Ex: 5

PRACTICAL - 6

AIM:-

To write a program to implement error detection and correction using HAMMING code concept. Make a test run to input data stream and verify error correction feature.

Error Correction at Data Link Layer:-

Hamming code is a set of error-correction codes that can be used to detect and correct the errors that can occur when the data is transmitted from the sender to the receiver. It is a technique developed by R.W. Hamming for error detection correction

Create a receiver program with below features.

1. Receiver program should read the input from channel file.
2. Apply hamming code on the binary data to check for errors

3. If there is an error, display the position of the error.

4. Else remove the redundant bits and convert the binary data to ascii and display the output.

Student Observation :- Receiver Program :-

- Apply hamming code on the binary data to check for errors.
- If there is any error, display the position of the error.

```
def hamming-check (hamming code):-
    n = len (hamming code)
    r = 0
    while (2**r) < n+1 :-
        r += 1
    syn = 0
    parity = [ ]
    for i in range (r):
        parity-pos = 2**i
        parity-val = 0
        for k in range (1, n+1):
            if k & parity-pos:
```

```
        parity_val ^= int(hamming_code[-i])
    parity.append(parity_val)
    syn |= (parity_val << i)
syndbits = ''.join(str(x) for x in reversed(parity))
return syndbits, syn

code = input("Enter received hamming code: ")
res, error = hamming_check(code)
print('Error bits:', res)
if error == 0:
    print('No error detected')
else:
    print('Error detected at bit position:', error)
```

STUDENT OBSERVATION:

→ INPUT:- & OUTPUT:-

Enter binary data: 1001101
Hamming code = 10011100101

Enter received Hamming code: 10010100101
Error syndrome bits: 0111
Error detected at bit position: 7

[- k])

Enter received Hamming code : 10011100101
Error syndrome bits : 0000
No error detected.

Result:-

Sender and receiver program for hamming code concept was executed and got the output.

18/9/20