Sistemas Digitais I

LESI - 2º ano

Lesson 8 - Sequential Design Practices

Prof. João Miguel Fernandes
(miguel@di.uminho.pt)

Dept. Informática

UNIVERSIDADE DO MINHO
ESCOLA DE ENGENHARIA

8. Sequential Practices

- Introduction (1) -

- Just about every real digital system is a sequential system.
- All it takes is one feedback loop, latch or flip-flop to make a circuit's present behaviour to depend on its past inputs.
- We typically deal with digital systems by partitioning them into data paths, registers and control units.
- A typical system has multiple functional units with well-defined interfaces and connections between them.
- Each functional unit may contain a hierarchy with several layers of abstraction.

8. Sequential Practices

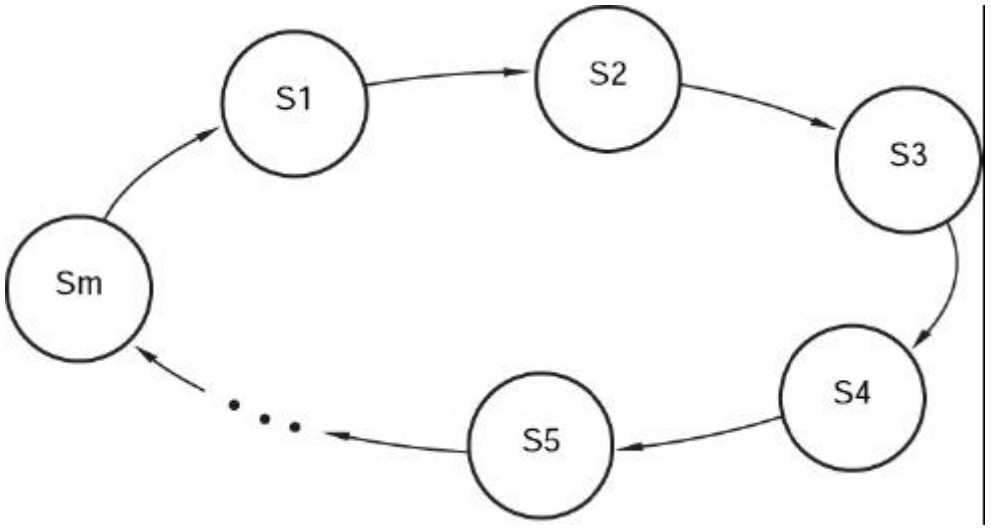
- Introduction (2) -

- The heart of any system is usually a state machine.
- During the 80s, designers wrote out state tables by hand and built corresponding circuits using the traditional synthesis methods.
- Today, most state tables are specified with an HDL.
- The HDL compiler then performs the equivalent of the synthesis methods and realizes the machine in a given target technology.

8. Sequential Practices

- Counters (1) -

- The name counter is used for any clocked sequential circuit whose state diagram contains a single cycle.

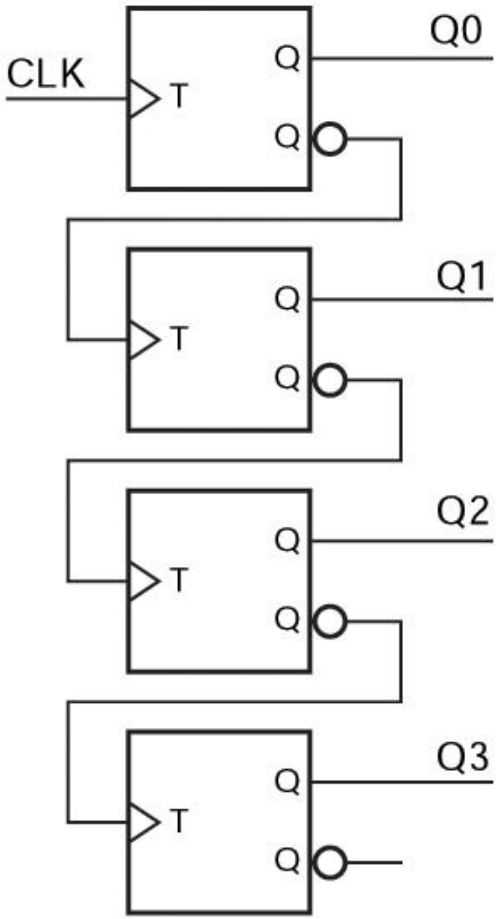


- The modulus of a counter is the number of states in the cycle.
- A counter with m states is called a modulo-m counter.
- A counter with a non-power-of-2 modulus has extra states that are not used in normal operation.

8. Sequential Practices

- Counters (2) -

- The most common counter is the n-bit binary counter.
- This counter has n flip-flops and has 2^n states, which are visited in the sequence 0, 1, 2,…, 2^n-1, 0, 1, …
- Each state is encoded as the corresponding n-bit binary integer.
- An n-bit binary counter can be constructed with just n flip-flops and no other components.
- Each bit of the counter toggles when the preceding bit changes from 1 to 0.
- The counter is called a ripple counter because the carry information ripples from the LSB to the MSB.



8. Sequential Practices

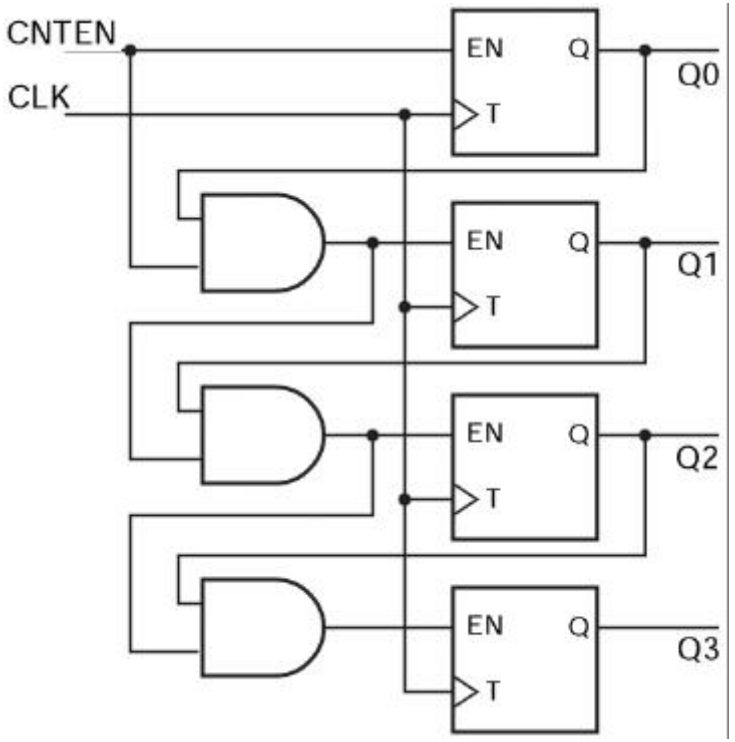
- Counters (3) -

- Although a ripple counter requires fewer components than any other type of counter, it is also slower.
- In the worst case, when the MSB must change, the output is not valid until time $n.t_{TQ}$ after the rising edge of clock.
  (t_{TQ} is the propagation delay from input to output of a T flip-flop).
- A synchronous counter connects all of its flip-flop clock inputs to the same common CLK signal.
- Thus, all of the flip-flops outputs change at the same time, after only t_{TQ} delay.

8. Sequential Practices

- Counters (4) -

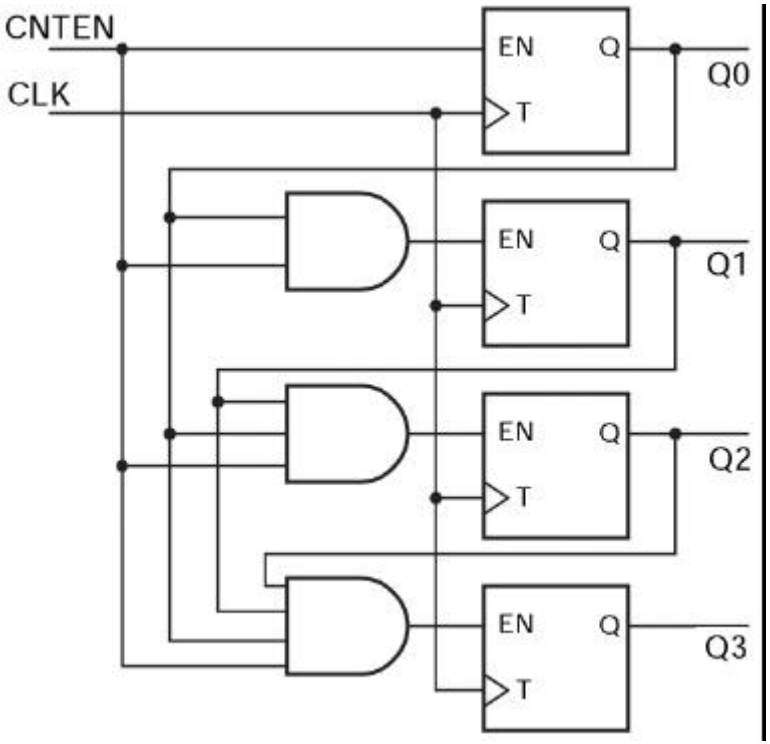
- This requires the use of T flip-flops with enable input.
- The output toggles on the rising edge of the T only if EN is asserted.
- Each T flip-flop toggles if CNTEN is asserted and all the lower-order bits are 1.
- A synchronous n-bit binary counter can be built with a fixed amount of logic per bit.
- In this case, a T flip-flop with enable and a 2-input AND gate.
- The counter is a synchronous serial counter because the enable signals propagate serially from the LSB to the MSB.



8. Sequential Practices

- Counters (5) -

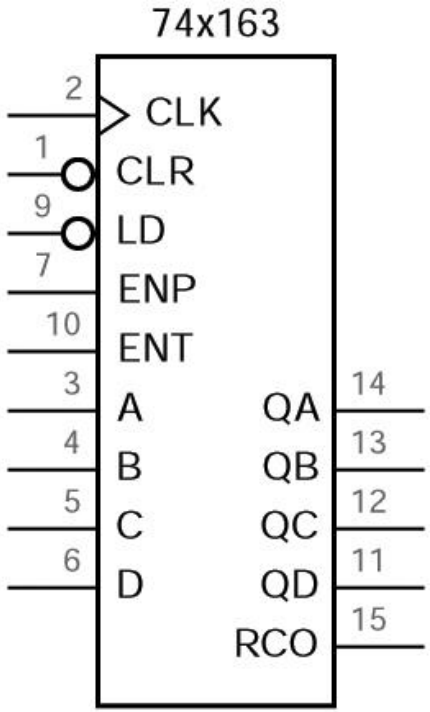
- If the clock period is too short, there may be no time for a change in the LSB to propagate to the MSB.
- This problem is eliminated if each EN input is driven by a dedicated AND gate (1-level of logic).
- This is a synchronous parallel counter.
- It is the fastest binary counter structure.



8. Sequential Practices

- Counters (6) -

- 74x163 is a synchronous 4-bit binary counter.
- RCO=1 when all count bits are 1 and ENT is asserted.



Inputs				Current State				Next State			
CLR_L	LD_L	ENT	ENP	QD	QC	QB	QA	QD*	QC*	QB*	QA*
0	x	x	x	x	x	x	x	0	0	0	0
1	0	x	x	x	x	x	x	D	C	B	A
1	1	0	x	x	x	x	x	QD	QC	QB	QA
1	1	x	0	x	x	x	x	QD	QC	QB	QA
1	1	1	1	0	0	0	0	0	0	0	1
1	1	1	1	0	0	0	1	0	0	1	0
1	1	1	1	0	0	1	0	0	0	1	1
1	1	1	1	0	0	1	1	0	1	0	0
1	1	1	1	0	1	0	0	0	1	0	1
1	1	1	1	0	1	0	1	0	1	1	0
1	1	1	1	0	1	1	0	0	1	1	1
1	1	1	1	0	1	1	1	1	0	0	0
1	1	1	1	1	0	0	0	1	0	0	1
1	1	1	1	1	0	0	1	1	0	1	0
1	1	1	1	1	0	1	0	1	0	1	1
1	1	1	1	1	0	1	1	1	1	0	0
1	1	1	1	1	1	0	0	1	1	0	1
1	1	1	1	1	1	0	1	1	1	1	0
1	1	1	1	1	1	1	0	1	1	1	1
1	1	1	1	1	1	1	1	0	0	0	0

8. Sequential Practices

- Counters (7) -

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;

entity V74x163 is
    port ( CLK, CLR_L, LD_L, ENP, ENT: in STD_LOGIC;
           D: in UNSIGNED (3 downto 0);
           Q: out UNSIGNED (3 downto 0);
           RCO: out STD_LOGIC );
end V74x163;

architecture V74x163_arch of V74x163 is
signal IQ: UNSIGNED (3 downto 0);
begin
process (CLK, ENT, IQ)
  begin
    if (CLK'event and CLK='1') then
      if CLR_L='0' then IQ <= (others => '0');
      elsif LD_L='0' then IQ <= D;
      elsif (ENT and ENP)='1' then IQ <= IQ + 1;
      end if;
    end if;
    if (IQ=15) and (ENT='1') then RCO <= '1';
    else RCO <= '0';
    end if;
    Q <= IQ;
  end process;
end V74x163_arch;
```

8. Sequential Practices

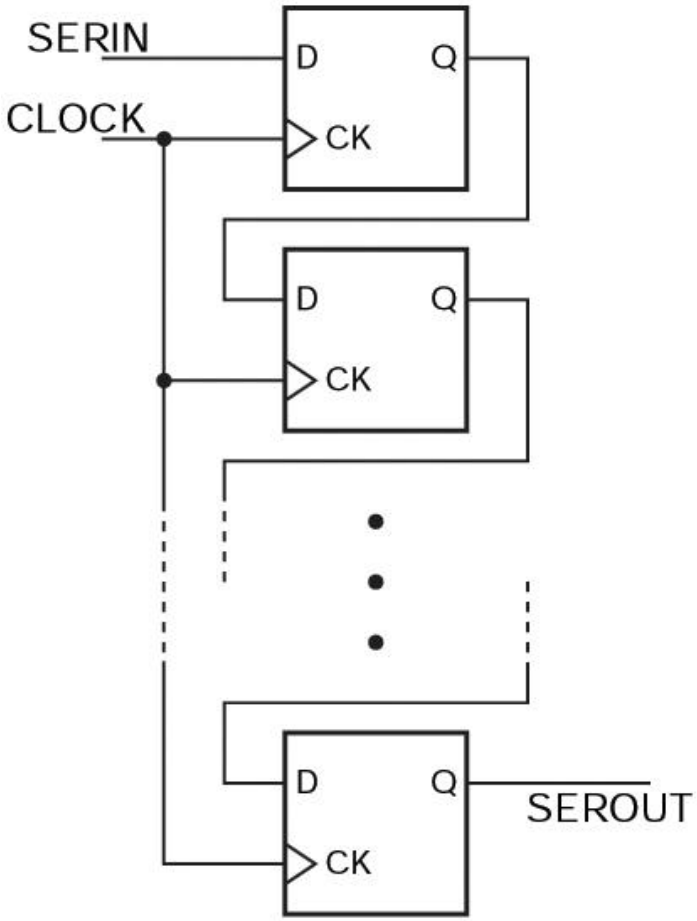
- Counters (7) -

```
architecture V74xs3_arch of V74x163 is
signal IQ: UNSIGNED (3 downto 0);
begin
process (CLK, ENT, IQ)
  begin
    if CLK'event and CLK='1' then
      if CLR_L='0' then IQ <= (others => '0');
      elsif LD_L='0' then IQ <= D;
      elsif (ENT and ENP)='1' and (IQ=12) then IQ <= ('0','0','1','1');
      elsif (ENT and ENP)='1' then IQ <= IQ + 1;
      end if;
    end if;
    if (IQ=12) and (ENT='1') then RCO <= '1';
    else RCO <= '0';
    end if;
    Q <= IQ;
  end process;
end V74xs3_arch;
```

8. Sequential Practices

- Shift Registers (1) -

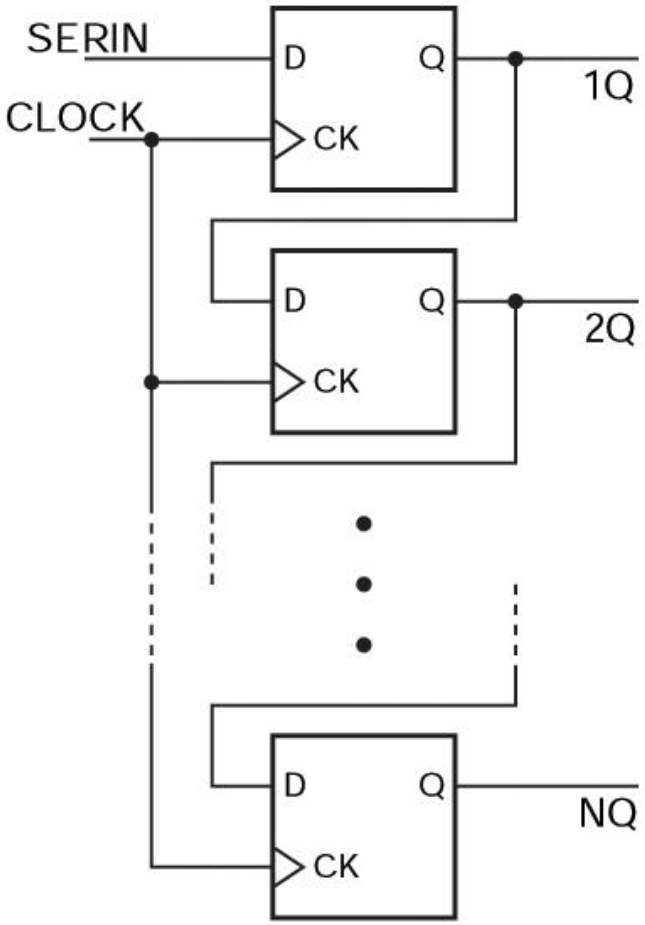
- A shift register is an n-bit register with a provision for shifting the stored data by one position at each clock pulse.
- A serial-in serial-out shift register has one input (SERIN) and one output (SEROUT).
- The SERIN input specifies a new bit to be shifted into one end, at each clock tick.
- That bit appears at the SEROUT output after n clock tick and is lost one tick later.
- An n-bit serial-in serial-out shift register can be used to delay a signal by n clock ticks.



8. Sequential Practices

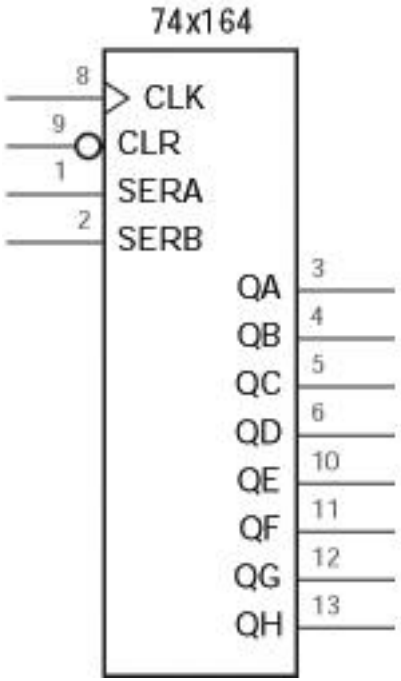
- Shift Registers (2) -

- A serial-in parallel-out shift register has outputs for all of its stored bits, making them available.
- Such a shift register can be used to perform serial-to-parallel conversions.
- Conversely, it is possible to build a parallel-in serial-out shift register.

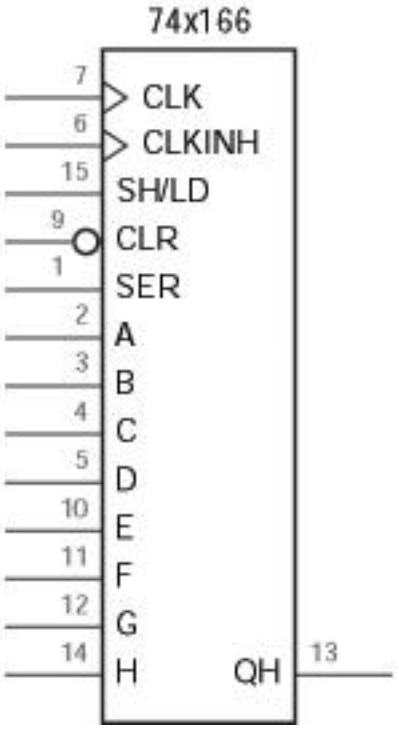


8. Sequential Practices

- Shift Registers (3) -



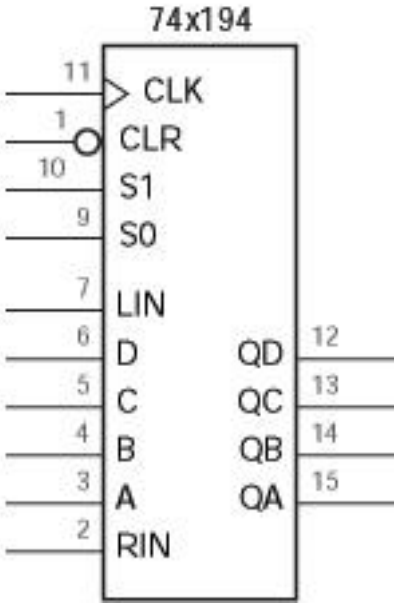
- These 8-bit shift registers are unidirectional because they shift in only one direction.
- The ‘164 is a serial-in, parallel-out device with an asynchronous clear input.
- The ‘166 is a parallel-in, parallel-out device, also with an asynchronous clear input.



8. Sequential Practices

- Shift Registers (4) -

- The 74x194 is an MSI 4-bit bidirectional, parallel-in, parallel-out shift register.
- This shift register is bidirectional because its contents may be shifted in either of two directions, depending on a control input.
- The ’194 is sometimes called an universal shift register, because it can be made to function like the less general previous ones.

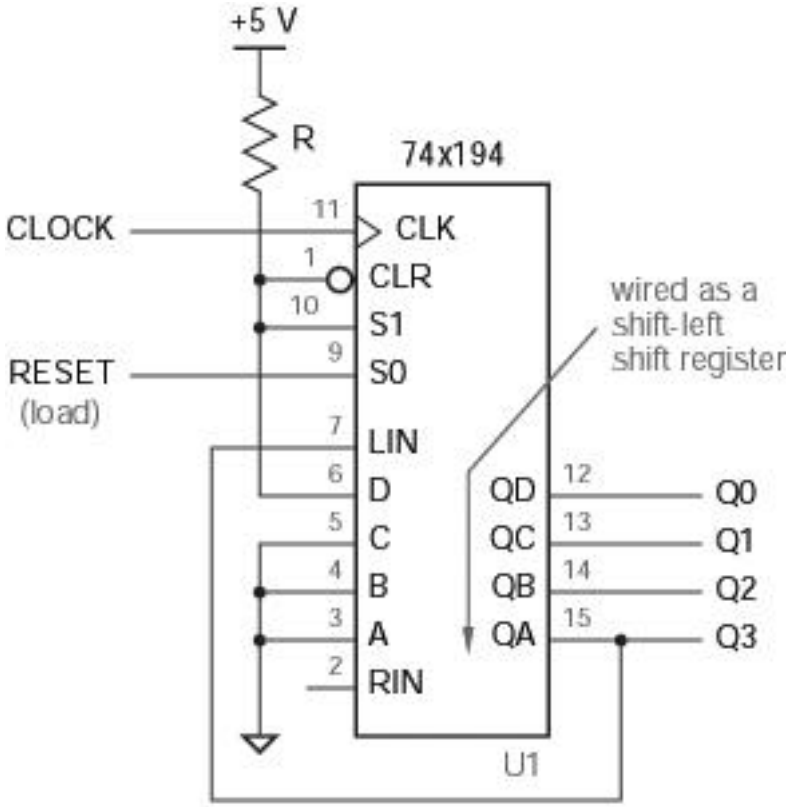


Function	Inputs		Next state			
	S1	S0	QA*	QB*	QC*	QD*
Hold	0	0	QA	QB	QC	QD
Shift right	0	1	RIN	QA	QB	QC
Shift left	1	0	QB	QC	QD	LIN
Load	1	1	A	B	C	D

8. Sequential Practices

- Shift Register Counters (1) -

- A shift register can be combined with combinatorial logic to form a state machine whose diagram is cyclic.
- Such a circuit is called a shift-register counter.
- Unlike a binary counter, a shift-register counter does not count in order, but it is useful in many control applications.
- The simplest shift-register counter uses an n-bit shift register to obtain a counter with n states, and is called a ring counter.

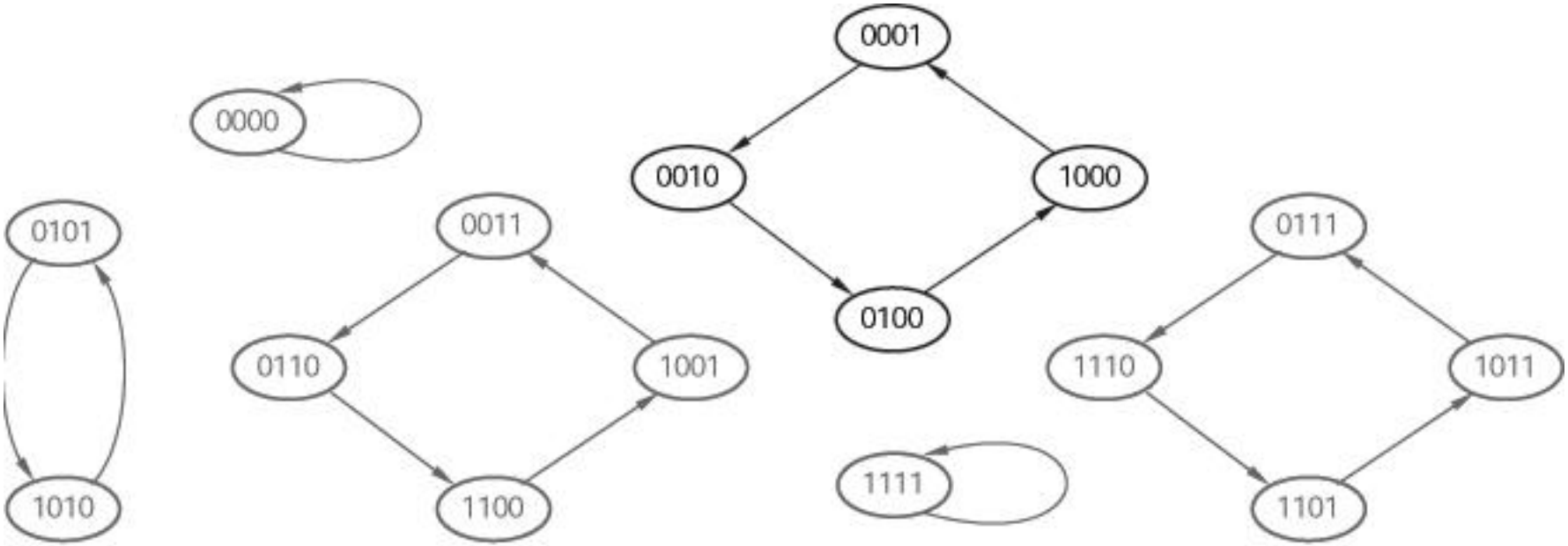


- A 4-bit ring counter can be obtained with a 74x149.

8. Sequential Practices

- Shift Register Counters (2) -

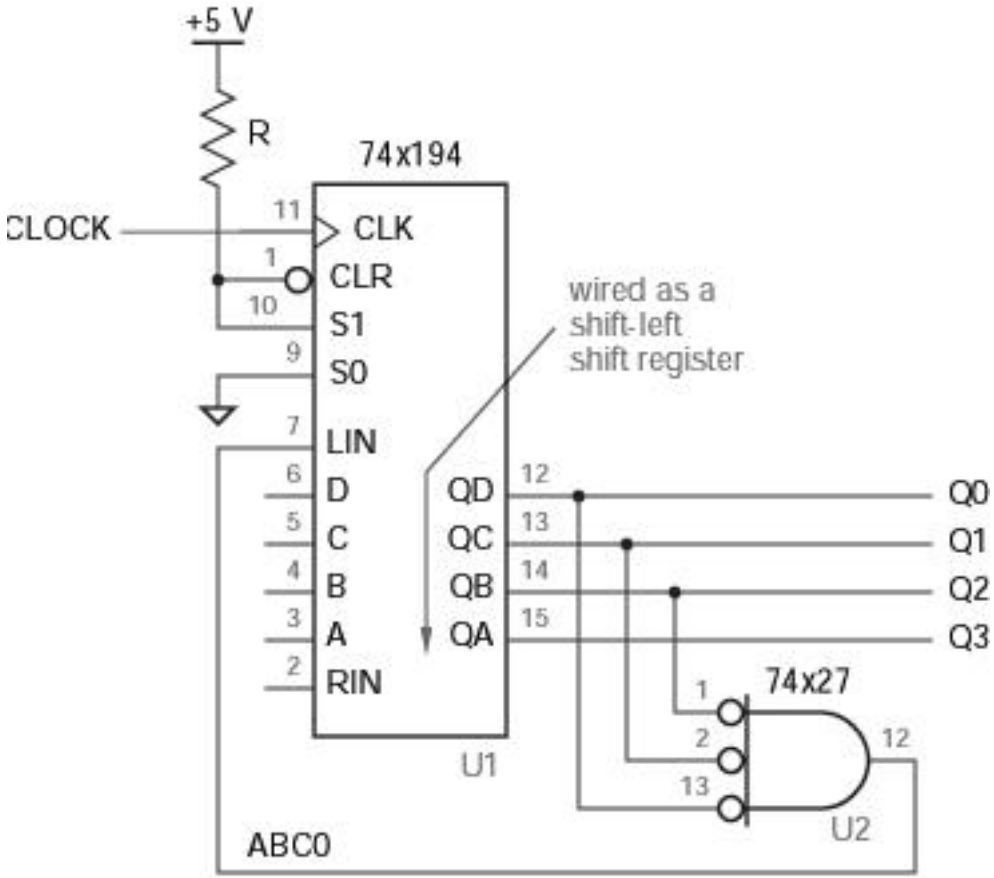
- When RESET is asserted, it loads 0001 (initial state).
- The circuit performs normally a shift left.
- This circuit is not robust. If the counter somehow gets off the normal cycle, it stays off it.



8. Sequential Practices

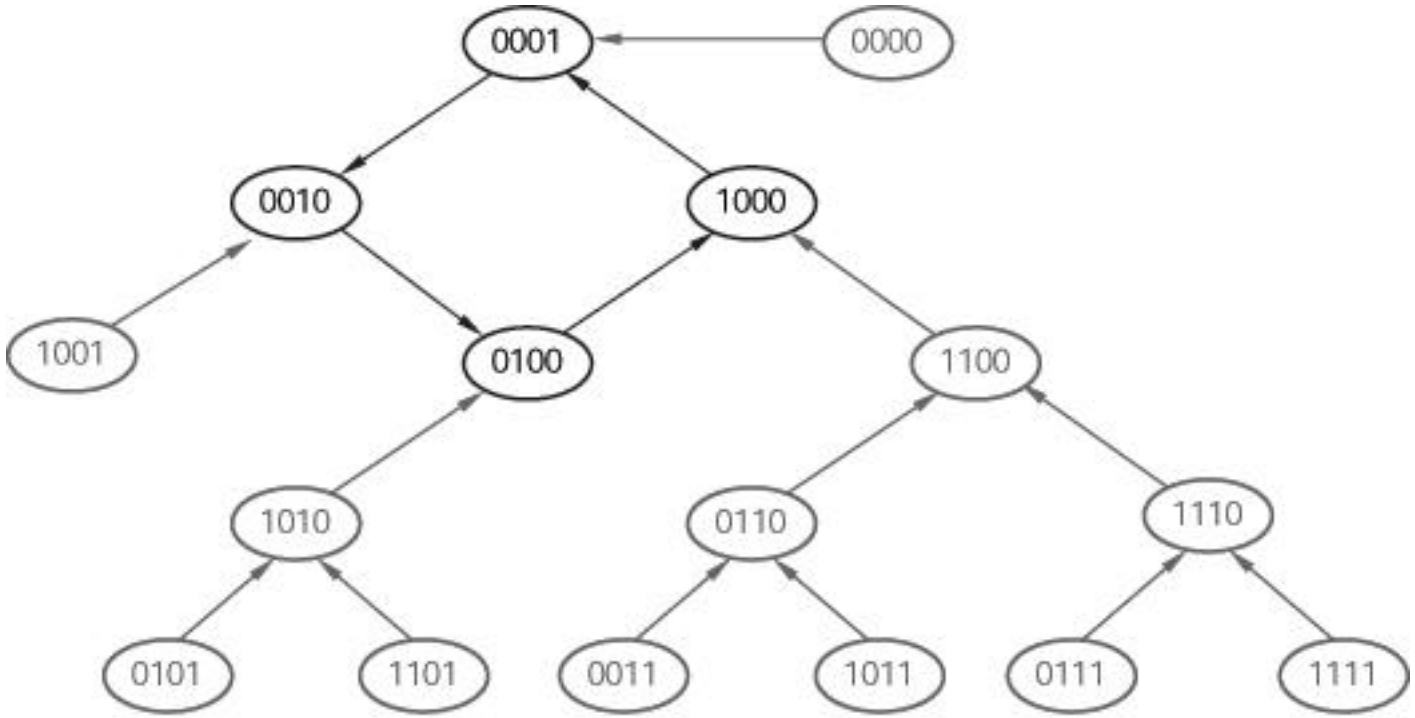
- Shift Register Counters (3) -

- A self-correcting counter is designed so that all abnormal states have transitions leading to normal states.
- A 4-bit ring counter can be obtained with a 74x149 and a NOR gate.
- The NOR gate is used to feed a 1 into LIN, only when the 3 LSBs are 0.



8. Sequential Practices

- Shift Register Counters (4) -



- All abnormal states lead back into the normal cycle.
- Regardless of the reached state, state 0001 is reached within 4 clock cycles, so a RESET signal is not necessarily required.