Slotted and Pure ALOHA

Find throughput.

Date ______
Page ______

Throughput

Slotted ALOHA / Pure ALOHA

$T_{fr} = \frac{\text{Frames}}{\text{Bandwidth}}$ (Usually has 10^3) $= \_\_\_\_$ ms

$\_\_\_\_$ frames per sec

$\_\_\_\_ \times 10^{-3}$ frames per ms $= G$

$\eta = G * e^{-G}$ ← Slotted

$\eta = G * e^{-2G}$ ← Pure

Throughput $= \eta \times$ no. of frames

UGC NET | SUBSCRIBE | Computer4Quant | June 2018

41. A slotted ALOHA network transmits 200-bit frames using a shared channel with a 200 Kbps bandwidth. Find the throughput of the system, if the system (all stations put together) produces 250 frames per second :

(1) 49 (2) 368 (3) 149 (4) 151

$T_t \begin{cases} L = 200 \text{ bit} \\ BW = 200 \times 10^3 \text{ b/s} \end{cases}$

$T_t = \frac{L}{BW} = \frac{200}{200 \times 10^3} = 1 \text{ ms}$

1 sec — 250 frames

1 msec → $250 \times 10^{-3} = \frac{250}{1000} = \frac{1}{4} = G$

Join Telegram: @computer4quant

UGC NET | SUBSCRIBE | Computer4Quant | June 2018

41. A slotted ALOHA network transmits 200-bit frames using a shared channel with a 200 Kbps bandwidth. Find the throughput of the system, if the system (all stations put together) produces 250 frames per second :
(1) 49 (2) 368 (3) 149 (4) 151

$T_t \begin{cases} L = 200 \text{ bit} \\ BW = 200\times10^{3} \text{ b/s} \end{cases}$

$T_t = \frac{L}{BW} = \frac{200}{200\times10^{3}} = 1ms$

1 sec — 250 frames

1 msec → $250\times10^{-3} = \frac{250}{1000} = \frac{1}{4} = G$

$\eta = G * e^{-G}$
$= \frac{1}{4} * e^{-1/4}$
$= 0.195$

$\eta = \eta *$ No. of frames
$= 0.195 * 250$
$= 48.75 = 49$

Join Telegram: @computer4quant

Q. A pure Aloha network transmits 200 bit frames on a shared channel of 200kbps. What is the throughput if the system (all stations together) produces

1) 1000 frame/sec 2) 500 frame/sec 3) ~~250 frames~~/sec

Solution (a) $T_{fr} = \frac{200}{200 \times 10^3} = 10^{-3} = 1ms$

1) If the system creates 1000 frames per second.

this is 1 frame per millisecond $(1000 \times 10^{-3}) = 1$

$\therefore G = 1$

Throughput $S = G \times e^{-2G} = 1 \times e^{-2} = \frac{1}{e^2}$

2) If the system creates 500 frames per second.

$G = 500 \times 10^{-3} = \frac{500}{1000} = \frac{1}{2}$

~~Throughput~~ $S = G \times e^{-2G} = \frac{1}{2} \times e^{-2 \times \frac{1}{2}} = \frac{1}{2} \times e^{-1}$

$= \frac{1}{2e} = 0.184$ (18.4 %)

~~That~~ $\therefore$ throughput $= 500 \times 0.184 = 92$

That means only 92 frames out of 500 will probably survive.

Minimum hamming distance
Error detection and correction

Minimum Hamming Distance Vs Error Detection & Error Correction

→ $c_1 - 0\boxed{0}\boxed{1}\boxed{0}$ ← Hamming dist^ce = 3
$c_2 - 0\boxed{1}\boxed{0}\boxed{1}$

- **Hamming Distance**
  - It is a difference in terms of bits between two codewords.
- **Minimum Hamming Distance**
  - It is a minimum difference in terms of bits between two codewords.
- **Minimum Hamming Distance and Error Detectio**
  - For S error detection, the minimum hamming distance must
- **Minimum Hamming Distance and Error Correcti**
  - For t error correction, the minimum hamming distance must
- **Minimum Hamming Distance of Hamming Code**
  - $d_{min} = n - k + 1$

- It is a minimum difference in terms of bits between two codewords.

Minimum Hamming Distance and Error Detection

- For S error detection, the minimum hamming distance must be S + 1. $d_{min} = S+1$

Minimum Hamming Distance and Error Correction

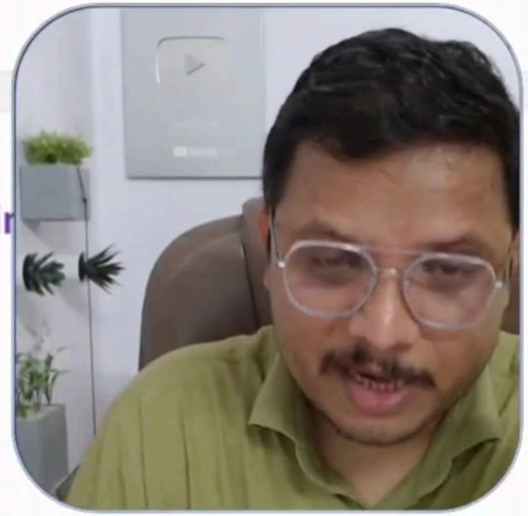
- For t error correction, the minimum hamming distance must be 2t + 1. $d_{min} = 2t+1$

Minimum Hamming Distance of Hamming Code (n,k)

- dmin = n – k + 1

Examples of Hamming Distance

1. If two codewords are 11001101 and 10100101 then fir
   between these two codewords.



- For S error detection, the minimum hamming distance must be S + 1. $d_{min} = S+1$

❖ Minimum Hamming Distance and Error Correction

- For t error correction, the minimum hamming distance must be 2t + 1. $d_{min} = 2t+1$

❖ Minimum Hamming Distance of Hamming Code (n,k)

- dmin = n – k + 1

Total bits   message bits

❖ Examples of Hamming Distance

1. If two codewords are 11001101 and 10100101 then fir
between these two codewords.

2. If the minimum hamming distance is 9 then how many e
corrected by those codewords?

❖ Examples of Hamming Distance

1. If two codewords are 11001101 and 10100101 then find the hamming distance between these two codewords.

$$
\begin{array}{rl}
 & c_1 - 11001101 \\
\oplus & c_2 - 10100101 \\
\hline
 & 01101000 \leftarrow \underline{\underline{3\text{ bits}}}
\end{array}
$$

2. If the minimum hamming distance is 9 then how many errors can be detected and corrected by those codewords?

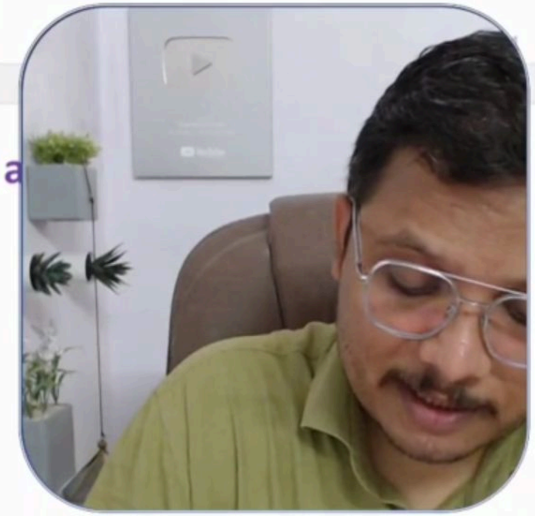
2. If the minimum hamming distance is 9 then how many errors can be detected an corrected by those codewords?

$\rightarrow$ $d_{min} = 9$

$d_{min} = s + 1 \Rightarrow s = 9 - 1 = 8$ — detection

$d_{min} = 2t + 1 \Rightarrow t = \frac{9-1}{2} = 4$ — Correction.

3. For (11,5) hamming code, how many errors can be detected a



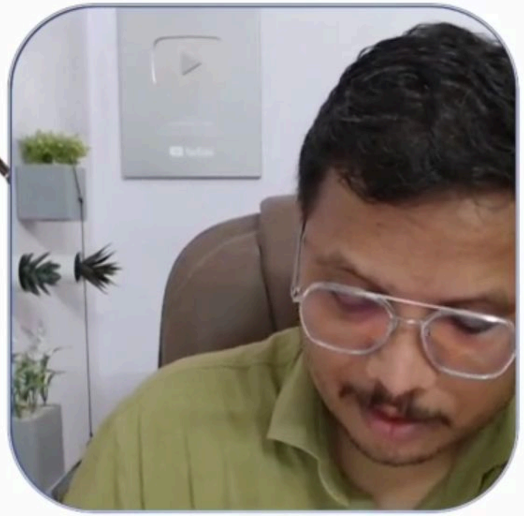
3. **For (11,5) hamming code, how many errors can be detected and corrected?**

$n = 11$ $k = 5$

$$d_{min} = n - k + 1$$
$$= 11 - 5 + 1$$
$$= 7$$

$d_{min} = s + 1 \Rightarrow s = 7 - 1 = 6$ — detected

$d_{min} = 2t + 1 \Rightarrow t = \frac{7-1}{2} = 3$ — Corrected



Q) Find the minimum Hamming distance for the given code words: 00000, 01011,10101, 11110

Between 00000 and 01011:

- 00000
- 01011
  Difference in positions: 2, 3, 4
  Hamming Distance = 3

Between 00000 and 10101:

- 00000
- 10101
  Difference in positions: 1, 3, 5
  Hamming Distance = 3

Between 00000 and 11110:

- 00000
- 11110
  Difference in positions: 1, 2, 3, 4
  Hamming Distance = 4

Between 01011 and 10101:

- 01011
- 10101
  Difference in positions: 1, 2
  Hamming Distance = 2

Between 01011 and 11110:

- 01011
- 11110
  Difference in positions: 1, 3, 4
  Hamming Distance = 3

Between 10101 and 11110:

- 10101
- 11110
  Difference in positions: 2
  Hamming Distance = 1

Cyclic Redundancy Check

Cyclic Redundancy check (CRC)

$\overline{)1010101010}$

→ Based on binary division

total bits = $(m+r)$

$\underbrace{x^4 + x^3 + 1}_{\text{Polynomial}}$

→ Polynomial should not be divisible by x

→ also not with $(x+1)$

→ Can detect all odd errors, Single bit, burst error of length equal to polynomial degree

Cyclic Redundancy check (CRC)

→ Based on binary division

Total bits = $(\overbrace{m}^{\text{message}} + \overbrace{r}^{\text{Redun.}})$

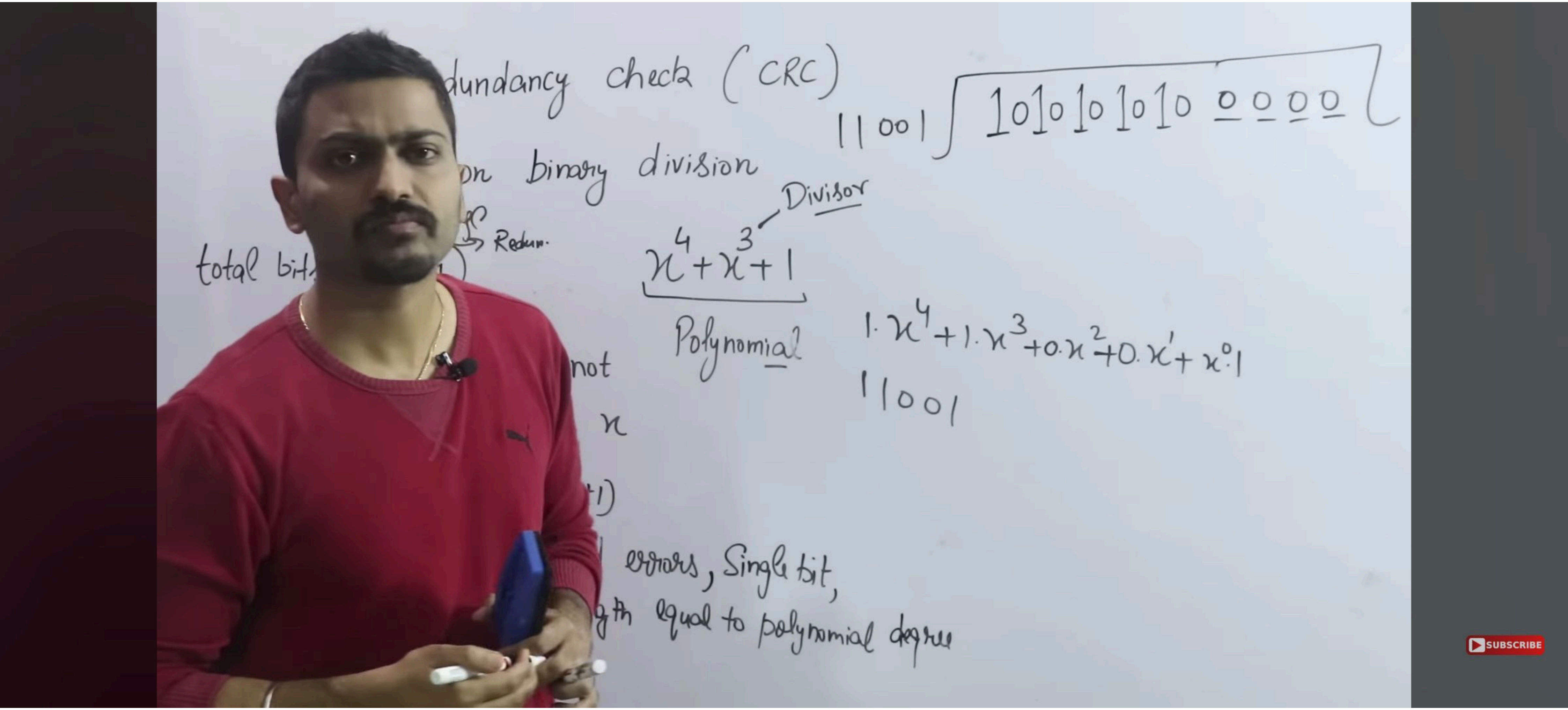
$\underbrace{x^4 + x^3 + 1}_{\text{Polynomial}}$ — Divisor

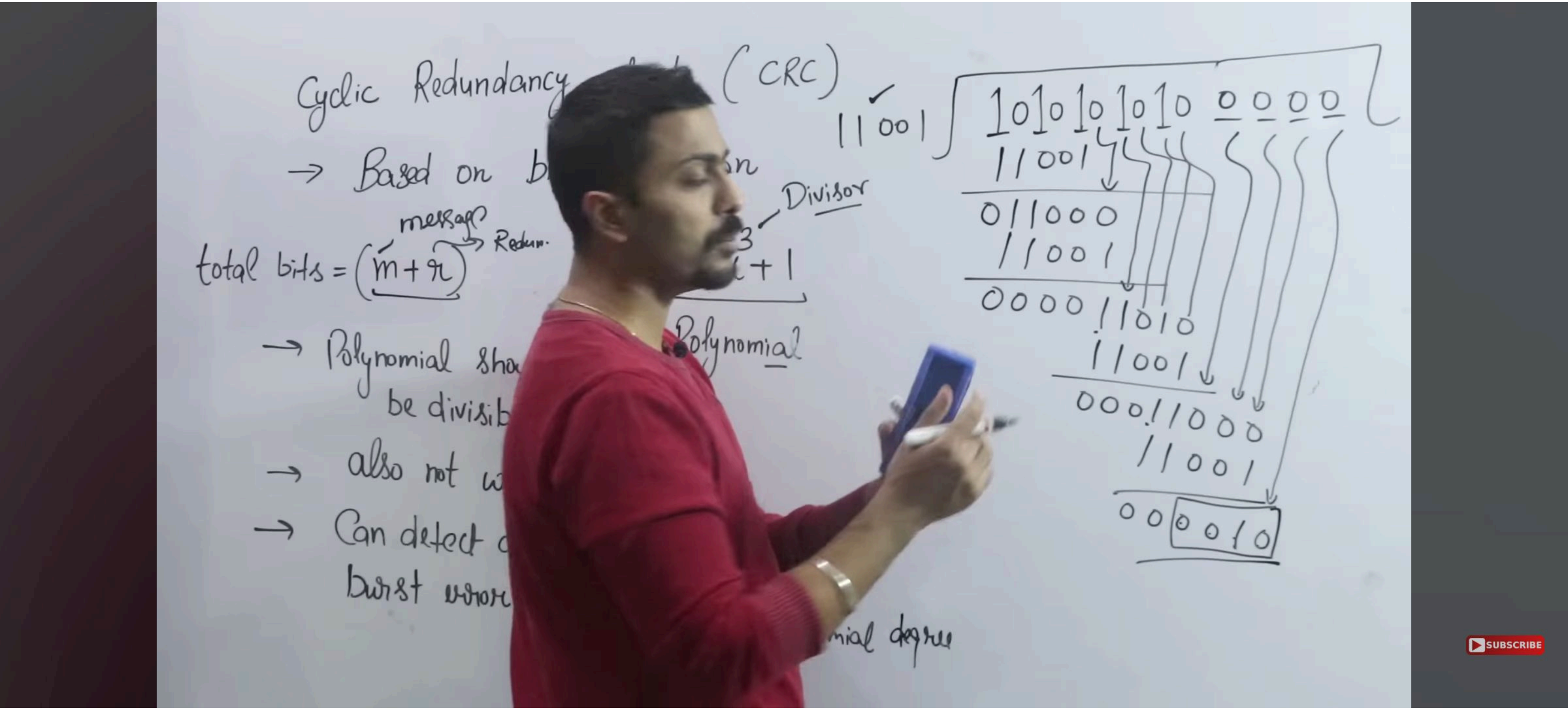
→ Polynomial should not be divisible by x

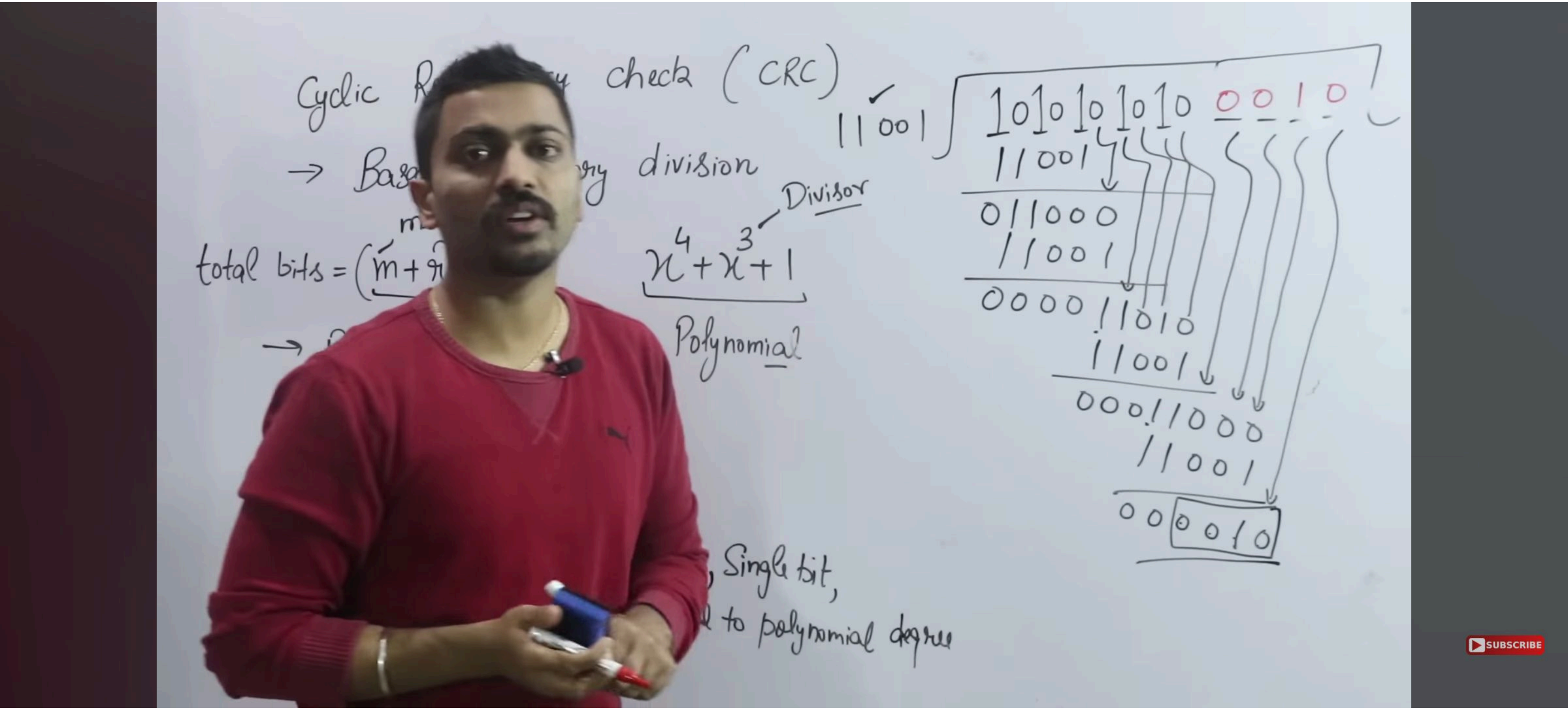
→ also not with $(x+1)$

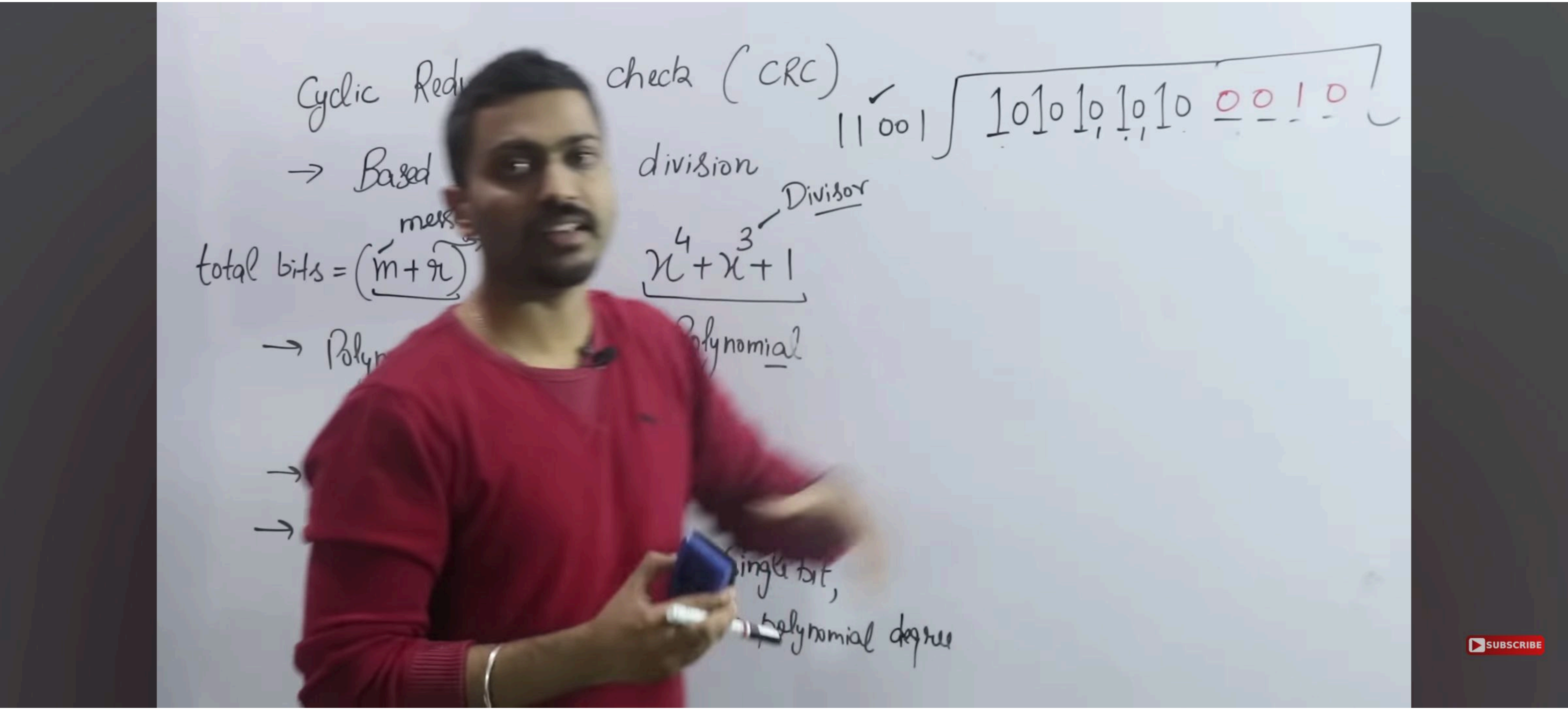
→ Can detect all odd errors, Single bit, burst error of length equal to polynomial degree

$\overline{)\,1010101010}$

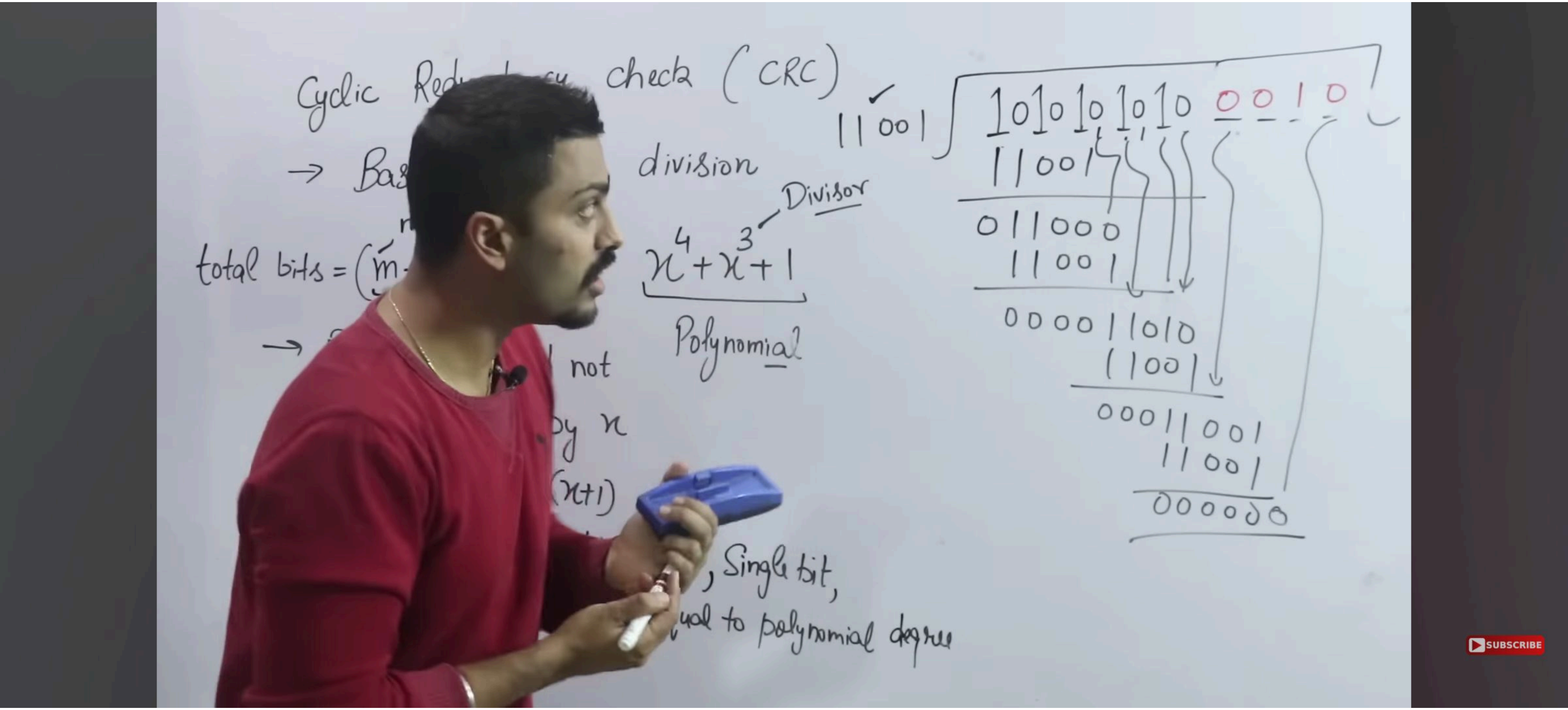

dundancy check (CRC)
11001
1010101010 0000
binary division
Divisor
Redun.
total bit
x^4+x^3+1
Polynomial
$1 \cdot x^4+1 \cdot x^3+0 \cdot x^2+0 \cdot x^1+x^0 \cdot 1$
11001
not
errors, Single bit,
equal to polynomial degree
SUBSCRIBE


Cyclic Redundancy (CRC)
→ Based on
message
Redun.
total bits = (m + r)
Divisor
3
+1
Polynomial
→ Polynomial
be divisib
→ also not
→ Can detect
burst error
degree
11001
1010101010 0000
11001
011000
11001
000011010
11001
00011000
11001
000010
SUBSCRIBE

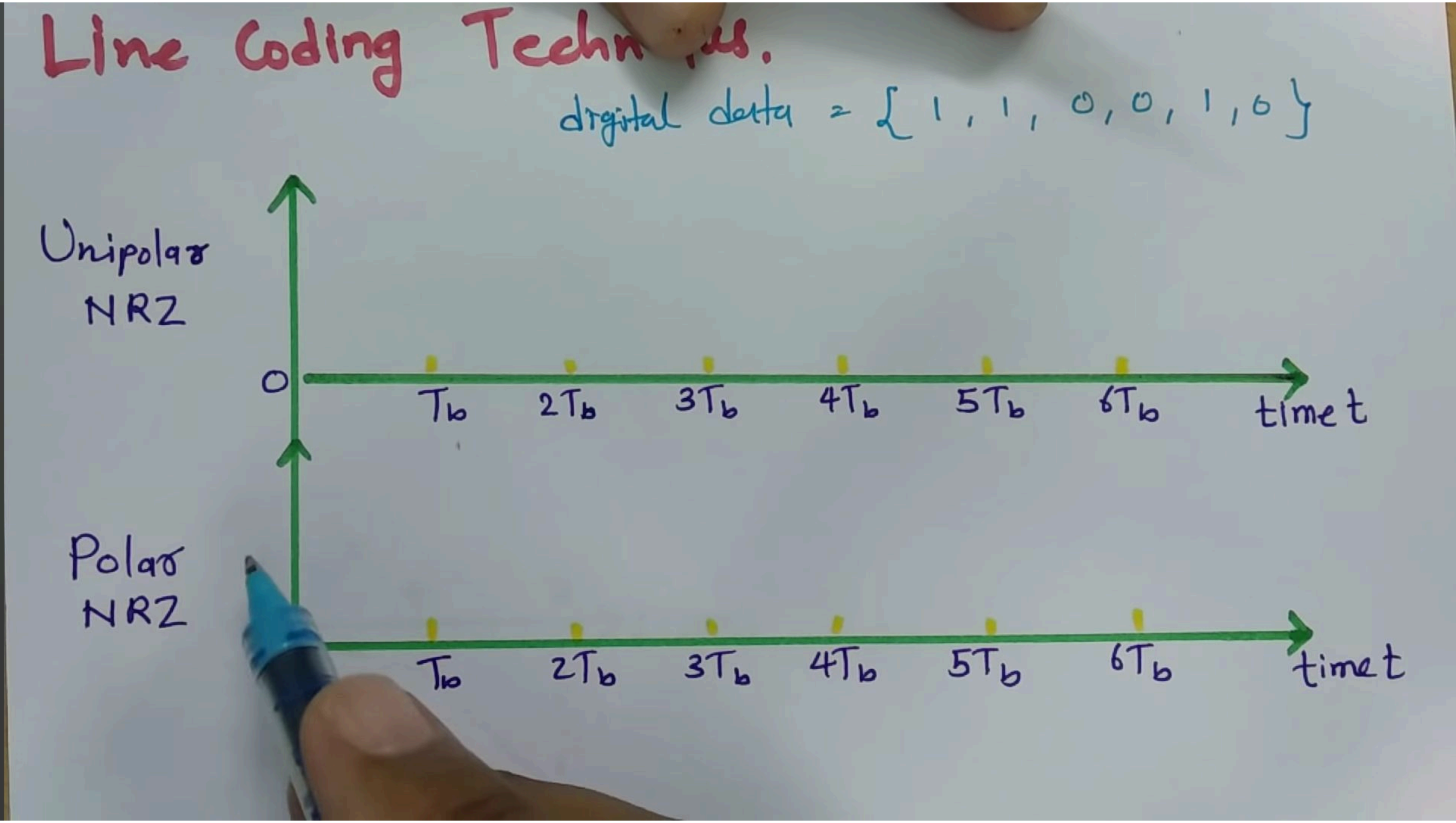

check (CRC)
division
Divisor
total bits =
$x^4 + x^3 + 1$
Polynomial
Single bit,
to polynomial degree
11001
1010101010 0010
11001
011000
11001
000011010
11001
00011000
11001
0000010
SUBSCRIBE

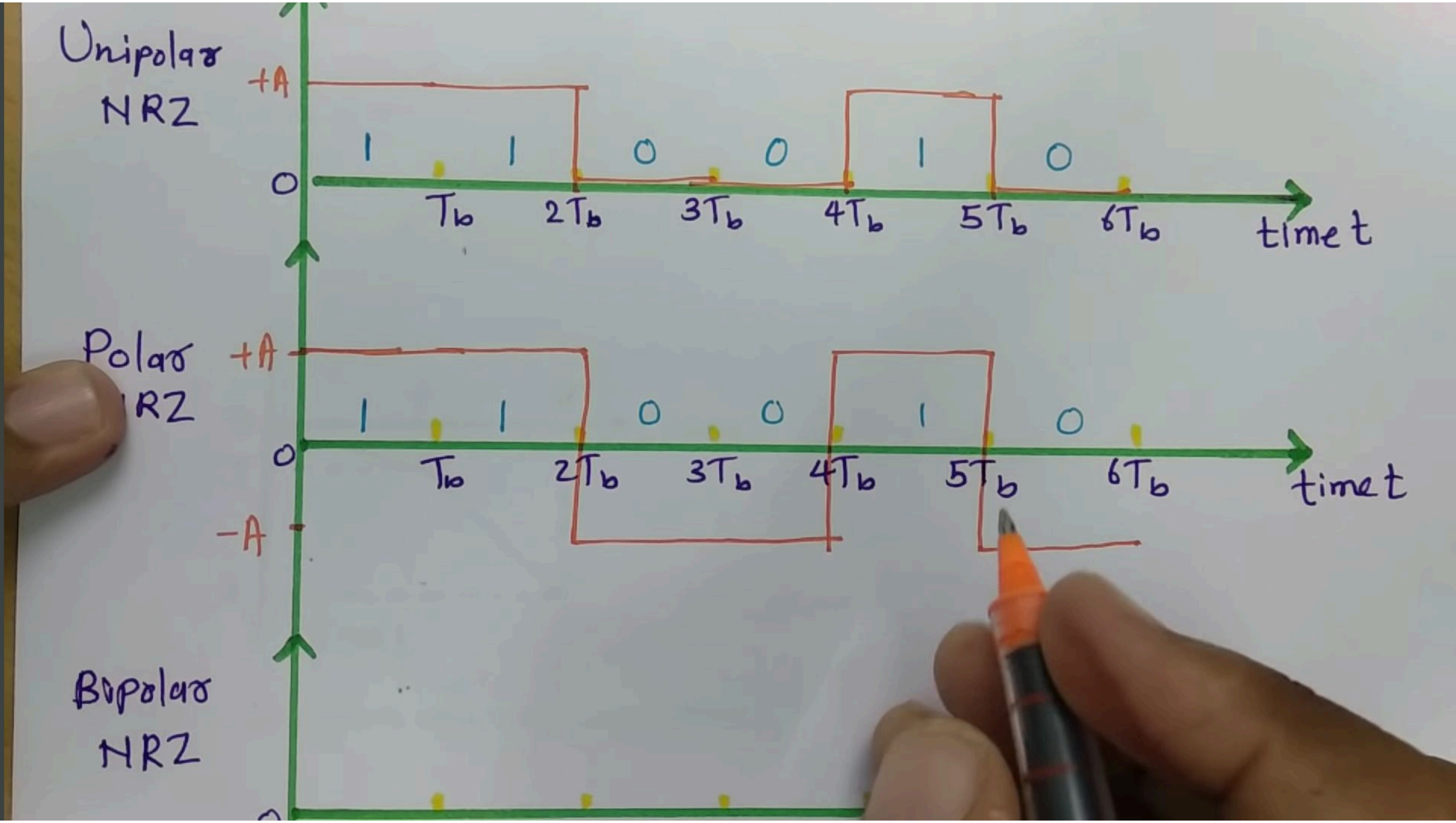


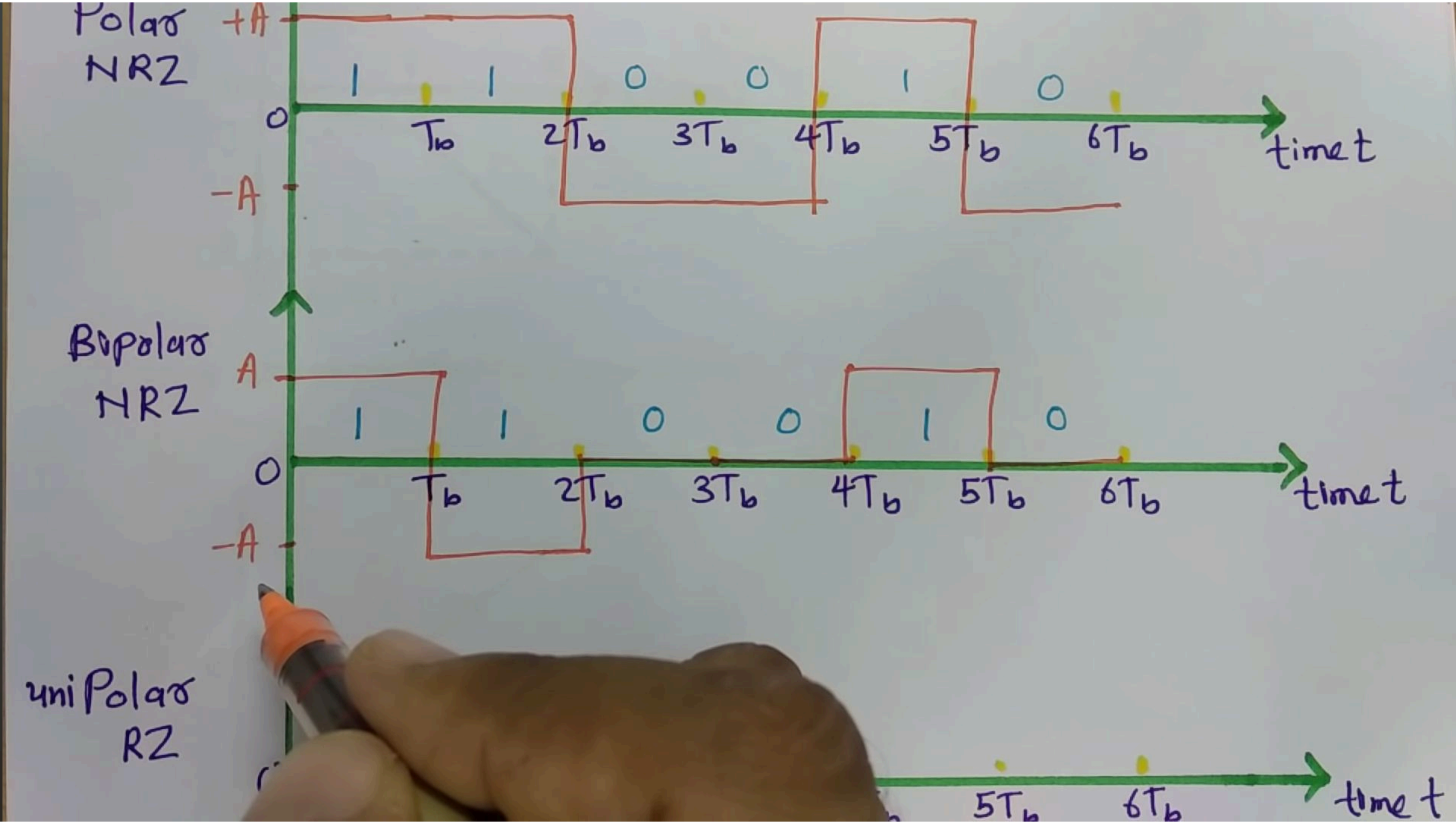
Cyclic Red check (CRC)
11001
1010101010 0010
→ Based
division
Divisor
total bits = (m + r)
x⁴ + x³ + 1
Polynomial
single bit,
polynomial degree
SUBSCRIBE

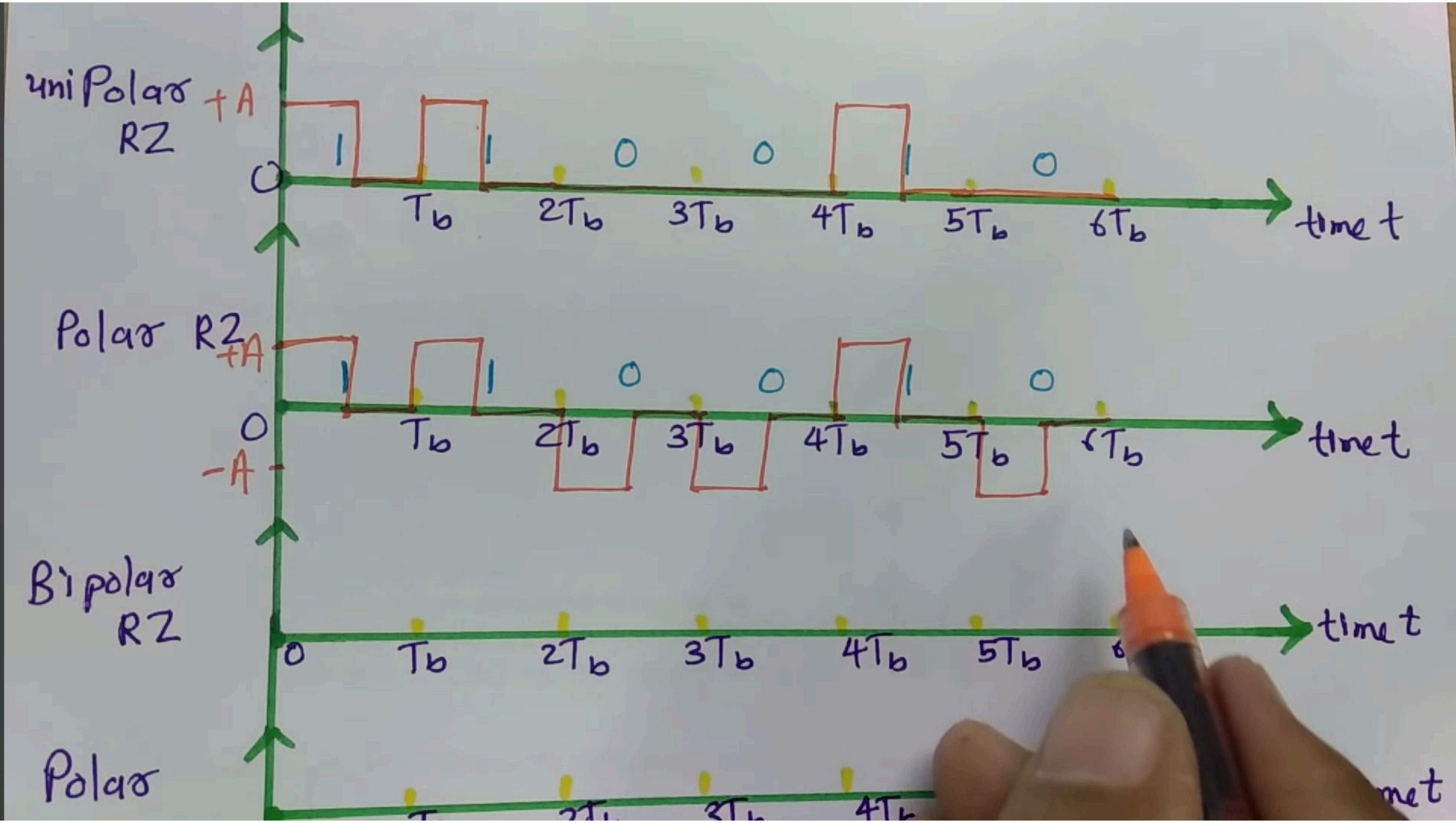

Cyclic Red
check (CRC)
→ Bas
division
Divisor
total bits = (m
$x^4 + x^3 + 1$
Polynomial
→
not
by x
(x+1)
, Single bit,
ual to polynomial degree
11001
1010101010 0010
11001
011000
11001
0000011010
11001
00011001
11001
000000
SUBSCRIBE

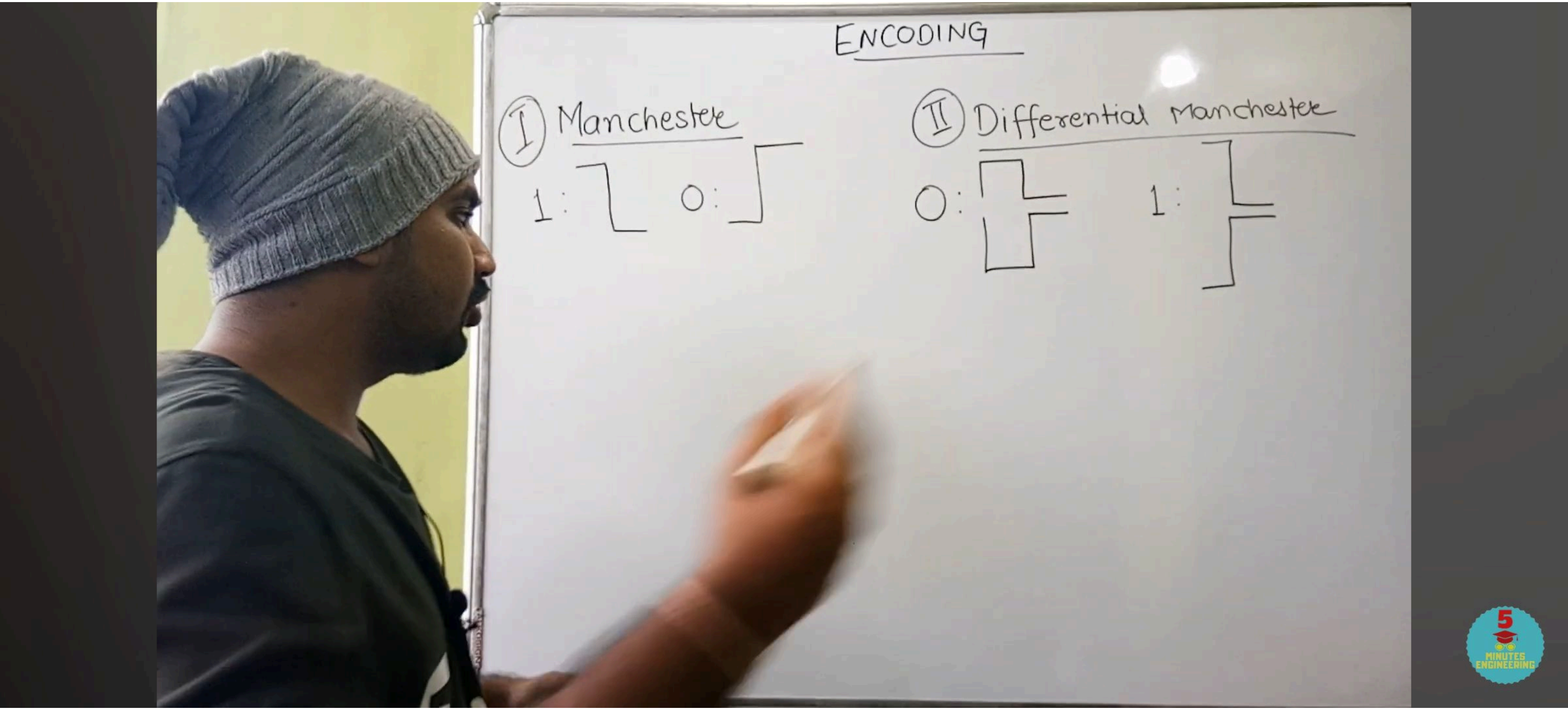
Line coding Waveforms

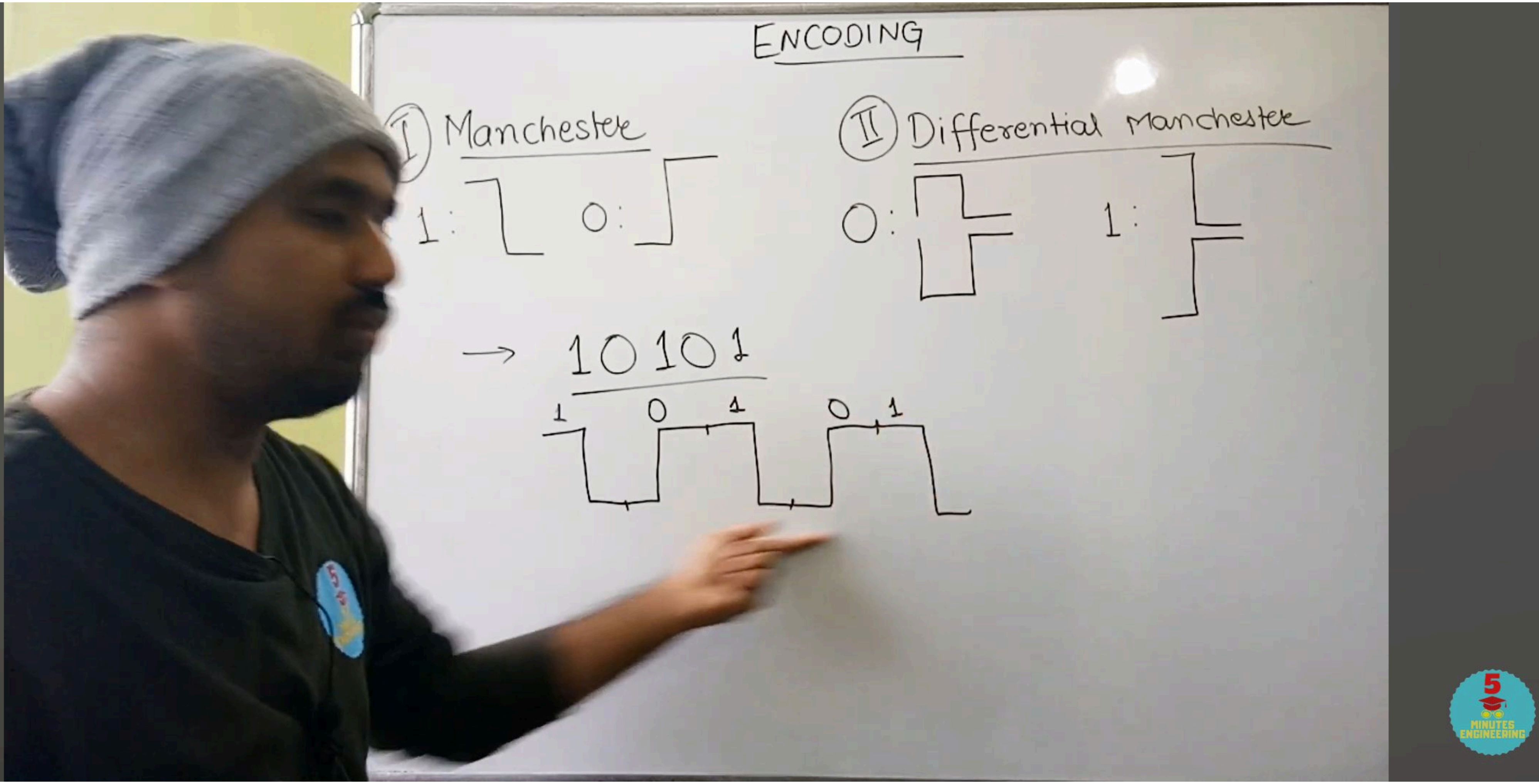

Line Coding Techn es.
digital data = { 1, 1, 0, 0, 1, 0 }
Unipolar
NRZ
0
Tb
2Tb
3Tb
4Tb
5Tb
6Tb
time t
Polar
NRZ
Tb
2Tb
3Tb
4Tb
5Tb
6Tb
time t

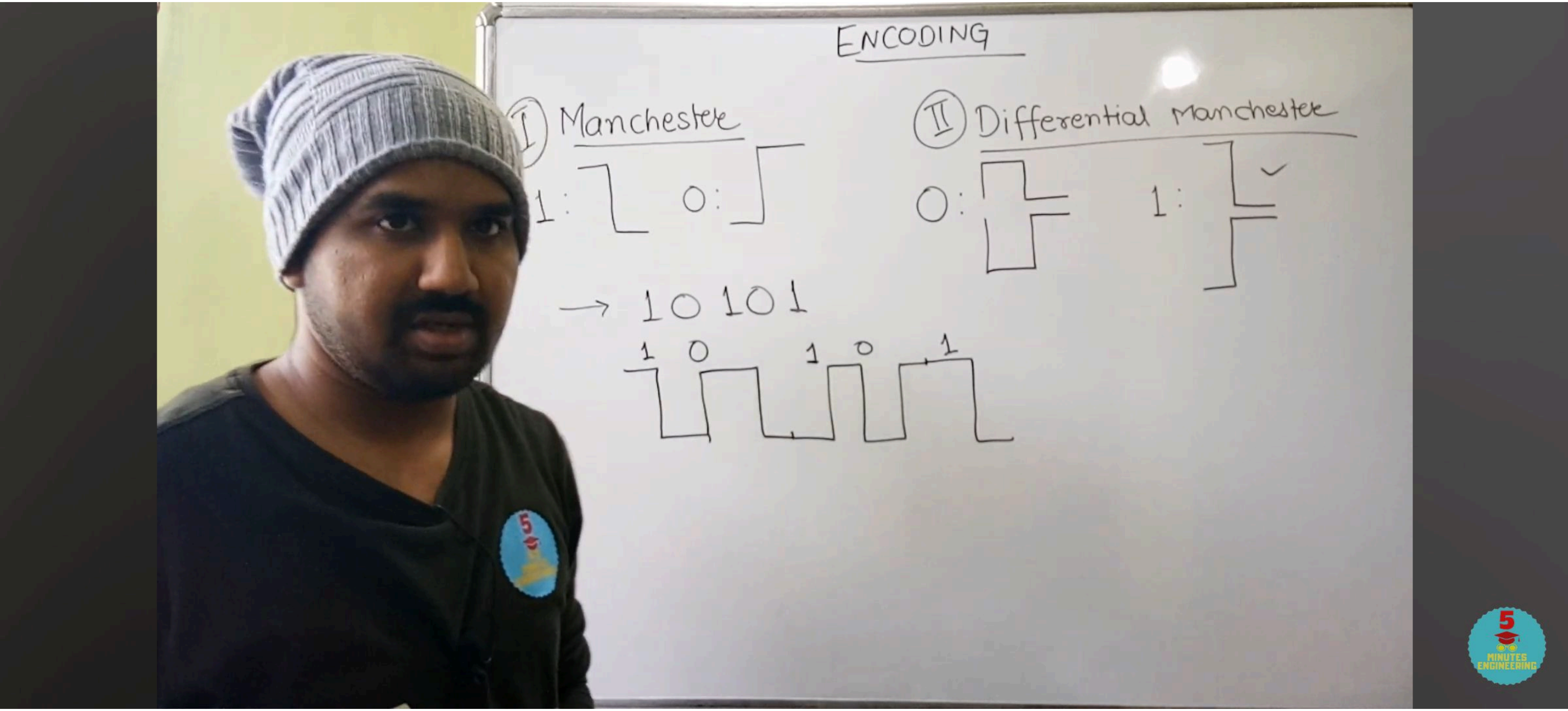

Unipolar
NRZ
+A
0
1 1 0 0 1 0
Tb 2Tb 3Tb 4Tb 5Tb 6Tb
time t
Polar
RZ
+A
0
-A
1 1 0 0 1 0
Tb 2Tb 3Tb 4Tb 5Tb 6Tb
time t
Bipolar
NRZ


Polar NRZ
+A
0
-A
1 1 0 0 1 0
Tb 2Tb 3Tb 4Tb 5Tb 6Tb
time t
Bipolar NRZ
A
0
-A
1 1 0 0 1 0
Tb 2Tb 3Tb 4Tb 5Tb 6Tb
time t
uniPolar RZ
5Tb 6Tb
time t


uniPolar RZ
+A
0
1
1
0
0
1
0
Tb
2Tb
3Tb
4Tb
5Tb
6Tb
time t
Polar RZ
+A
0
-A
1
1
0
0
1
0
Tb
2Tb
3Tb
4Tb
5Tb
6Tb
time t
Bipolar RZ
0
Tb
2Tb
3Tb
4Tb
5Tb
time t
Polar


ENCODING
I Manchester
1:
0:
II Differential Manchester
0:
1:
5 MINUTES ENGINEERING


ENCODING
I Manchester
1 :
0 :
II Differential Manchester
0 :
1 :
→ 10101
1 0 1 0 1
5 MINUTES ENGINEERING


ENCODING
I Manchester
1 :
0 :
II Differential Manchester
0 :
1 :
→ 1 0 1 0 1
1 0 1 0 1
5 MINUTES ENGINEERING