EAST WEST UNIVERSITY

EAST WEST UNIVERSITY

Department of Computer Science & Engineering

Course Title: Computer Architecture

Course code: CSE360

Semester: spring 2021

Section: 02

Project Report

Title: Analysis and Design of Collision Detection and Avoidance Algorithm CSMA/CD for Bus Architecture

Submitted by

Name: Md Fahim Faez Abir
Id: 2018-1-60-032
Name: Kamrul Hassan
Id: 2018-1-60-039

Department: Computer Science & Engineering

Submitted to

Dr. Ahmed Wasif Reza

Associate Professor,

Department of Computer Science & Engineering

East West University

Date of Submission: 5.23.2021

Executive summary: This report provides an analysis and describe that how the algorithm Carrier Sense Multiple Access with Collision Detection (CSMA/CD) detect collision and avoid the collision in Bus topology. Methods of analysis include limitation of Carrier Sense Multiple Access with Collision Detection (CSMA/CD), algorithm and its implementation using programming language discussed in methodology, necessary calculation of the algorithm and the calculation found in appendices. Results and algorithm analysis show that the algorithm perfectly works for almost every test case in this report except for some limitations.

Introduction: A local area network (LAN) is a collection of computers connected by physical wires. You should focus on the position of other machines within the network rather than the actual distance between them [1].

For designing a local area network there are different types of Topology. Topology is a structure that connects multiple devices to a network in an efficient way. There are different types of topology in computer networks.

1. Bus topology
2. Ring topology
3. Star topology
4. Mesh topology

In this report we are going to focus on Bus topology and its collision detection and avoidance method.

The bus topology is set up in such a way that all of the stations are linked together by a single backbone cable. If a station wants to transmit data, if no data is detected in the bus the station can send data easily. If two stations transmit data in the same transmission time, a collision will occur and the station will start transmitting data again. Collision is detected using Carrier Sense Multiple Access with Collision Detection (CSMA/CD) [1][2]. CSMA/CD prevents the collisions on bus networks. If there is any collision CSMA/CD detects the collision and sends a signal to another station and after a random time wait the station starts transmitting data again. CSMA/CD has three different types of Approach [3].

1. 0-persistent
2. 1-persistent
3. P-persistent

In this report we are going to discuss the P-persistent approach to implement CSMA/CD. P-persistent is an approach between 1-persistent and 0-persistent in CSMA/CD [3].

Methodology: For executing our project we are flowing Carrier Sense Multiple Access with Collision Detection (CSMA/CD) algorithm.

- ➢ Description of algorithm (CSMA/CD):
  - Step 1: Initially k = 1.

- Step2: If any station is willing to transmit data, call p-persistent and check if the station is busy or free. If the station is busy, it will check again and again. If the station is free, transmission will be started.
- Step 3: Check whether collisions occur or not. If there is no collision, transmission will succeed. Otherwise, send a jamming signal.
- Step 4: If k is greater than kmax transmission will be aborted. Otherwise, follow the next step.
- Step 5: Update transmission time.
- Step 6: Flow the step 2.

➢ Flowchart (CSMA/CD):

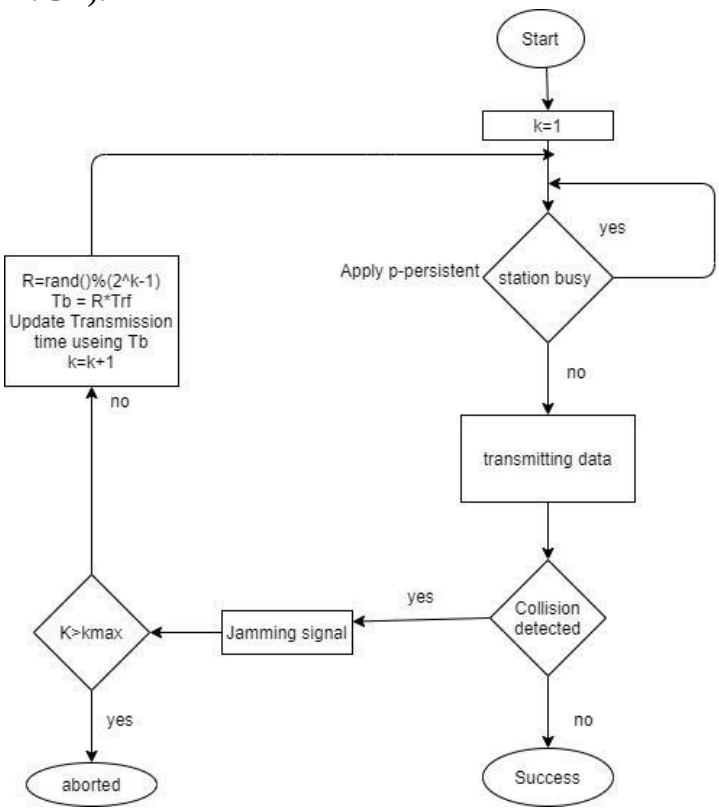


Fig: Flow-Chart (CSMA/CD)

We used the programming language C++ to implement the algorithm. And the tool used to write code is Code::Blocks 20.03.

Analysis: In this project we used Carrier Sense Multiple Access with Collision Detection (CSMA/CD) to detect collision in bus architecture. After detecting collisions this code will automatically try to avoid collision using a back-off process. In the back-off process we update the starting time of transmission time using back-off time [4].

back-off time, Tb = R * Tfr

here, Tfr = average transmitting time of a frame.

In this project p-persistent check whether the station is busy or not. It checks the station until it detects the station will be free. When it's got free then check a random value r with a probability value 0.50. Here r is between 0 to 1. When r < 0.50 only then the data will be loaded in the channel. We use five stations as five channels. In this code data passing happens from one station to the next station. There is no intermediate channel between the two stations.

Results: Some Sample input and output for our project

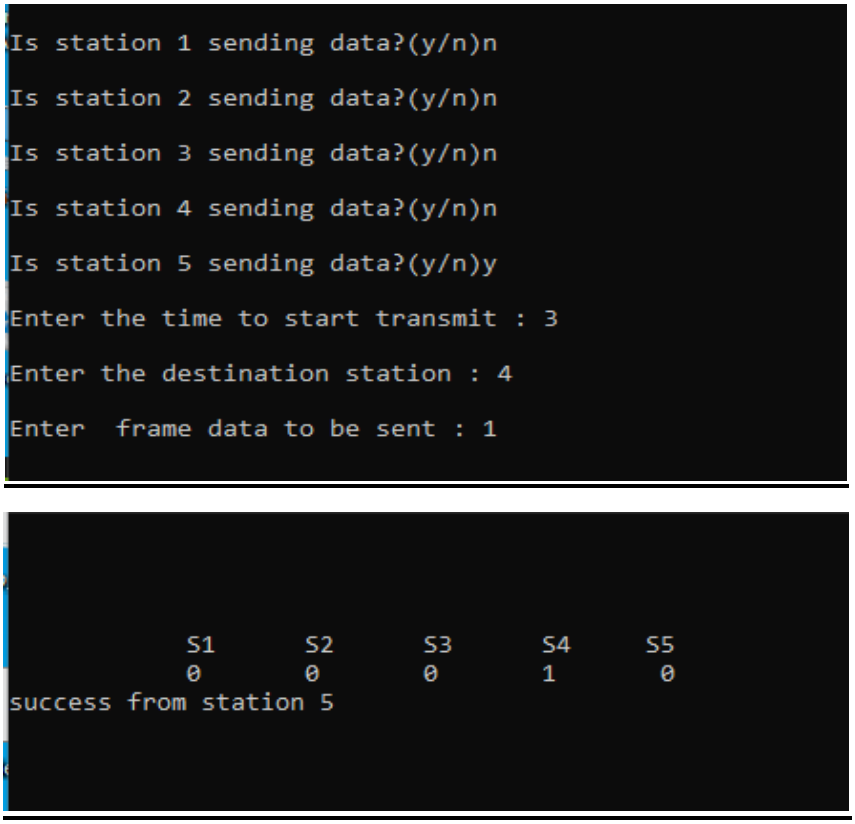
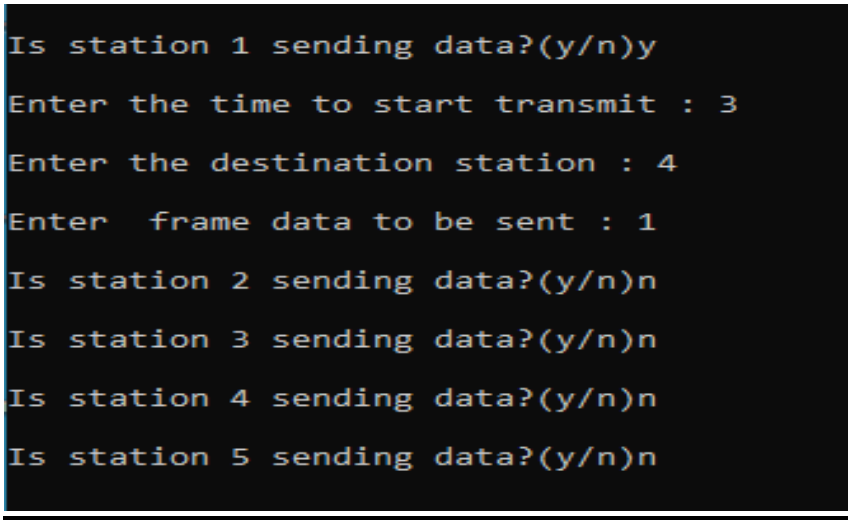


Fig: Test case 1



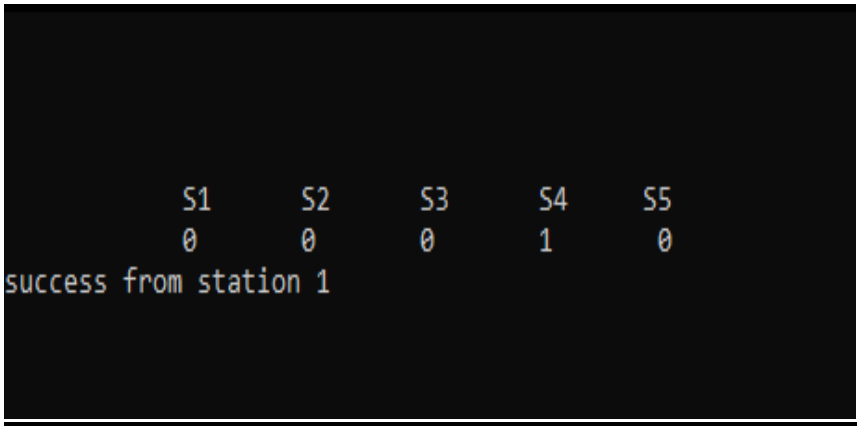
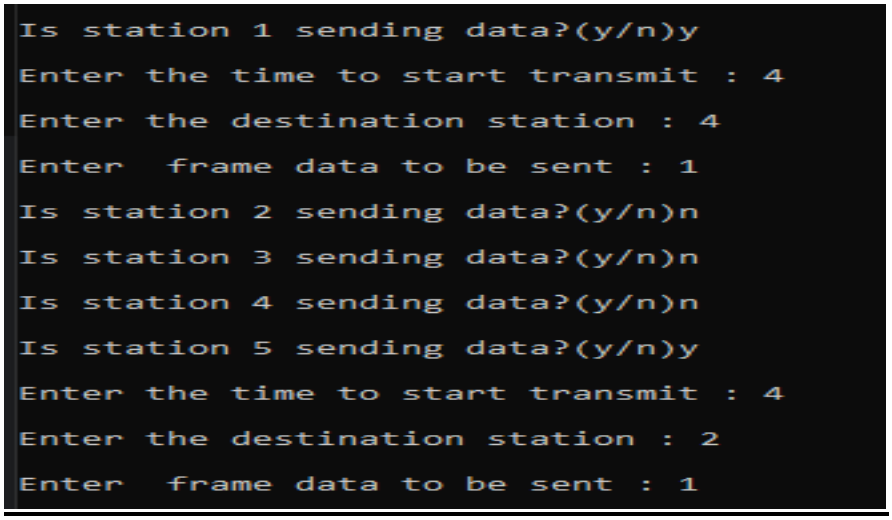
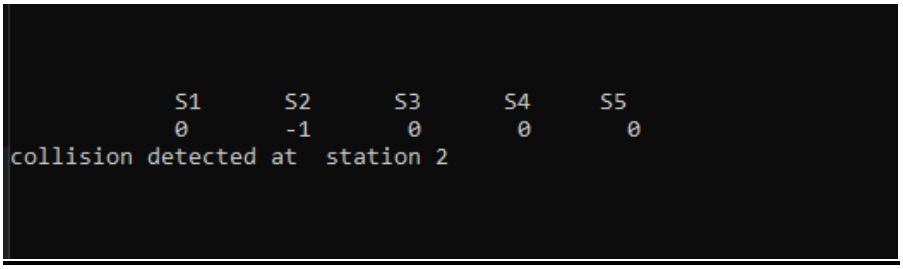
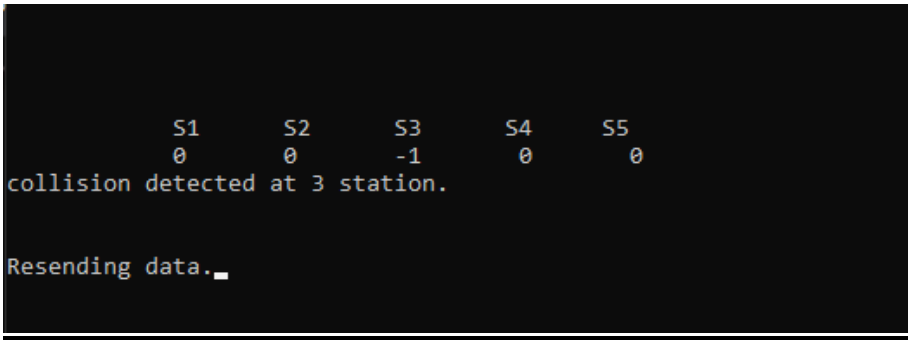


Fig: Test Case 2







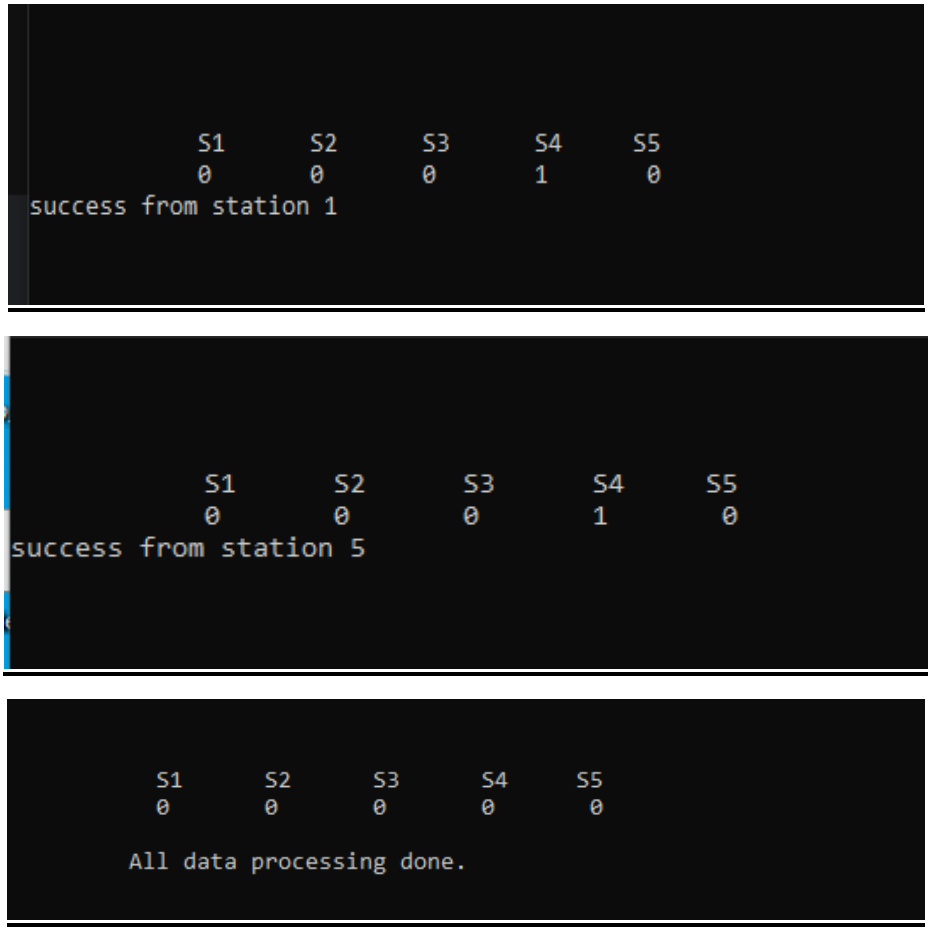


Fig: Test Case 3

Limitations:

There is a limitation for avoiding collision. Sometimes it takes too much time for resending data again as this algorithm updates transmission time with a random value. It's only worked for bus architecture. In wireless systems it cannot be used.

Recommendations: In future we can improve the code for better performance. After developing more steps, it can be used in real life bus architecture for data passing.

Conclusions: Collision is a big issue in bus architecture. In this project we are trying to solve the problem using Carrier Sense Multiple Access with Collision Detection (CSMA/CD). CSMA/Cd detect the collision and help the sender to avoid the collision. In this project we implement the CSMA/CD algorithm practically using code and we learn how CSMA/CD work in bus Architecture.

Appendices:

```
62  int p_persistence(int i)
63  {
64      if(channel[i]!=0)
65      {
66          return 0;
67      }
68      else
69      {
70          float p=0.5;
71
72          float r=float(rand()%11)/10.0;
73
74          if(r<=p)
75          {
76              ii[i]=1;
77
78              return 1;
79          }
80          else
81          {
82
83              return 0;
84          }
85      }
86  }
```

Fig: p-persistent Function

In this Function check whether the station is busy or not

```
87  void back_off(int st_no)
88  {
89      pp[st_no]=0;
90      if(k>kmax)
91      {
92          cout<<"Transmission from "<<st_no<<" is aborted."<<endl;
93          mwait();
94          tt[st_no]=0;
95          ns1++;
96      }
97      else
98      {
99          int s=pow(2,k)-1;
100         int r=(rand()%s);
101         tt[st_no]+=r*t_fr;
102         k++;
103
104         ii[st_no]=0;
105         l[st_no]=0;
106     }
107     mark1 = 0;
108 }
```

Fig: Back-off Function

This function updates the transmission time

References:

[1]"Fundamentals of communication and networking: Network topology - Wikibooks, open books for an open world", *En.wikibooks.org*, 2021. [Online]. Available: https://en.wikibooks.org/wiki/A-level_Computing/AQA/Paper_2/Fundamentals_of_communication_and_networking/Network_topology#:~:text=Bus%20Networks,-Bus%20layout&text=If%20two%20or%20more%20computers,again%20at%20a%20different%20time. [Accessed: 23- May- 2021].

[2]"Computer Network Topologies - javatpoint", *www.javatpoint.com*, 2021. [Online]. Available: https://www.javatpoint.com/computer-network-topologies#:~:text=What%20is%20Topology%3F,the%20nodes%20in%20a%20network. [Accessed: 23- May- 2021].

[3]"Carrier-sense multiple access - Wikipedia", *En.wikipedia.org*, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Carrier-sense_multiple_access#:~:text=To%20improve%20performance%2C%20engineers%20developed,%2C%20and%20slotted%20p%2Dpersistence.&text=CSMA%2FCD%20is%20used%20to,a%20retry%20can%20be%20attempted. [Accessed: 23- May- 2021].

[4]2021.[Online].Available: https://www.researchgate.net/figure/Flow-diagram-for-the-CSMA-CD_fig3_323511648. [Accessed: 23- May- 2021].