

Name	Identity	Dual
Idempotent	$xx = x$	$x + x = x$
Complement	$x\bar{x} = 0$	$x + \bar{x} = 1$
Commutativity	$xy = yx$	$x + y = y + x$
Associativity	$x(yz) = (xy)z$	$x + (y + z) = (x + y) + z$
Distributivity	$x(y + z) = xy + xz$	$x + (yz) = (x + y)(x + z)$
Absorption	$x + xy = x$	$x(x + y) = x$
Redundancy	$x + \bar{x}y = x + y$	$x(\bar{x} + y) = xy$
Combination	$xy + x\bar{y} = x$	$(x + y)(x + \bar{y}) = x$
DeMorgan's	$\overline{xy} = \bar{x} + \bar{y}$	$\overline{x + y} = \bar{x}\bar{y}$
Consensus	$xy + \bar{x}z + yz = xy + \bar{x}z$	$(x + y)(\bar{x} + z)(y + z) = (x + y)(\bar{x} + z)$

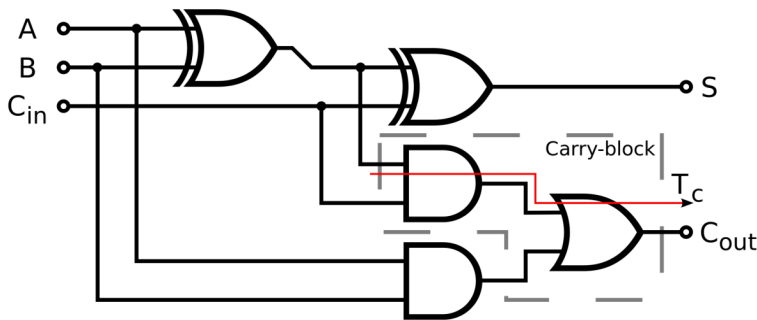
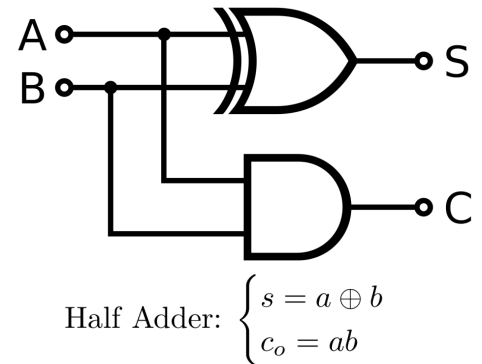
$x_3 \backslash x_1x_2$	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

$x_3x_4 \backslash x_1x_2$	00	01	11	10
00	m_0	m_4	m_{12}	m_8
01	m_1	m_5	m_{13}	m_9
11	m_3	m_7	m_{15}	m_{11}
10	m_2	m_6	m_{14}	m_{10}

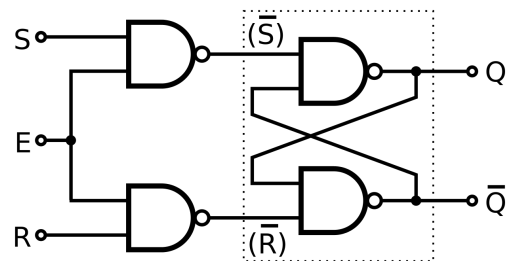
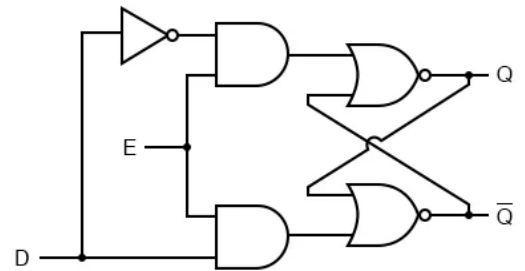
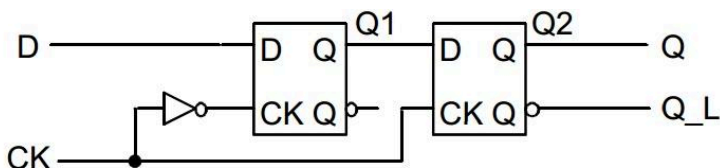
$x_2 \backslash x_1$	0	1
0	m_0	m_2
1	m_1	m_3

$$a \oplus b = \bar{a}b + a\bar{b}$$

Op	NAND	Circuit	Op	NOR	Circuit
\bar{x}	\overline{xx}		\bar{x}	$\overline{x+x}$	
xy	$\overline{\overline{xy}}$		xy	$\overline{\bar{x} + \bar{y}}$	
$x + y$	$\overline{\bar{x}\bar{y}}$		$x + y$	$\overline{\overline{x+y}}$	



$$\text{Full Adder: } \begin{cases} s = a \oplus b \oplus c_i \\ c_o = c_i a + c_i b + ab \end{cases}$$



Using Interrupts:

1. Configure interrupt source
2. `csrrw zero, utvec, <handler_addr_reg>`
3. `csrrsi zero, ustatus, 0x1`
4. `csrrsi zero, uie, <mask>` (bit 4 for timer)
5. In handler, save *all* registers, then `uret`

$$\Delta = \Delta_{\text{sink}} - \Delta_{\text{src}}$$

$$t_{\text{min}} \geq t_{CQ} + t_{\text{logic,max}} + t_{su} - \Delta$$

$$t_h + \Delta \leq t_{CQ} + t_{\text{logic,min}}$$

- Use `<=` for assignment in `always_ff`.
- Mux: $m = \bar{s}x + sy$, $m = x$ when 0, $m = y$ when 1.
- Implicant: any product term covered by f .
- Prime Implicant: an implicant that can't have any literal removed without making it invalid.
- Essential PI: a PI that covers at least one minterm not covered by any other PI.
- RS latches oscillate when R and S are pulled high then low at the same time.
- Caller save: `t0-t7`, `a0-a7`, `sp`; callee save: `s0-s11`
- Overflow occurs when MSB of result and carry-out are different (except positive + negative)