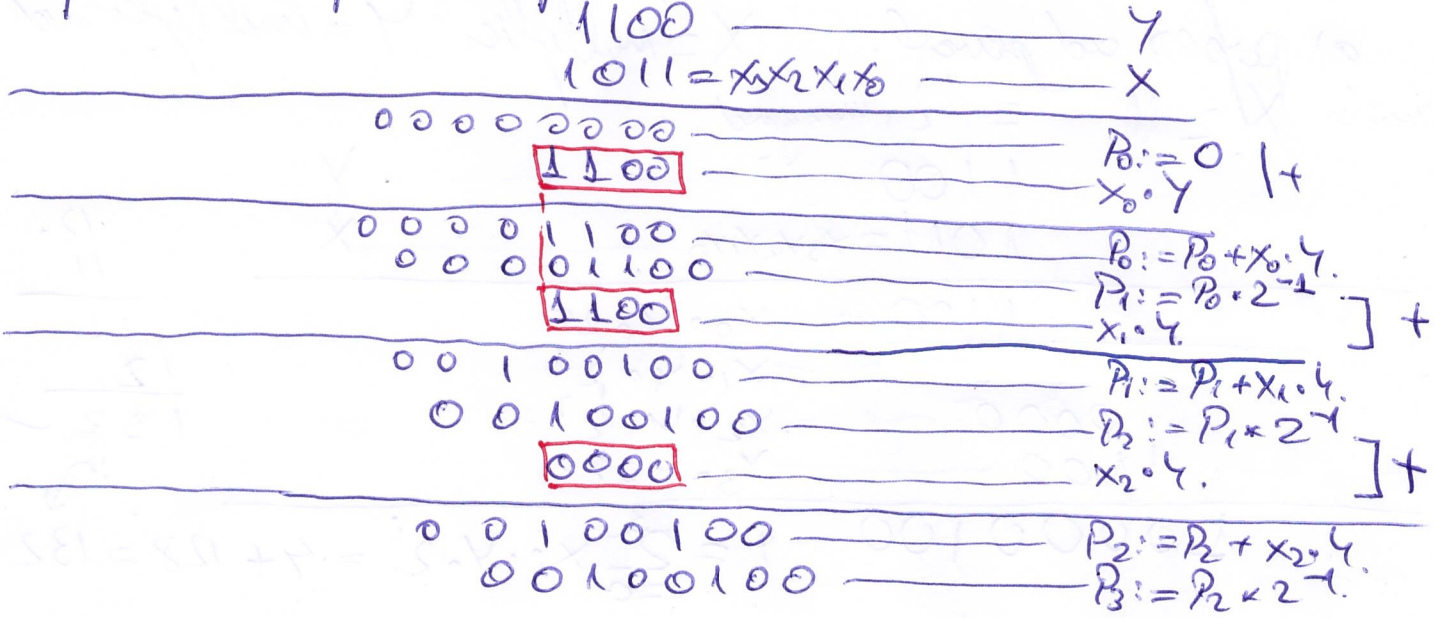
Iteration step:

$$P_{i+1} := P_i + x_i \cdot Y \cdot 2^i, \text{ for } i \geq 0, P_0 := 0$$

HW investment:

- 2 4-bit registers for X, Y
- 8-bit registers for partial products
- 8-bit adder
- Shifter for the 4-bit products.

c) Keep the 1-bit products fixed.



00100100 ———— $P_3 := P_2 * 2^{-1}$
1100 ———— $x_3 \cdot Y$] +
10000100 ———— $P_3 := P_3 + x_3 \cdot Y$
10000100 ———— $P_4 := P_3 * 2^{-1} = P$

iteration step: $\begin{cases} P_i := P_i + x_i \cdot Y \\ P_{i+1} := P_i * 2^{-1} \end{cases}$ for $i \geq 0$, $P_0 := 0$.

Hardware investment
- 2 4-bit registers for X, Y.
- 8-bit register for the p. products; RShift capabilities
- 4-bit adder

4.2. Sequential Binary Multiplication for Sign-Magnitude numbers

Let $X = x_7 . x_6 x_5 x_4 x_3 x_2 x_1 x_0$

$Y = y_7 . y_6 y_5 y_4 y_3 y_2 y_1 y_0$

sign: y_7; weight: 2^{-1} 2^{-2} 2^{-3} 2^{-4} 2^{-5} 2^{-6} 2^{-7}

X, Y – SM, on 8 bit.
– fractionals
|a fractional| < 1

$P = X * Y = P_{15} . P_{14} P_{13} —— P_2 P_1 P_0$

multiply 2 unsigned on 7 bits, for the magnitude part
⟹ result on 14 bits (unsigned)
⟹ add the sign. $P_{15} = x_7 \oplus y_7$

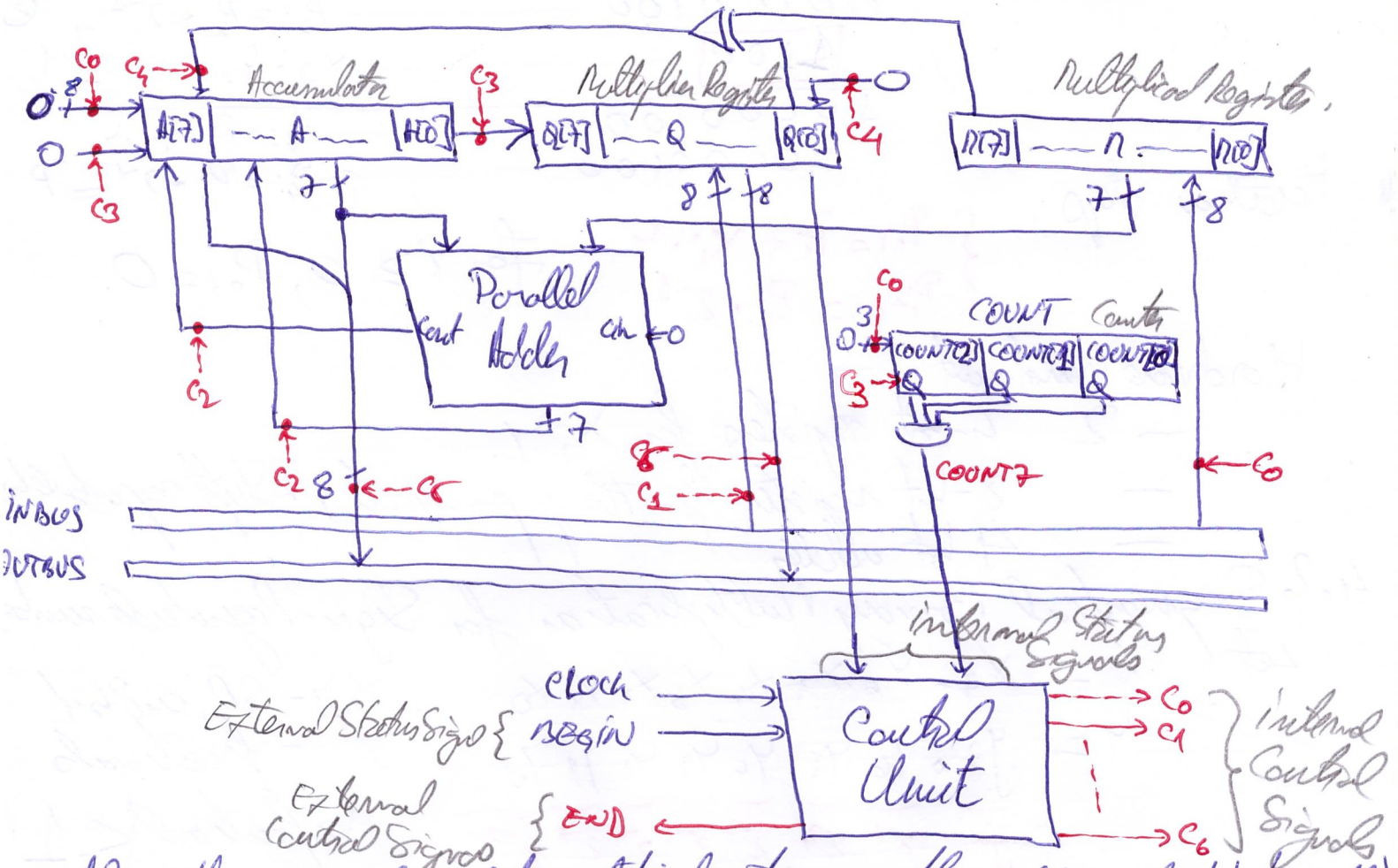
multiplication of 2 unsigned nos on p bits
⟹ 2p bit result
ex: 15 * 15 = 225
4 bit 4 bit unsigned. 8-bit unsgd.

– In computers we work with values on n_8 bits
⟹ add $P_0 = 0$ fractional extension

0.12 → 0.120

multiplier2
declare registers A[7:0], Q[7:0], M[7:0], COUNT[2:0];
declare bus INBUS[7:0], OUTBUS[7:0];

BEGIN: A := 0, COUNT := 0, } ———— {c0}
INPUT: M := INBUS;
Q := INBUS; ———— {c1}
TEST1: if Q[0] = 0 then go to RShift;
ADD: A[7:0] := A[6:0] + M[6:0]; ———— {c2}
RShift: A[7] := 0, A[6:0].Q := A.Q[7:1], } ———— {c3}
INCREMENT: COUNT := COUNT + 1;
TEST2: if COUNT ≠ 1 then go to TEST1;
SIGN: A[7] := Q[0] xor M[7], Q[0] := 0; ———— {c4}
OUTPUT: OUTBUS := A; ———— {c5}
OUTBUS := Q; ———— {c6}
END:



Algorithm uses the 3rd multiplication method (keep the 1-bit product)

iteration step
$$\begin{cases} P_i := P_i + x_i \cdot Y \rightarrow \text{label TEST1 and ADD} \\ P_{i+1} := P_i \cdot 2^{-1} \rightarrow \text{label RShift} \end{cases}$$

TEST1: if x_i is 0: avoid adding 0 to P_i and jump to
$P_{i+1} := P_i \cdot 2^{-1}$

! - at any given time x_i: the current bit of X is stored in Q[0]

Partial products:
- at the beginning of the algorithm is in A. ($P_0 := 0$)
- with each iteration, P_{i+1} advances into register Q one bit at a time, during RShift

⇒ X is right shifted by loosing the last significant bits, one at a time, at RShift.

Parallel Adder:
- updates the partial product
- on 7 bits: because P_i and Y are in S-7, at addition of S-M no just ignore the signs.
- addition's carry out is stored into A[7].
⇒ avoid overflow

COUNTER: - counts the 7 iterations (7 magnitude bits in Y)
- incremented at each R shift

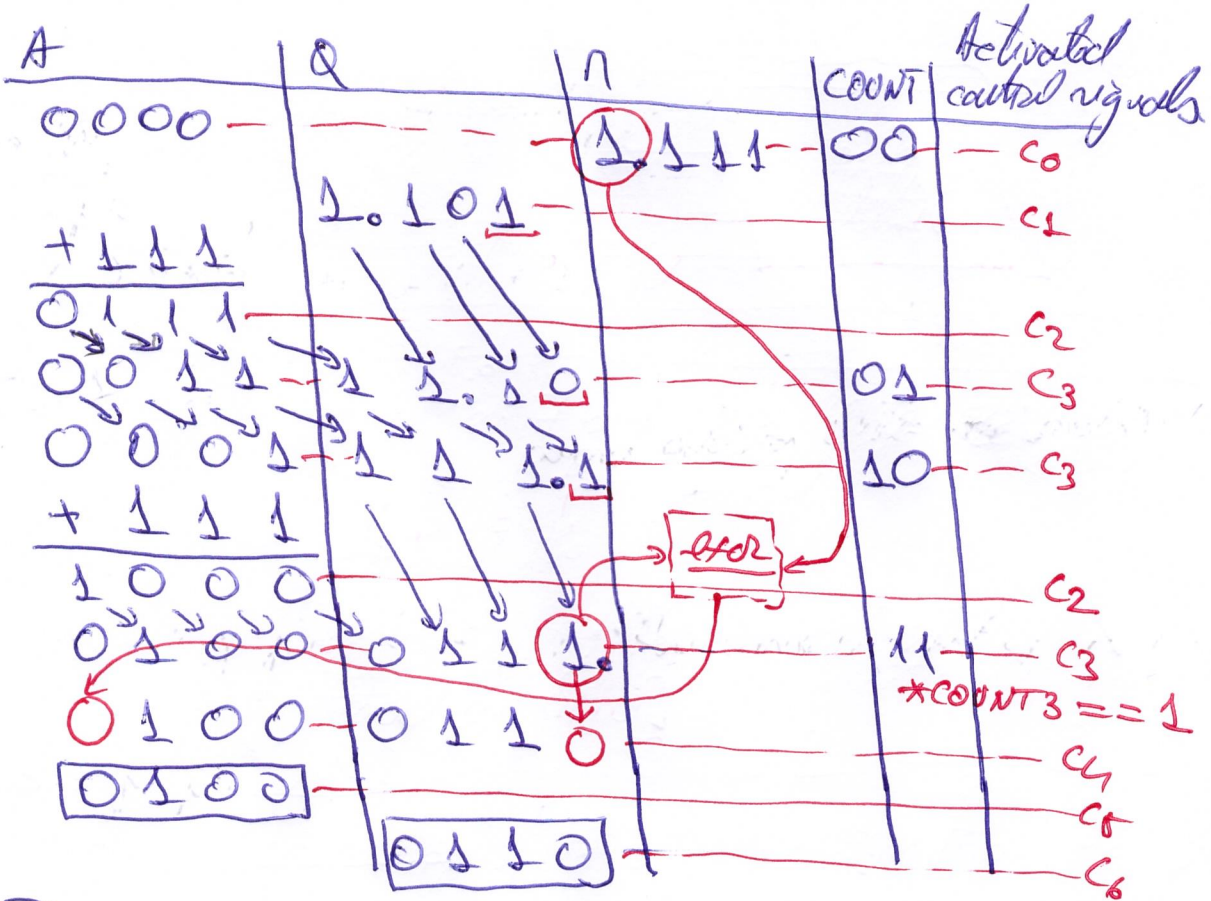
COUNT7 - activated when COUNT's state becomes 111 (value of 7)

CONTROL UNIT: sequences the operations by activating the control signals
- each clock cycle one control signal is active.
- control signals are represented with ----→
- data lines are solid. ——
- control signals affect ↔ data lines, or storage elements.

Example:

$X = -0.625 = -5 * 2^{-3} = \underbrace{1}_{\text{sign}}.\underbrace{101}_{5}{}_{SM}$

$Y = -0.875 = -7 * 2^{-3} = 1.111_{SM}$



$P = 0.1000110 = 100011 * 2^{-6} = 35 * 2^{-6}$

$X = -5 * 2^{-3}$ $Y = -7 * 2^{-3}$ $P = X * Y = 35 * 2^{-6}$ ✓