Encoding using (n-k)bit shift register

In order to obtain remainder polynomial $b(x)$, we have to perform the division of $x^{n-k} D(x)$ by the generator polynomial $g(x)$.

This division can be accomplished using dividing circuit consist of feedback shift register.

Encoder (n,k) cyclic code

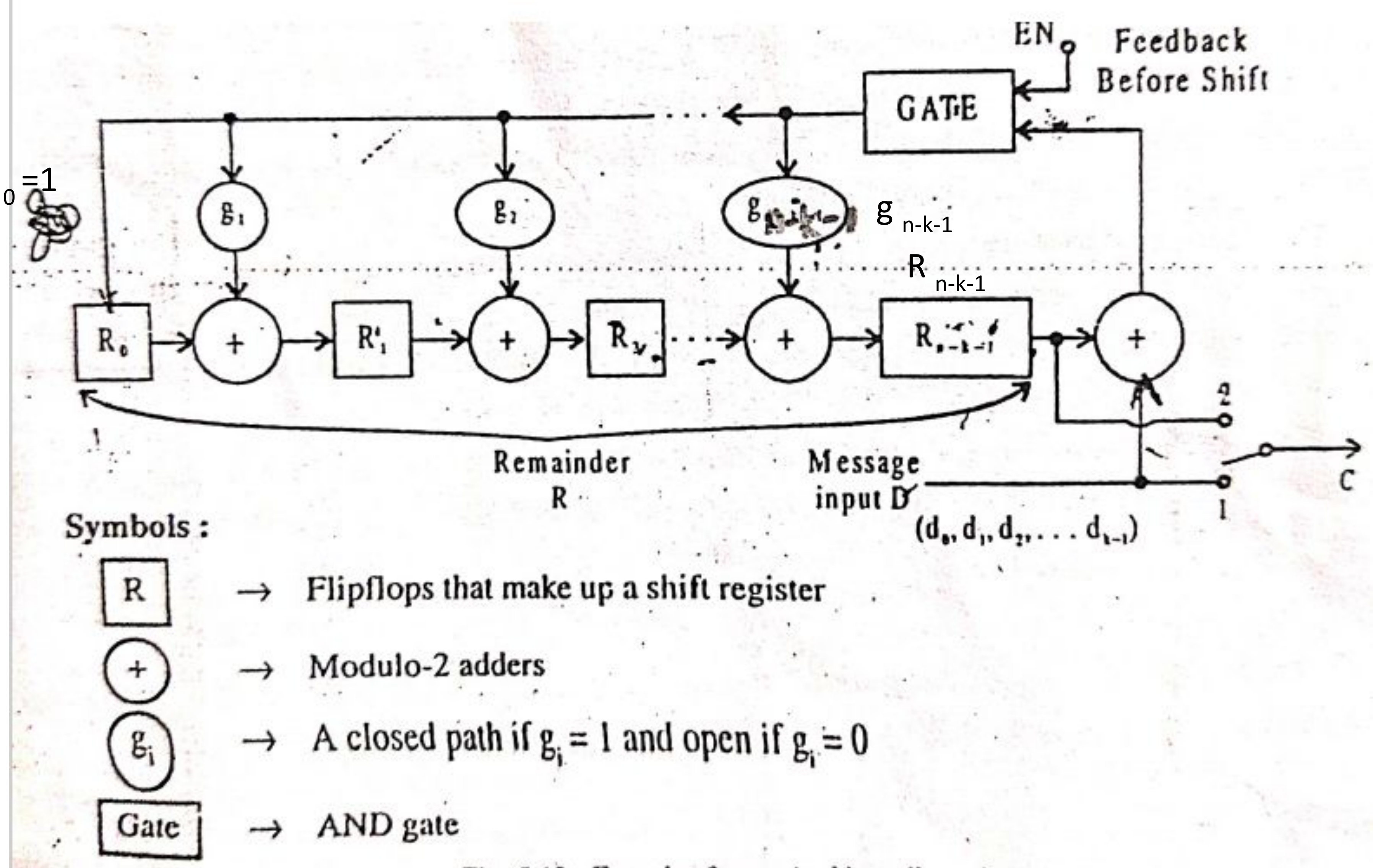

Fig. 5.13 : Encoder for an (n, k) cyclic code

Operation of encoder

Function:

It is assumed that at the occurrence of clock pulse, register inputs are shifted into register and appear at the end of the clock pulse.

1. With the gate turned ON and switch in position1 the information digits ($d_0 d_1 d_2 \dots d_{k-1}$) are shifted into register (with d_{k-1} first) and simultaneously into the communication channel.

As soon as k information digits have been shifted into the register, register contains parity check bits ($R_0\ R_1 \dots R_{n-k-1}$)

2. With the gate turned OFF and the switch in position 2, the contents of the shift register are shifted into the channel. Thus the code-vector $(R_0, R_1, \ldots\ldots, R_{n-k-1}, d_0, d_1, \ldots, d_{k-1})$ is generated and sent over the channel.

Example

Design an encoder for (7,4) Bcyclic code generated by $g(x)$;
$g(x) = 1 + x + x^3$. verify its operation using the message vectors
1001 and 1011.

Solution

$$g(x) = g_0 + g_1 x + g_2 x^2 + \dots g_{n-k} x^{n-k}.$$

$$= 1 + g_1 x + g_2 x^2 + \dots x^{n-k}.$$

$$g_0 = g_{n-k} = 1$$

Given.

$g(x) = 1 + x + x^3$ for $(7,4)$ cyclic code.

comparing coeffecients we have

$$g_0 = 1 \quad g_1 = 1 \quad g_3 = 1 \quad g_2 = 0$$

Encoder circuit

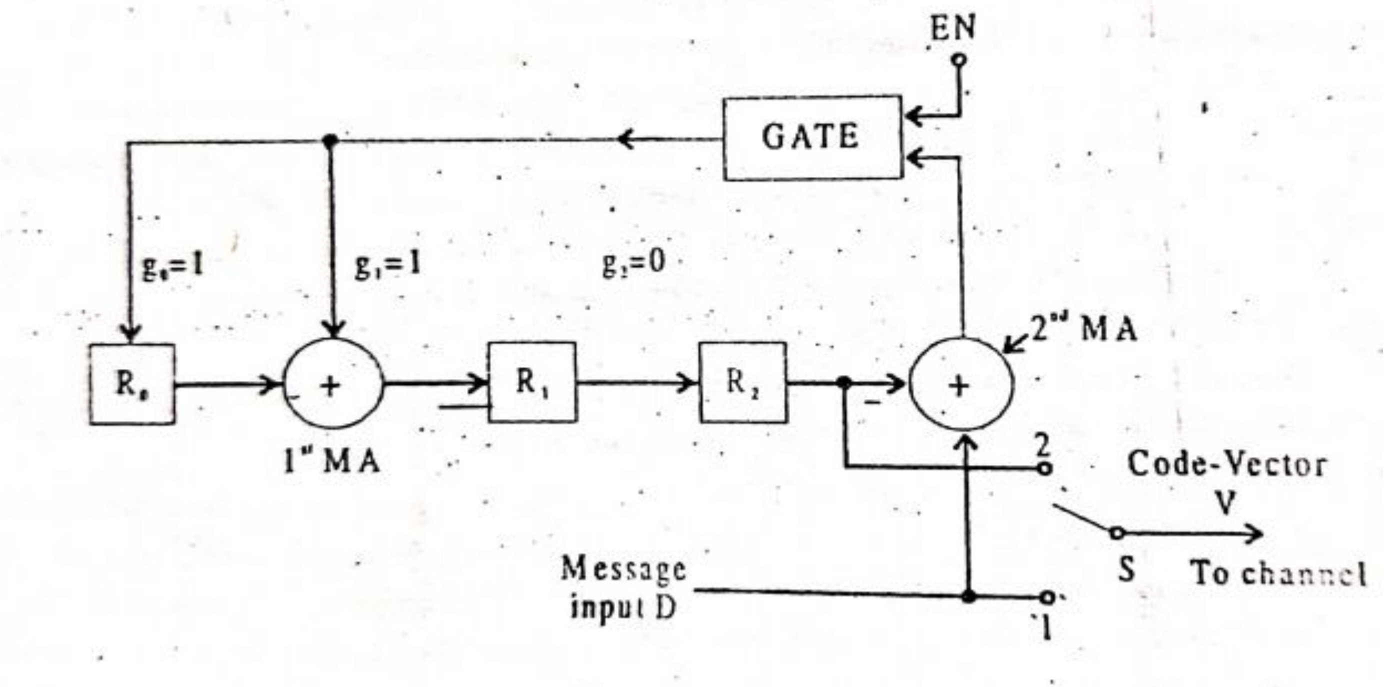

Fig. 5.14 : Encoder for (7, 4) cyclic code of example 5.24

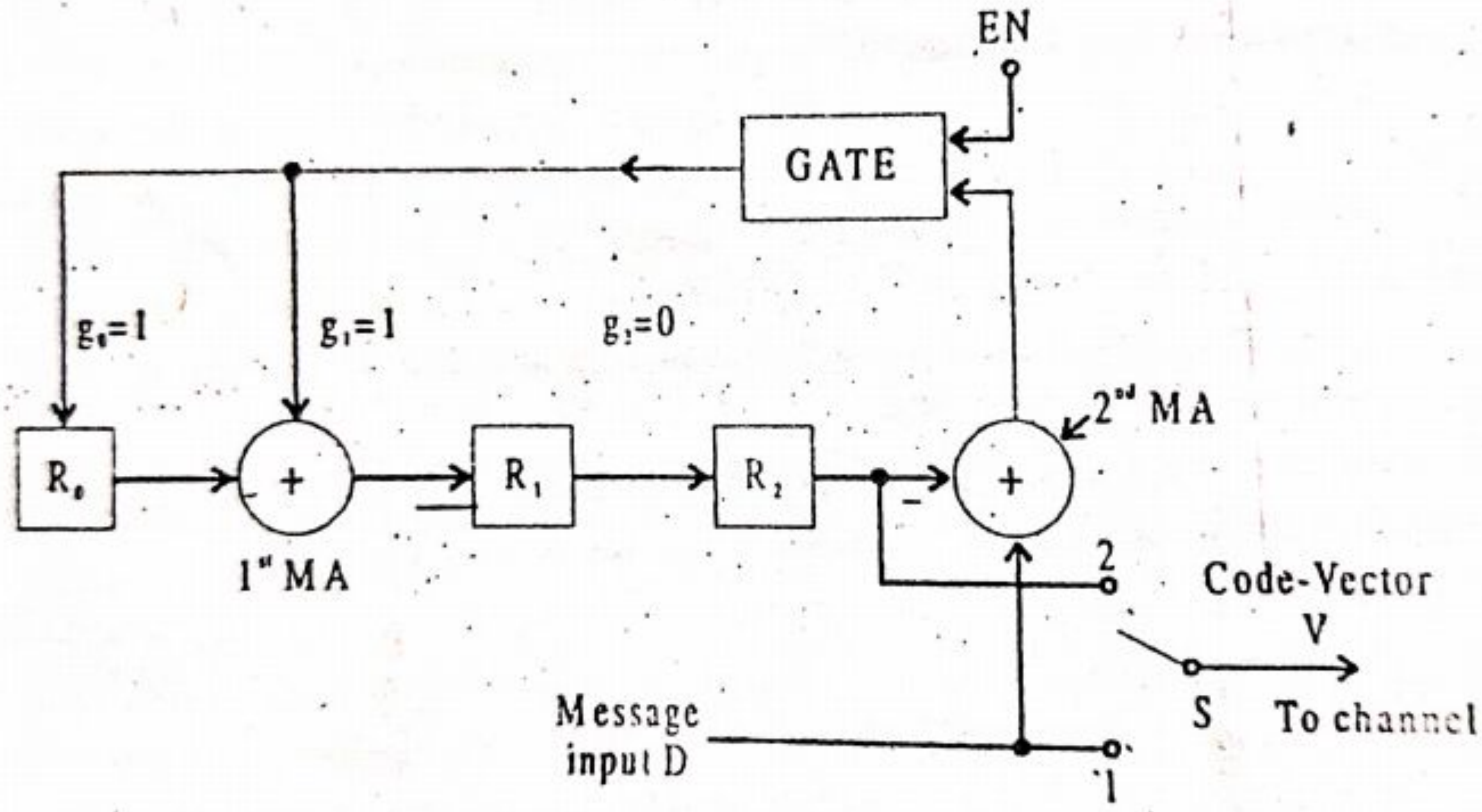

Fig. 5.14 : Encoder for (7, 4) cyclic code of example 5.24

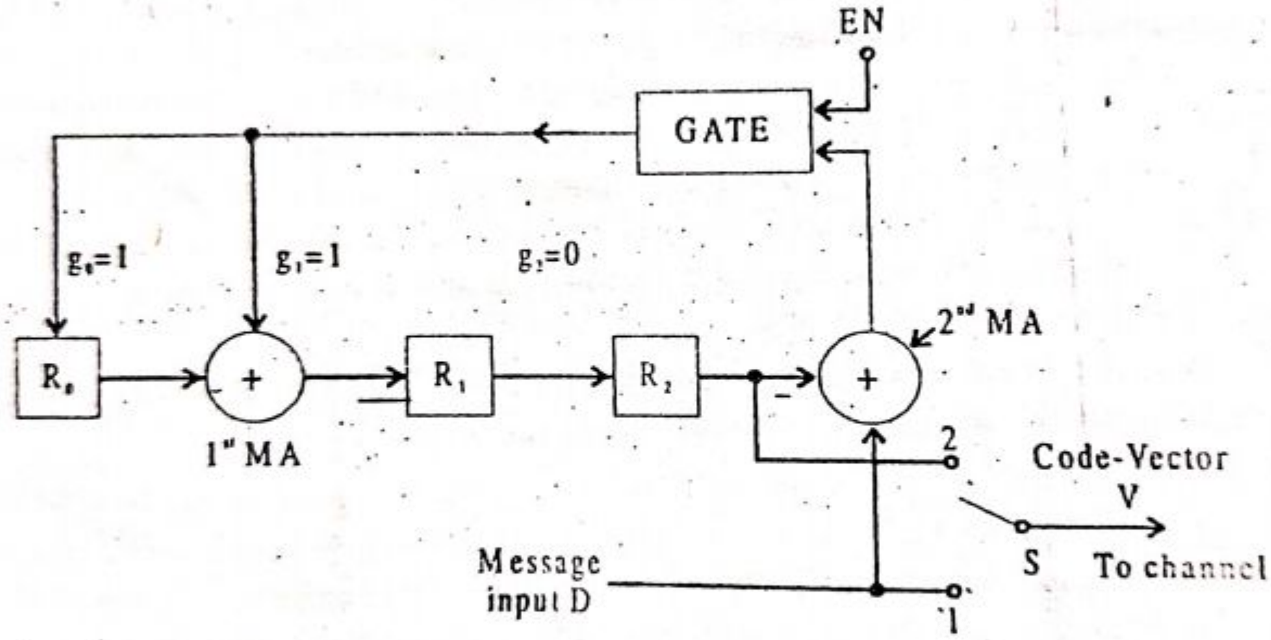

Fig. 5.14 : Encoder for (7, 4) cyclic code of example 5.24

(i) For the message D = [1 0 0 1], the shift register contents are shown in table 5.22.

Number of shifts	Input D	Shift Register Contents R_0	R_1	R_2	Remainder bits → R
Initialisation → switch S is in position-1 and gate is turned ON		0	0	0	-
1	1	1	1	0	-
2	0	0	1	1	-
3	0	1	1	1	-
4	1	0	1	1	-
Switch S moves to position-2 and gate is turned OFF					
5	X	0	0	1	1 (R_2)
6	X	0	0	0	1 (R_1)
7	X	0	0	0	0 (R_0)

Table 5.22 Contents of shift register in the encoder of figure 5.14 for message sequence D = 1001

Mechanism of operation

Initialisation → clear $R_0 R_1 R_2$ . .

ie $R_0 R_1 R_2 = 0\ 0\ 0$.

Then S is in position 1 to read inputs · Gate is on.

2· if data $d_0 d_1 d_2 d_3 = 1001$ · moving D_3 first to 2^{nd} M.A ·

$R_2 = 0$ o/p of 2^{nd} MA = 1.

3. o/p of gate is moved to R_0 - and $R_0 = 1$. (gate = 1) ·

The previous value of $R_0 = 0$ was ~~moved~~ now shifted to R_1 ·M A (1) ·

and $R_1 = 0 \oplus 1 = 1$ ·

$R_1 = 1$

Previous value of R_1 is moved to R_2 directly. $R_2 = 0$.

So $\underline{\text{II}}^{nd}$ sequence $R_0 R_1 R_2 = 110$.

4. Next i/p is $D_2 = 0$. Move D_2 to 2^{nd} MA.

$0 \oplus 0 = 0$. gate $= 0 \rightarrow$ moves to $R_0 = 0$.

$1 \oplus 0 = 1$ $R_1 = \underline{1}$. $R_2 =$ previous $R_1 = 0$.

$\underline{\text{III}}$ rd sequence is ~~001~~. 011.

5. Next input is 0. gate is $0 \oplus 1 = 1$.

$R_0 = 1$. $R_1 = 0 \oplus 1 = 1$. $R_2 = 1$.

$\underline{\text{IV}}^{th}$ seq $= \underline{111}$

6). Next (&last) input $d_0 = 1$.

Modulo adder MA(2) will have inputs (

gate is zero.

$R_0 = 0$.

$R_1 = 1 \oplus 0 = 01$. consider ~~gate~~ value g_1 has same gate value

$R_2 = 1$

Last $\underline{\underline{011}}$.

when all data bits are moved into the register; final contents of shift register is ~~00~~ $\underline{011}$ These are coefficients of polynomial $R(x)$.

Now switch S is shifted to position 2 and gate is turned OFF (EN=0) & contents of SR are shifted into channel using 3 more shifts. The codevector is then

$$\boxed{0111001}$$

The codevector generated is sent over the channel.