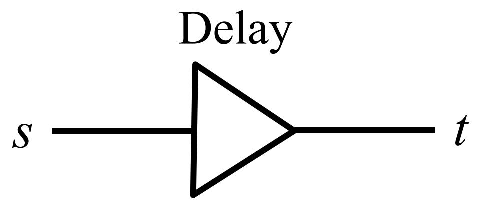
Sequential logic

- Sequential logic has memory: past values of a signal influence the present values
- We add a way for the past to influence the present: a Delay gate

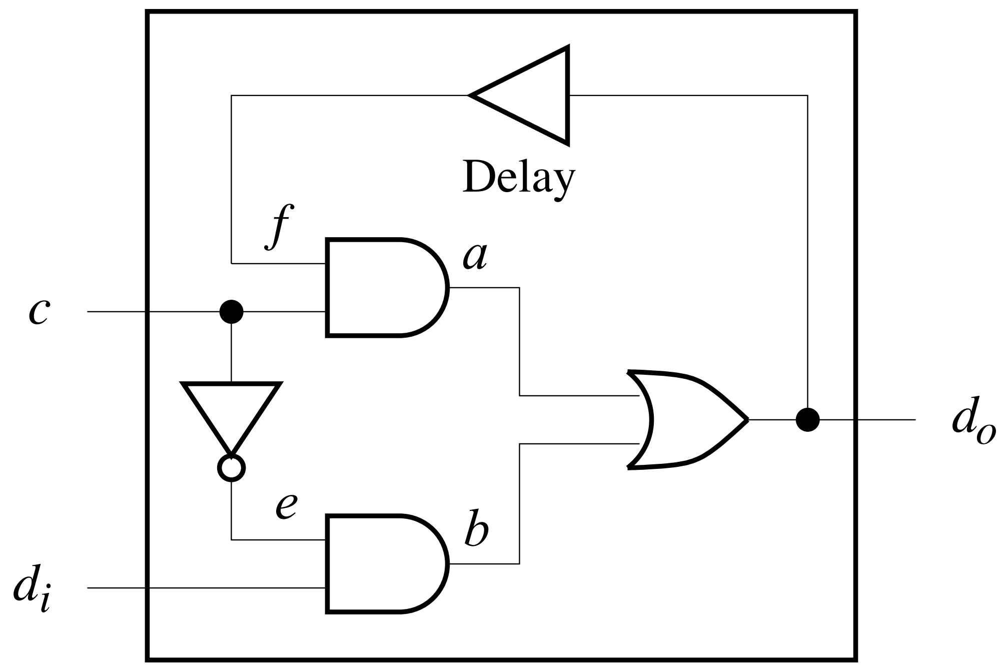
$S=a_0|a_1|a_2|\ldots|a_i|\ldots$
$T=b_0|b_1|b_2|\ldots|b_i|\ldots$



```
fun {DelayG S} 0|S end
```

$b_i=a_{i-1} \Rightarrow T=0|S$

Latch specification



- A latch is a simple circuit with memory; it has two stable states and can memorize its input
- Output d_o follows input d_i and freezes when c is 1

Latch implementation

- Latch creation as a three-argument component:

```
proc {Latch C Di Do}
   A B E F
in
   F={DelayG Do}
   A={AndG C F}
   E={NotG C}
   B={AndG E Di}
   Do={OrG A B}
end
```