Dayananda Sagar College of Engineering

# Department of Electronics and Communication Engineering

### Shavige Malleshwara Hills, Kumaraswamy Layout, Bengaluru – 560 078.

**(An Autonomous Institute affiliated to VTU, Approved by AICTE & ISO 9001:2008 Certified)**

***Accredited by National Assessment and Accreditation Council (NAAC) with ‘A’ grade***

# Assignment

## Program: B.E. Branch: ECE

Course: Digital Communication Systems Semester: 5 Section: C Course Code: 18EC5DCDCS Date: 17. 12. 2020

### A Report on

**Error Correction and Syndrome Calculation (linear block codes) using MATLAB**

**Submitted by**

**USN NAME**

1DS18EC156 VISHVENDRA SINGH

1DS18EC143 SAMEER GAUTAM

# Faculty In-charge Prof. Vidyashree K N

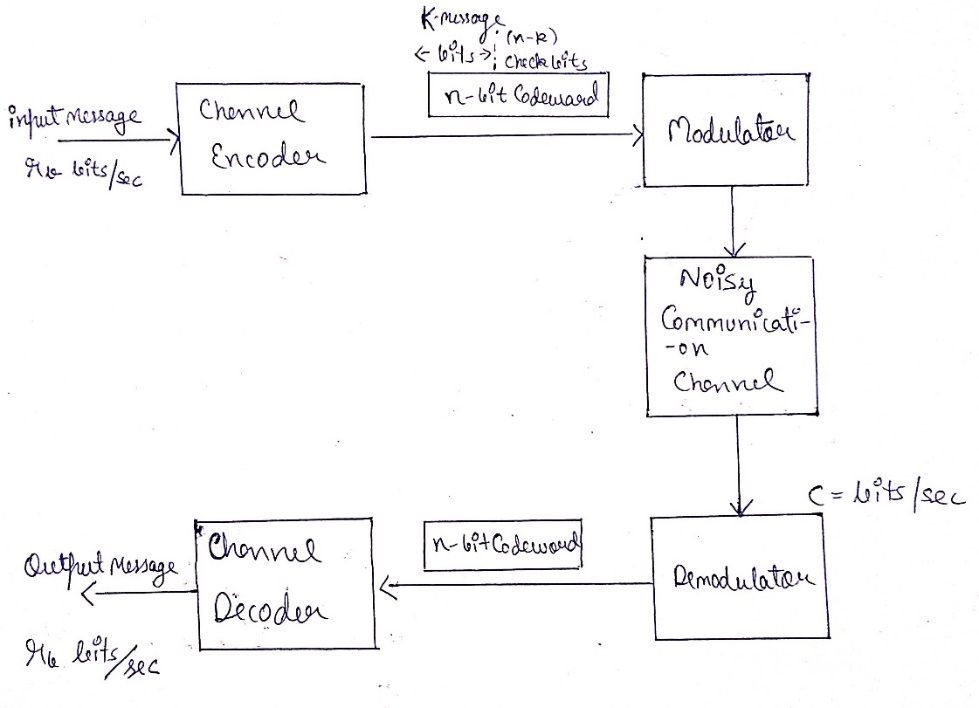
**Signature of Faculty In-charge**

**Introduction and Concept**

Noise or Error is the main problem in the signal while transmitting information from one source to another, which disturbs the reliability of the communication system. Error is a condition when the output information does not match with the input information. During transmission, digital signals suffer from noise that can introduce errors in the binary bits travelling from one system to other. That means a 0 bit may change to 1 or a 1 bit may change to 0. Error control coding is the coding procedure done to control the occurrences of errors. These techniques help in Error Detection and Error Correction. There are many different error correcting codes depending upon the mathematical principles applied to them. But, generally these codes have been classified into Linear block codes and Convolution codes.

Error control coding is nothing but the calculated use of redundancy. The functional blocks that are used are channel encoder at the transmitter and channel decoder at the receiver.

The Channel encoder at the Transmitter systematically adds the digit to transmitted message bits. These additional digits carry no information but makes it possible for the channel decoder to detect the error and correct the error in information bearing digits. This reduces the overall probabilities of error, thereby achieving the desired error-free output signal. These additional digits which carry no information are called REDUNDANT digits and the process of adding these digits is called REDUNDANCY.



The above diagram is the Block diagram of Digital Communication System employing error control coding. The main functional blocks are channel encoder, channel decoder, modulator, demodulator, and noisy communication channel. The source generates a message block at a rate of rb bits/sec and feeds it to the channel encoder. The channel encoder than adds (n-k) number of redundant bits to these k-bit messages to form n-bit Codeword. These (n-k) number of additional bits are also called as Check bits which do not carry any information but helps the channel decoder to detect and correct the errors. The channel decoder then decodes this message to get back the information block at the receiver.

A block of n-coded bits generated by encoder from k information bits is called as (n,k) block codes. If the convolution sum of any two code words(n-bits) belonging to a set of (n, k) block code is also a n-bit code word belonging to the same set of (n,k) block code, then such a block code is called (n, k) linear block code. The coded bits are also called as code word symbols**.**

When a code-vector is transmitted over a noisy communication channel belonging to a (n, k) linear block code, then the error-vector or error pattern E is defined as the difference between Received vector and Code vector. The receiver does the decoding operation by determining an (n-k) vector S. This vector S is called ‘error syndrome’ of Received Vector.

k- information digits digits

n-encoded digits

Rate, *R=k/n*

**Block Coder**

n- digit Codeword

**Information digits**

**Parity**

**digits**

n-k

k

**Algorithm/Flowchart**

**Inputs**

**Generator Matrix and Parity Check Matrix**

**Code Vectors using the Message Vectors**

**Calculation of Syndrome S**

**Correction of Error Bit**

**Outputs**

1. The Parity Matrix P and the values of n and k are entered as user input.
2. The Received vector R is entered.
3. The (k x k) Identity Matrix I is generated by using the function eye(k).
4. The Generator Matrix G is generated by combining Identity Matrix and Parity Matrix.
5. The Parity Check Matrix H is generated by combining the transpose of Parity Matrix (stored in Matrix L) and the Identity Matrix.
6. The message vector U denoting all information sequences is generated which has 2^k messages by iterating through a unit - spaced vector.
7. The code vector CV is generated by using rem function on message vector U and generator matrix G.
8. The transpose of Parity Check Matrix H is generated by using the transpose(H) function.
9. The syndrome S using the rem function on transpose of H and the received vector R.
10. The error bit is found by matching the value of calculated syndrome in each row of H transpose matrix. The index of the matching row is same as that of the index of the error bit in received vector from the MSB.
11. The bit which has an error and the corrected codeword is displayed.

**Program with Comments**

*[This code has been simulated in MATLAB R2020b]*

% Enter Parity Matrix

P = input("Enter the P matrix: ");

% Enter the values of n and k

n = input("Enter the value of n: ");

k = input("Enter the value of k: ");

% Enter Received Vector

r = input("Enter the r vector: ");

r

% Generating I Matrix of size k\*k

I = eye(k);

% Generating G Matrix

G = [I P]

% Generating H Matrix

L = P';

% Generating I Matrix of size (n-k)

I1 = eye(n-k)

H=[L I1]

% Generate U data vector, denoting all information sequences

% Number of message vectors present

no = 2 ^ k;

% Iterate through an Unit-Spaced Vector

for i = 1 : 2^k

% Iterate through Vector with Specified Increment

for j = k : -1 : 1

if rem(i - 1, 2 ^ (-j + k + 1)) >= 2 ^ (-j + k)

u(i, j) = 1;

else

u(i, j) = 0;

end

end

end

u

% Generate CodeVectors

cv = rem(u \* G, 2)

%Find Syndrome

Ht = transpose(H)

s = rem(r \* Ht, 2)

for i = 1 : 1 : size(Ht)

if(Ht(i,1:3)==s)

r(i) = 1-r(i);

break;

end

end

disp('The Error is in bit:')

disp(i)

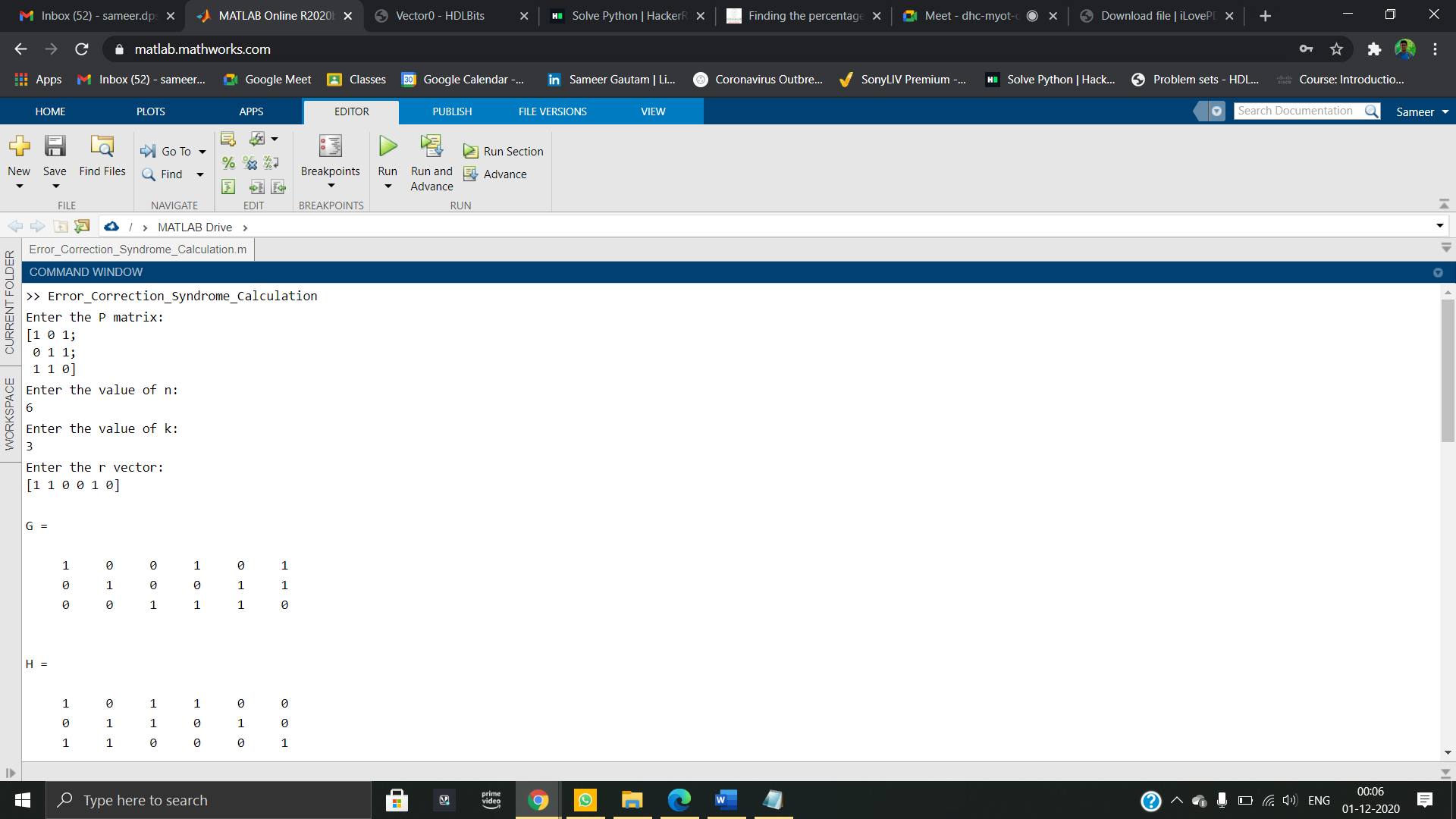
disp('The Corrected Codeword is :')

disp(r)

**MATLAB Execution Output**

(with execution snapshots)

Error\_Correction\_Syndrome\_Calculation

Enter the P matrix:

[1 0 1;

0 1 1;

1 1 0]

Enter the value of n:

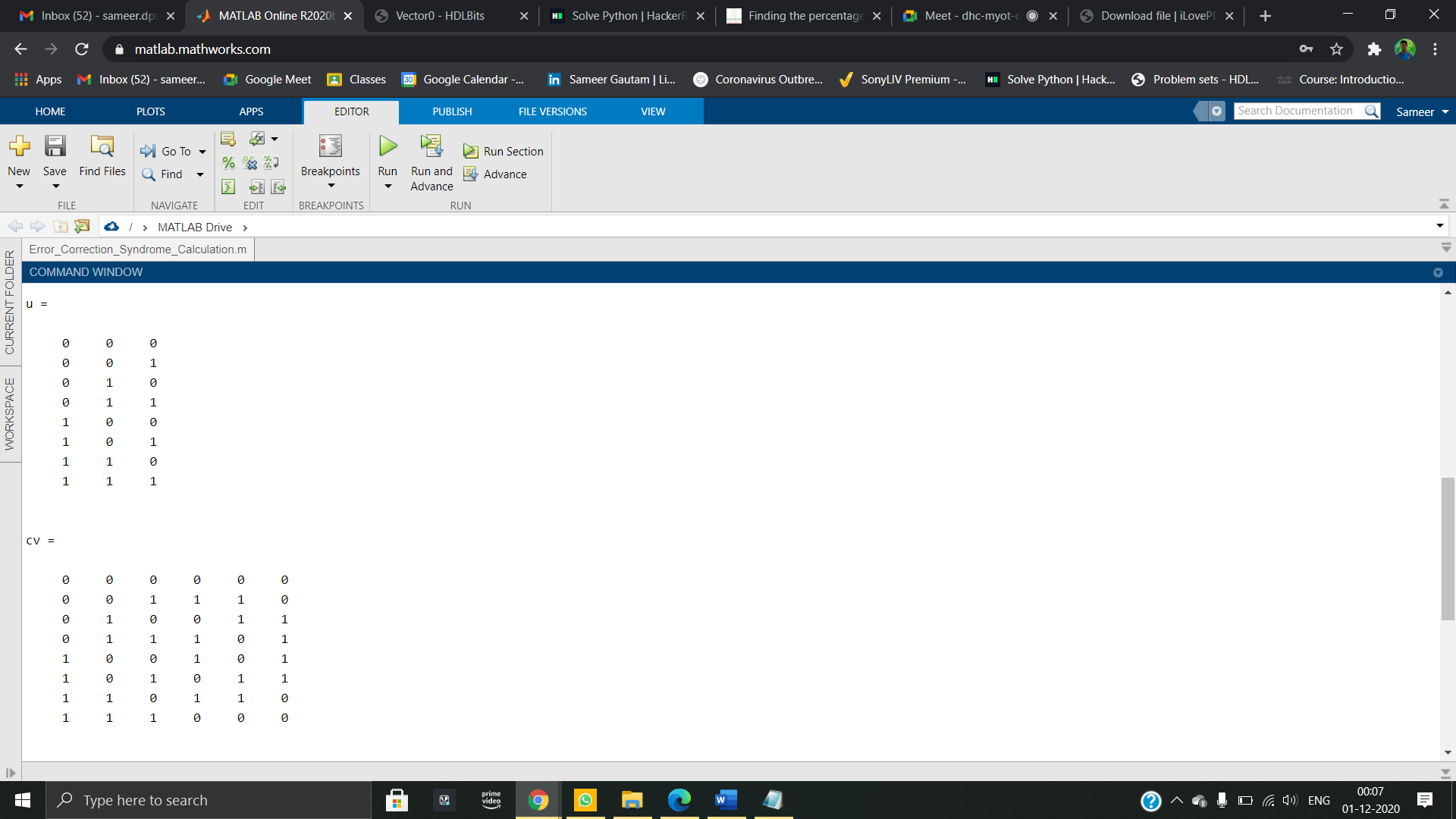
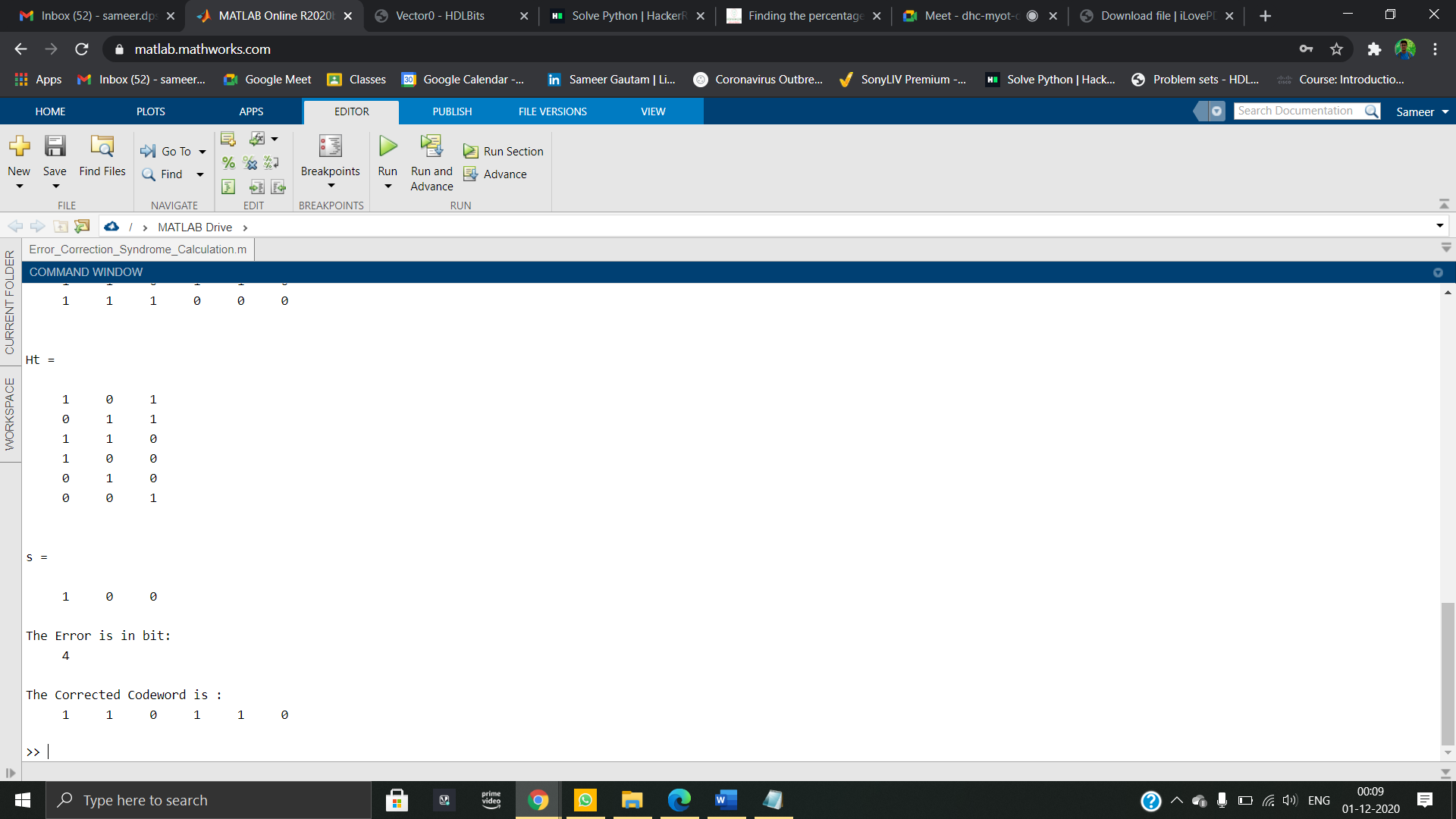
6

Enter the value of k:

3

Enter the r vector:

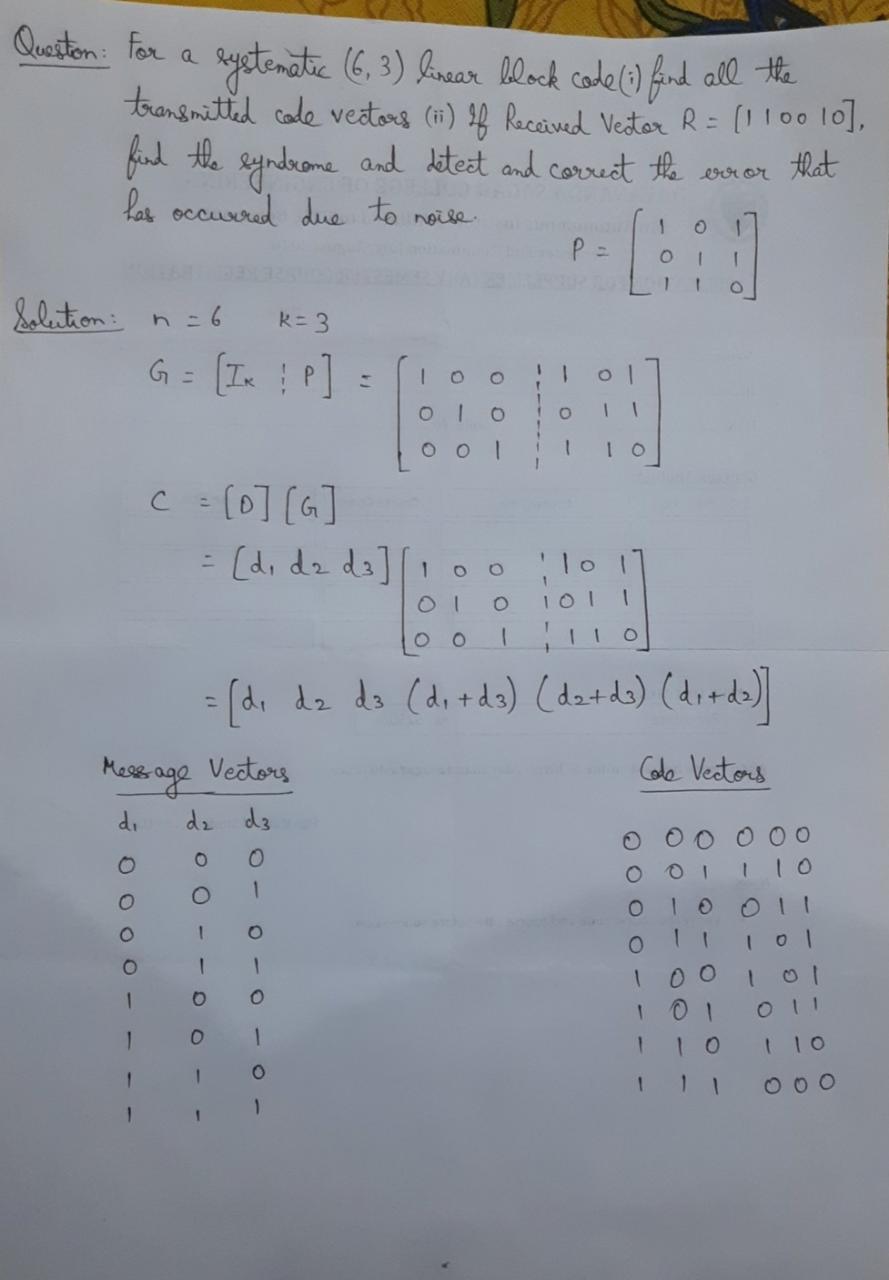
[1 1 0 0 1 0]

  
G =  
  
 1 0 0 1 0 1  
 0 1 0 0 1 1  
 0 0 1 1 1 0  
  
  
H =  
  
 1 0 1 1 0 0  
 0 1 1 0 1 0  
 1 1 0 0 0 1  
  
  
u =  
  
 0 0 0  
 0 0 1  
 0 1 0  
 0 1 1  
 1 0 0  
 1 0 1  
 1 1 0  
 1 1 1  
  
  
cv =  
  
 0 0 0 0 0 0  
 0 0 1 1 1 0  
 0 1 0 0 1 1  
 0 1 1 1 0 1  
 1 0 0 1 0 1  
 1 0 1 0 1 1  
 1 1 0 1 1 0  
 1 1 1 0 0 0  
  
  
Ht =  
  
 1 0 1  
 0 1 1  
 1 1 0  
 1 0 0  
 0 1 0  
 0 0 1

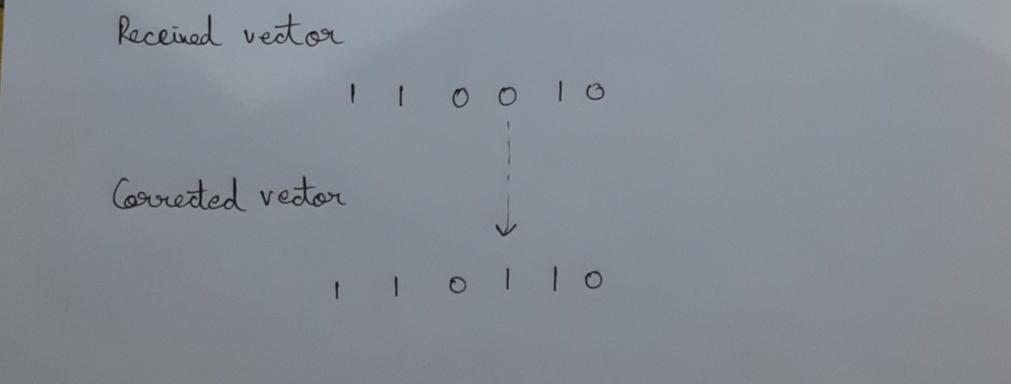
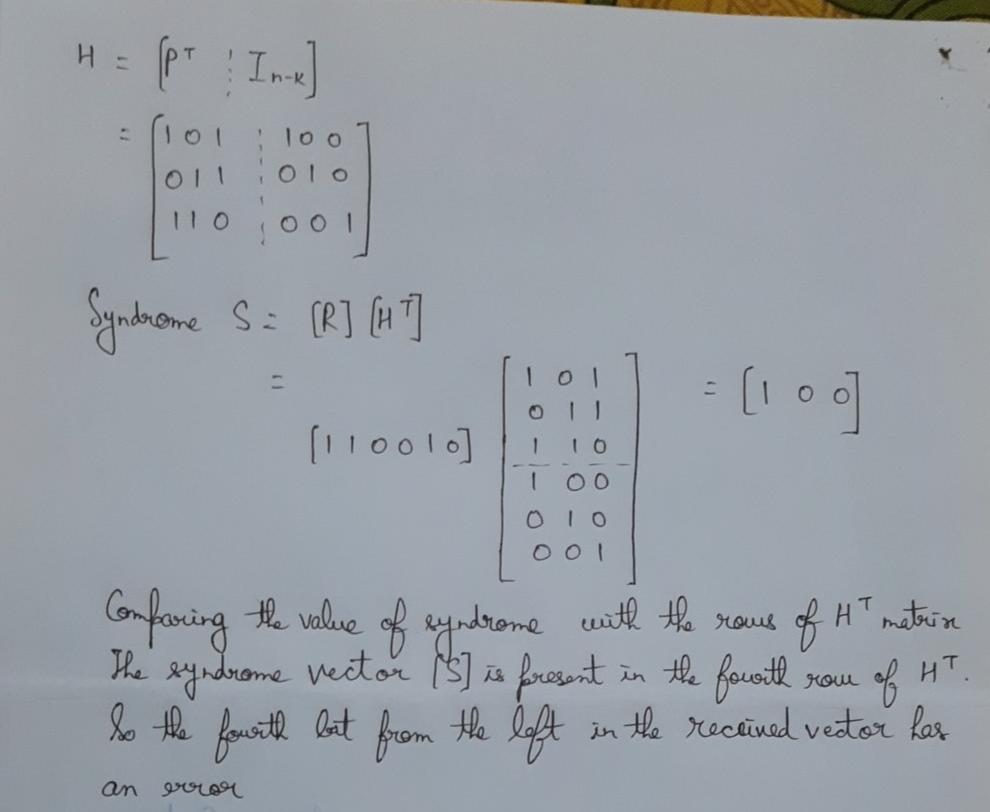
s =  
  
 1 0 0  
  
The Error is in bit:  
 4  
  
The Corrected Codeword is :  
 1 1 0 1 1

**Verification of Results**

1.



2.



**Reference**

[1]. <https://www.tutorialspoint.com/digital_communication/digital_communication_error_control_coding.htm>

[2]. <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-02-introduction-to-eecs-ii-digital-communication-systems-fall-2012/lecture-slides/MIT6_02F12_lec05.pdf>

[3]. Digital Communication System 5th Semester class notes.