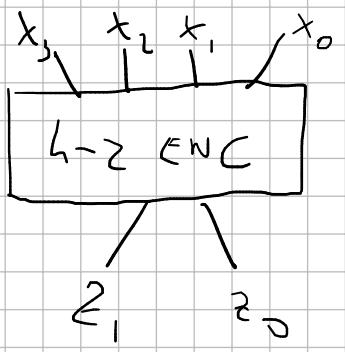
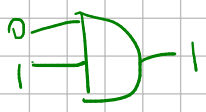
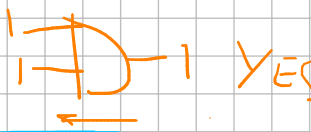
ENCODER



Reverse function of the ENCODER

Can gate be reversed?

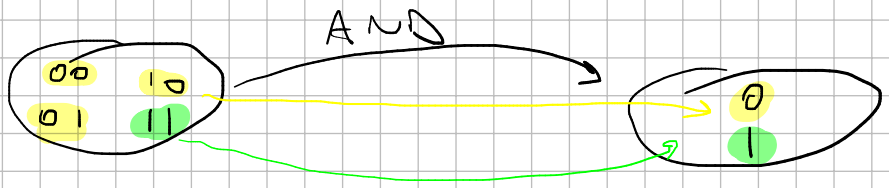
A) NOT  YES!

B) AND, OR   YES! only if 1

NO!

---

2 INPUTS 1 OUTPUT



4 INDIVIDUALS (BITS)

2 DESTINATION (NETS)

ENCODER IMPLEMENTATION

x_3	x_2	x_1	x_0	z_1	z_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

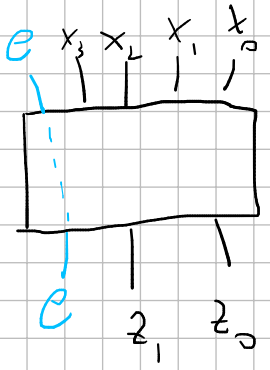
2 rows missing! ↕ ?

PRIORITY ENCODER

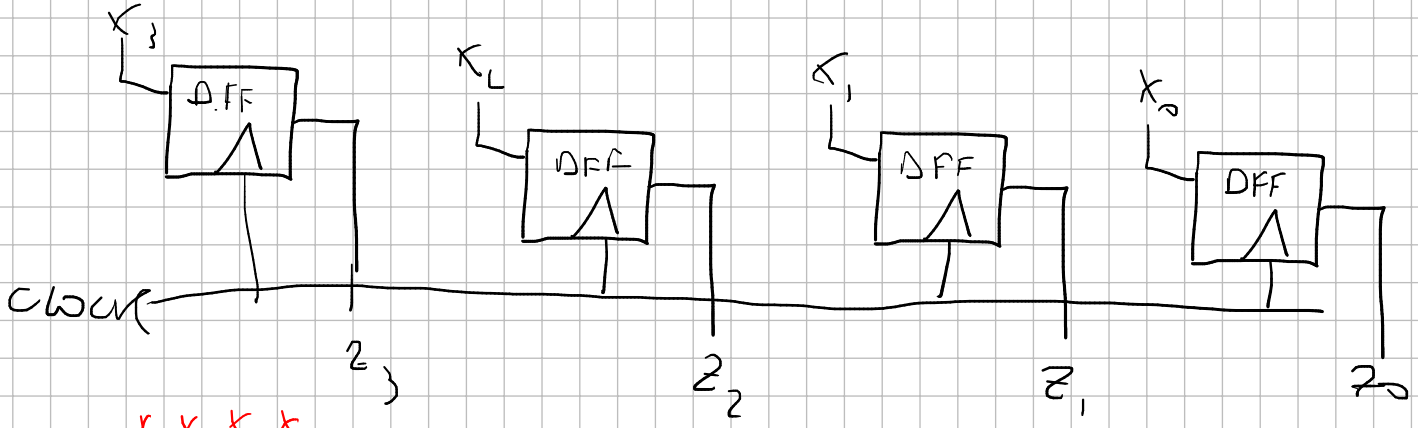
e	x_3	x_2	x_1	x_0	z_1	z_0	e
1	0	0	0	1	0	0	← 1
1	0	0	1	—	0	1	1
1	0	1	—	—	1	0	1
1	1	—	—	—	1	1	1
0	—	—	—	—	0	0	← 0

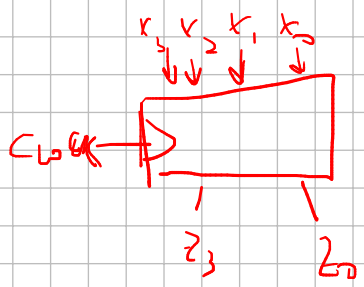
1 row missing:

$\phi\ \phi\ \phi\ \phi$



REGISTER

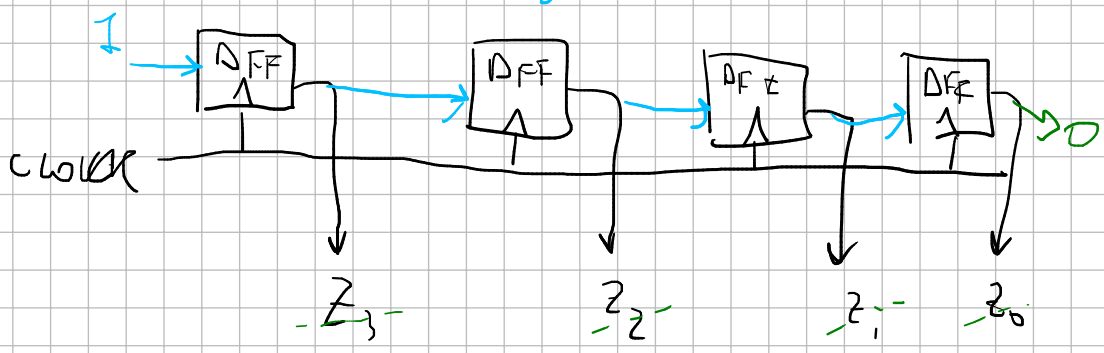




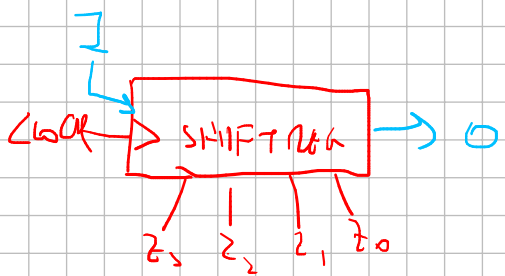
PARALLEL LOAD → all together

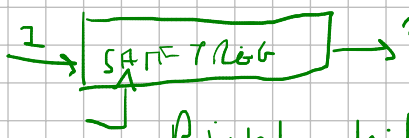
SHIFT REGISTER (4 BIT)

left to right

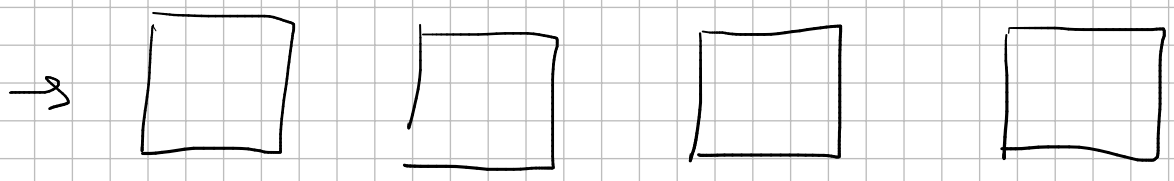


Right shift register with parallel output





Right shift reg no parallel



RIGHT SHIFT

WANT

1234

START 4

START 1 → 4 3 2 1

START 4

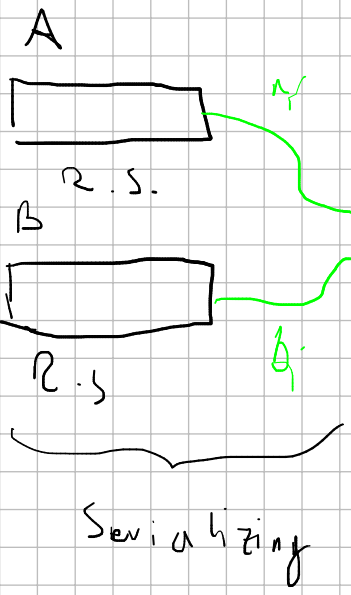
4 → 4 - - -

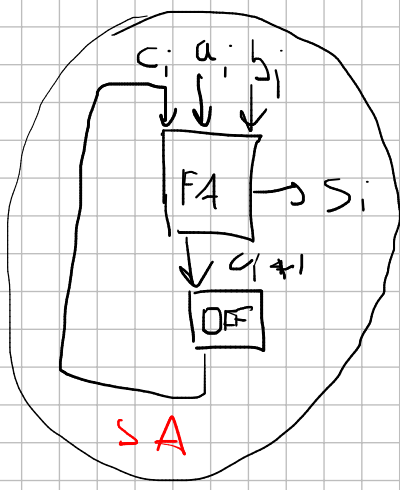
3 → 3 4 - -

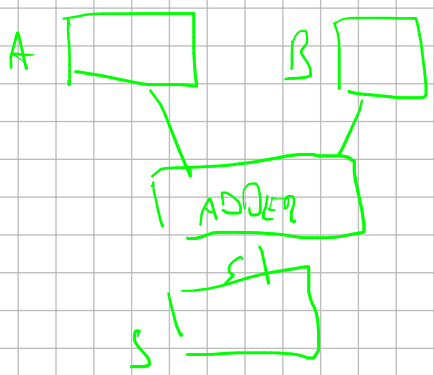
2 → 2 3 4 -

1 → 1 2 3 4

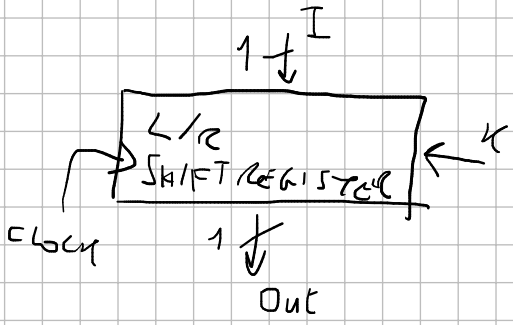
SERIALIZER





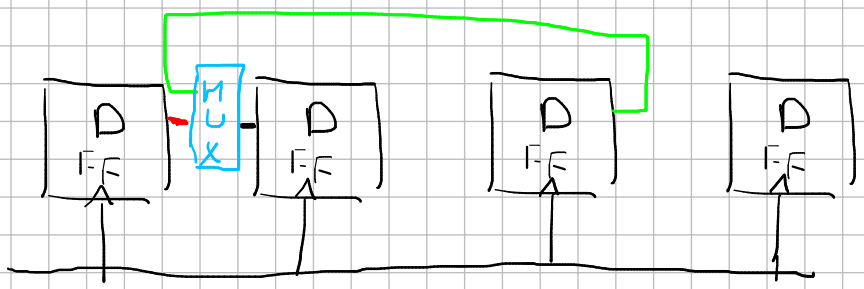


L/R REGISTER

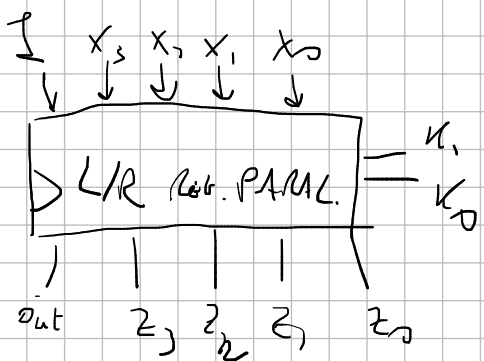


K = 0 right shift

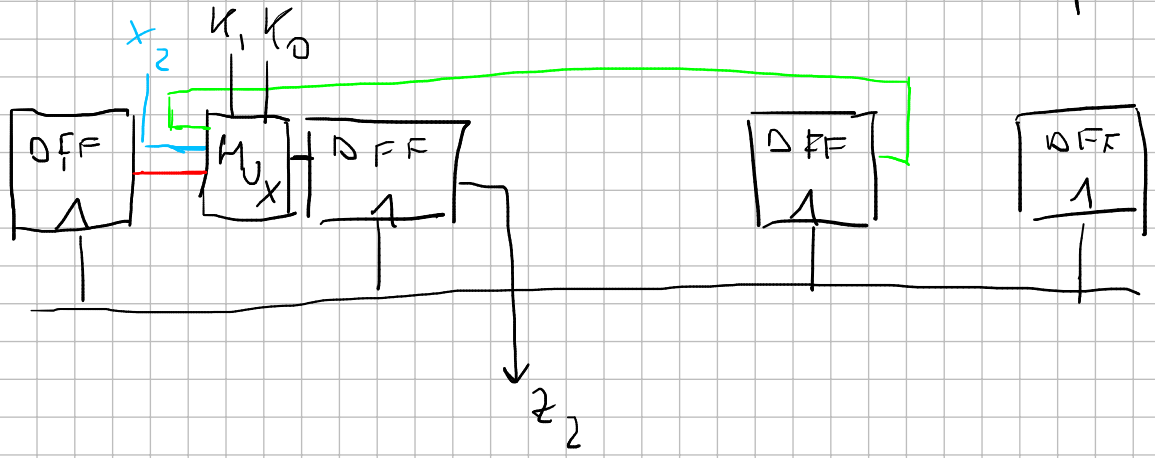
K = 1 left shift



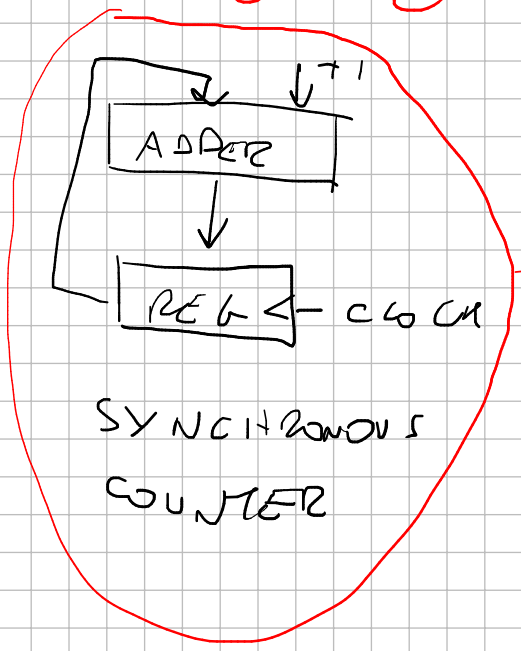
L/R REGISTER with parallel load

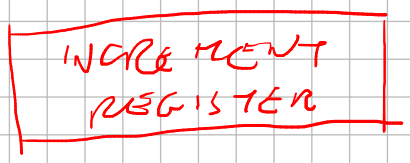


K_1	K_0	
0	0	shift right
0	1	shift left
1	–	parallel load



COUNTER



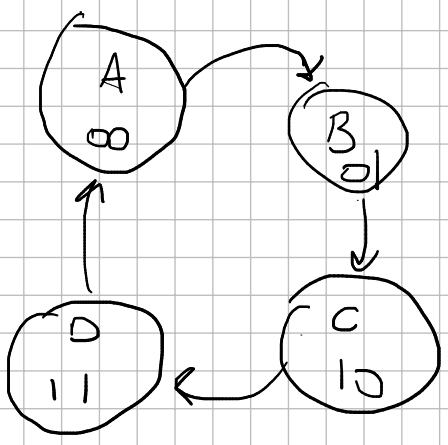


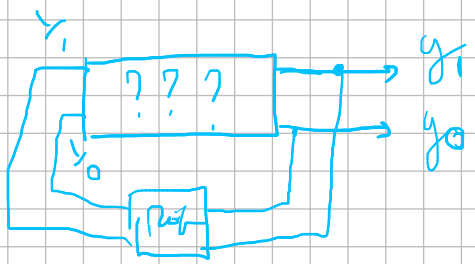
Program Counter / Instruction Pointer

AMA

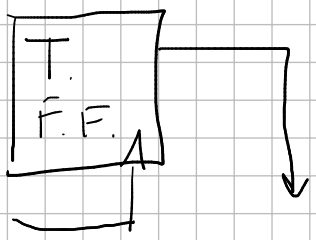
2 BIT COUNTER

Y_1	Y_0	y_1	y_0



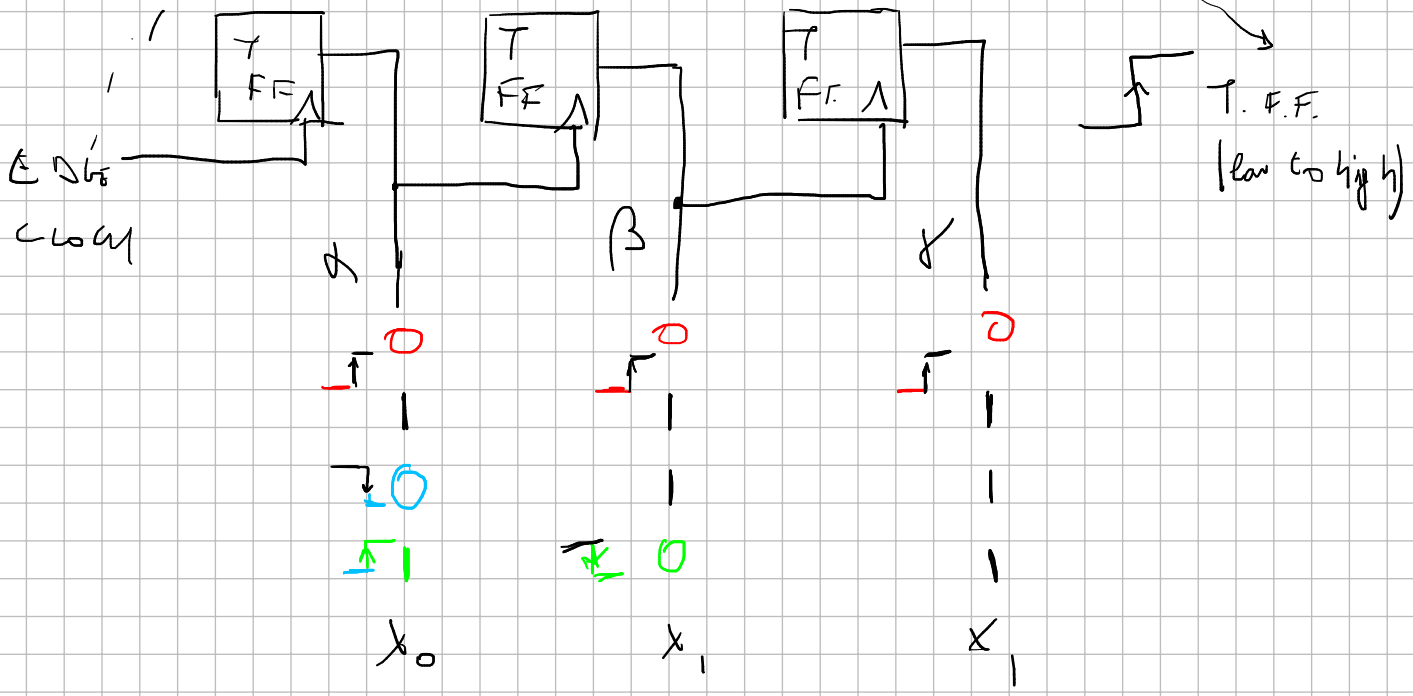


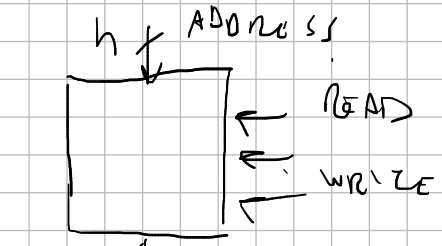
ASYNCHRONOUS COUNTER



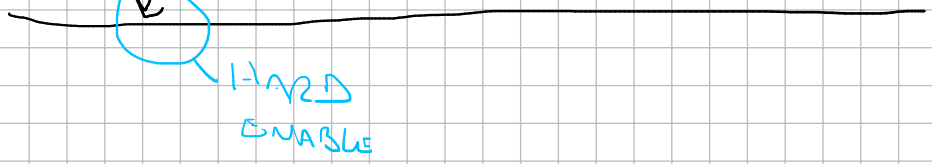
Negate previously stored value

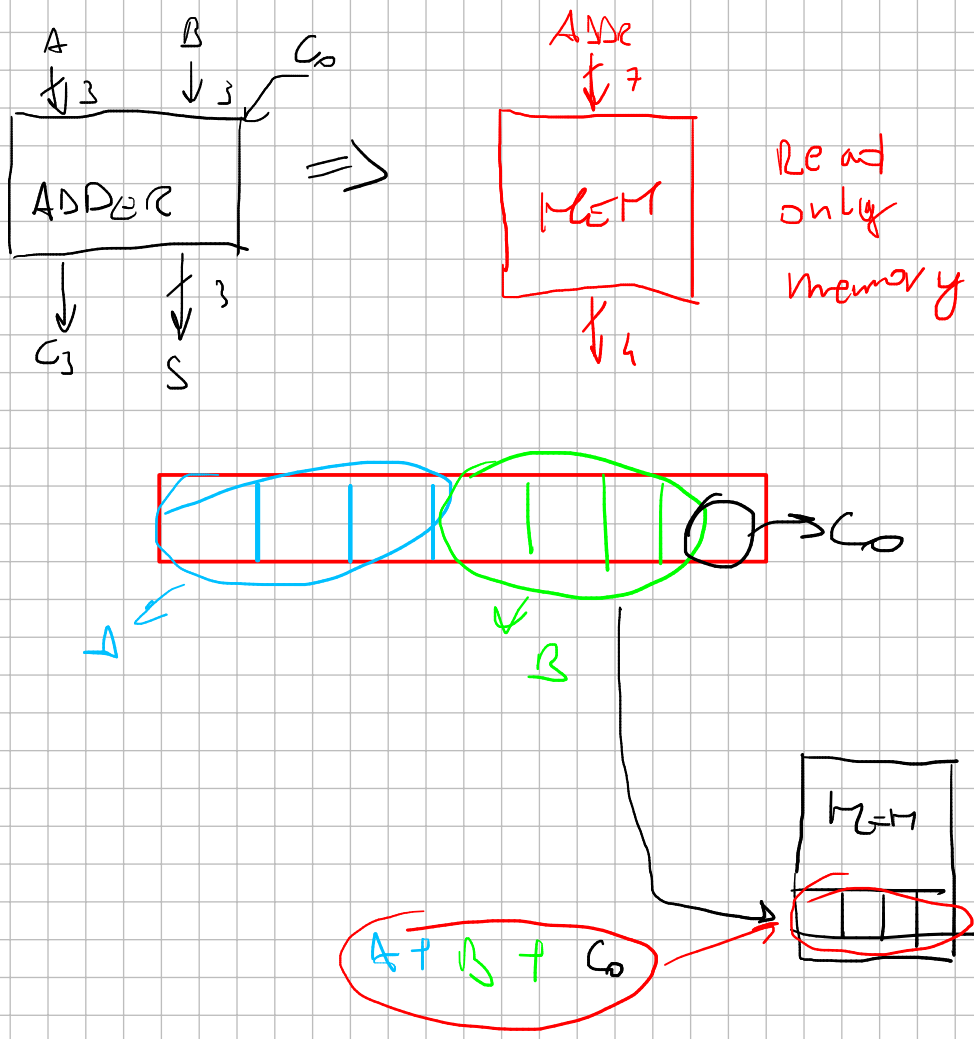
EDGE CLOCK





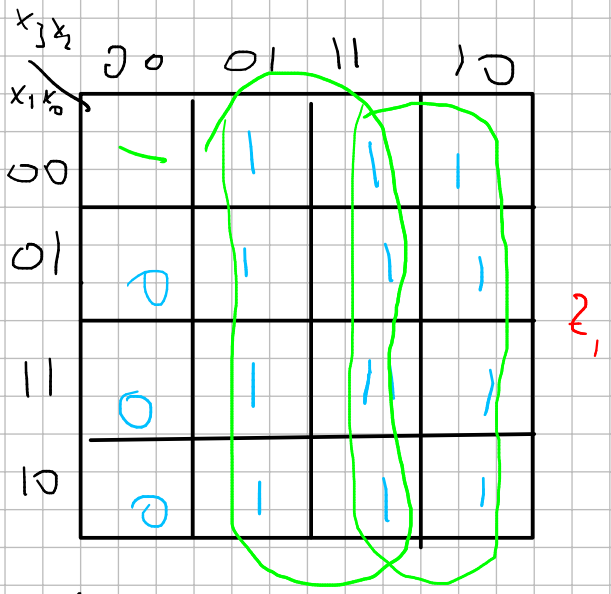
2^n entries each hosting m bits

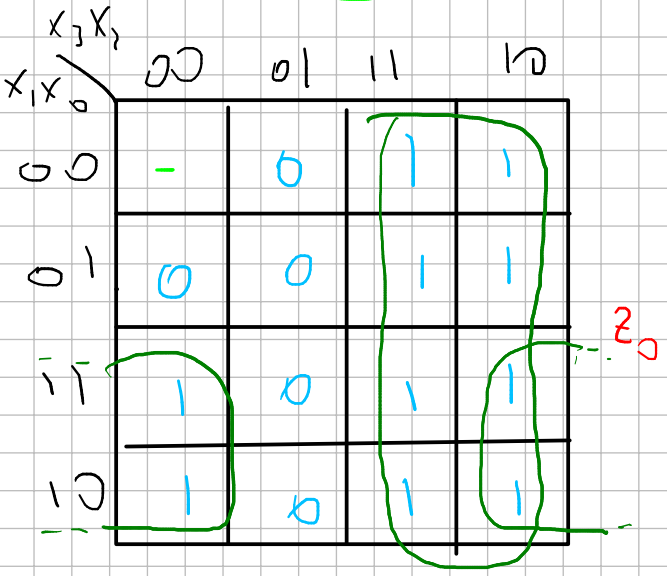



A
B
C_0
3
3
ADDER
C_1
3
S
Addr
7
MEM
4
Read only memory
C_0
A
B
MEM
A + B + C_0

PRIORITY ENCODER

X_3	X_2	X_1	X_0	Z_1	Z_0
0	0	0	1	0	0
0	0	1	–	0	1
0	1	–	–	1	0
1	–	–	–	1	1





$$\begin{cases} Z_1 = X_2 + X_3 \\ Z_0 = X_3 + \bar{X}_2 X_1 \end{cases}$$