Registers

What is a Register?

- A Register is a collection of flip-flops with some common function or characteristic
  - Control signals - common clock, clear, load, etc.
  - Function - part of multi-bit storage, counter, or shift register

- At a minimum, we must be able to:
  - Observe the stored binary value
  - Change the stored binary value

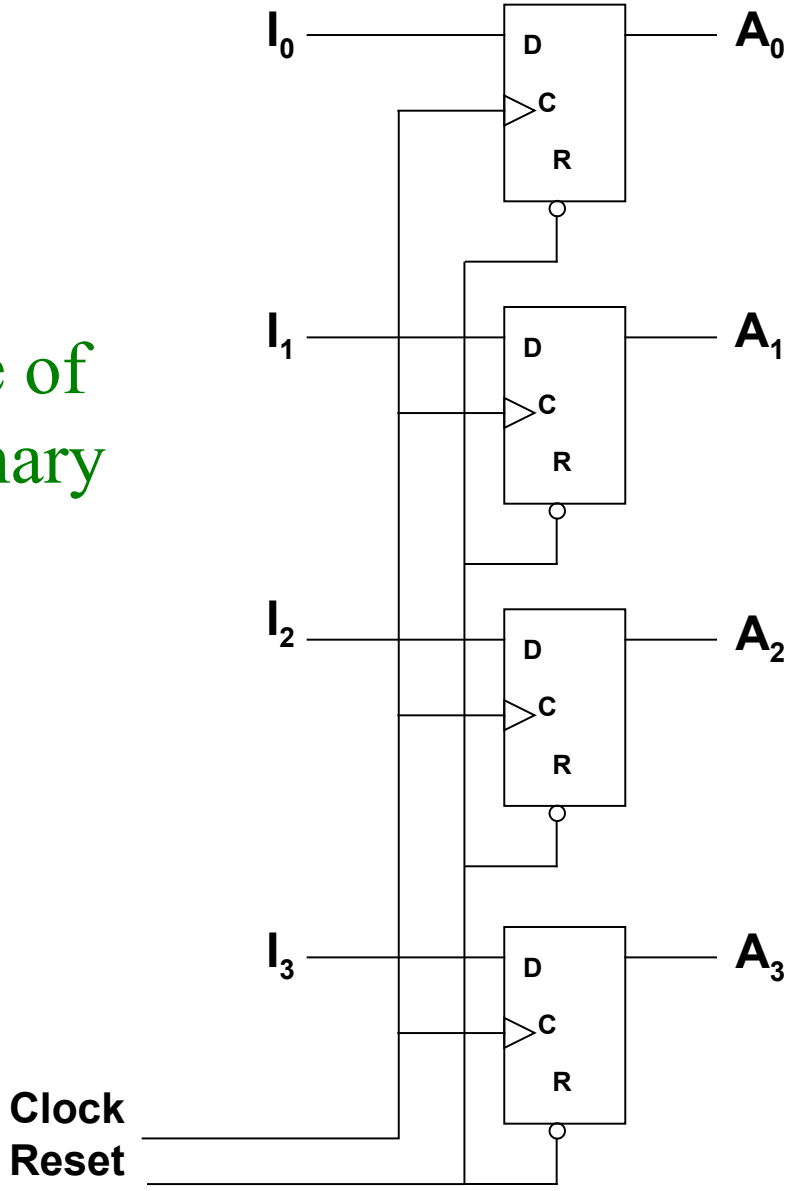
- Clocked sequential circuits
  - a group of flip-flops and combinational gates
  - connected to form a feedback path

    Flip-flops + Combinational gates

    (essential) (optional)
- Register:
  - a group of flip-flops
  - gates that determine how the information is transferred into the register
- Counter:
  - a register that goes through a predetermined sequence of states

Registers

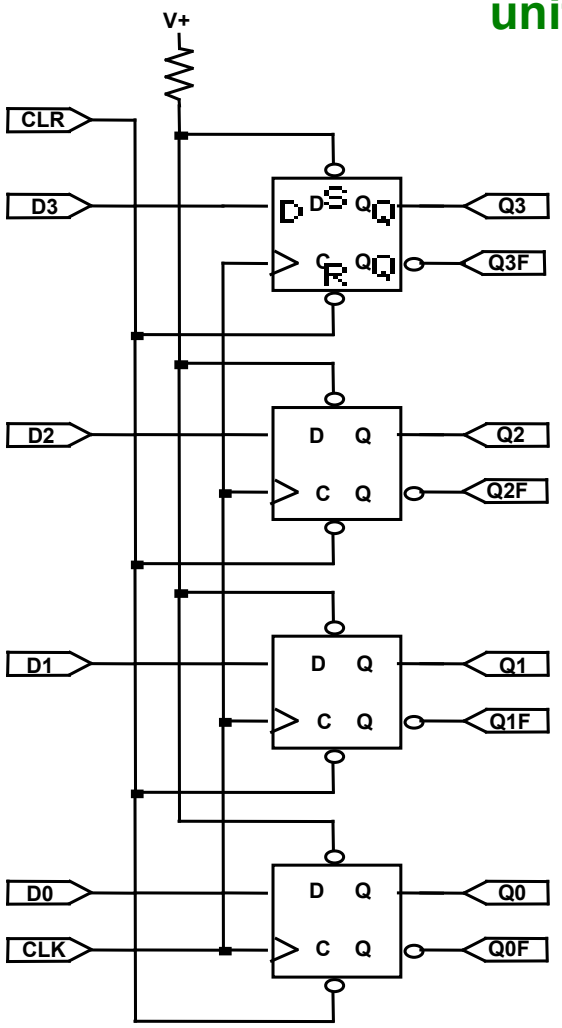
- A n-bit register
  - n flip-flops capable of storing n bits of binary information
  - 4-bit register



Kinds of Registers

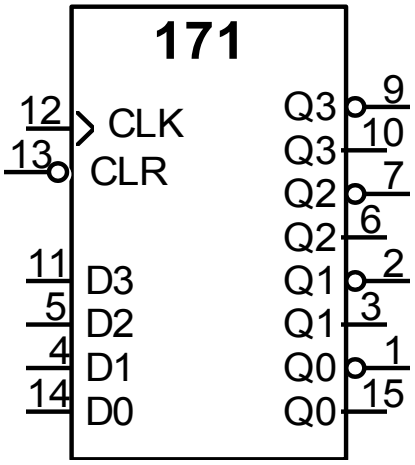
Storage Register

Group of storage elements read/written as a unit



4-bit register constructed from 4 D FFs
Shared clock and clear lines

Schematic Shape

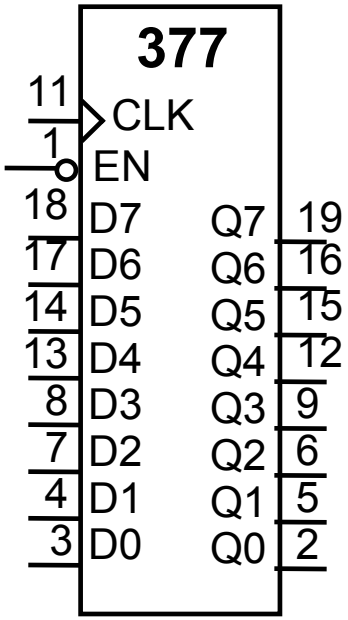


TTL 74171 Quad D-type FF with Clear
(Small numbers represent pin #s on package)

Kinds of Registers

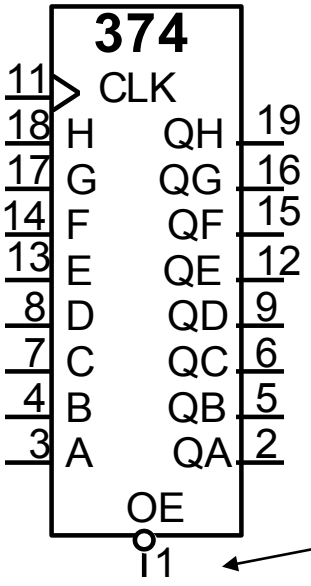
Input/Output Variations

Selective Load Capability
Tri-state or Open Collector Outputs
True and Complementary Outputs



74377 Octal D-type FFs with input enable

EN enabled low and lo-to-hi clock transition to load new data into register

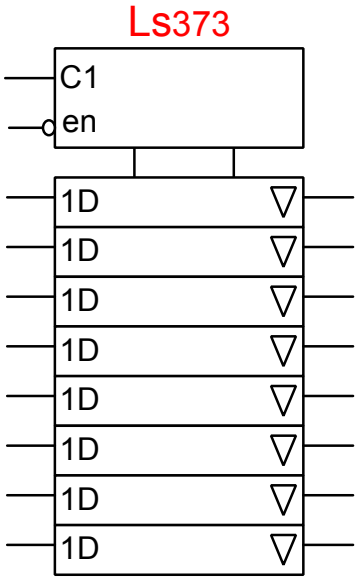


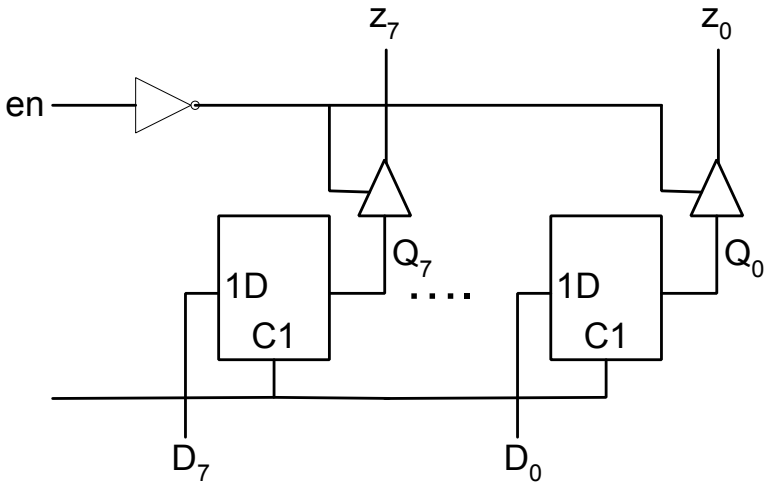
74374 Octal D-type FFs with output enable

OE asserted low presents FF state to output pins; otherwise high impedence

Kinds of Registers

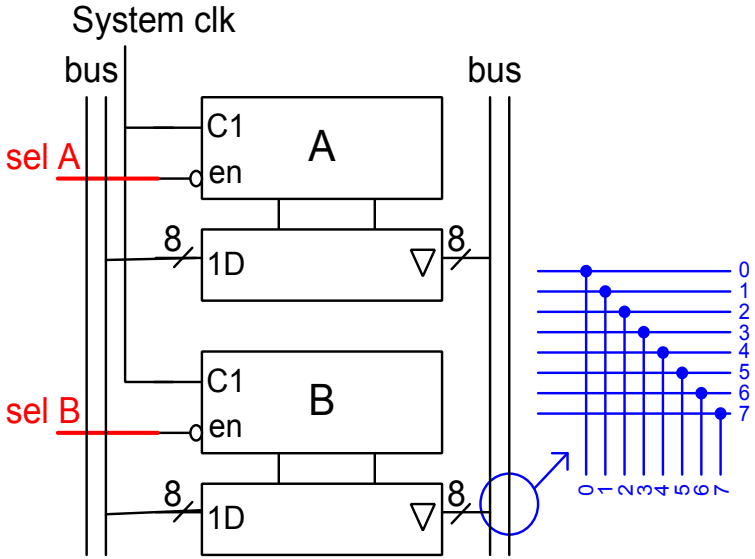
- We will be discussing the 7400 register series which is a rather popular series of registers.
- 74ls373: This register is made up of 8 latches and to have a clock enable the following structure is used:





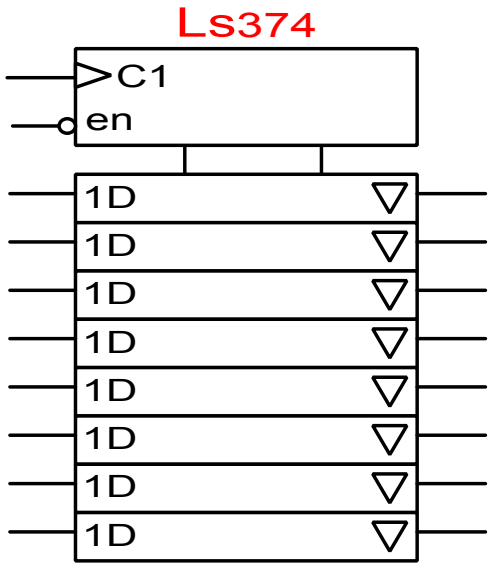
Kinds of Registers

- When the enable signal is high although clocking is done, the values of the latches won't appear on the output lines. The following figure shows us how this enable line can be used to select which one of the numerous registers' value is to be set on the output bus:



Kinds of Registers

- 74ls374: This package is very similar to that of the 74ls373. The only particular difference is that here we are allowed to feeding of outputs through combinational logic back into the register and this is because we have flip flops instead of latches in the 74ls373:



Yükleme Kontrollü Register

- Yüklemenin daha güvenli yapılabilmesi yolu

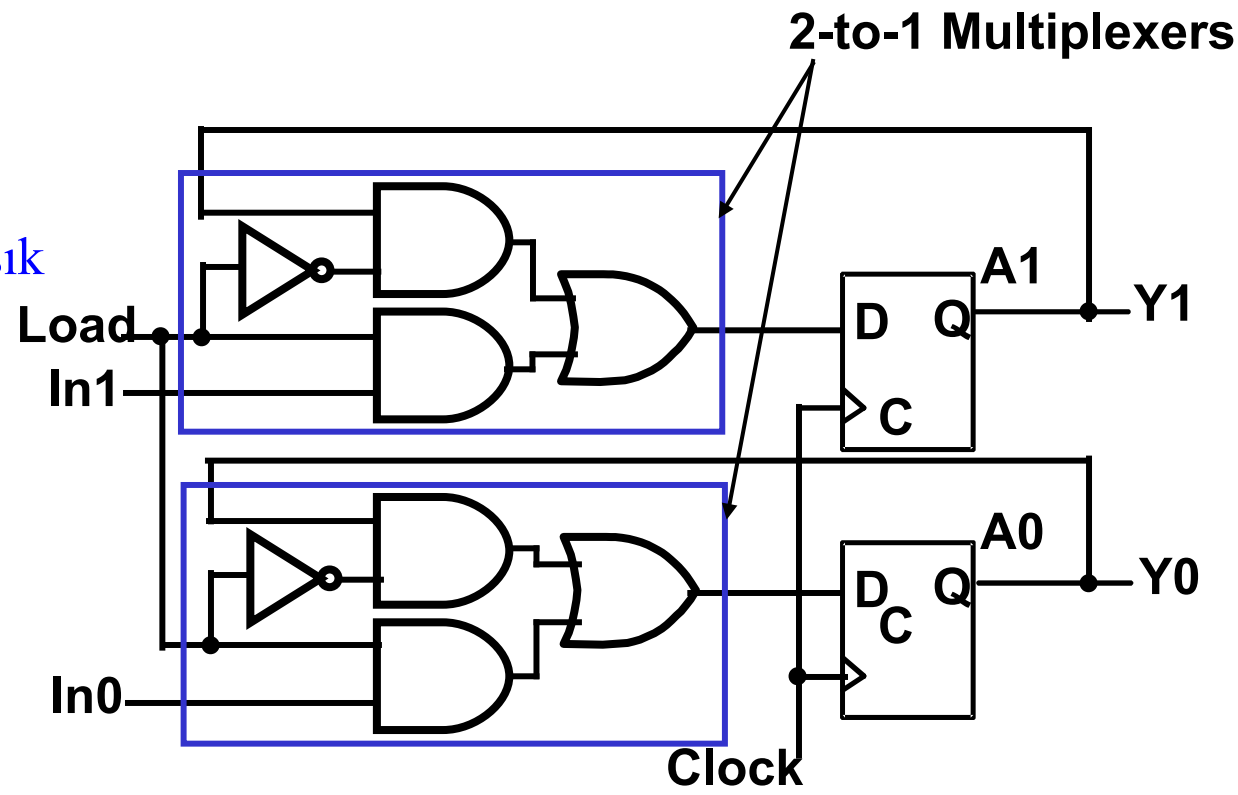
Register içeriğini değiştirmek için bir yükleme kontrol devresi kullanmaktır.

Örnek: 2-bit register
Load =0 için,
içerik korunur

- Load = 1 için,
lyeni içerik yüklenir
- Donanım daha karmaşık

ancak zamanlama

problemi yok



4-bit register with parallel load

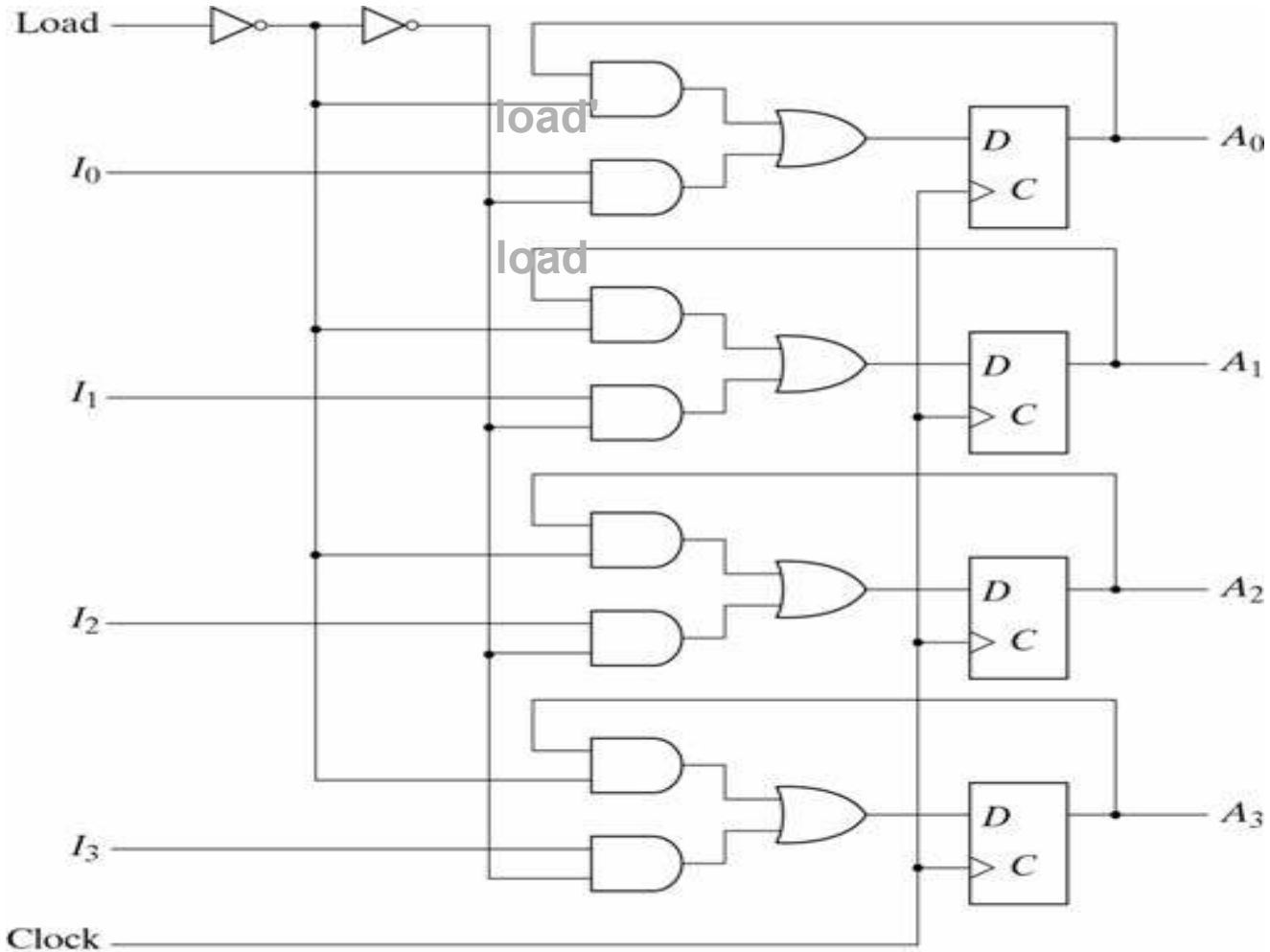


Fig. 6-2 4-Bit Register with Parallel Load

Shift Registers

- Shift register
  - a register capable of shifting its binary information in one or both directions
- Simplest shift register

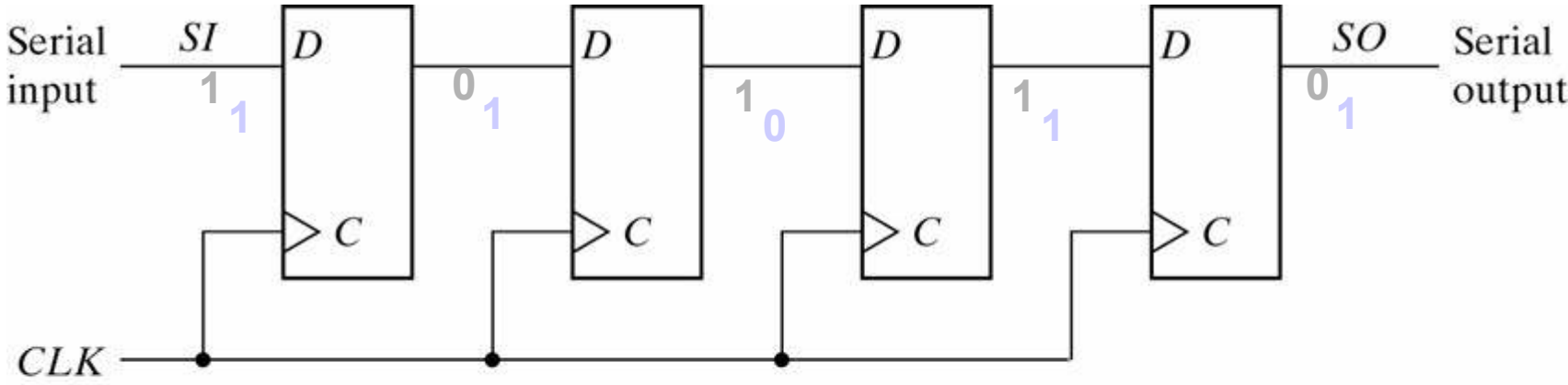
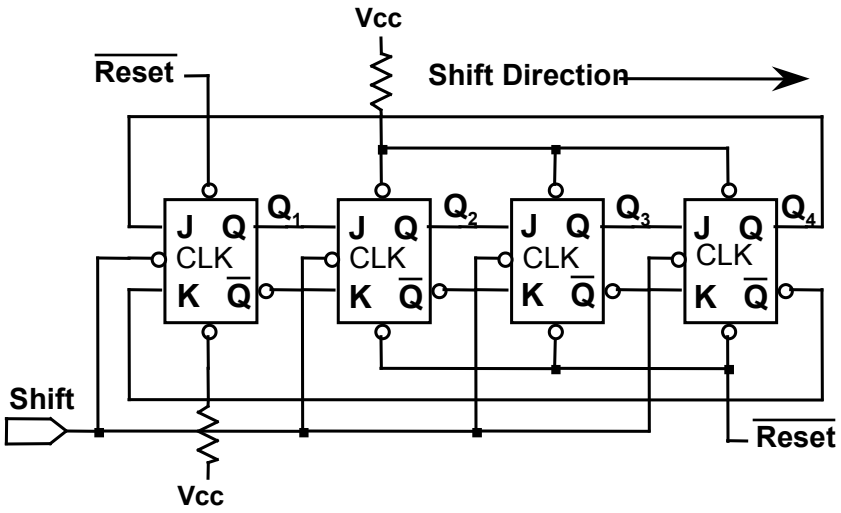


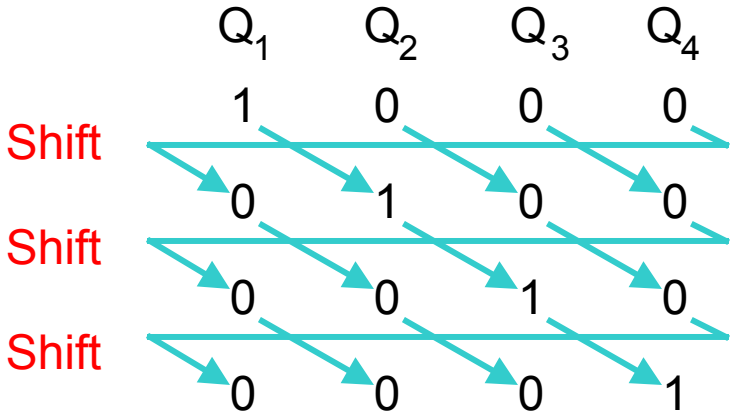
Fig. 6-3 4-Bit Shift Register

Kinds of Registers

Shift Registers

Storage + ability to circulate data among storage elements





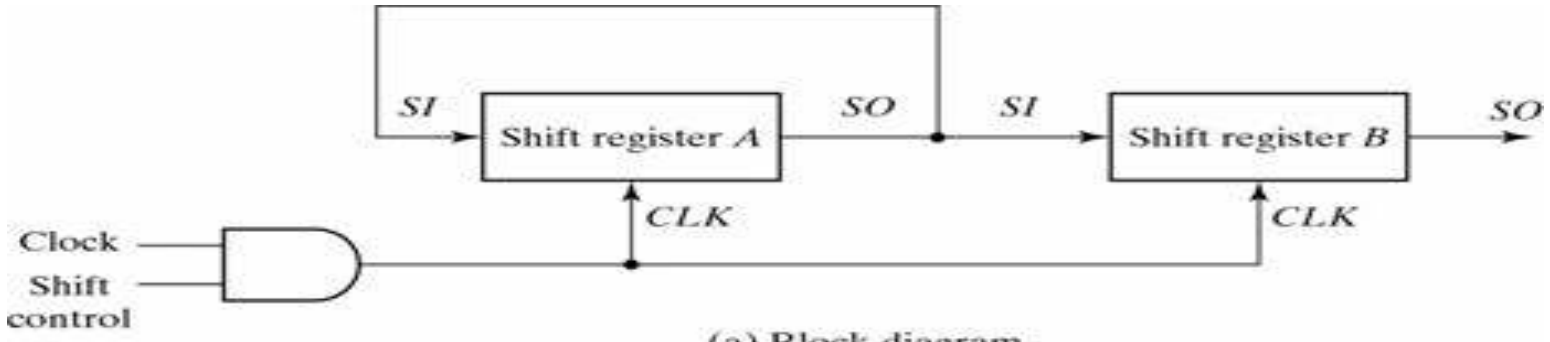
Shift from left storage element to right neighbor on every lo-to-hi transition on shift signal

Wrap around from rightmost element to leftmost element

Serial transfer vs. Parallel transfer

- Serial transfer
  - Information is transferred one bit at a time
  - shifts the bits out of the source register into the destination register
- Parallel transfer:
  - All the bits of the register are transferred at the same time

Example: Serial transfer from reg A to reg B



(a) Block diagram

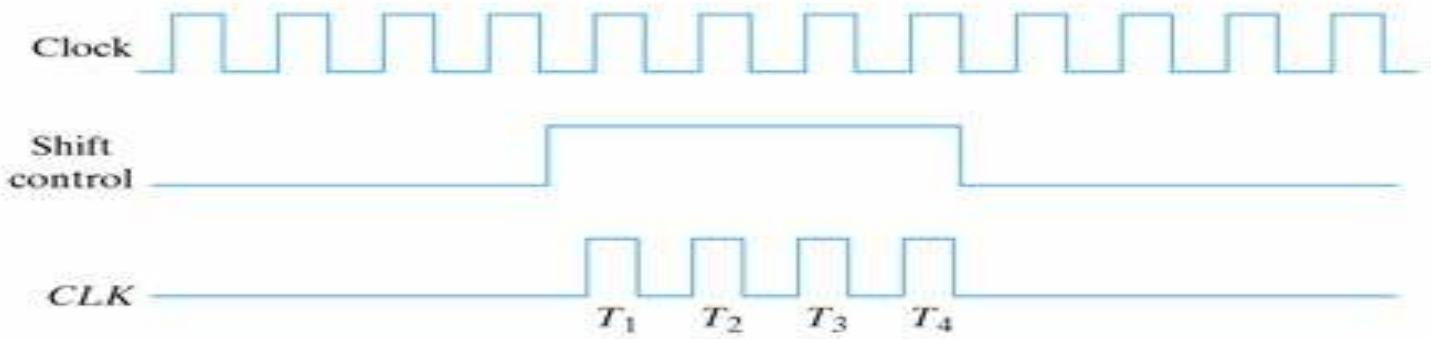


Table 6-1
Serial-Transfer Example

Timing Pulse	Shift Register A				Shift Register B			
Initial value	1	0	1	1	0	0	1	0
After T_1	1	1	0	1	1	0	0	1
After T_2	1	1	1	0	1	1	0	0
After T_3	0	1	1	1	0	1	1	0
After T_4	1	0	1	1	1	0	1	1

- Serial addition using D flip-flops

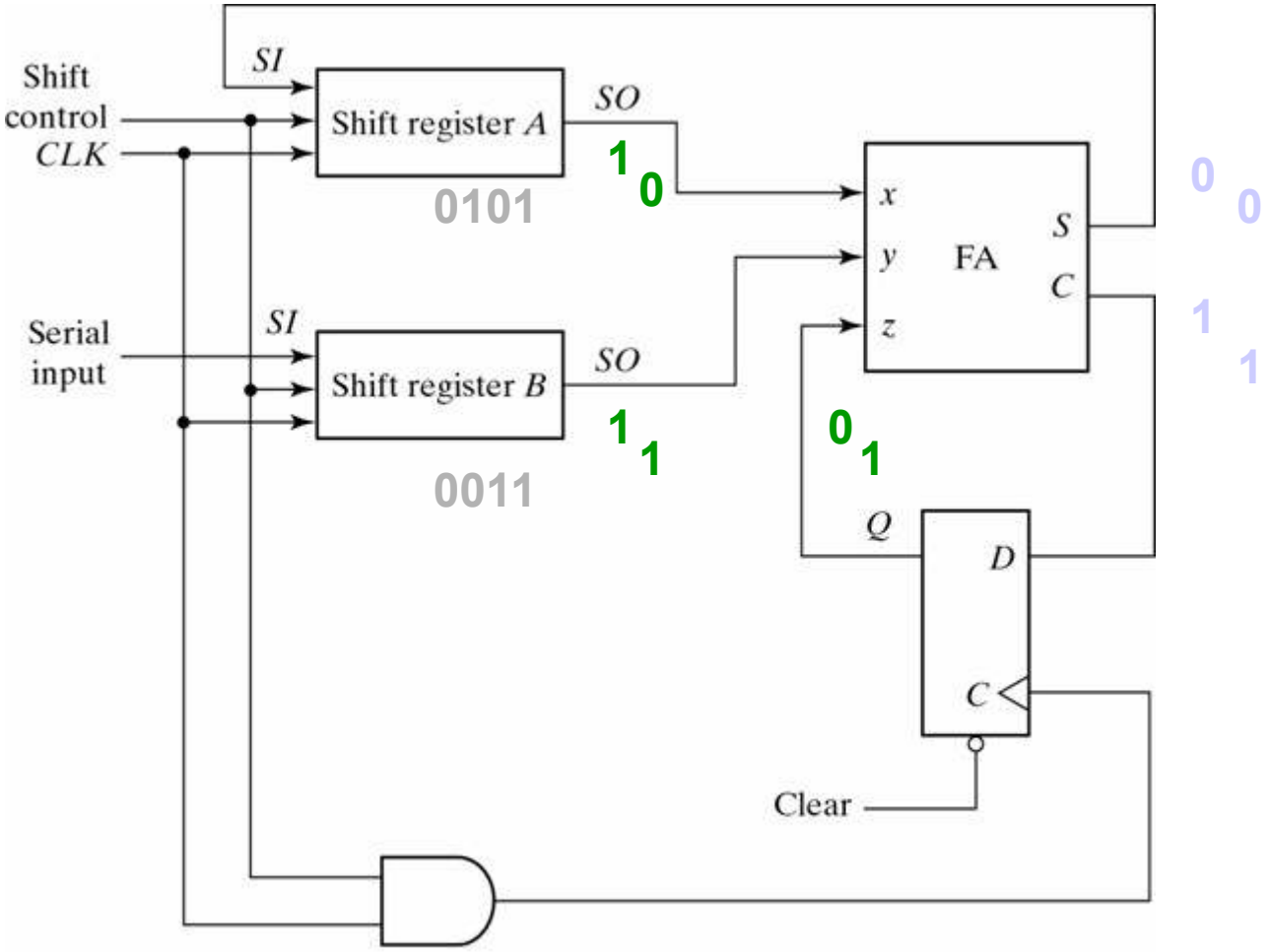


Fig. 6-5 Serial Adder

- Serial adder using JK flip-flops

Table 6-2
State Table for Serial Adder

Present State	Inputs		Next State	Output	Flip-Flop Inputs	
Q	X	y	Q	S	J_Q	K_Q
0	0	0	0	0	0	X
0	0	1	0	1	0	X
0	1	0	0	1	0	X
0	1	1	1	0	1	X
1	0	0	0	1	X	1
1	0	1	1	0	X	0
1	1	0	1	0	X	0
1	1	1	1	1	X	0

$$J_Q = x\,y$$
$$K_Q = x'\,y' = (x + y)'$$
$$S = x \oplus y \oplus Q$$

Circuit diagram

$$J_Q = x\,y$$

$$K_Q = x'\,y' = (x + y)'$$

$$S = x \oplus y \oplus Q$$

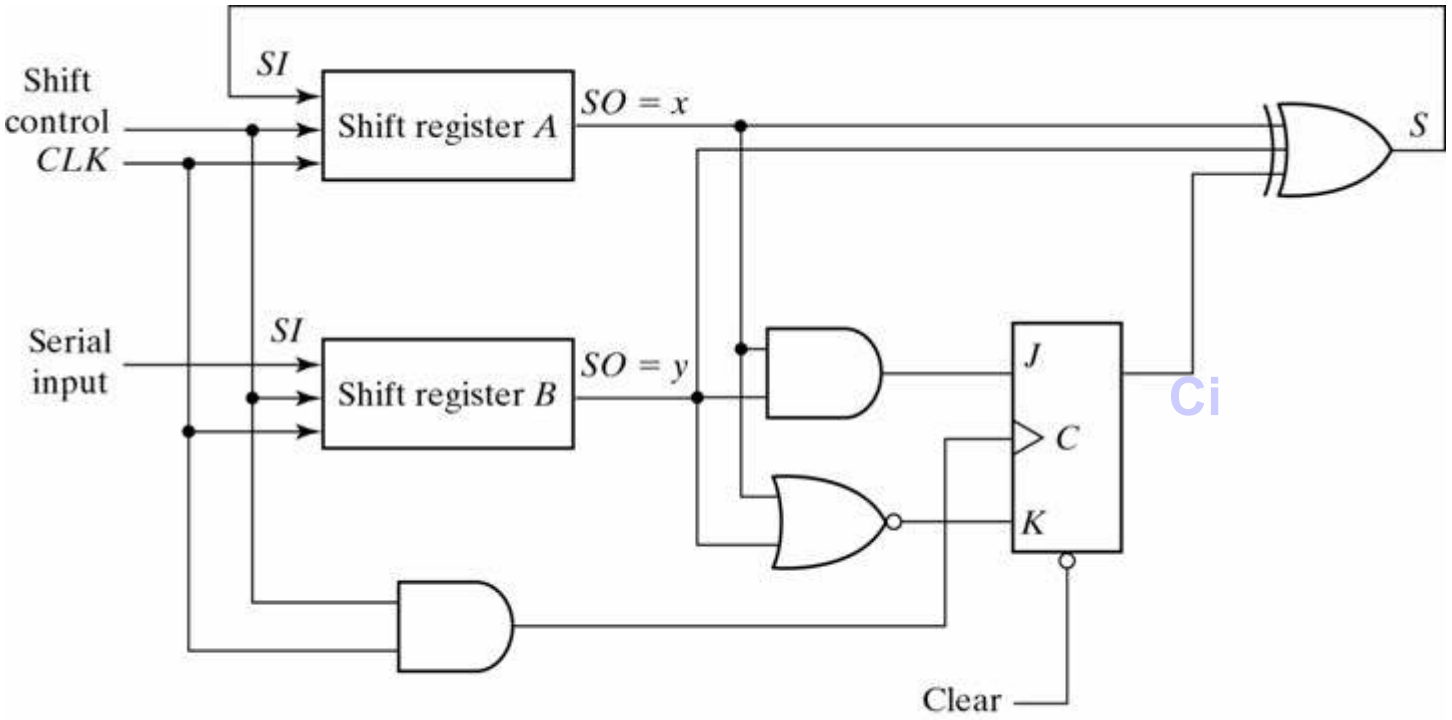
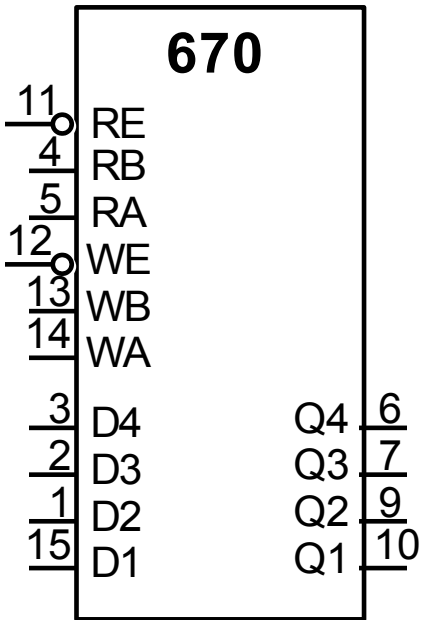


Fig. 6-6 Second form of Serial Adder

Kinds of Registers

Register Files

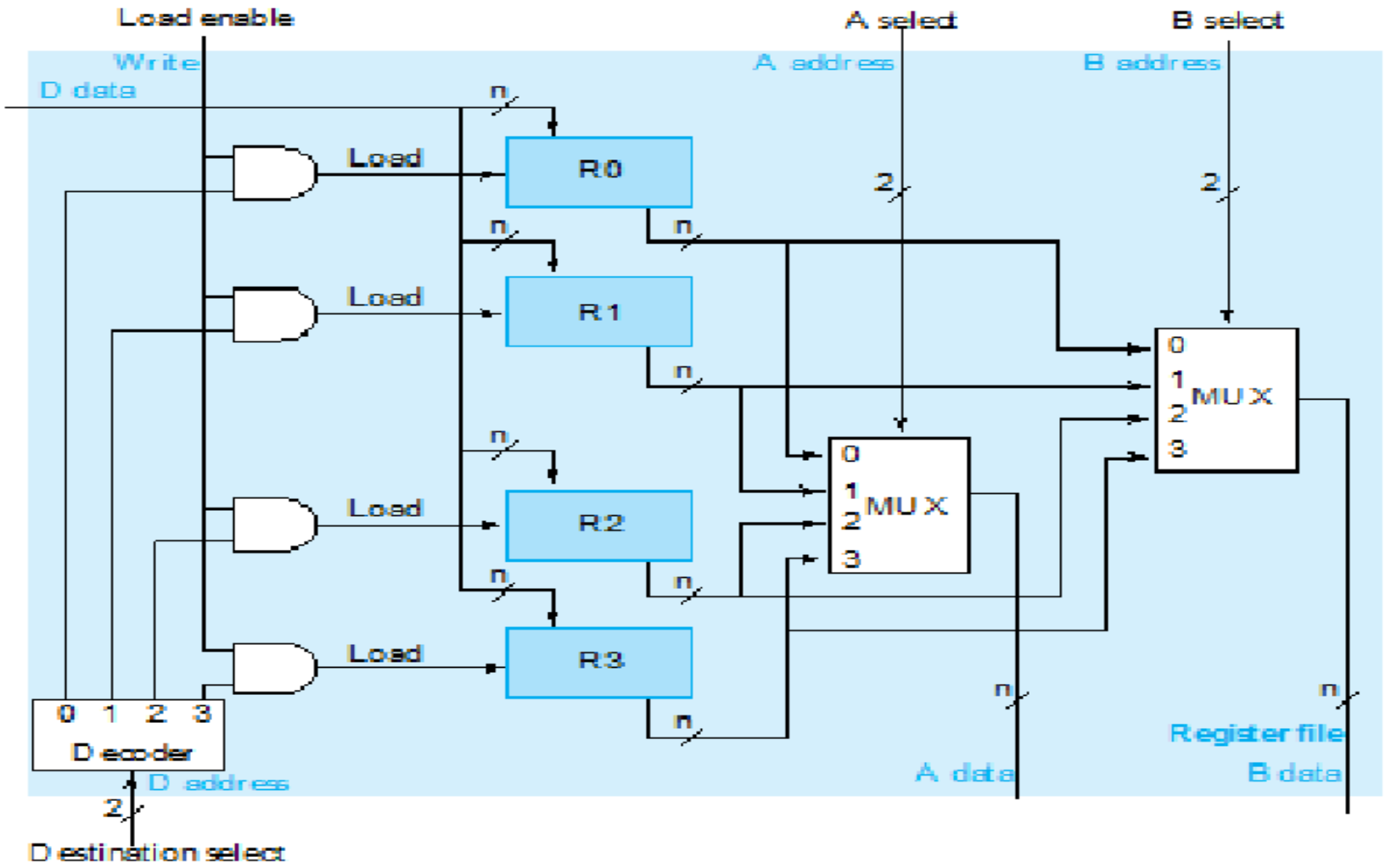
Two dimensional array of flip-flops
Address used as index to a particular word
Word contents read or written



74670 4x4 Register File with Tri-state Outputs

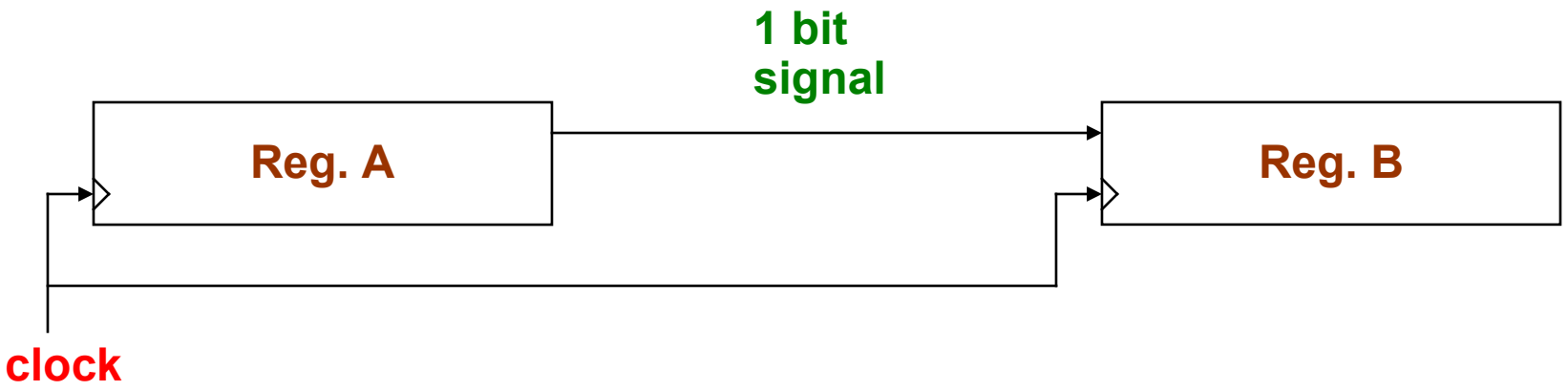
Separate Read and Write Enables
Separate Read and Write Address
Data Input, Q Outputs

Contains 16 D-ffs, organized as four rows (words) of four elements (bits)


Load enable
A select
B select
Write
D data
A address
B address
n
Load
R0
R1
R2
R3
2
0
1
2
3
MUX
0 1 2 3
Decoder
D address
A data
B data
Register file
Destination select

Serial Data Transfer

Serial transfer moves data bits from A to B one bit per clock
Rx and Tx have single wire between the two.
For ‘n’ bit registers, it takes ‘n’ clocks for data move

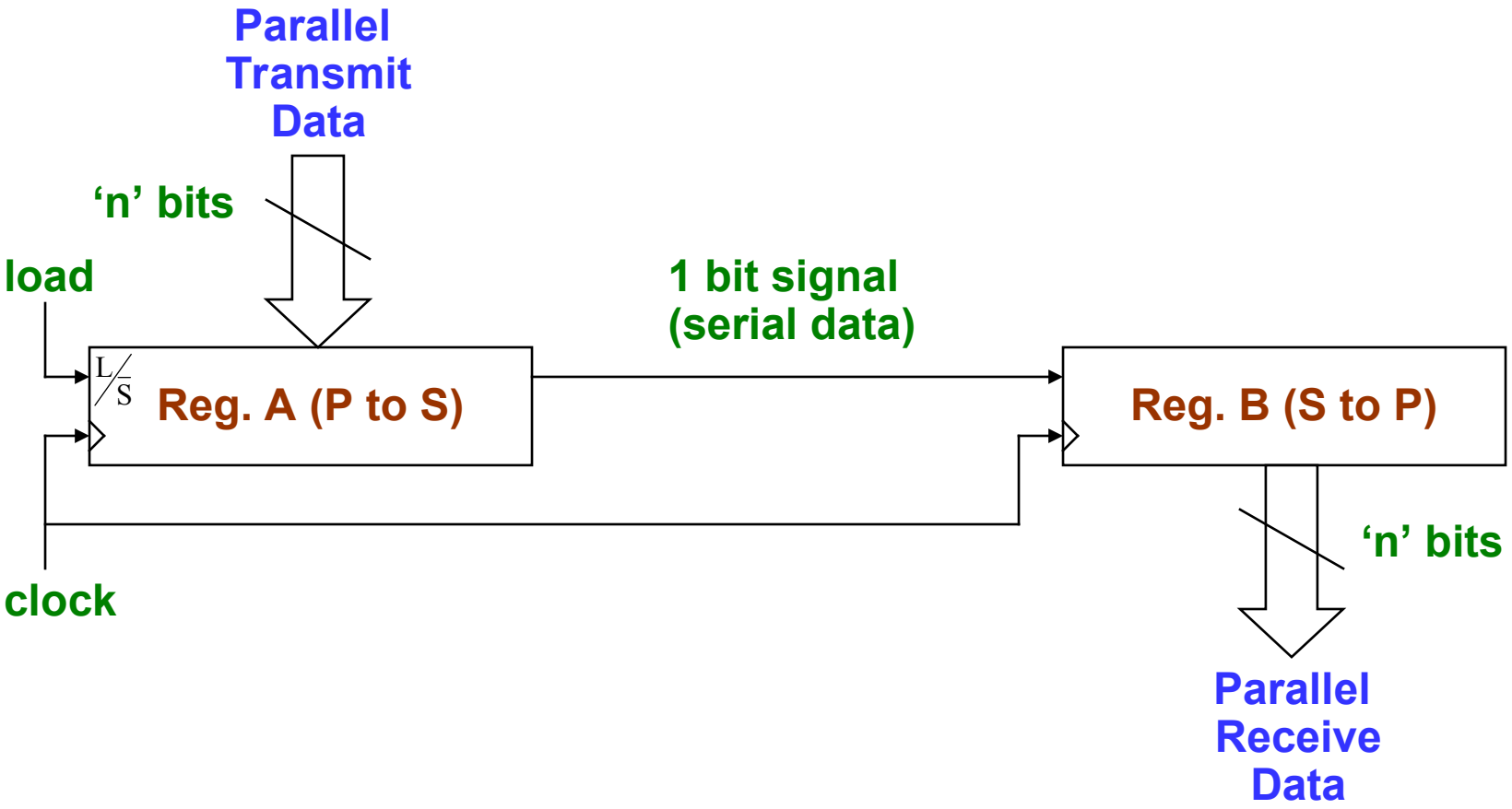


Usual implementation is with a shift register.

Serial Data Transfer

- Typical serial transfer is a multi-step process
  - Load transmit shift register with data to send
  - Shift data bit by bit from transmit to receive SR
  - Transfer received data to other registers
- The transmit SR must have parallel load
  - parallel to serial shift register
- The receive SR must have parallel outputs
  - serial to parallel shift register
- Other control/timing signals usually needed

Serial Data Transfer



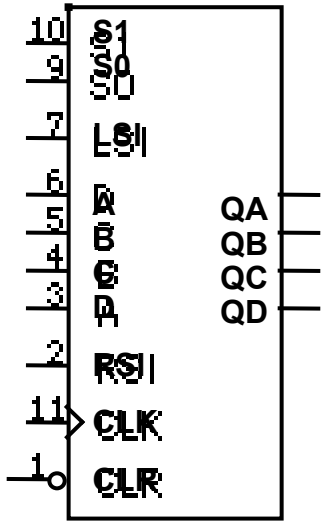
Serial Data Transfer

- Serial data transfer used where data rate is relatively slow and/or parallel bit transfer channels are expensive
  - PC serial port and USB interfaces
  - wireless/fiber optic data transmissions
    - Cell phones
    - Wireless networks
    - Satellite telephone/TV

Typical Multi-Function Shift Register

Shift Register I/O

Serial vs. Parallel Inputs
Serial vs. Parallel Outputs
Shift Direction: Left vs. Right



74194 4-bit Universal Shift Register

Serial Inputs: LSI, RSI
Parallel Inputs: D, C, B, A
Parallel Outputs: QD, QC, QB, QA
Clear Signal
Positive Edge Triggered Devices

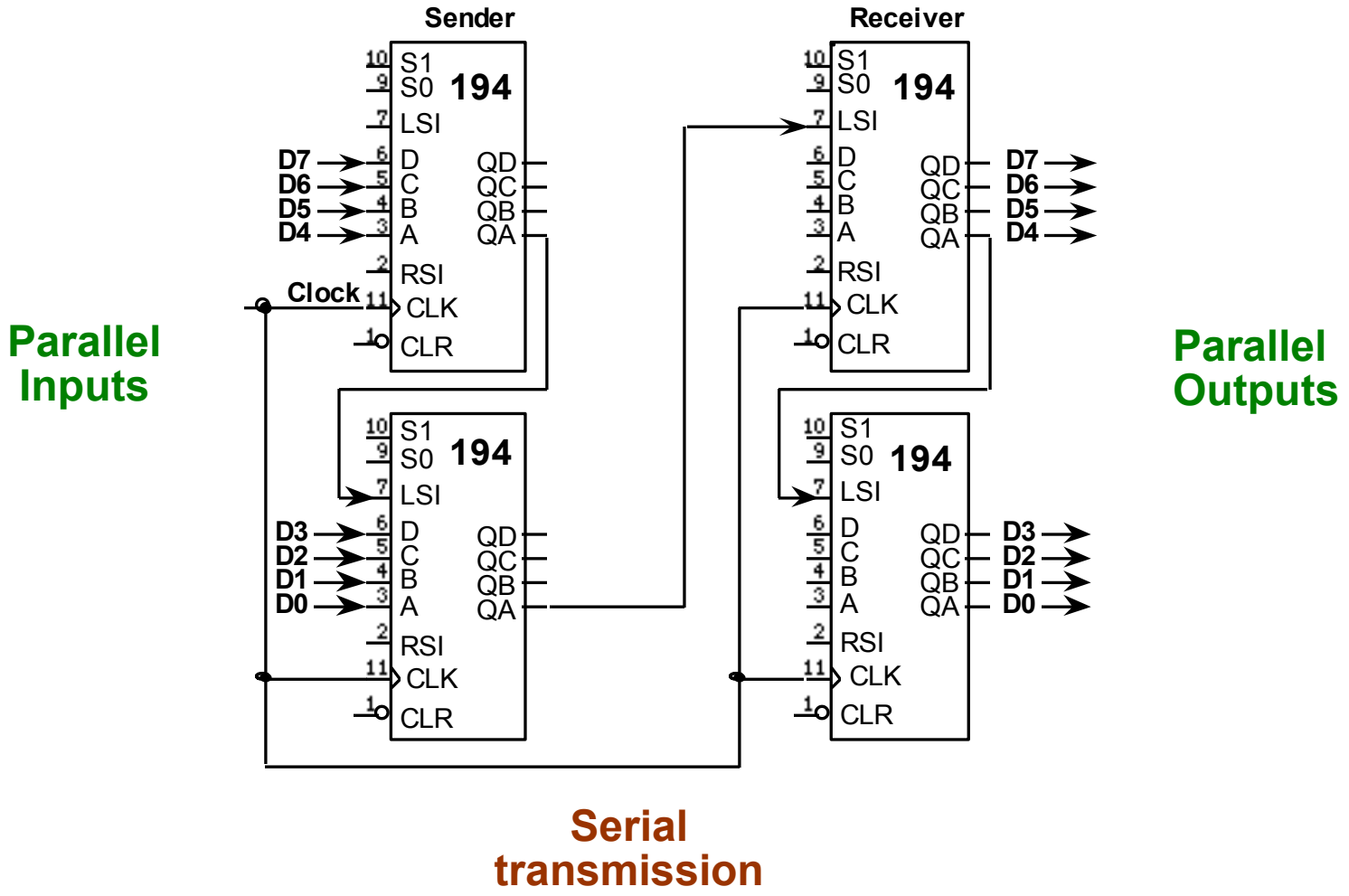
S1,S0 determine the shift function
S1 = 1, S0 = 1: Load on rising clk edge
synchronous load
S1 = 1, S0 = 0: shift left on rising clk edge
LSI replaces element D
S1 = 0, S0 = 1: shift right on rising clk edge
RSI replaces element A
S1 = 0, S0 = 0: hold state

Multiplexing logic on input to each FF!

Shifters well suited for serial-to-parallel conversions, such as terminal to computer communications

Serial Transfer with Shift Registers

Shift Register Application: Parallel to Serial Conversion



Universal Shift Register

- Unidirectional shift register
- Bidirectional shift register
- Universal shift register:
  - has both direction shifts & parallel load/out capabilities

Capability of a universal shift register:

1. A *clear* control to clear the register to 0.

2. A *clock* input to synchronize the operations.

3. A *shift-right* control to enable the shift right operation

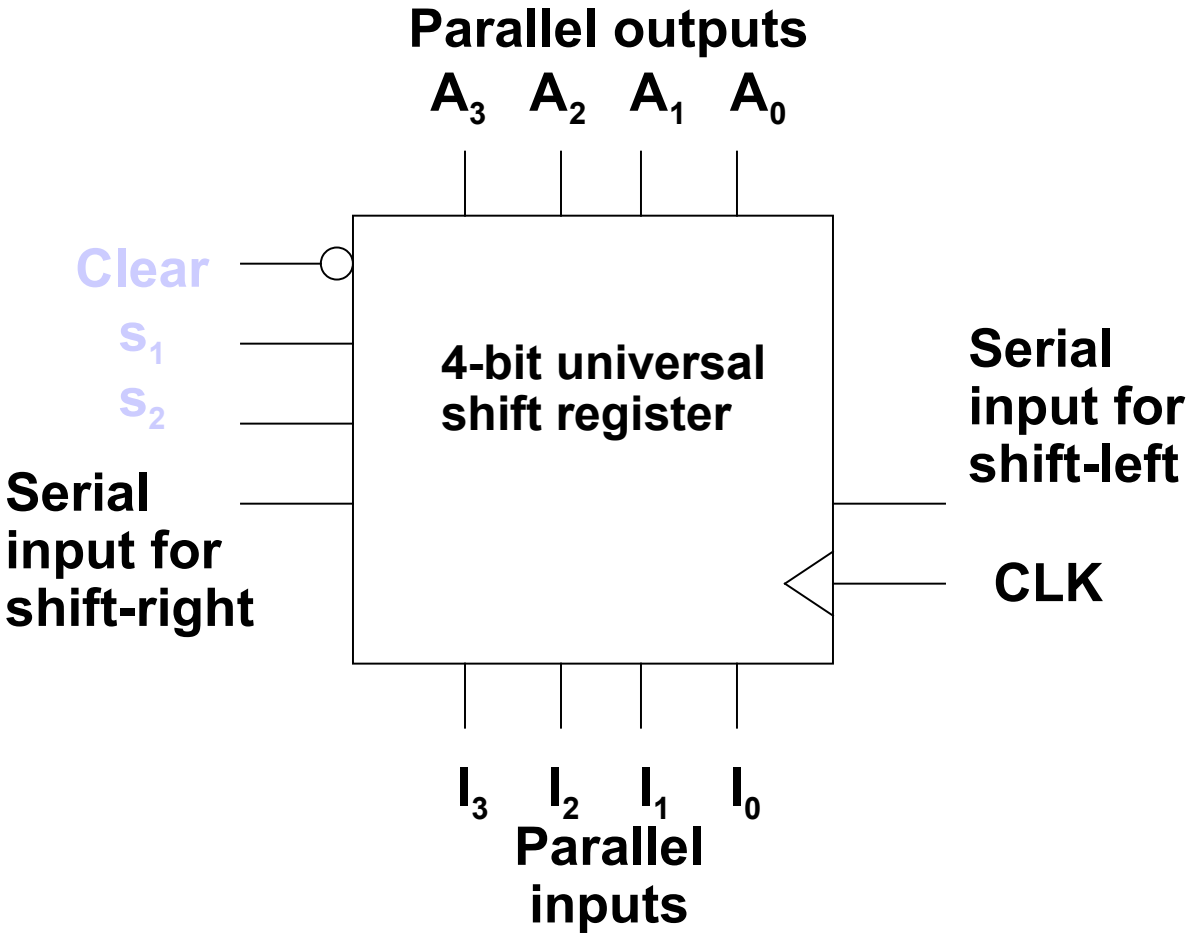
4. A *shift-left* control to enable the shift left operation

5. A *parallel-load* control to enable a parallel transfer and the *n parallel input* lines associated with the parallel transfer.

6. *n parallel output* lines.

7. A control state that leaves the information in the register unchanged in the presence of the clock.

- Example: 4-bit universal shift register



- Function table

Clear	s_1	s_0	A_3^+	A_2^+	A_1^+	A_0^+	(operation)
0	×	×	0	0	0	0	Clear
1	0	0	A_3	A_2	A_1	A_0	No change
1	0	1	sri	A_3	A_2	A_1	Shift right
1	1	0	A_2	A_1	A_0	sli	Shift left
1	1	1	I_3	I_2	I_1	I_0	Parallel load

4-bit universal shift register

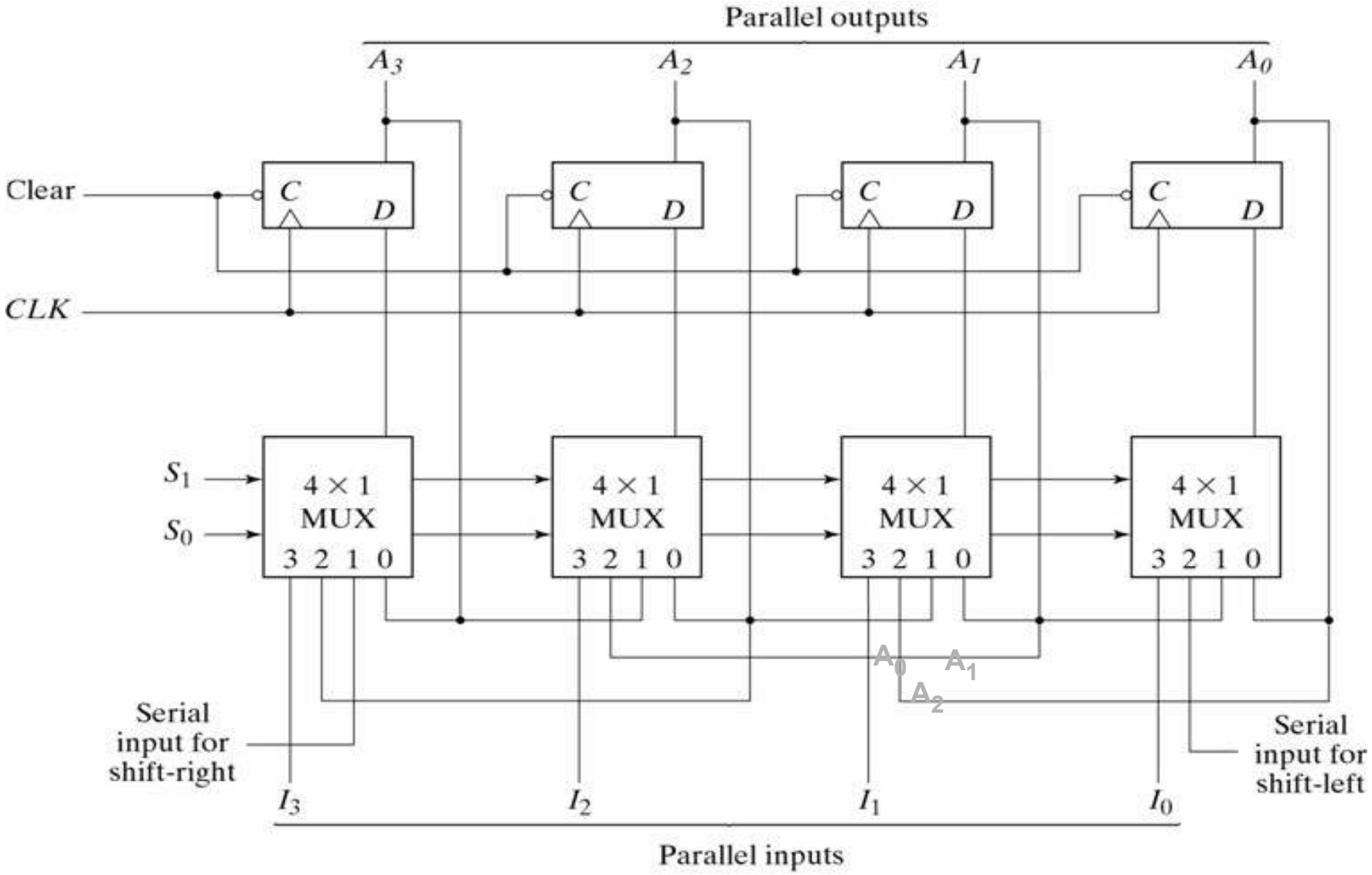


Fig. 6-7 4-Bit Universal Shift Register