

Project 05

Lab 08

4-bit adder

Sequential Logic

1-bit register

4-bit register

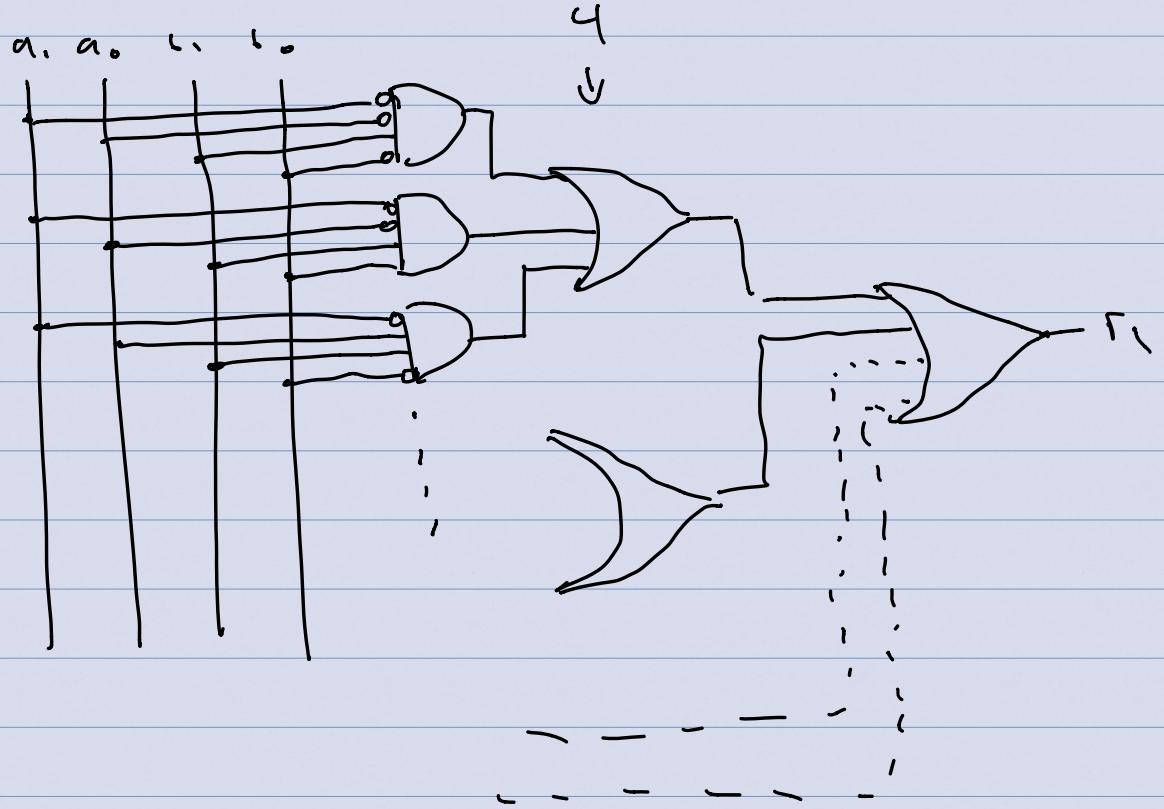
[4-bit counter]

Lab 08 Part 3

a b r $r = \max(a, b)$ 2-bit values

| $\overrightarrow{a_1\ a_0}$ | $\overrightarrow{b_1\ b_0}$ | $r_1\ r_0$ |
|-----------------------------|-----------------------------|------------|
| 0 0 | 0 0 | 0 0 |
| 0 0 | 0 1 | 0 1 |
| x 0 | 0 1 | 1 0 |
| x 0 | 1 - | 1 1 |
| 0 1 | 0 0 | 0 1 |
| 0 1 | 0 1 | 0 1 |
| x 0 | 1 0 | 1 0 |
| x 0 | 1 1 | 1 1 |
| x 1 | 0 0 | 1 0 |
| x 1 | 0 1 | 1 0 |
| x 1 | 1 0 | 1 0 |
| x 1 | 1 1 | 1 1 |
| x 1 | 0 0 | 1 1 |
| x 1 | 0 1 | 1 1 |
| x 1 | 1 0 | 1 1 |
| x 1 | 1 1 | 1 1 |
| 1 1 | 0 0 | 1 1 |
| 1 1 | 0 1 | 1 1 |
| 1 1 | 1 0 | 1 1 |
| 1 1 | 1 1 | 1 1 |

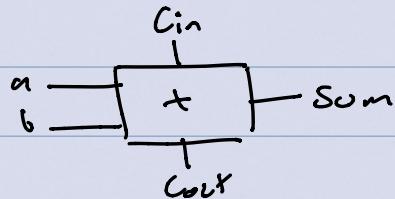
$$\begin{aligned} r_1 &= (\bar{a}_1 \cdot \bar{a}_0 \cdot b_1 \cdot \bar{b}_0) + (\bar{a}_1 \cdot \bar{a}_0 \cdot b_1 \cdot b_0) + (\bar{a}_1 \cdot a_0 \cdot b_1 \cdot \bar{b}_0) \\ &\quad + (\bar{a}_1 \cdot a_0 \cdot b_1 \cdot b_0) + (a_1 \cdot \bar{a}_0 \cdot \bar{b}_1 \cdot \bar{b}_0) + (a_1 \cdot \bar{a}_0 \cdot \bar{b}_1 \cdot b_0) \\ &\quad + (a_1 \cdot \bar{a}_0 \cdot b_1 \cdot \bar{b}_0) + (a_1 \cdot \bar{a}_0 \cdot b_1 \cdot b_0) + (a_1 \cdot a_0 \cdot \bar{b}_1 \cdot \bar{b}_0) \\ &\quad + (a_1 \cdot a_0 \cdot \bar{b}_1 \cdot b_0) + (a_1 \cdot a_0 \cdot b_1 \cdot \bar{b}_0) + (a_1 \cdot a_0 \cdot b_1 \cdot b_0) \end{aligned}$$



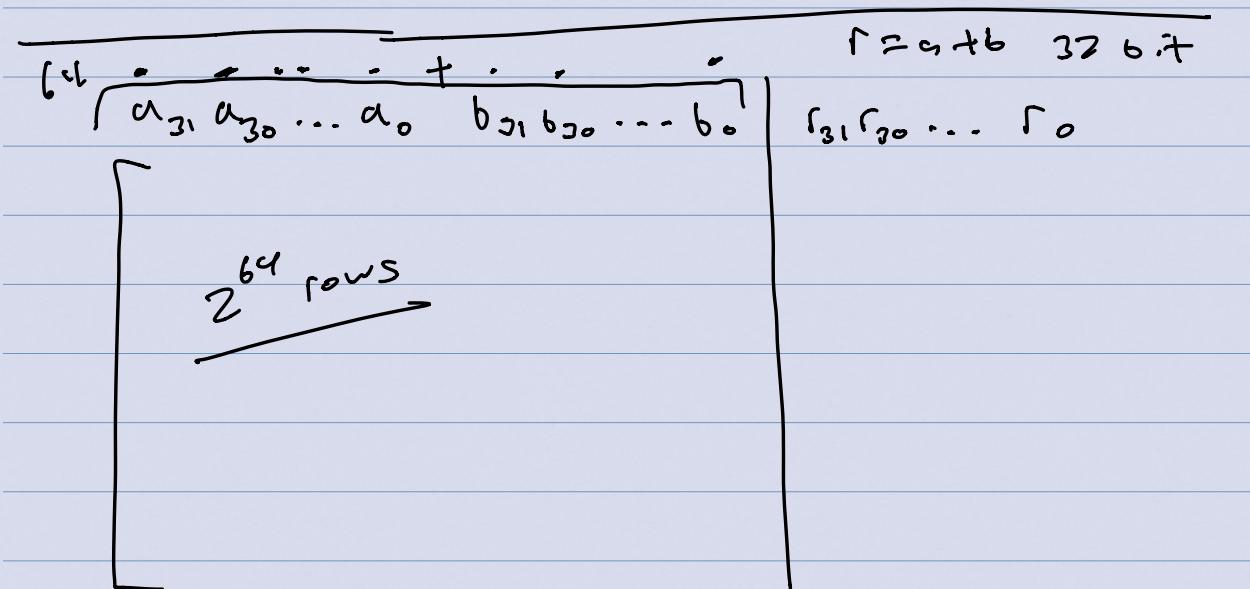
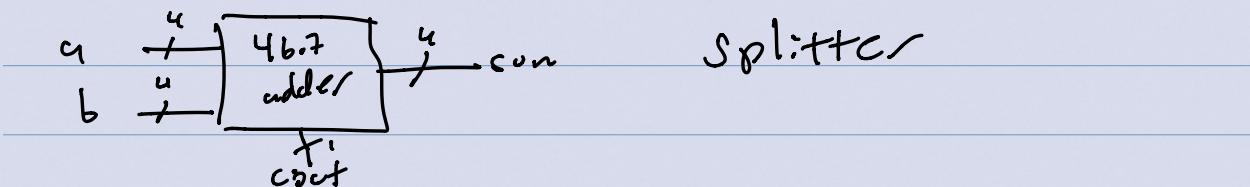
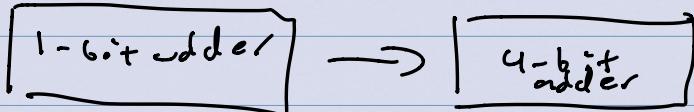
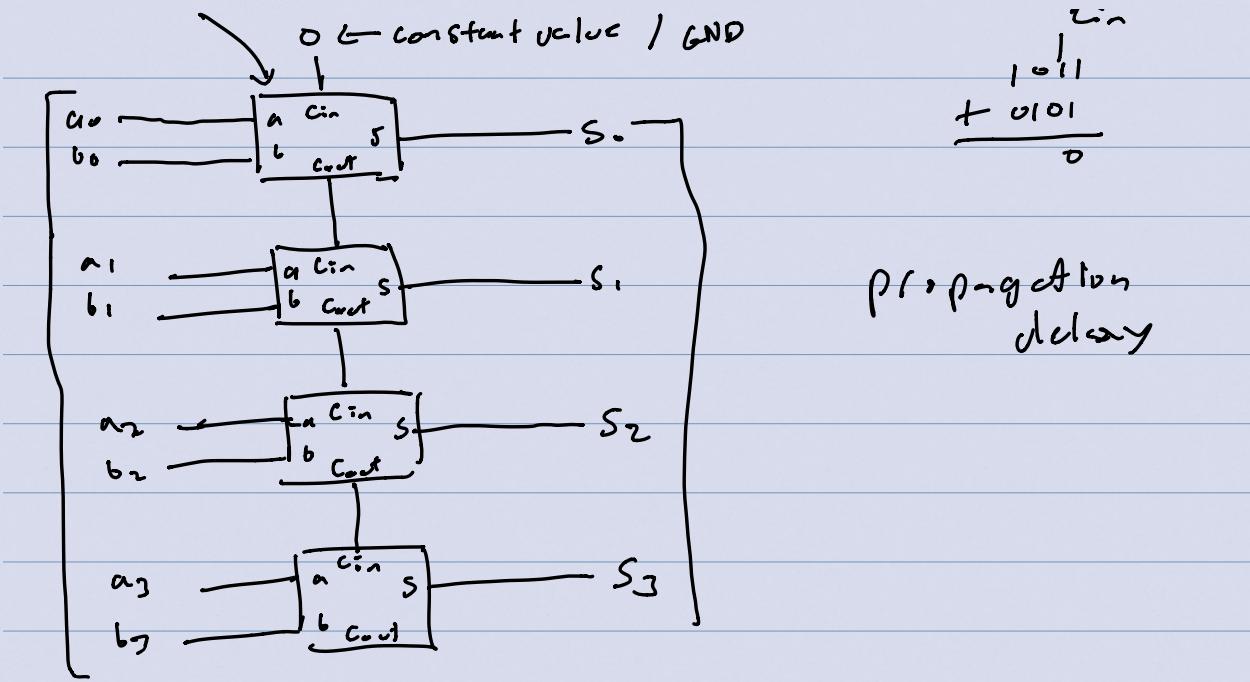
Can override 5 input max for gates

adder

1-bit full adder

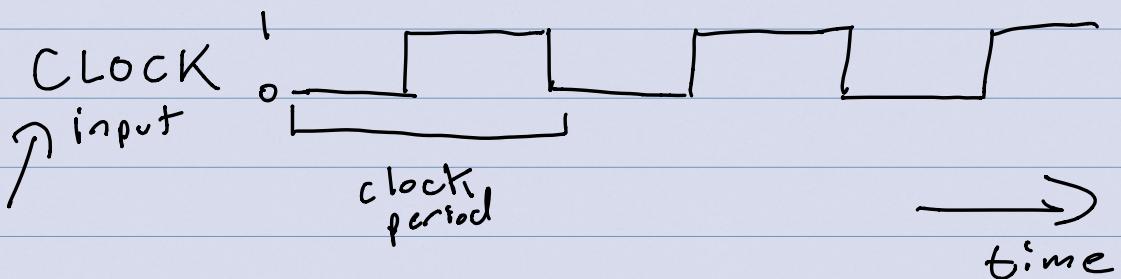
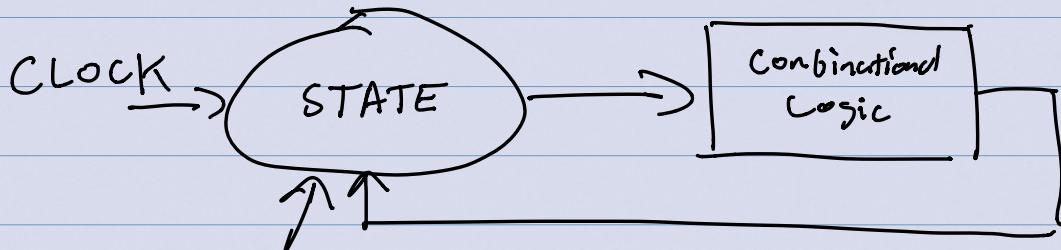


Ripple-carry adder



Carry Lookahead adder

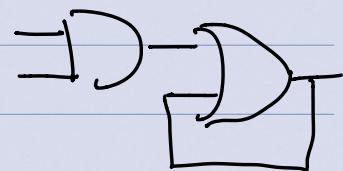
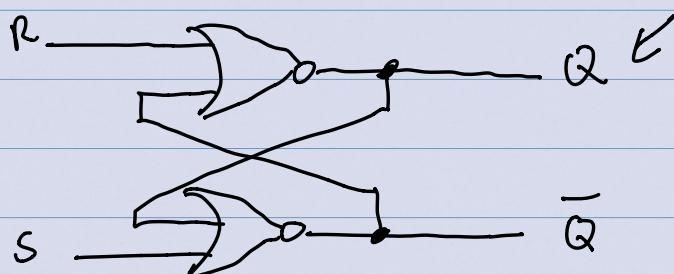
Sequential Logic Processor



STATE?

How to store 1 bit of information

SR Latch Set/Reset

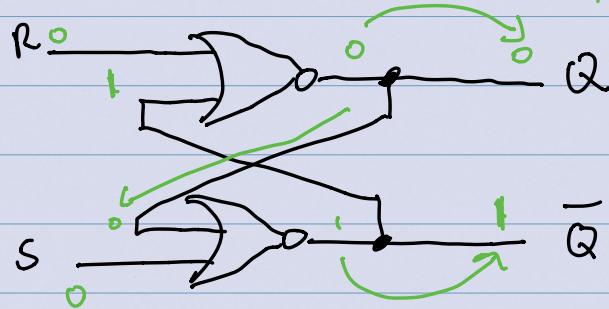
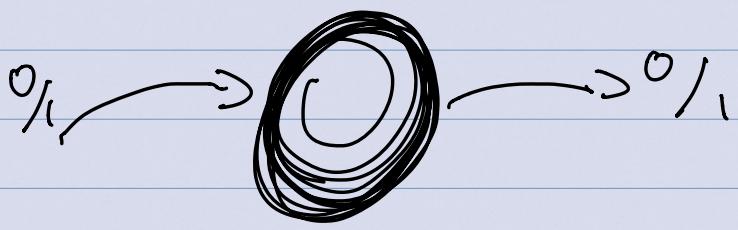


NOR



If $R = 1$ (high) \Rightarrow set latch to 0

If $S = 1$ (high) \Rightarrow set latch to 1



$$\begin{array}{r}
 R \quad S \quad Q \quad \bar{Q} \\
 \hline
 0 \quad 0 \quad 0 \quad 1 \\
 0 \quad 1 \quad 1 \quad 0 \\
 0 \quad 0 \quad 1 \quad 0
 \end{array}$$

