

Latch: CC-BY Rberteig@flickr

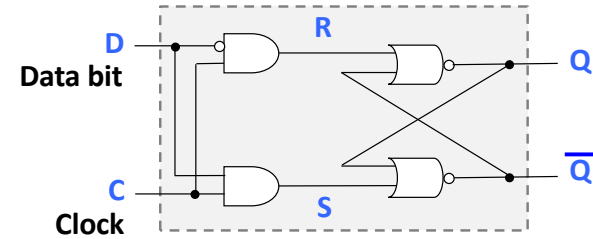


Latches, Flip-flops, Registers, Memory

Sequential logic: elements to store values
Output depends on inputs *and stored values*.

(vs. combinational logic: output depends only on inputs)

D latch



if $C = 0$, then SR latch stores current value of Q.

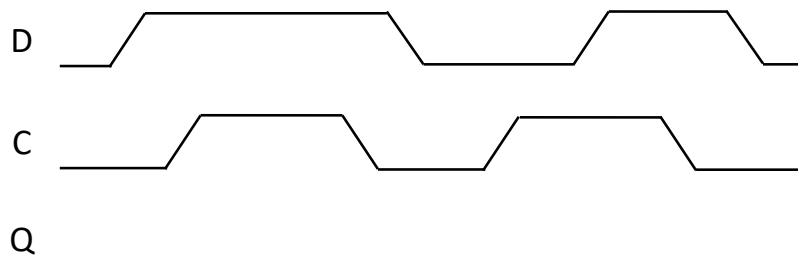
if $C = 1$, then D flows to Q:

if $D = 0$, then $R = 1$ and $S = 0$, $Q = 0$

if $D = 1$, then $R = 0$ and $S = 1$, $Q = 1$

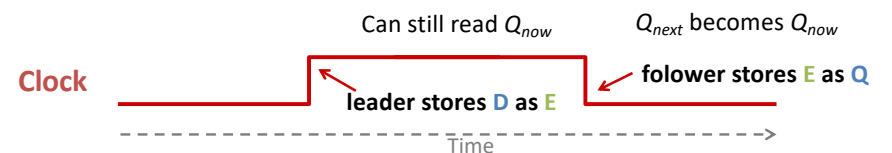
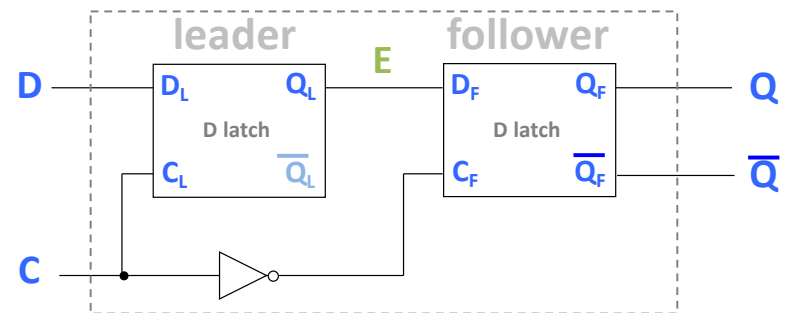
Time matters!

ex



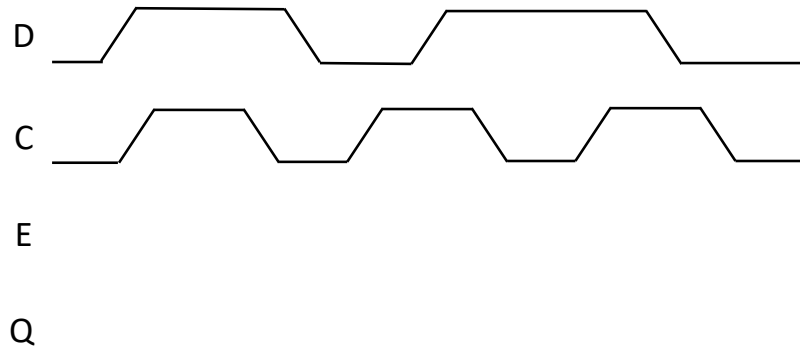
Assume Q has an initial state of 0

D flip-flop with falling-edge trigger



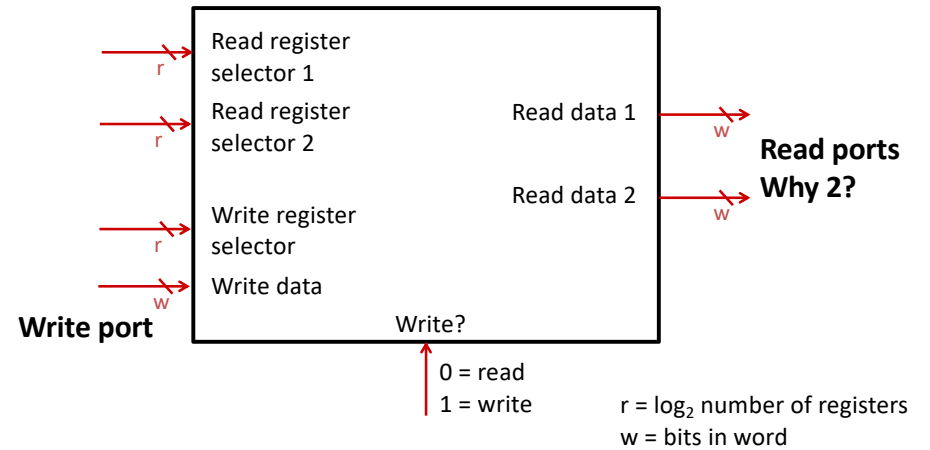
Time matters!

ex



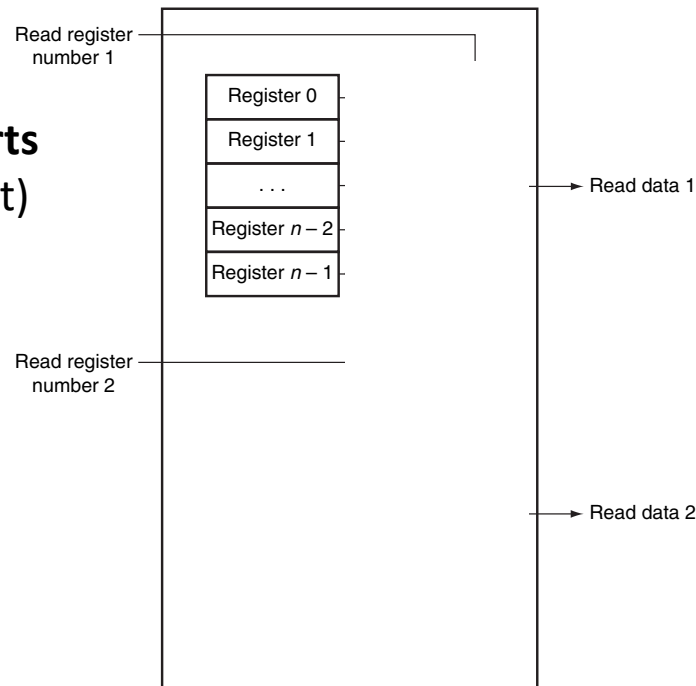
Assume Q and E have an initial state of 0

Register file



Array of registers, with register selectors, write/read control, input port for writing data, output ports for reading data.

Read ports (data out)



Write port (data in)

