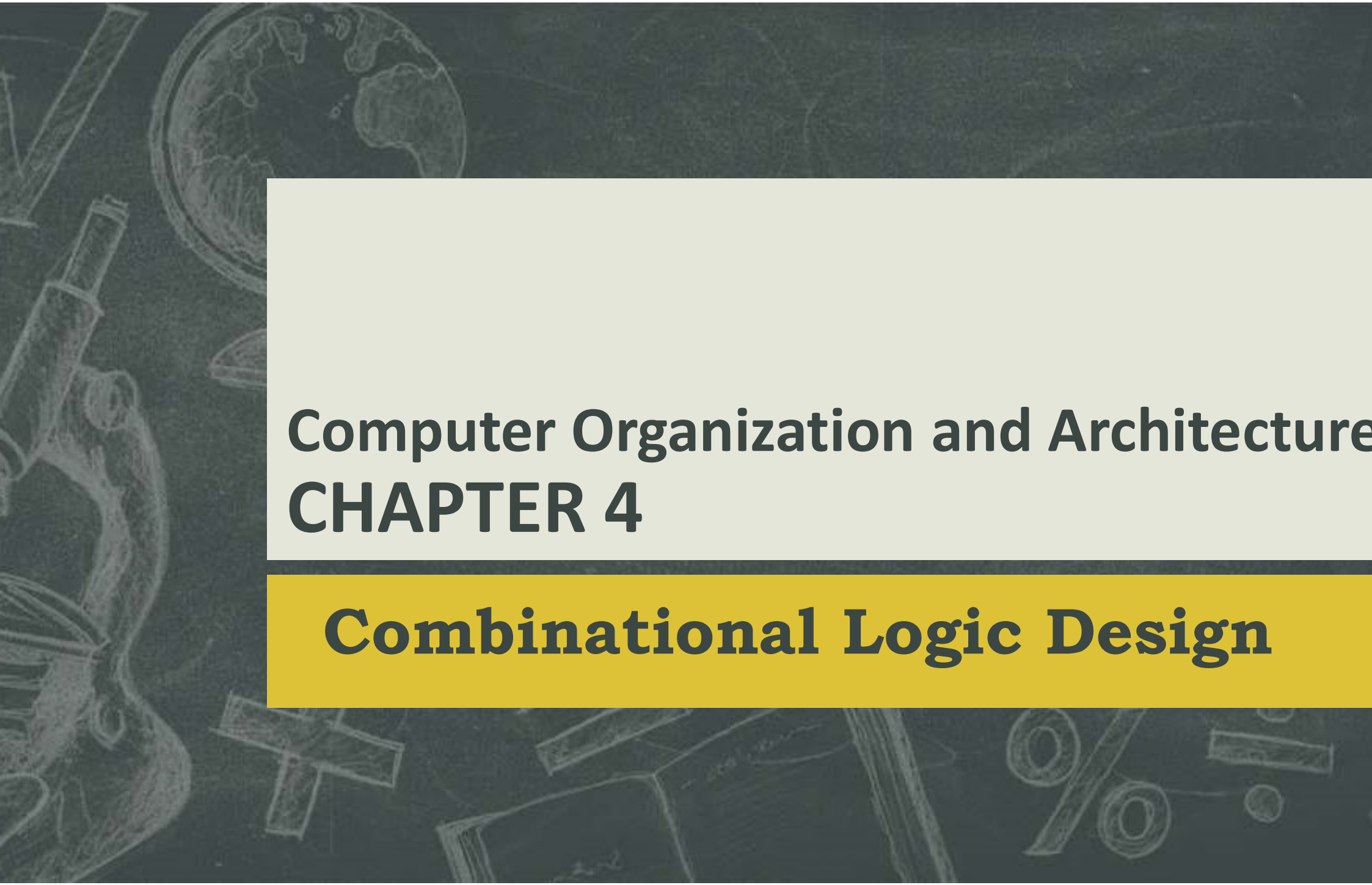


Computer Organization and Architecture

CHAPTER 4

Combinational Logic Design

Agenda

- Encoder
- Decoder
- Multiplexer
- Demultiplexer

Introduction

In a digital system, discrete quantities of information are represented wi
binary codes. Binary code of N digits can be used to store 2^N distin
elements of coded information.

This is what encoders and decoders are used for. **Encoders** convert 2^N lin
of input into a code of N bits and **Decoders** decode the N bits into 2^N lines.

A binary code of n bits can represent up to 2^n distinct elements of the code
information.

Encoder and Decoder

Encoder

An Encoder is a **combinational circuit** that performs the reverse operation of Decoder.

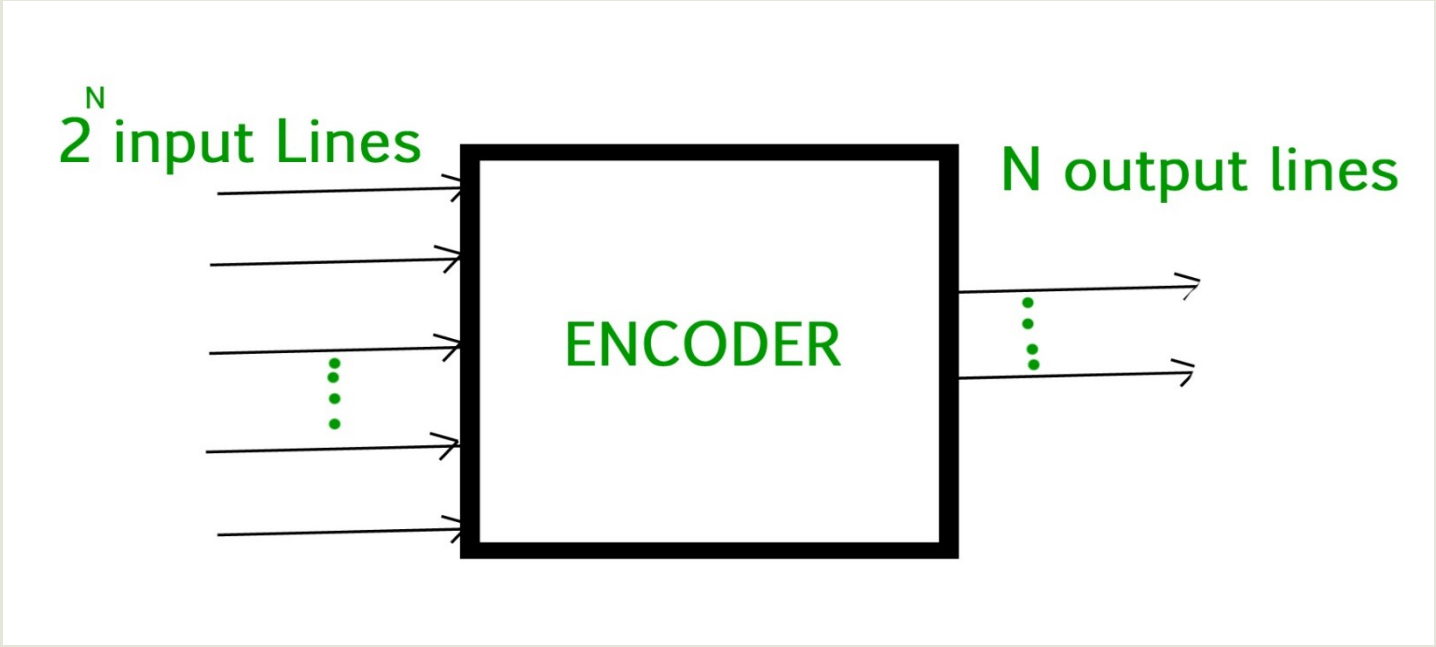
It has maximum of **2^n input lines** and **'n' output lines**, hence it encodes the information from 2^n inputs into an n-bit code.

It will produce a binary code equivalent to the input. Therefore, the encoder encodes 2^n input lines with 'n' bits.

The encoders and decoders play an essential role in digital electronics projects; encoders & decoders are used to convert data from one form to another form.

Encoder and Decoder

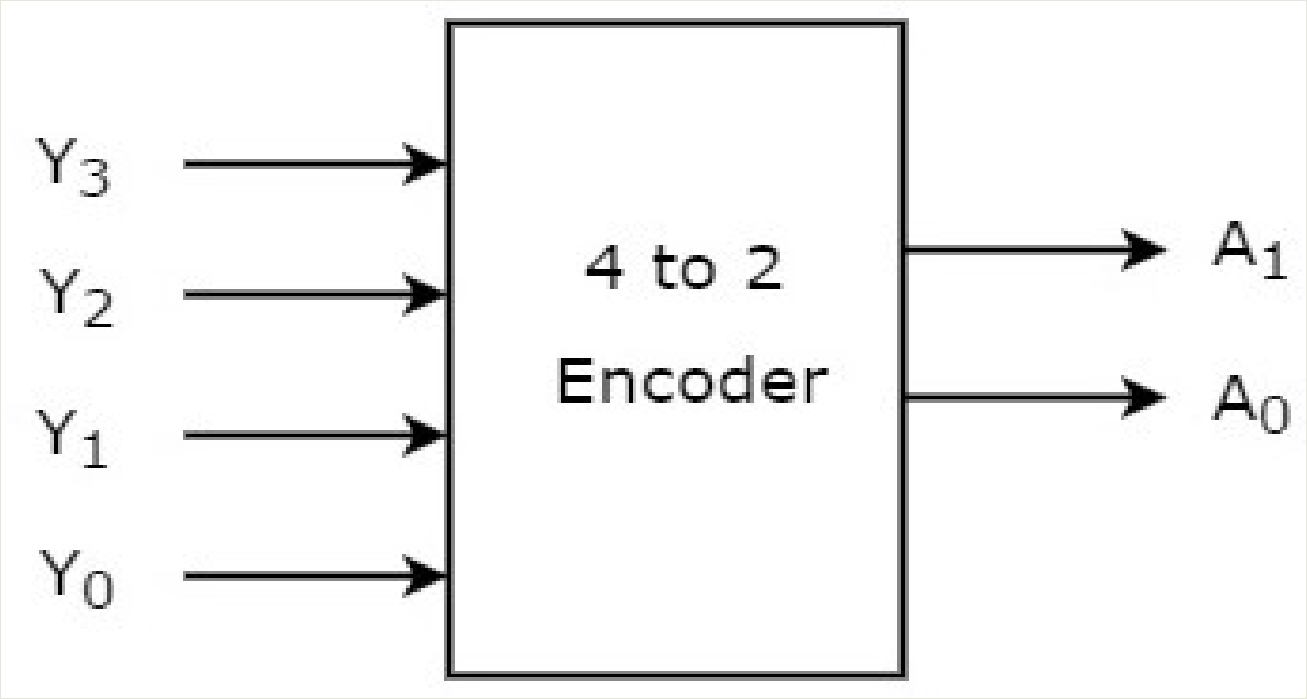
hese are frequently used in communication system such
elecommunication, networking, etc. to transfer data from one en
he other end.



Encoder and Decoder

to 2 Encoder

to 2 Encoder has four inputs Y_3, Y_2, Y_1 & Y_0 and two outputs
A_0. The **block diagram** of 4 to 2 Encoder is shown in
llowing figure.



Encoder and Decoder

t any time, only one of these 4 inputs can be '1' in order to
he respective binary code at the output. The **Truth table** of 4
ncoder is shown below.

Inputs				Outputs	
Y_3	Y_2	Y_1	Y_0	A_1	A_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

Encoder and Decoder

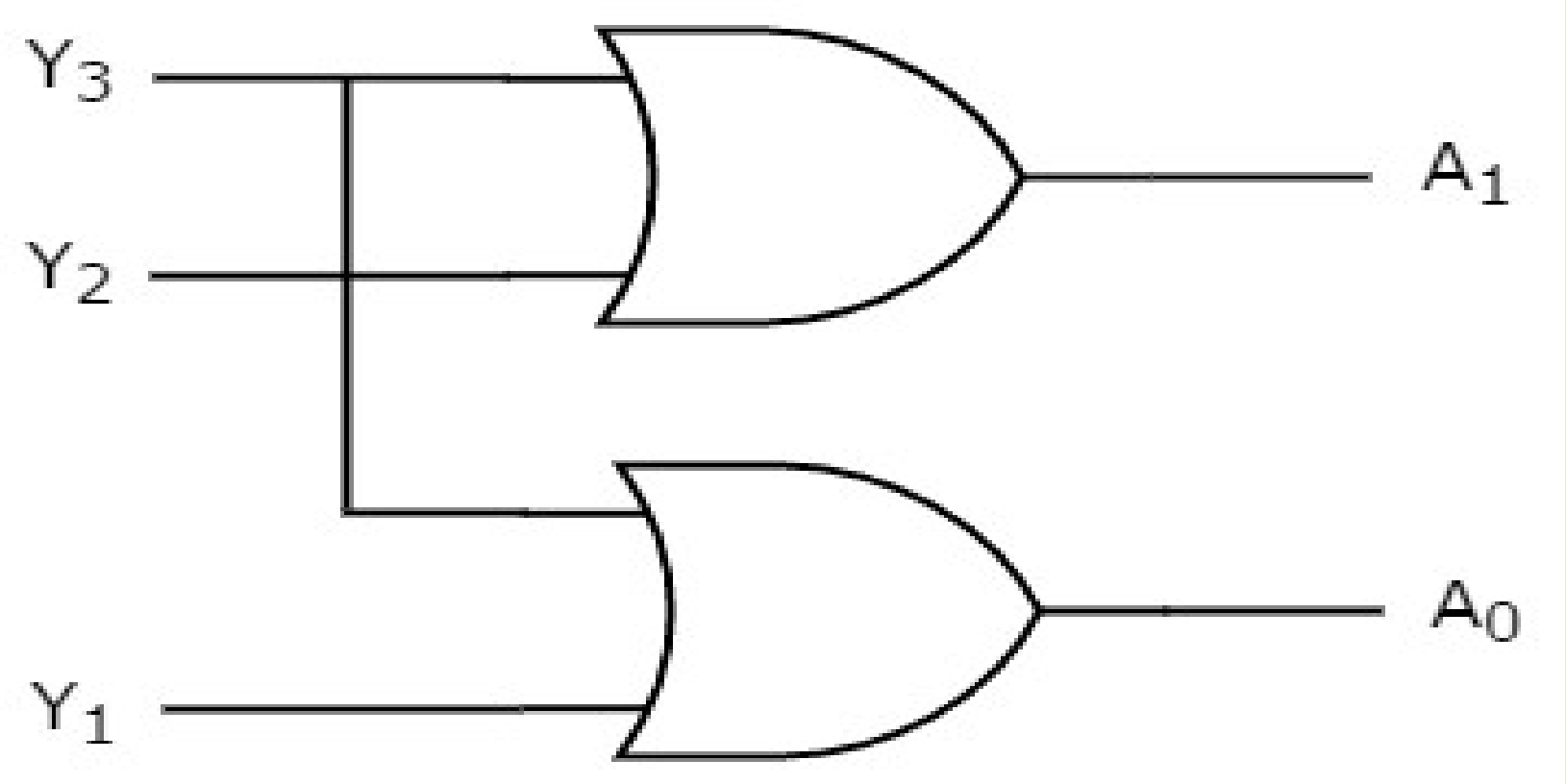
From Truth table, we can write the **Boolean functions** for each output as

$$A_1 = Y_3 + Y_2$$

$$A_0 = Y_3 + Y_1$$

We can implement the above two Boolean functions by using two input OR gates. The **circuit diagram** of 4 to 2 encoder is shown in the following figure.

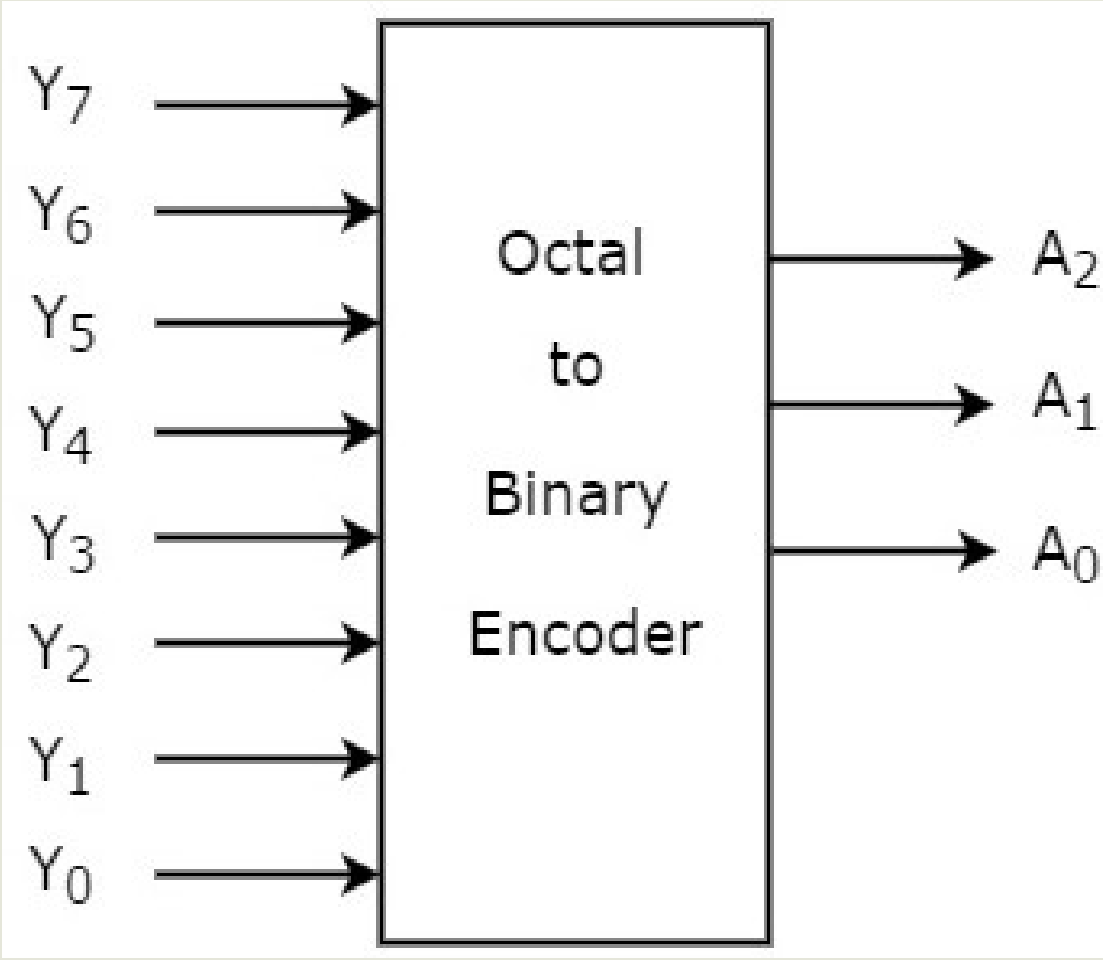
Encoder and Decoder



Encoder and Decoder

ctal to Binary Encoder

ctal to binary Encoder has ght inputs, Y_7 to Y_0 and ree outputs A_2, A_1 & A_0. ctal to binary encoder is thing but 8 to 3 encoder. e **block diagram** of octal to nary Encoder is shown in e following figure.



Encoder and Decoder

any time, only
e of these eight
puts can be '1'
order to get
e respective
nary code. The
uth table of
tal to binary
coder is shown
low.

Inputs								Outputs	
Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0	A_2	A_1
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	1	0	0	0	1
0	0	0	0	1	0	0	0	0	1
0	0	0	1	0	0	0	0	1	0
0	0	1	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	1
1	0	0	0	0	0	0	0	1	1

Encoder and Decoder

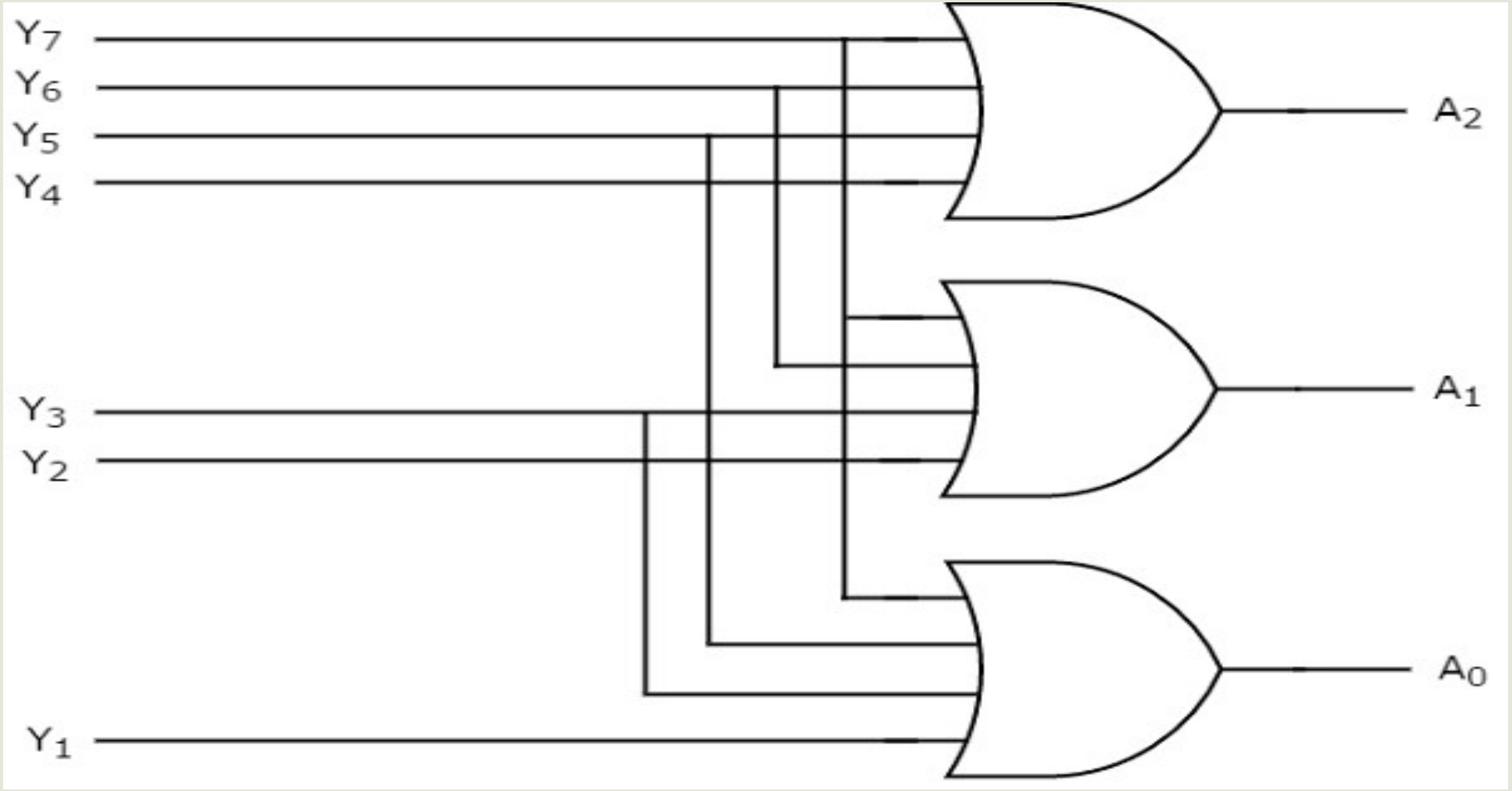
From Truth table, we can write the **Boolean functions** for each output as

$$A_2 = Y_7 + Y_6 + Y_5 + Y_4$$

$$A_1 = Y_7 + Y_6 + Y_3 + Y_2$$

$$A_0 = Y_7 + Y_5 + Y_3 + Y_1$$

Encoder and Decoder



Encoder and Decoder

rawbacks of Encoder

here is an ambiguity, when all outputs of encoder are equal to zero. Becaus
ould be the code corresponding to the inputs, when only least significant inp
ne or when all inputs are zero.

more than one input is active High, then the encoder produces an output, w
ay not be the correct code.

or **example**, if both Y_3 and Y_6 are '1', then the encoder produces 111 at the ou
his is neither equivalent code corresponding to Y_3, when it is '1' nor the equiva
ode corresponding to Y_6, when it is '1'.

Encoder and Decoder

o, to overcome these difficulties, we should assign prioritie
ach input of encoder.

hen, the output of encoder will be the (binary) c
orresponding to the active High input(s), which has hig
riority.

his encoder is called as **priority encoder**.

Encoder and Decoder

Priority Encoder

A 4 to 2 priority encoder has four inputs Y_3, Y_2, Y_1 & Y_0 and two outputs A_1 & A_0

Here, the input, Y_3 has the highest priority, whereas the input, Y_0 has the lowest priority.

In this case, even if more than one input is '1' at the same time, the output will be the (binary) code corresponding to the input, which is having **higher priority**.

Encoder and Decoder

We considered one more **output, V** in order to know, whether the co
available at outputs is valid or not.

- If at least one input of the encoder is '1', then the code available
  outputs is a valid one. In this case, the output, V will be equal to 1.
- If all the inputs of encoder are '0', then the code available at outputs
  not a valid one. In this case, the output, V will be equal to 0.

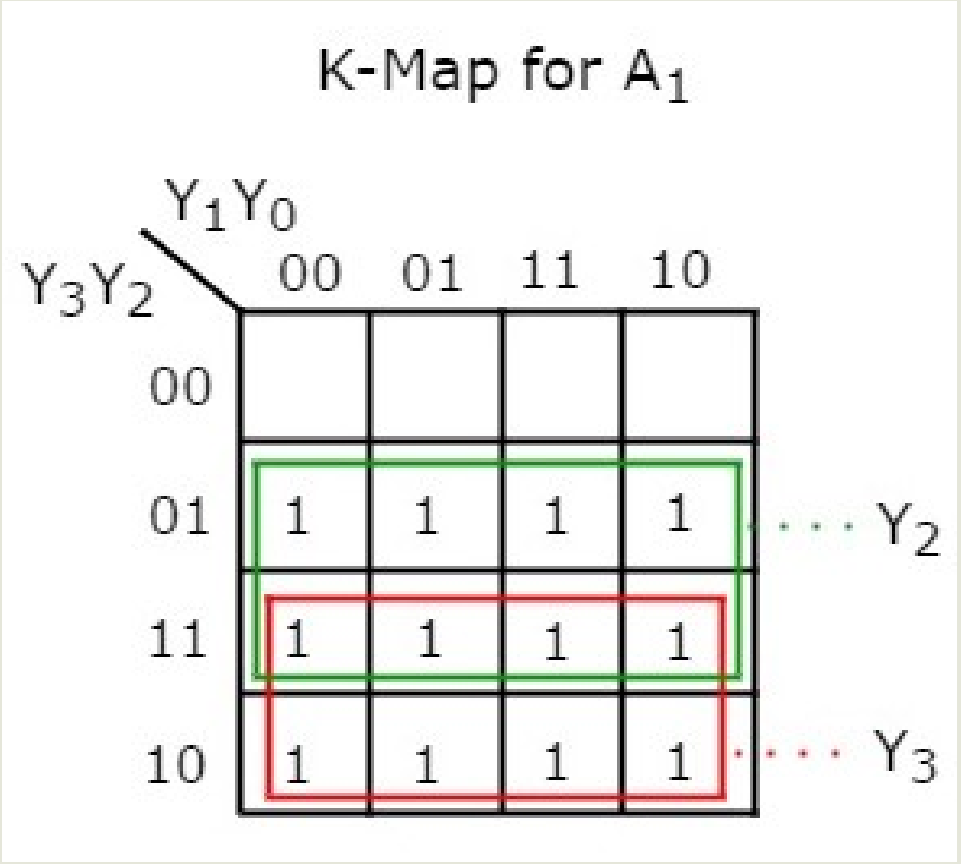
Encoder and Decoder

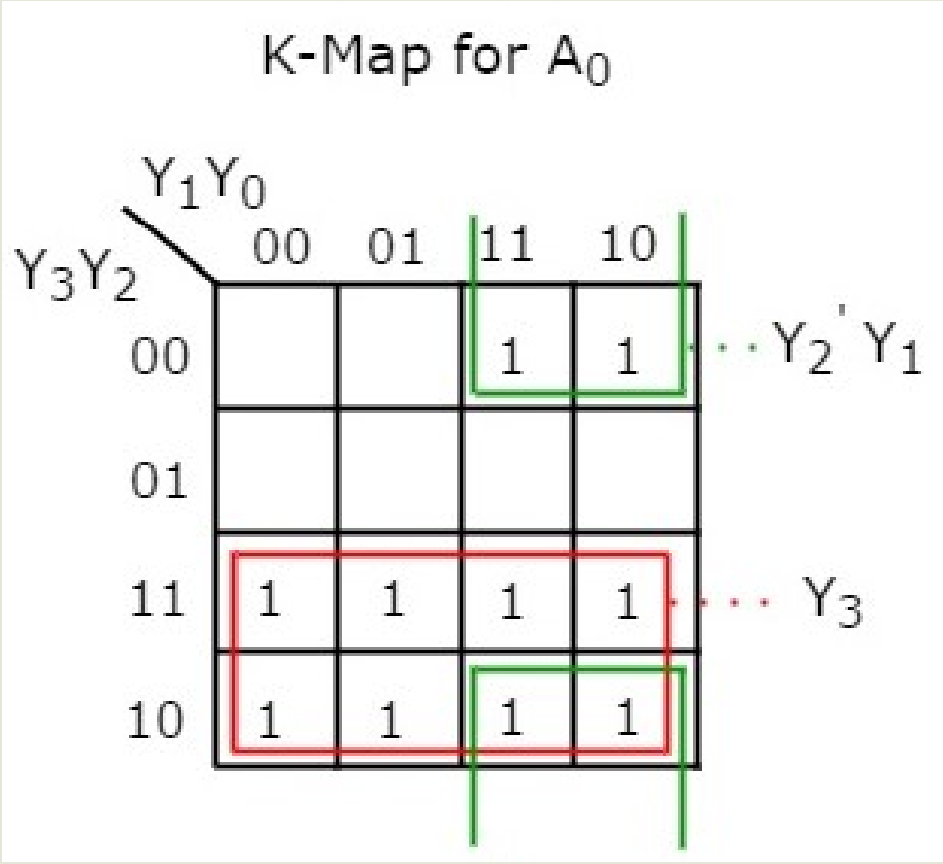
uth table of 4 to 2 priority encoder is shown below.

Inputs				Outputs		
Y_3	Y_2	Y_1	Y_0	A_1	A_0	V
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	x	0	1	1
0	1	x	x	1	0	1
1	x	x	x	1	1	1

Encoder and Decoder

e **4 variable K-maps** for getting simplified expressions for each output.





Encoder and Decoder

The simplified **Boolean functions** are

$$A_1 = Y_3 + Y_2$$

$$A_0 = Y_3 + Y_2'Y_1$$

Similarly, we will get the Boolean function of output, V as

$$V = Y_3 + Y_2 + Y_1 + Y_0$$