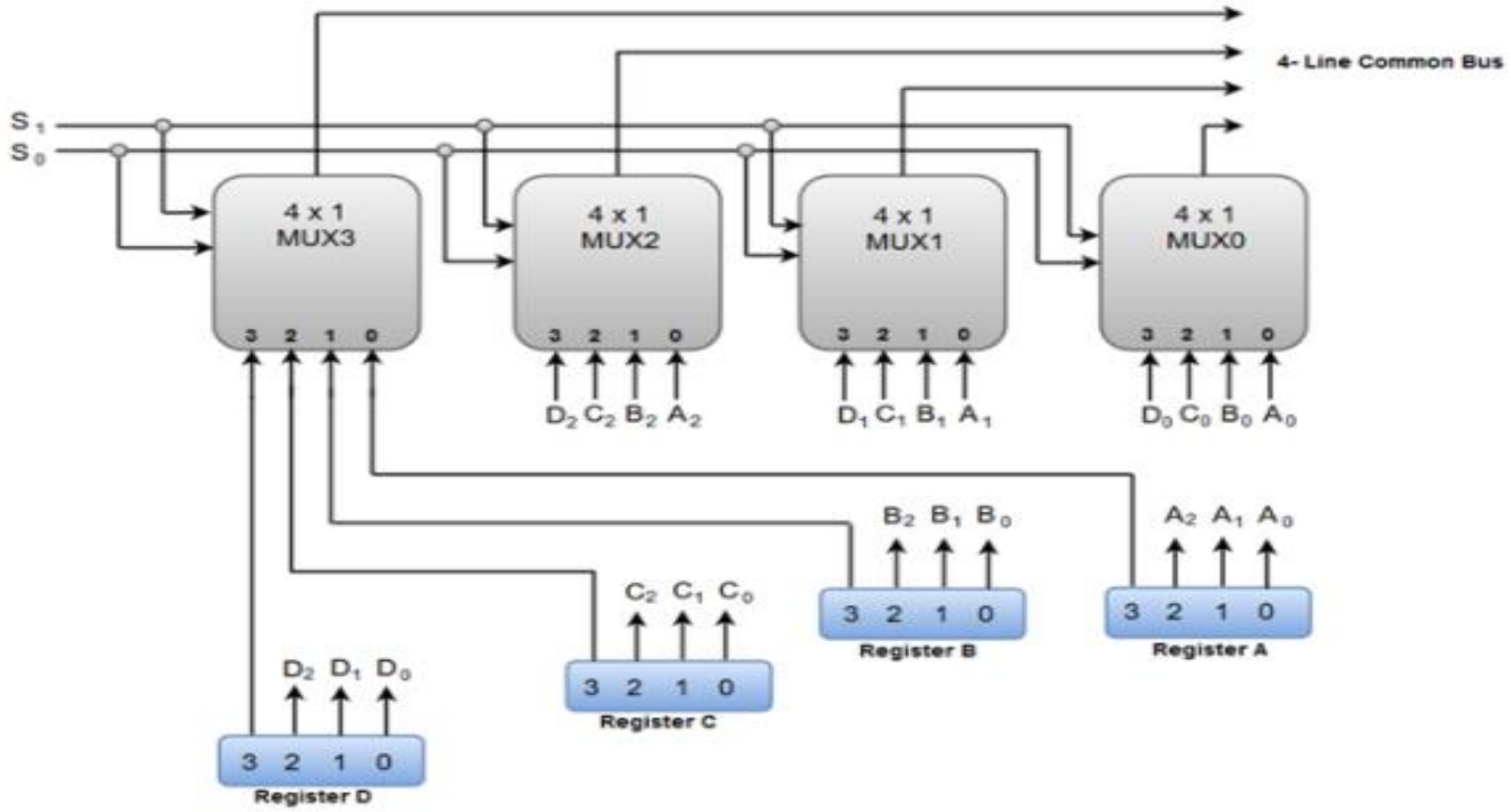
# Bus and Memory Transfers

- A shared communication path consisting of one or more connection lines is known as bus.
- Bus transfer : The transfer of data through bus is known as bus transfer.
- Memory Transfer : When a data is read from memory or is stored in memory is referred to as memory transfer.
- A more efficient scheme for transferring information between registers in a multiple-register configuration is a Common Bus System.
- A common bus consists of a set of common lines, one for each bit of a register, through witch binary information is transferred one at a time.
- Control signals determine which register is selected by the bus during each particular register transfer.
- Different ways of constructing a Common Bus System
  - Using Multiplexers
  - Using Tri-state Buffers

# Common bus system with multiplexers

- The multiplexers select the source register whose binary information is then placed on the bus.
- The construction of a bus system for four registers is shown in below Figure



# Common bus system is with multiplexers cont.

- The bus consists of four 4 x 1 multiplexers each having four data inputs, 0 to 3, and two selection inputs, S1 and S0.
- It also consists four 4 bit register each having four outputs, 0 to 3, that are connected to the inputs of MUX.
- For example, output 1 of register A is connected to input 0 of MUX 1 because this input is labelled A1.
- The diagram shows that the bits in the same significant position in each register are connected to the data inputs of one multiplexer to form one line of the bus.
- Thus MUX 0 multiplexes the four 0 bits of the registers, MUX 1 multiplexes the four 1 bits of the registers, and similarly for the other two bits.
- The two selection lines S1 and S0 are connected to the selection inputs of all four multiplexers.
- The selection lines choose the four bits of one register and transfer them into the four-line common bus.

# Common bus system is with multiplexers cont.

- When S1S0 = 00, the 0 data inputs of all four multiplexers are selected and applied to the outputs that form the bus.
- This causes the bus lines to receive the content of register A since the outputs of this register are connected to the 0 data inputs of the multiplexers.
- Similarly, register B is selected if S1S0 = 01, and so on.
- Below table shows the register that is selected by the bus for each of the four possible binary value of the selection lines.

| $S_1$ | $S_0$ | Register selected |
|---|---|---|
| 0 | 0 | *A* |
| 0 | 1 | *B* |
| 1 | 0 | *C* |
| 1 | 1 | *D* |

# Common bus system is with multiplexers cont.

- In general a bus system has
  - Multiplex “k” Registers
  - each register of “n” bits

  to produce “n-line bus” require
  - no. of multiplexers required = n
  - size of each multiplexer = k x 1
- When the bus is includes in the statement, the register transfer is symbolized as follows:
  - P: BUS← C, R1← BUS  (if P=1)
- The content of register C is placed on the bus, and the content of the bus is loaded into register R1 by activating its load control input. If the bus is known to exist in the system, it may be convenient just to show the direct transfer.

**A digital computer has a common bus system for 16 registers of 32 bits each. (i) How many selection input are there in each multiplexer? (ii) What size of multiplexers is needed? (iii) How many multiplexers are there in a bus?**

(i) How many selection input are there in each multiplexer?

$2^n$=No. of Registers; n=selection input of multiplexer

$2^n$=16; here n=4

Therefore 4 selection input lines should be there in each multiplexer.

(ii) What size of multiplexers is needed?

size of multiplexers= Total number of register X 1

= 16 X 1

Multiplexer of 16 x 1 size is needed to design the above defined common bus.

(iii) How many multiplexers are there in a bus?

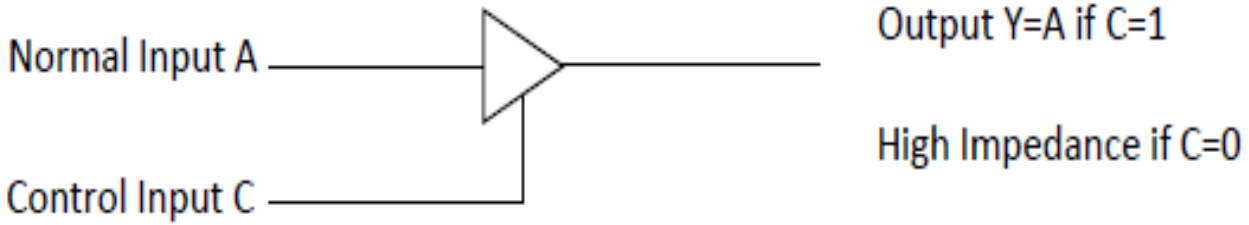
No. of multiplexers = bits of register

= 32

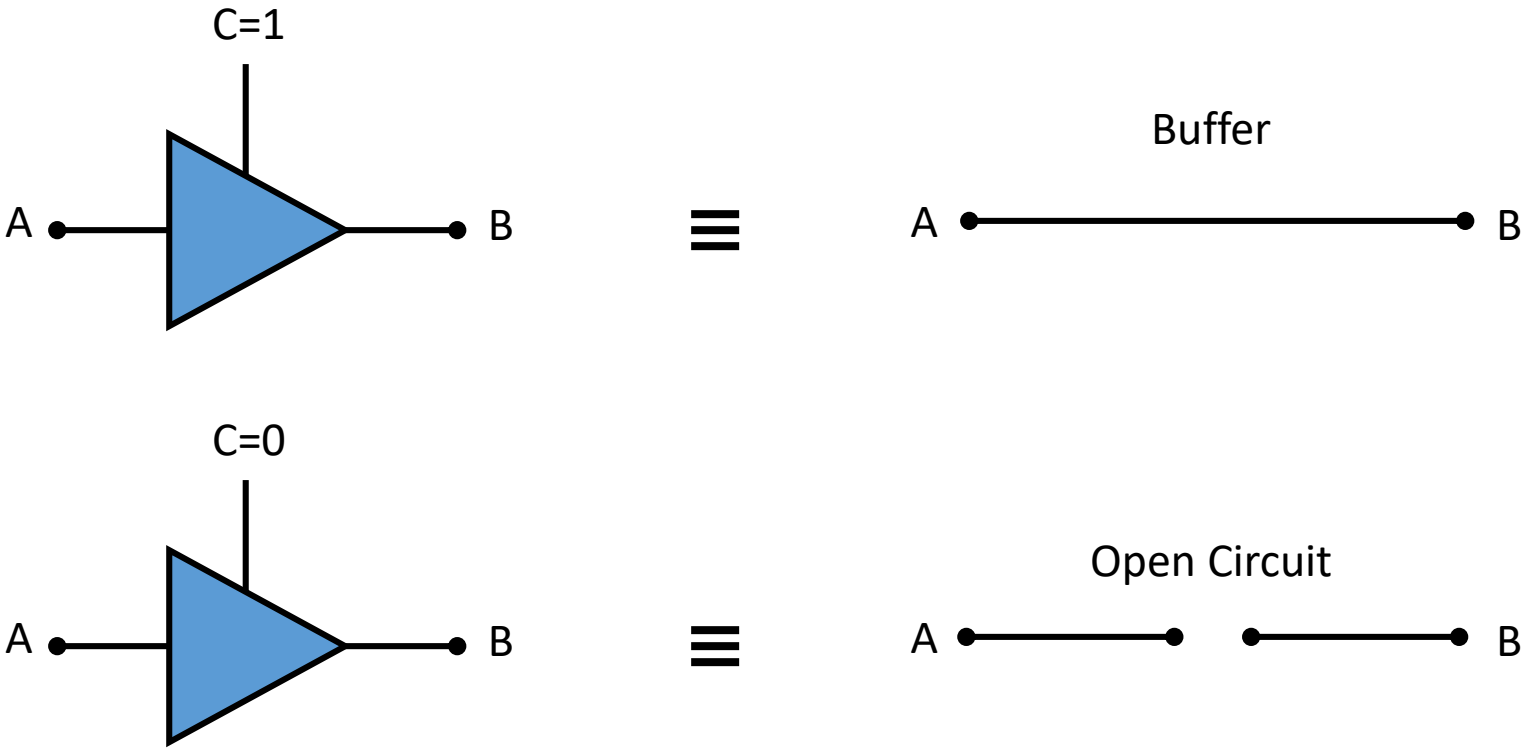
32 multiplexers are needed in a bus.

# Bus and Memory Transfers: Three-State Bus Buffers

- A bus system can be constructed with three-state gates instead of multiplexers.
- A three-state gate is a digital circuit that exhibits three states.
  - State 1:-Control Signal equivalent to logic 1.
  - State 2:-Control Signal equivalent to logic 0.
  - State 3:- High-impedance state.
- The high-impedance state behaves like an open circuit, which means that the output is disconnected and does not have logic significance.
- The most commonly used design of a bus system is the buffer gate.
- The graphic symbol of a three-state buffer gate is shown in Fig.



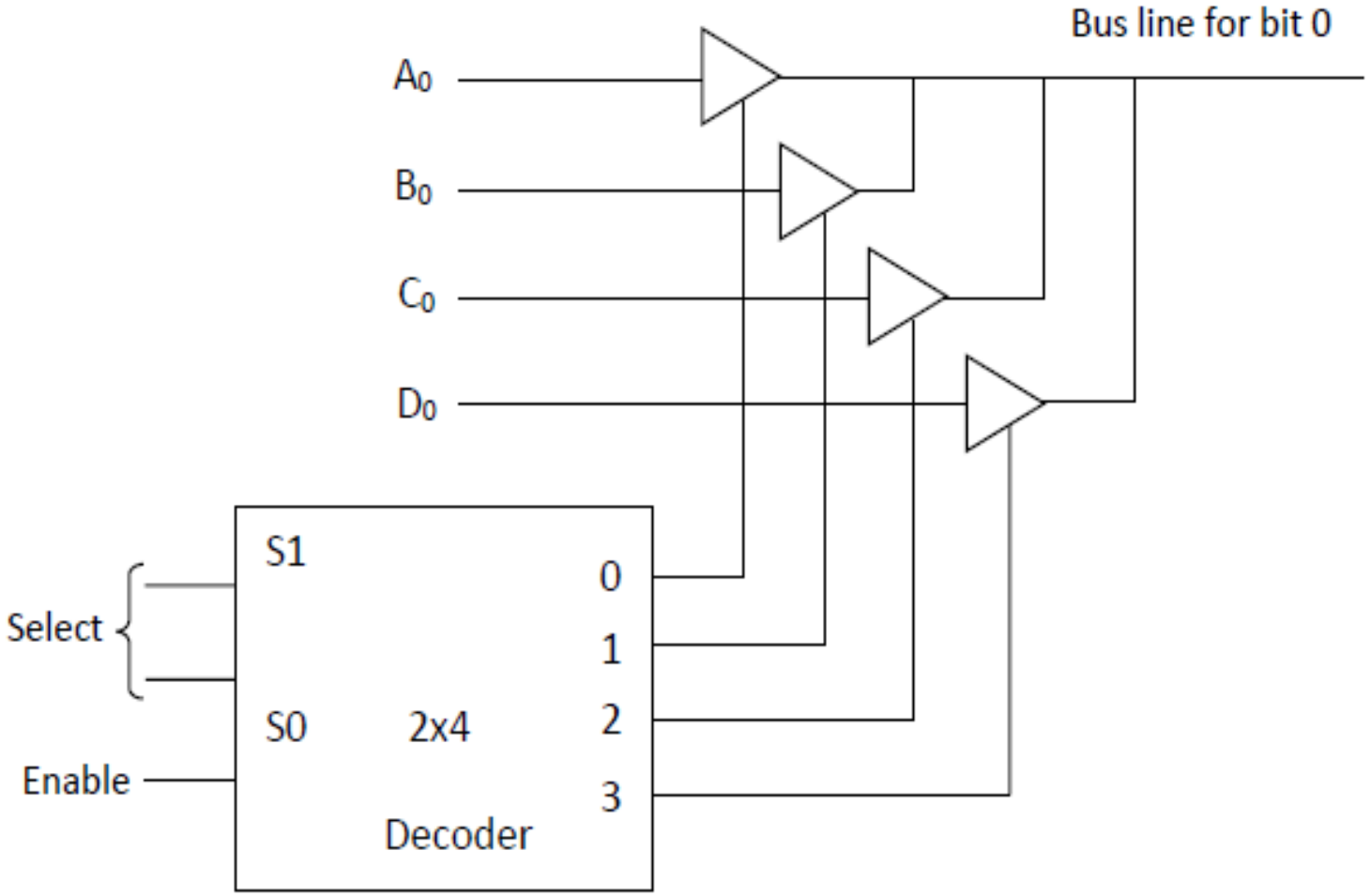
# Bus and Memory Transfers: Three-State Bus Buffers cont.



# Bus and Memory Transfers: Three-State Bus Buffers
cont.

- It is distinguished from a normal buffer by having both a normal input and a control input.
- The control input determines the output state. When the control input is equal to 1, the output is enabled and the gate behaves like any conventional buffer, with the output equal to the normal input.
- When the control input is 0, the output is disabled and the gate goes to a high-impedance state, regardless of the value in the normal input.
- The construction of a bus system with three-state buffers is shown in Fig.

# Bus Line with Three-State Bus Buffers



# Bus and Memory Transfers: Three-State Bus Buffers cont.

- The outputs of four buffers are connected together to form a single bus line.
- The control inputs to the buffers determine which of the four normal inputs will communicate with the bus line.
- No more than one buffer may be in the active state at any given time.
- The connected buffers must be controlled so that only one three-state buffer has access to the bus line while all other buffers are maintained in a high impedance state.
- One way to ensure that no more than one control input is active at any given time is to use a decoder, as shown in the diagram.
- When the enable input of the decoder is 0, all of its four outputs are 0, and the bus line is in a high-impedance state because all four buffers are disabled.
- When the enable input is active, one of the three-state buffers will be active, depending on the binary value in the select inputs of the decoder.

# Bus and Memory Transfers: Three-State Bus Buffers cont.

- To construct a common bus for four registers of n bits each using three-state buffers, we need n circuits with four buffers in each as shown in figure :Bus line with three state buffers
- Each group of four buffers one significant bit from the four registers.
- Each common output produces one of the line for the common bus for a total of n lines.
- Only one decoder is necessary to select between the four registers.

# Memory Transfers

- Read Operation: The transfer of information from a memory word to the outside environment is called a *read* operation.
- Write Operation: The transfer of new information to be stored into the memory is called a *write* operation.
- A memory word will be symbolized by the letter M.
- The particular memory word among the many available is selected by the memory address during the transfer.
- It is necessary to specify the address of M when writing memory transfer operations.
- This will be done by enclosing the address in square brackets following the letter M.
- Consider a memory unit that receives the address from a register, called the address register, symbolized by AR.
- The data are transferred to another register, called the data register, symbolized by DR.
- The read operation can be stated as follows:

**Read: DR<- M [AR]**

# Memory Transfers

- This causes a transfer of information into DR from the memory word M selected by the address in AR.
- The write operation transfers the content of a data register to a memory word M selected by the address. Assume that the input data are in register R1 and the address is in AR.
- The write operation can be stated as follows:

**Write: M [AR] <- R1**

- This causes a transfer of information from R1 into memory word M Selected by address AR