ST. ANNE'S

COLLEGE OF ENGINEERING AND TECHNOLOGY

(Approved by AICTE New Delhi, Affiliated to Anna University, Chennai)

(An ISO 9001:2015 Certified Institution)

ANGUCHETTYPALAYAM, PANRUTI - 607 106

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

EC8563 – COMPUTER NETWORKS LABORATORY

Regulation: 2017

Year / Semester: III / V

Jun 2019 - Nov 2019

PREPARED BY

Mrs. Z. ASMATHUNNISA, M.TECH.,
AP/CSE

EC8563 COMPUTER NETWORKS LABORATORY

LIST OF EXPERIMENTS

1. **Implementation of Error Detection / Error Correction Techniques**
2. **Implementation of Stop and Wait Protocol and sliding window**
3. **Implementation and study of Goback-N and selective repeat protocols**
4. **Implementation of High Level Data Link Control**
5. **Implementation of IP Commands such as ping, Traceroute, nslookup**
6. **Implementation of IP address configuration.**
7. **To create scenario and study the performance of network with CSMA / CA protocol and compare with CSMA/CD protocols.**
8. **Network Topology - Star, Bus, Ring**
9. **Implementation of distance vector routing algorithm**
10. **Implementation of Link state routing algorithm**
11. **Study of Network simulator (NS) and simulation of Congestion Control Algorithms using NS**
12. **Implementation of Encryption and Decryption Algorithms using any programming language**

Total :60 Periods

Table of Contents

Ex No: 1. Implementation of Error detection/Error correction techniques

AIM:

To implement and check the error detection/error correction techniques in networks using a c program.

APPARATUS REQUIRED:

1. Pc-ino
2. C/c++compiler

THEORY:

Error Detection

- Bit errors occur in frames due to electrical interference or thermal noise.
- Detecting errors is one part of the problem; correcting errors is the other.
- What happens when an error is detected?
- Two basic approaches:
  - Notify the sender that message is corrupt so the sender can retransmit it; (most often used in every day applications)
  - Use an error-correcting code to reconstruct the correct message

Transmission Errors

· External electromagnetic signals can cause incorrect delivery of data

- · Data can be received incorrectly
- · Data can be lost
- · Unwanted data can be generated

· Any of these problems are called *transmission errors*

Error Detection

- Detecting Transmission Errors: basic idea is to add redundant information to a frame that can determine if errors have been introduced.

Error Correction or Error Detection?

- When error is detected, frame is discarded and resent, using bandwidth and causing latency, waiting for its arrival.
- Error correction requires additional bit to be sent with every frame.
- Correction is useful when
  - 1) errors are probable or
  - 2) the cost of retransmission is too high

PROCEDURE:

- Start the process.
- Give the data which is the message.
- Compile and run the program.
- Enter the received hamming code.
- The error is corrected codeword.

PROGRAM FOR CODE GENERATION FOR ERROR DETECTION AND CORRECTION

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
Int main()
{
Int I,j,k,count,err-pos=0,flag=0;
Char dw[20],cw[20],data[20];
Printf("enter data as binary bit stream(7 bits):\n");
Scanf("%s",data);
For(i=1,j=0,k=0;i<12;i++)
{
If(i==(int)pow(2,j))
{
Dw[i]='?';
J++;
}
Else
{
Dw[i]=data[k];
K++;
}
}
For(i=0;j<4;i++)
{
Count=0;
For(j=(int)pow(2,i);j<12;j<12;j+=(int)pow(2,i))
{
For (k=0;k<(int)pow(2,i);k++)
{
If(dw[j]=='1')count++;j++;
}
}
If(count%2==0)
Dw[(int)pow(2,i)]='0';
else
Dw[(int)pow(2,i)]='1';
}
Printf("in code word is\n\n");
For(i=1;i<12;i++)
Printf("%c",dw[i]);
Printf("\n\n enter the received hamming code\n\n");
Scanf("%s",cw);
For(i=12;i>0;i--)
Cw[i]=cw[i-1];
For(i=0;i<4;i++)
{
Count=0;
For(j=(int)pow(2,i);j<12;j+=(int)pow(2,i))
{
```

```
For(k=0;k<(int)pow(2,i);k++)
{
If(cw[j]==’1’)count++;j++;
}
}
If (count%2!=0)
Err-pos=err-post+(int)pow(2,i);
}
If(err-pos==0)
printf(“\n\n there is no error in received code word \n”);
else
{
If(cw[err-pos]==dw[err-pos])
{
Printf(“\n\n there are 2 or more errors in received code……\n\n”);
Printf(“sorry…! Hamming code cannot correct 2 or more errors….\n”);
Flag=1;
}
Else
Printf(“in there is an error in bit position %d of received code word \n”,err-pos);
If(flag==0)
{
Cw[err-pos]=(cw[err-pos]==’1’)?’0’:’1’;
printf(“\n\n corrected code word is \n\n”);
for(i=1;i<12;i++)
printf(“%c”,cw[i]);
}
}
Printf(“\n\n”);
}
```

OUTPUT:

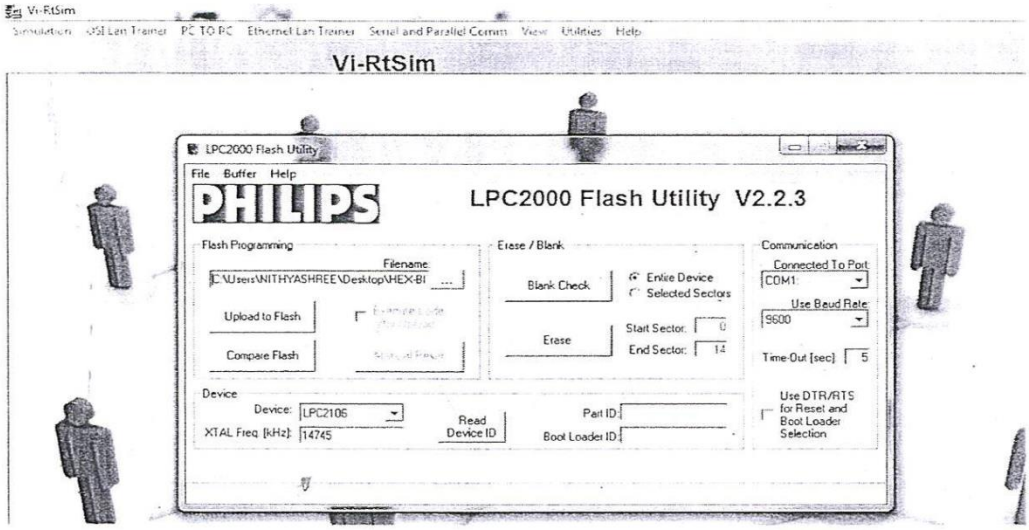
```
Enter data as binary bit stream(7 bits):
1110110
Code word is
11101100110
Enter the received hamming code
10101100110
There is an error in bit position 2 of received code word corrected code word is
11101100110
Enter data as binary bit stream(7 bits)
11101110
Code word is
11101100110
Enter the received hamming code
00101100110
There are 2 or more error in received code…
Sorry…!
```

RESULT:

Thus the error detection/error correction techniques were implemented successfully.

IMPLEMENTATION AND STUDY OF STOP AND WAIT PROTOCOL

MENU BAR



Ex No: 2. Implementation and Study of Stop and Wait Protocoland sliding window protocol

AIM:

To implement and Study the performance of Stopand Wait Protocol using LAN trainer.

APPARATUS REQUIRED:

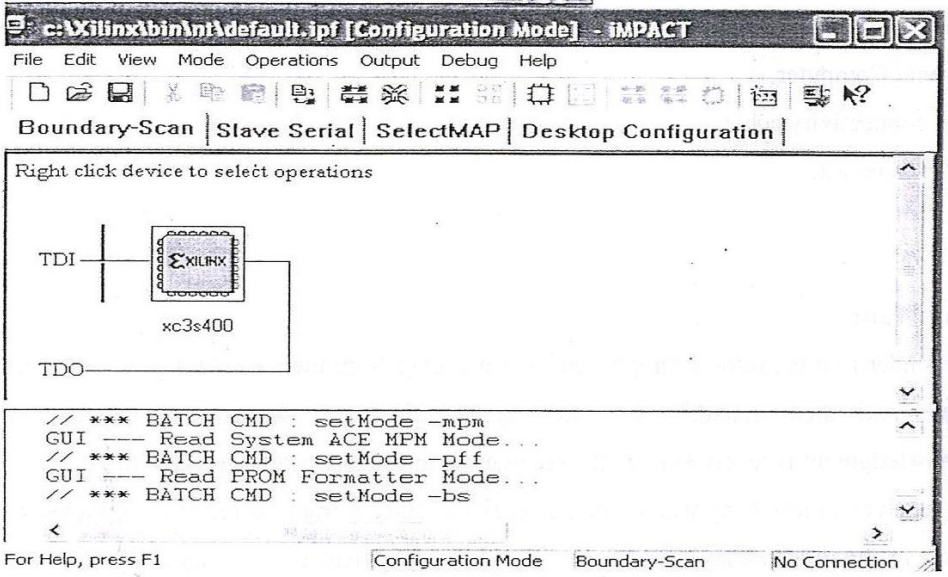
1. VI-RTSIM software.
2. Personal Computer.
3. LAN connectivity cable.
4. LAN trainer kit.

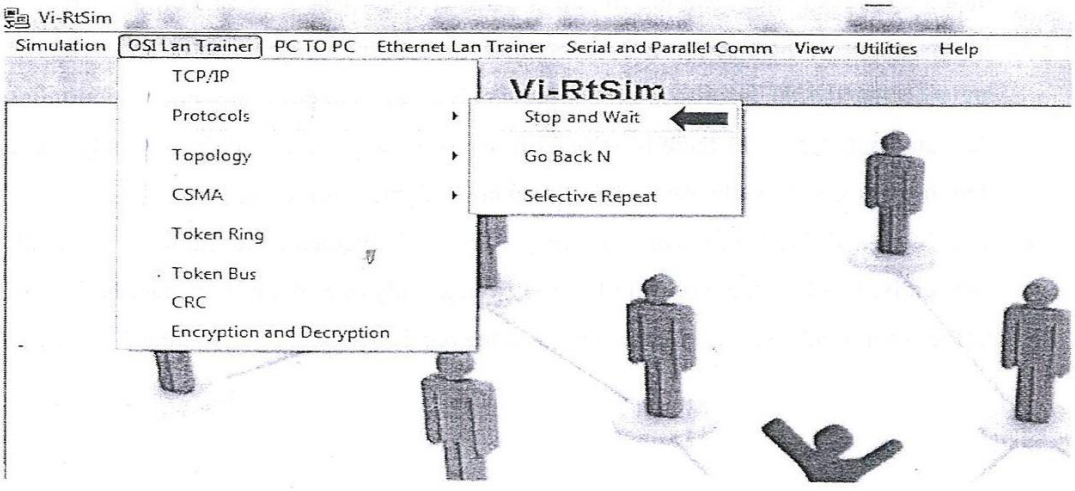
THEORY:

Stop and Wait:

- The sender in this protocol simply retrieves a packet from the network layer, copies it into a frame, and then transmits it. After transmission, the sender busy waits until an acknowledgment is received from the receiver, then the loop starts over again.
- The receiver simply busy waits a frame received. Once a frame is received it passes the data packet to the network layer and sends an acknowledgment for the frame it just received. It then loops back to busy waiting and the process continues until the End of File is reached.
- In this protocol, there can only be one outstanding frame at a time so no sequence numbers are required and the acknowledgment the receiver sends back to the sender in nothing more than an empty frame, as there is no other possibility then acknowledging the only frame sent. Another frame will not be sent until this acknowledgment is received.
- The Stop and Wait protocol was very easy to implement and runs very quickly and efficiently. It solves the problem of congestion, as only one frame is outstanding at any time, frames cannot be lost due to congestion and the receiver will not be swamped by the sender


XILINX®
DESIGN SOLUTIONS
iMPACT
Release Version: 7.1.02i
Application Version: H.40
Registration ID: UNKNOWN
Copyright (c) 1995-2005 Xilinx, Inc.
All rights reserved.


c:\Xilinx\bin\nt\default.ipf [Configuration Mode] - iMPACT
File Edit View Mode Operations Output Debug Help
Boundary-Scan Slave Serial SelectMAP Desktop Configuration
Right click device to select operations
TDI
XILINX
xc3s400
TDO
// *** BATCH CMD : setMode -mpm
GUI --- Read System ACE MPM Mode...
// *** BATCH CMD : setMode -pff
GUI --- Read PROM Formatter Mode...
// *** BATCH CMD : setMode -bs
For Help, press F1
Configuration Mode
Boundary-Scan
No Connection


Vi-RtSim
Simulation
OSI Lan Trainer
PC TO PC
Ethernet Lan Trainer
Serial and Parallel Comm
View
Utilities
Help
TCP/IP
Protocols
Topology
CSMA
Token Ring
Token Bus
CRC
Encryption and Decryption
Vi-RtSim
Stop and Wait
Go Back N
Selective Repeat

PROCEDURE

Downloading "ARM"

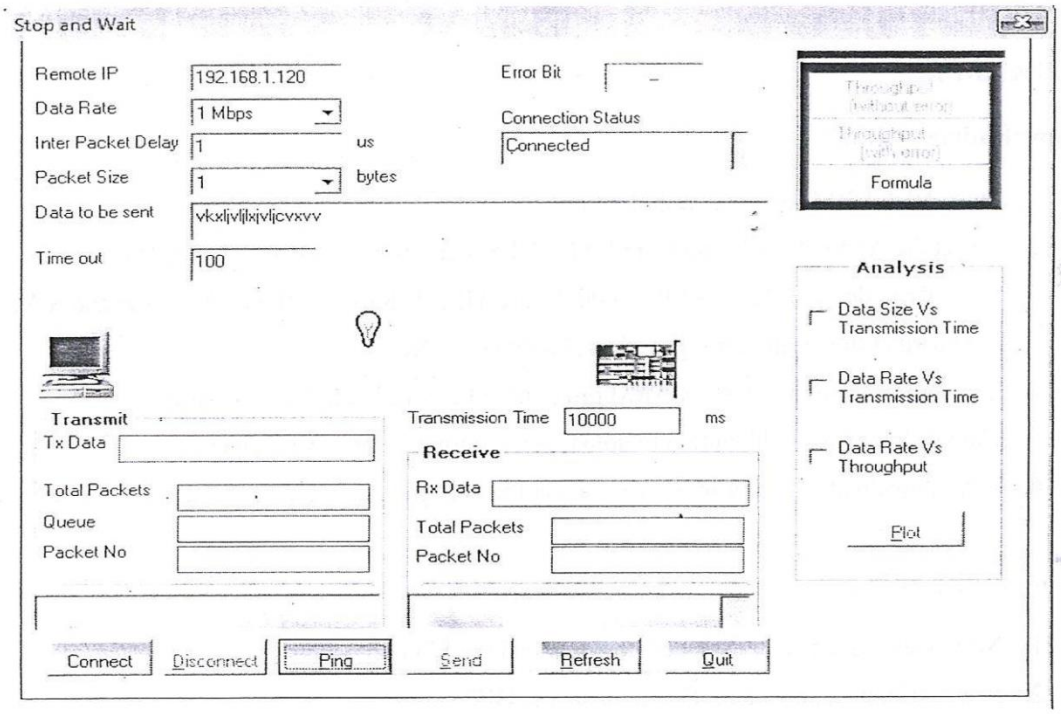
1. Open VI-RTSIM software from desktop
2. Select the ARM Downloader from UTILITIES menu bar to download the ARM.
3. Click Read device ID in the LPC 2000 flash UTILITIES to read the LAN trainer and reset the board at the programming mode and then click OK.
4. Select the TOKENBUS.HEX ARM program to be downloaded to the trainer kit.
5. Click "Upload to flash" button to upload the program to the LAN trainer.
6. After downloading the program we can get the "Program uploaded successfully" message.

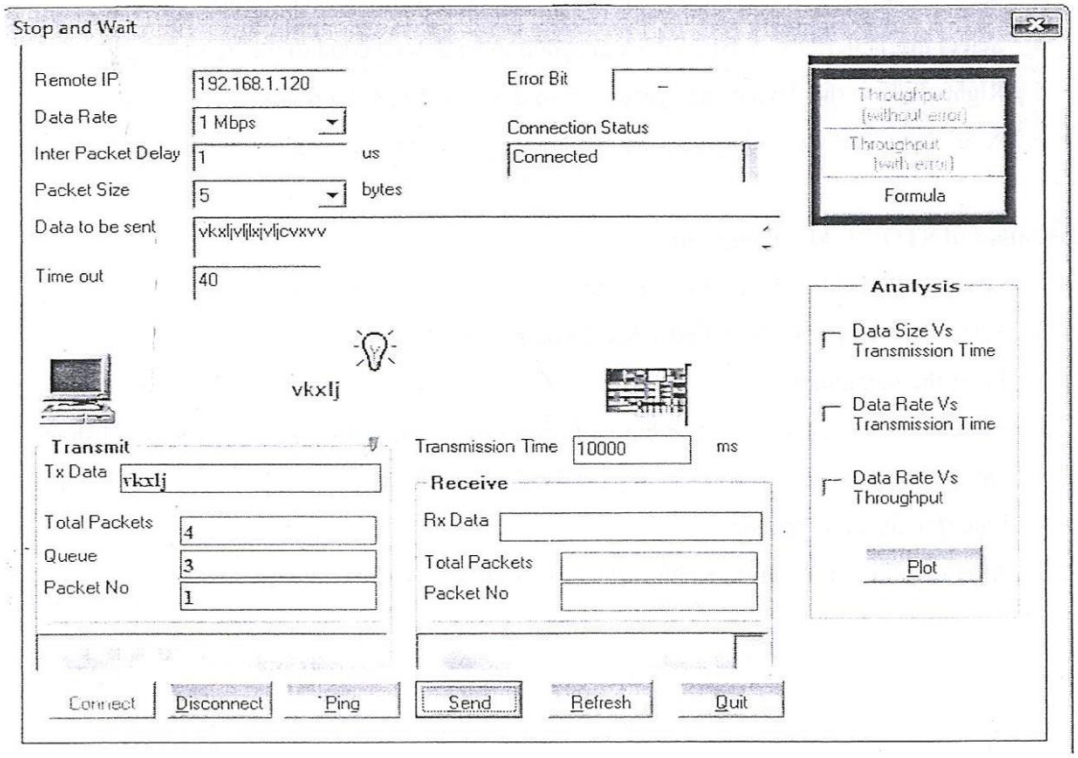
Downloading "FPGA"

1. Now open the software "Impact" to download the FPGA.]
2. Select "Create a new project" and clock on "OK".
3. Select "configure devices using boundary scan (J-TAG) and click "finish" button.
4. Select the TOKENBUS BIT FPGA program to be downloaded to the trainer kit.
5. Right click on the device and "program" to download FPGA program.
6. After downloading the program we can get the "program successfully" message.

Execution of STOPWAIT program:

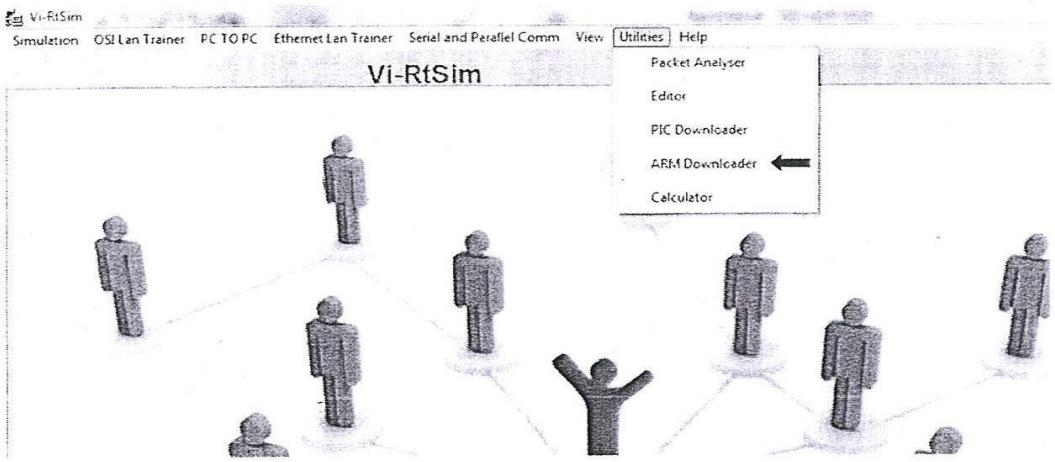
1. Open VI-RTSIM software from desktop.
2. Select Stop and Wait from OSI LAN Trainer menu bar.
3. Enter the Parameter.
4. At default setting press connect button to conform whether the connection is established or not. After connection is successfully established press start button.
5. Enter the data on the field.
6. After data field is entered, Sender waits for ACK after each frame transmission.

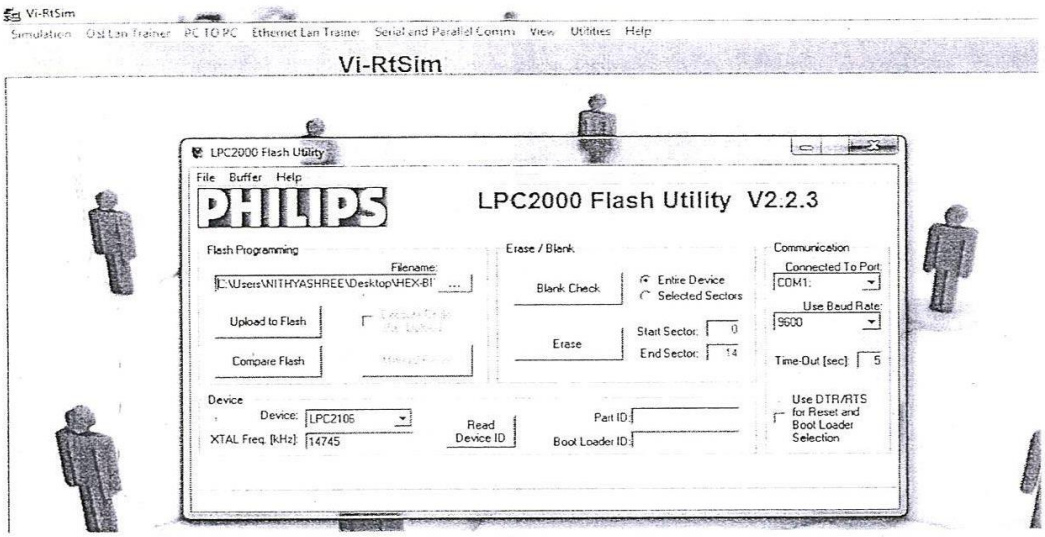

Stop and Wait
Remote IP
192.168.1.120
Error Bit
Data Rate
1 Mbps
Connection Status
Inter Packet Delay
1
us
Connected
Packet Size
1
bytes
Data to be sent
vkxljvljlxjvljcvxvv
Time out
100
Formula
Analysis
Data Size Vs Transmission Time
Data Rate Vs Transmission Time
Data Rate Vs Throughput
Plot
Transmit
Transmission Time
10000
ms
Tx Data
Receive
Rx Data
Total Packets
Queue
Packet No
Total Packets
Packet No
Connect
Disconnect
Ping
Send
Refresh
Quit


Stop and Wait
Remote IP
192.168.1.120
Error Bit
Data Rate
1 Mbps
Connection Status
Inter Packet Delay
1
us
Connected
Packet Size
5
bytes
Data to be sent
vkxljvljlxjvljcvxvv
Time out
40
Formula
Analysis
Data Size Vs Transmission Time
Data Rate Vs Transmission Time
Data Rate Vs Throughput
Plot
vkxlj
Transmit
Transmission Time
10000
ms
Tx Data
vkxlj
Receive
Rx Data
Total Packets
4
Queue
3
Packet No
1
Total Packets
Packet No
Connect
Disconnect
Ping
Send
Refresh
Quit

RESULT:

Thus the Stop and Wait protocol has been implemented and its performance is analyzed.


Vi-RtSim
Simulation
OSI Lan Trainer
PC TO PC
Ethernet Lan Trainer
Serial and Peralfiel Comm
View
Utilities
Help
Vi-RtSim
Packet Analyser
Editor
PIC Downloader
ARM Downloader
Calculator


Vi-RtSim
Simulation
OSI Lan Trainer
PC TO PC
Ethernet Lan Trainer
Serial and Peralfiel Comm
View
Utilities
Help
Vi-RtSim
LPC2000 Flash Utility
File
Buffer
Help
PHILIPS
LPC2000 Flash Utility V2.2.3
Flash Programming
Filename:
C:\Users\NITHYASHREE\Desktop\HEX-BI
Upload to Flash
Compare Flash
Erase / Blank
Blank Check
Entire Device
Selected Sectors
Erase
Start Sector:
End Sector:
Communication
Connected To Port:
COM1:
Use Baud Rate:
9600
Time-Out [sec]
Device
Device:
LPC2106
XTAL Freq. [kHz]:
14745
Read Device ID
Part ID:
Boot Loader ID:
Use DTR/RTS for Reset and Boot Loader Selection

Ex No: 3 a) Implementation and Study of GO BACK-N Protocol

AIM:

To implement and Study the performance of GO BACK-NProtocol using LAN trainer.

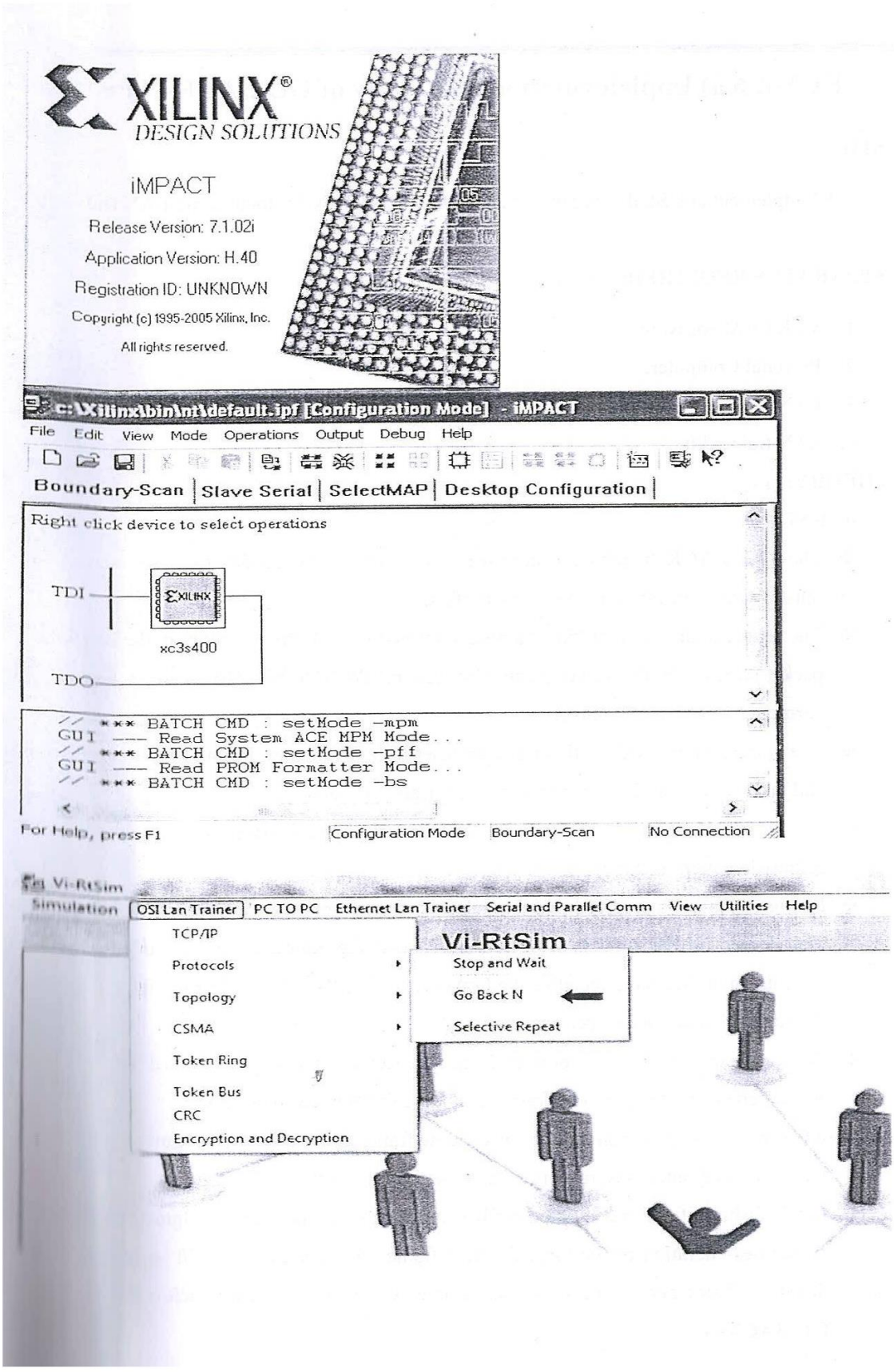
APPARATUS REQUIRED:

1. VI-RTSIM software.
2. Personal Computer.
3. LAN connectivity cable.
4. LAN trainer kit.

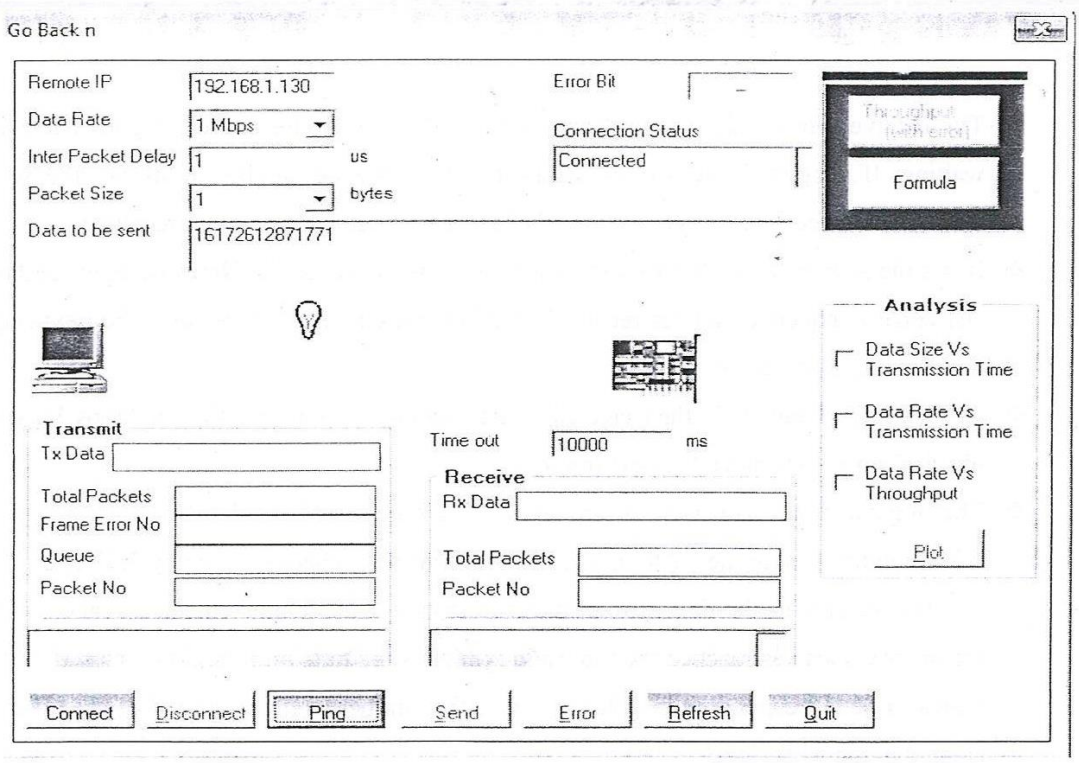
THEORY:

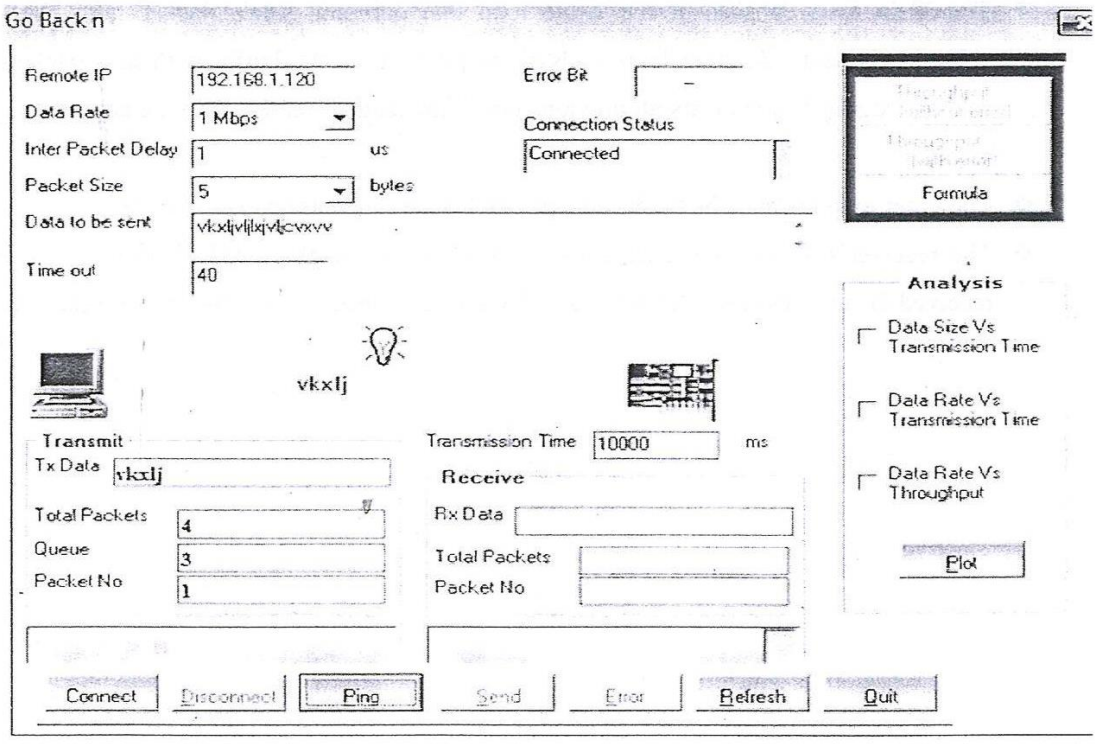
GO BACK-N.

- The **GO BACK-N** protocol improves by allowing the sender to have more than one outstanding frame at a time by using buffers.
- The sender maintains a buffer of a predetermined size. IF there is room in the buffer it gets a packet stores it in the correct empty slot (seq_nr%WINDOWSIZE) creates a frame with the correct seq_nr and transmits it.
- The corresponding logical timer is then reset to 0. The Upper Bound of the window is then slid up by circularly incrementing the next_frame_to_send.
- At this point or if no buffers are empty, the physical layer is checked to see if an acknowledgment is there.
- If a good frame is received and the acknowledgment number is within the current window then decrement the number of buffers used, reset the logical timer (to a negative value) to indicate an unused slot, and slide the Lower Bound of the window by circularly incrementing the acknowledgment number expected.
- The procedure runs is a loop until ack_expected equals ack_received (this clears the ack_received and the previous frames that haven't been acknoedged yet).
- After this, or if a bad frame or out of window frame arrives, the logical timers are updated. If frame is timed out, it's retransmitted and the timer is reset to 0.
- In this fashion if a frame is timed out because it was lost, no acknowledgment will arrive. So on the next iteration of the loop, the next frame will time out and will be resent. Thus the timed out frame and all the subsequent frames will be transmitted which is the definition of **GO BACK-N**


XILINX®
DESIGN SOLUTIONS
iMPACT
Release Version: 7.1.02i
Application Version: H.40
Registration ID: UNKNOWN
Copyright (c) 1995-2005 Xilinx, Inc.
All rights reserved.
c:\Xilinx\bin\nt\default.ipf [Configuration Mode] - iMPACT
File Edit View Mode Operations Output Debug Help
Boundary-Scan Slave Serial SelectMAP Desktop Configuration
Right click device to select operations
TDI
XILINX
xc3s400
TDO
// *** BATCH CMD : setMode -mpm
GUI --- Read System ACE MPM Mode...
// *** BATCH CMD : setMode -pff
GUI --- Read PROM Formatter Mode...
// *** BATCH CMD : setMode -bs
For Help, press F1
Configuration Mode
Boundary-Scan
No Connection
Vi-RtSim
Simulation
OSI Lan Trainer
PC TO PC
Ethernet Lan Trainer
Serial and Parallel Comm
View
Utilities
Help
TCP/IP
Protocols
Topology
CSMA
Token Ring
Token Bus
CRC
Encryption and Decryption
Vi-RtSim
Stop and Wait
Go Back N
Selective Repeat

- The receiver busy waits until a frame arrives. If a bad frame arrives, it goes back to busy waiting. If a good frame arrives, it checks the sequence number, if its not the sequence number expected it resends an ack for the last correct sequence number received.
- If it's the sequence number expected, it passes the packet to the Network layer, updates the last correct sequence number received variable and circularly increments the next sequence number expected variable.
- An acknowledgement is then created and transmitted for it and then it loops back to the physical layer to retrieve the next frame.
- The logical timer consists of an array of integers. When the value is negative the corresponding slot in the buffer is unused. Anytime after checking the physical layer a loop is run that increments the timers of all the used slots by one, it simultaneously checks if any of the timer values has reached the threshold (which is the timeout time).The value stored in the logical time corresponding to a buffer slot is the number of times the buffered packet has in created in the while loop.
- The Main loop consists of the while loop that gets and sends packets and checks for acknowledgments. Another loop loads all empty slots in the buffer with new packets. The acknowledgment loop clears all previous unacknowledged frames up to the acknowledgment received.
- The timer loop updates the timers and checks for the timeouts and retransmits.
- The receiver's Main Loop checks for frames received in the physical layer. If a frame is received then an acknowledgment is sent for the last correct in sequence frame received.


Go Back n
Remote IP
192.168.1.130
Error Bit
Data Rate
1 Mbps
Connection Status
Inter Packet Delay
1
us
Connected
Packet Size
1
bytes
Formula
Data to be sent
16172612871771
Analysis
Data Size Vs Transmission Time
Data Rate Vs Transmission Time
Data Rate Vs Throughput
Plot
Transmit
Tx Data
Time out
10000
ms
Receive
Rx Data
Total Packets
Frame Error No
Queue
Packet No
Total Packets
Packet No
Connect
Disconnect
Ping
Send
Error
Refresh
Quit


Go Back n
Remote IP
192.168.1.120
Error Bit
Data Rate
1 Mbps
Connection Status
Inter Packet Delay
1
us
Connected
Packet Size
5
bytes
Formula
Data to be sent
vkxljvljlxjvljcvxvv
Time out
40
Analysis
Data Size Vs Transmission Time
Data Rate Vs Transmission Time
Data Rate Vs Throughput
Plot
vkxlj
Transmit
Transmission Time
10000
ms
Tx Data
vkxlj
Receive
Total Packets
4
Rx Data
Queue
3
Total Packets
Packet No
1
Packet No
Connect
Disconnect
Ping
Send
Error
Refresh
Quit

PROCEDURE

Downloading "ARM"

1. Open VI-RTSIM software from desktop
2. Select the ARM Downloader from UTILITIES menu bar to download the ARM.
3. Click Read device ID in the LPC 2000 flash UTILITIES to read the LAN trainer and reset the board at the programming mode and then click OK.
4. Select the TOKENBUS.HEX ARM program to be downloaded to the trainer kit.
5. Click "Upload to flash" button to upload the program to the LAN trainer.
6. After downloading the program we can get the "Program uploaded successfully" message.

Downloading "FPGA"

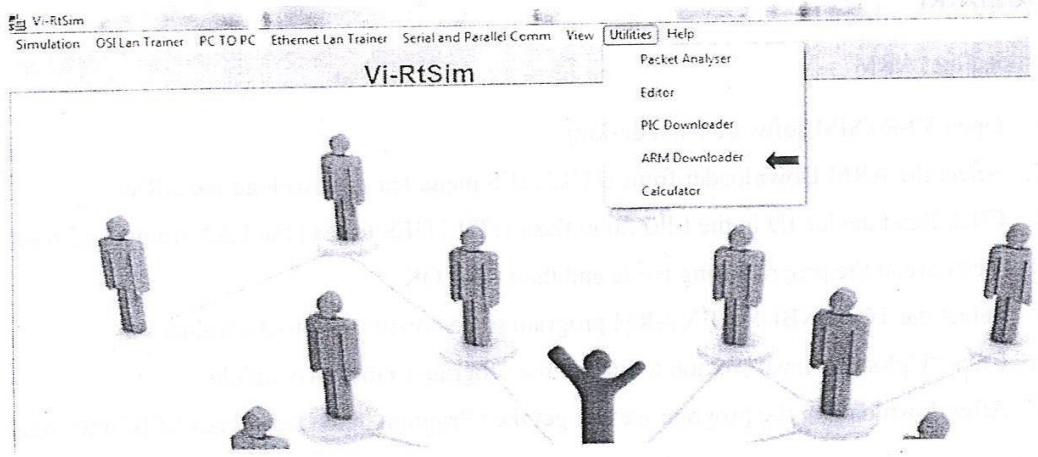
1. Now open the software "Impact" to download the FPGA.]
2. Select "Create a new project" and clock on "OK".
3. Select "configure devices using boundary scan (J-TAG) and click "finish" button.
4. Select the TOKENBUS BIT FPGA program to be downloaded to the trainer kit.
5. Right click on the device and "program" to download FPGA program.
6. After downloading the program we can get the "program successfully" message.

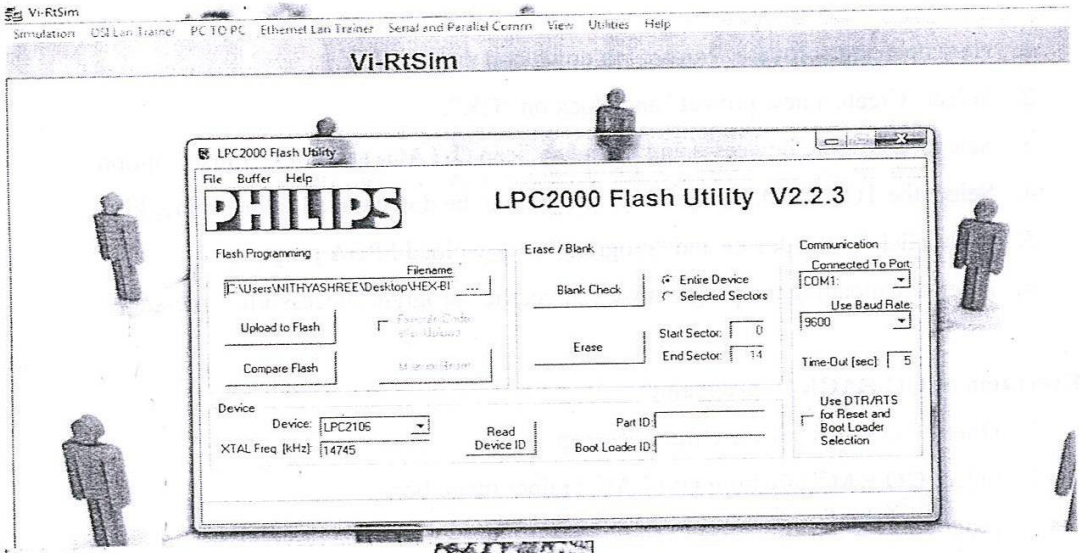
Execution of GO BACK-N program:

1. Open VI-RTSIM software from desktop.
2. Select **GO BACK-N** from OSI LAN Trainer menu bar.
3. Enter the Parameter.
4. At default setting press connect button to conform whether the connection is established or not. After connection is successfully established press start button.
5. Enter the data on data field, press Send button.
6. After the field is entered, Sender waits for ACK after each frame transmission.

RESULT:

Thus the Go back N protocol has been implemented and its performance is analyzed.


Vi-RtSim
Simulation
OSI Lan Trainer
PC TO PC
Ethernet Lan Trainer
Serial and Parallel Comm
View
Utilities
Help
Vi-RtSim
Packet Analyser
Editor
PIC Downloader
ARM Downloader
Calculator


Vi-RtSim
Vi-RtSim
LPC2000 Flash Utility
File
Buffer
Help
PHILIPS
LPC2000 Flash Utility V2.2.3
Flash Programming
Filename
C:\Users\NITHYASHREE\Desktop\HEX-BI
Upload to Flash
Compare Flash
Erase / Blank
Blank Check
Entire Device
Selected Sectors
Erase
Start Sector:
End Sector:
Communication
Connected To Port:
COM1:
Use Baud Rate
9600
Time-Out [sec]
5
Device
Device:
LPC2106
XTAL Freq [kHz]:
14745
Read Device ID
Part ID:
Boot Loader ID:
Use DTR/RTS for Reset and Boot Loader Selection


XILINX®
DESIGN SOLUTIONS
iMPACT
Release Version: 7.1.02i
Application Version: H.40
Registration ID: UNKNOWN
Copyright (c) 1995-2005 Xilinx, Inc.
All rights reserved.

Ex No:3 b) Implementation and study of Selective and Repeat Protocol

AIM:

To implement and Study the performance of Selective and RepeatProtocol using LAN trainer.

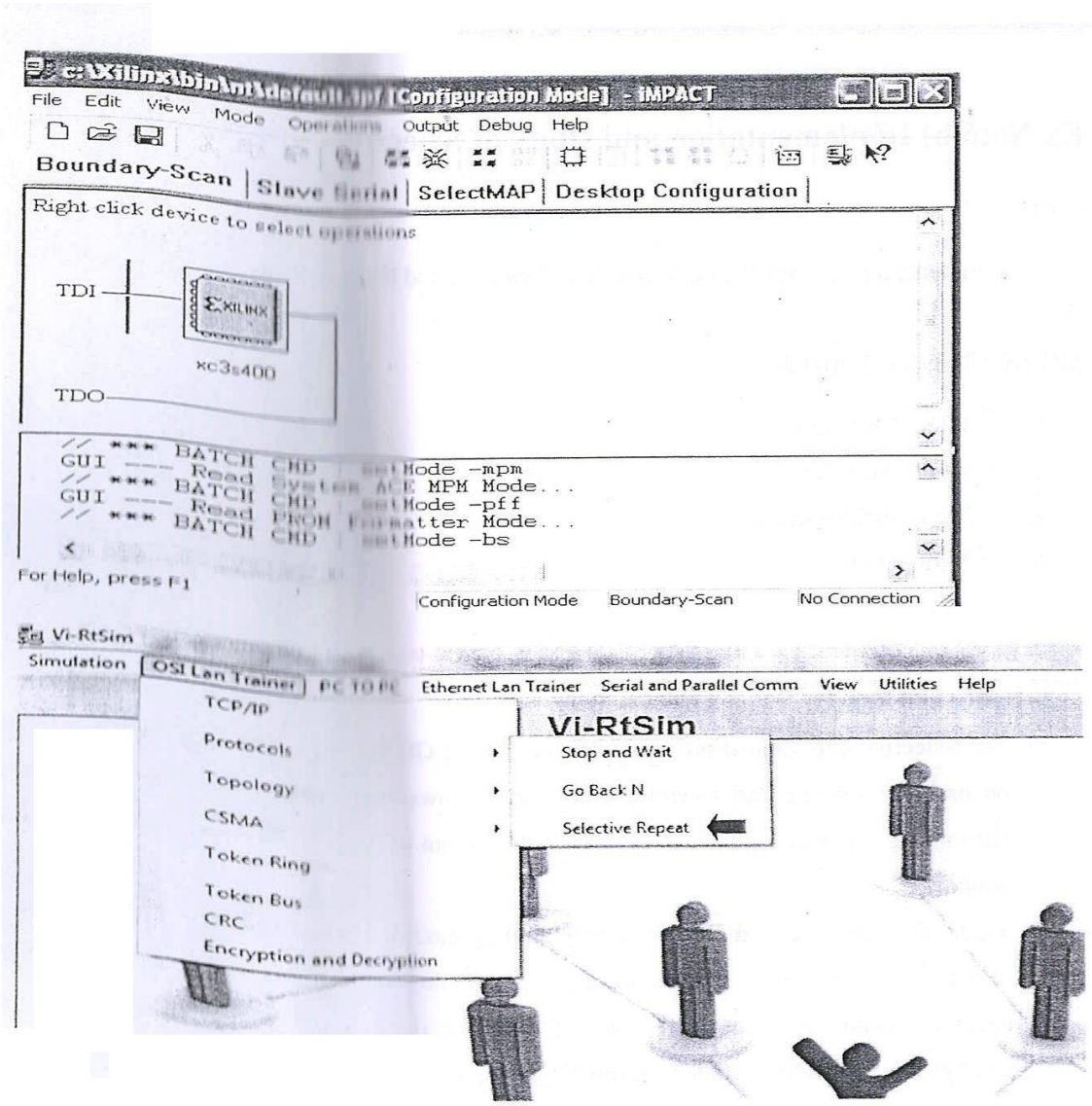
APPARATUS REQUIRED:

1. VI-RTSIM software.
2. Personal Computer.
3. LAN connectivity cable.
4. LAN trainer kit.

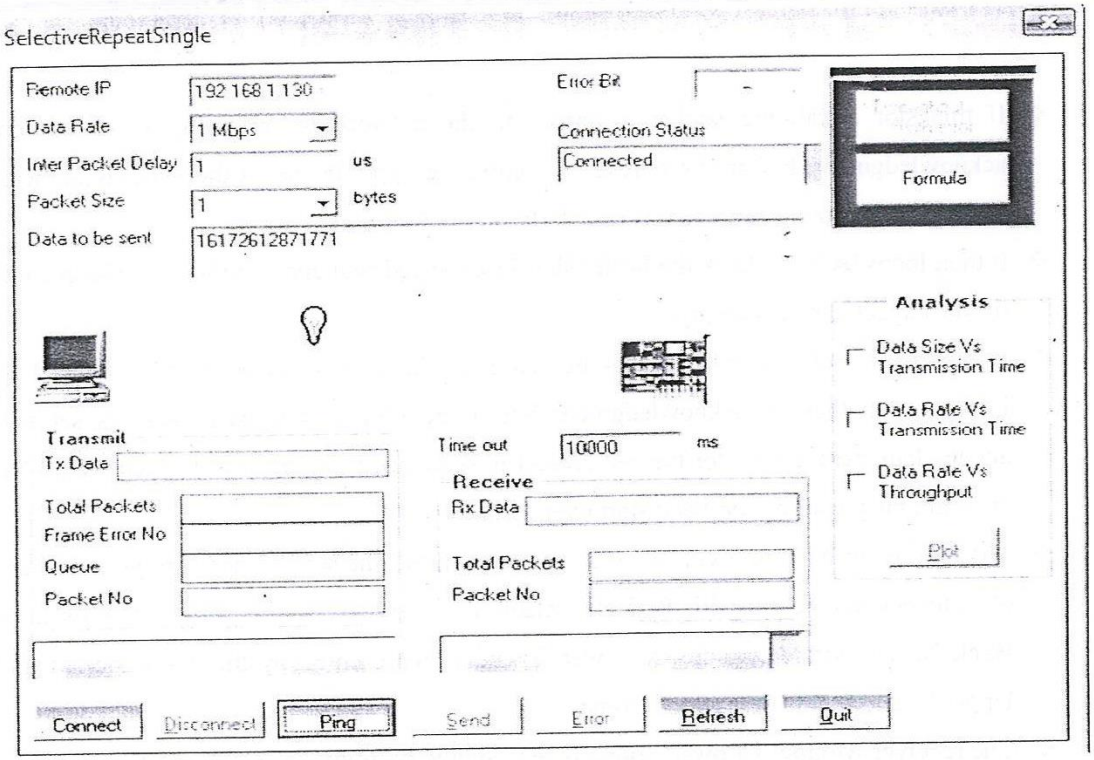
THEORY:

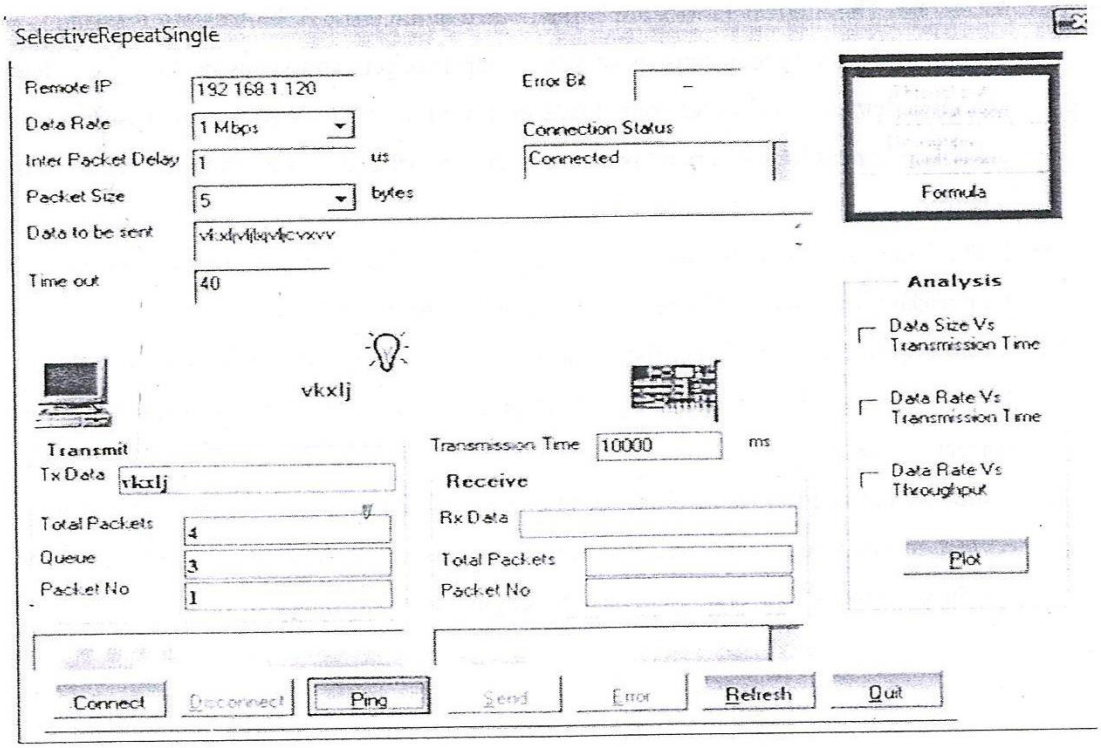
Selective and Repeat:

- The **Selective and Repeat** protocol improves on the GO Back N protocol by having buffers on both the sending and receiving sides. This allows the sender to have more than one outstanding frame at a time and receiver to accept out of order frames and store them in its window.
- Sender for **Selective and Repeat** is only slightly modified from that for **Go Back N.** The Maintenance of buffers and logical timers is exactly the same. The only difference is that if a negative acknowledgment is received, the sender retransmits the corresponding frame identified by the **NAK**. Other than this timeouts, loop iterations and retransmissions are all the same as **Go Back N.**
- This differs from **Go Back N** in that it retransmits only the frame for which a **NAK** is received and not all subsequent frames. As the receiver keeps a window of frames only the timed out frame needs to be retransmitted and not the whole series.
- The receiver busy waits until a frame arrives, if a timeout occurs or if a bad frame arrives, or if an out of sequence frame arrives, and a NAK has not been sent yet then a NAK is sent for the expected sequence number.
- If there is room in the receiver's buffer a packet is stored in the correct slot (sequence number%WINDOWSIZES) and the slot is flagged as used. Loop is run starting at buffer slot for expected sequence number.


File Edit View Mode Operations Output Debug Help
Boundary-Scan
Slave Serial
SelectMAP
Desktop Configuration
Right click device to select operations
TDI
XILINX
xc3s400
TDO
GUI BATCH CMD setMode -mpm
GUI Read System ACE MPM Mode...
GUI BATCH CMD setMode -pff
GUI Read PROM Formatter Mode...
GUI BATCH CMD setMode -bs
For Help, press F1
Configuration Mode
Boundary-Scan
No Connection
Vi-RtSim
Simulation
OSI Lan Trainer
Ethernet Lan Trainer
Serial and Parallel Comm
View
Utilities
Help
TCP/IP
Protocols
Topology
CSMA
Token Ring
Token Bus
CRC
Encryption and Decryption
Vi-RtSim
Stop and Wait
Go Back N
Selective Repeat

- If this slot is fall the packet is passed to the network layer, a flag is set to send an acknowledgment, buffer slot is reset to empty, the upper bound of the window is increased, and the lower bound expected) is circularly incremented.
- It then loops back to check the buffer slot for expected sequence number. The loop continues till the expected slot is empty.
- In this manner all buffered packets are passed to the network layer in order. If the flag that indicates whether an acknowledgment has to be sent or not (send_ack) is set then an acknowledgment is sent for the last correct in sequence frame received. Then we go back to the main busy waiting loop and start over again.
- The buffers on both ends consist of arrays of packets. The size of these arrays is 4, the range of sequence numbers used is 0-7. The selection c criteria were the same as that used for **Go Back N.**The sender window's Lower Bound is represented by the ack expected and the Upper Bound by the next frame to end.
- The receiver window's Lower Bound is represented by frame expected and the Upper Bound by too far. The Logical Timers are implemented in the same fashion as **Go Back N.**
- Main loop in the sender consists of while loop that gets and sends packets then checks for acknowledgments and other loop loads empty slots in the buffer with new packets. The acknowledgment loop clears all previous unacknowledged frames up to the acknowledgment received.
- If the frame is a NAK instead of entering the loop it retransmits the requested frame. Timer loop updates the timers and checks for timeouts to retransmit.
- Main loop in the receiver consist of a busy wait loop to retrieve frames from the physical layer. Data transfer loop passes buffered packets in order to the network layer when the correct in sequence frame is received.
- Timeouts and bad frames result in naks being sent. Otherwise an acknowledgment for the last correct in sequence frame received. This is necessary to keep the sender and receiver in synchronization when frames and acknowledgments are lost.


SelectiveRepeatSingle
Remote IP
192 168 1 130
Error Bit
Data Rate
1 Mbps
Connection Status
Inter Packet Delay
1
us
Connected
Packet Size
1
bytes
Formula
Data to be sent
16172612871771
Analysis
Data Size Vs Transmission Time
Data Rate Vs Transmission Time
Data Rate Vs Throughput
Plot
Transmit
Tx Data
Time out
10000
ms
Receive
Rx Data
Total Packets
Frame Error No
Queue
Packet No
Total Packets
Packet No
Connect
Disconnect
Ping
Send
Error
Refresh
Quit


SelectiveRepeatSingle
Remote IP
192 168 1 120
Error Bit
Data Rate
1 Mbps
Connection Status
Inter Packet Delay
1
us
Connected
Packet Size
5
bytes
Formula
Data to be sent
vkxlj
Time out
40
Analysis
Data Size Vs Transmission Time
Data Rate Vs Transmission Time
Data Rate Vs Throughput
Plot
vkxlj
Transmit
Tx Data
vkxlj
Transmission Time
10000
ms
Receive
Total Packets
4
Rx Data
Queue
3
Total Packets
Packet No
1
Packet No
Connect
Disconnect
Ping
Send
Error
Refresh
Quit

PROCEDURE

Downloading "ARM"

1. Open VI-RTSIM software from desktop
2. Select the ARM Downloader from UTILITIES menu bar to download the ARM.
3. Click Read device ID in the LPC 2000 flash UTILITIES to read the LAN trainer and reset the board at the programming mode and then click OK.
4. Select the TOKENBUS.HEX ARM program to be downloaded to the trainer kit.
5. Click "Upload to flash" button to upload the program to the LAN trainer.
6. After downloading the program we can get the "Program uploaded successfully" message.

Downloading "FPGA"

1. Now open the software "Impact" to download the FPGA.
2. Select "Create a new project" and clock on "OK".
3. Select "configure devices using boundary scan (J-TAG) and click "finish" button.
4. Select the TOKENBUS BIT FPGA program to be downloaded to the trainer kit.
5. Right click on the device and "program" to download FPGA program.
6. After downloading the program we can get the "program successfully" message.

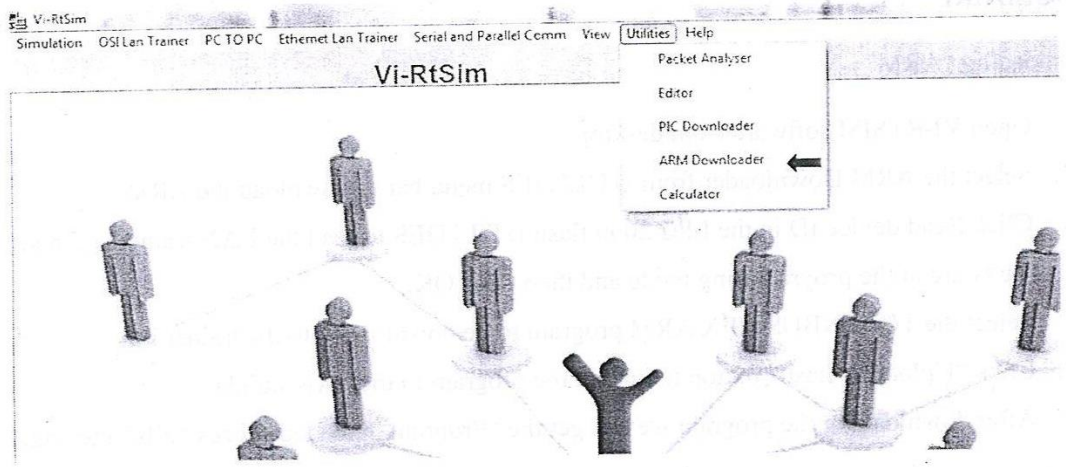
Execution of Selective and Repeat program:

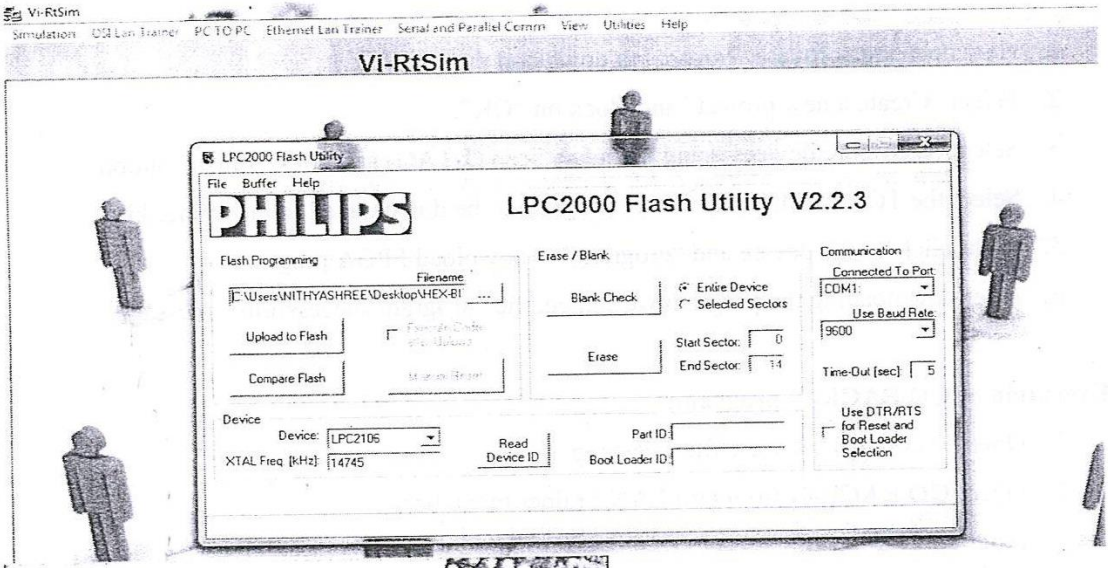
1. Open VI-RTSIM software from desktop.
2. Select Selective and repeat from OSI LAN Trainer menu bar.
3. Enter the Parameter.
4. At default setting press connect button to conform whether the connection is established or not. After connection is successfully established press start button.
5. Enter the error button, If error button pressed, Error window is displayed on LCD (Hardware unit). Where to create the frame error or Bit error.
6. Enter the data on data field, press Send button new data is moving to destination.
7. Sender receives the NAK when error data received at receiver during transmission.
8. After the transmission completed NAK data is retransmitted.

RESULT:

Thus the selective repeat protocol has been implemented and its performance is analyzed.

Implementation of high level data link control





Ex.No: 4 Implementation of high level data link control

Aim:

- To implement and test the high level data link control

Apparatus Required:

1. Vi-Rt sim software
2. C, C++ software
3. Pc-1 No

Theory:

- High-level Data Link Control (HDLC) is a bit-oriented protocol for communication over point-to-point and multipoint links.
- It is designed to support both half duplex and full duplex.
- Systems using HDLC can be characterized by their station types, their configurations , and their response modes.

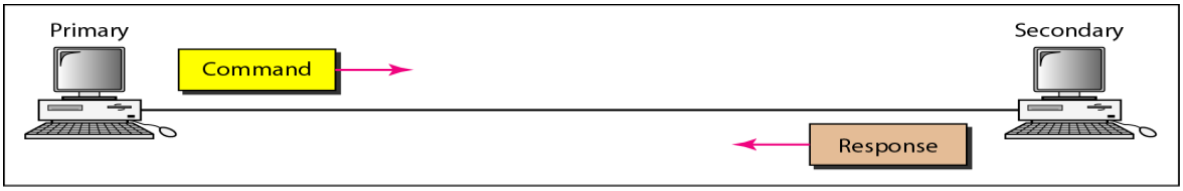
Station types:

- Stations in HDLC are of three types: primary, secondary, and combined.
- The primary send commands to the secondary station. The secondary station sends response. A combined station sends commands and response

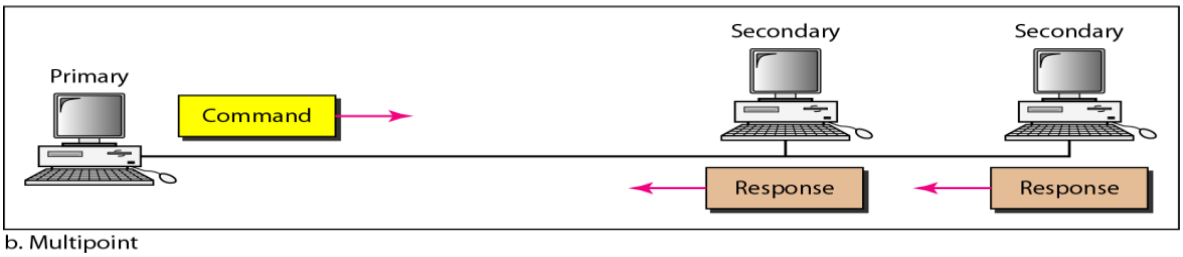
Configuration:

- The work configuration refers to the relationship of hardware devices on a link.
- Primary, secondary and combined stations can be configured in three ways: ***Unbalanced, symmetrical and balanced***
- Any of these configuration supports both half duplex and full duplex transmission

Unbalanced configuration :(master/slave configuration):



a. Point-to-point



b. Multipoint

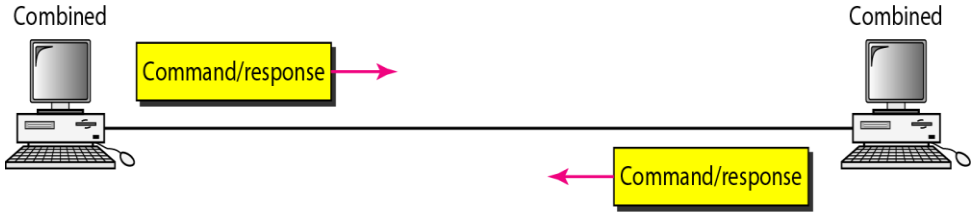
- In this device, one device is primary and other device is secondary.
- Unbalanced configuration can be point to point if only two devices are involved; more often they are multipoint, with one primary controlling several secondaries.

Symmetrical configuration:

- Here each physical station on a link consists of two logical stations, one a primary and other a secondary.
- A symmetrical configuration behaves like an unbalanced configuration expects that control of the link can shift between the two stations.

Balanced configuration:

- A balanced configuration is one in which both station in a point to point topology are of the combined type.
- The stations are linked by a single line that can be controlled by either station.
- HDLC does not support balanced multipoint.



Modes of communication:

- A mode in HDLC is the relationship between two devices involved in an exchange the mode describes who control the link.
- HDLC supports three modes of communication between stations.

(i) Normal response mode.(NRM)

(ii) Asynchronous response mode(ARM)

(iii) **Asynchronous Balanced mode(ABM**)

Normal response mode. (NRM):

- In normal response mode (NRM), the station configuration is unbalanced.
- We have one primary station and multiple secondary stations.
- A primary station can send commands; a secondary station can only respond.
- The NRM is used for both point-to-point and multiple-point links

Asynchronous response mode (ARM):

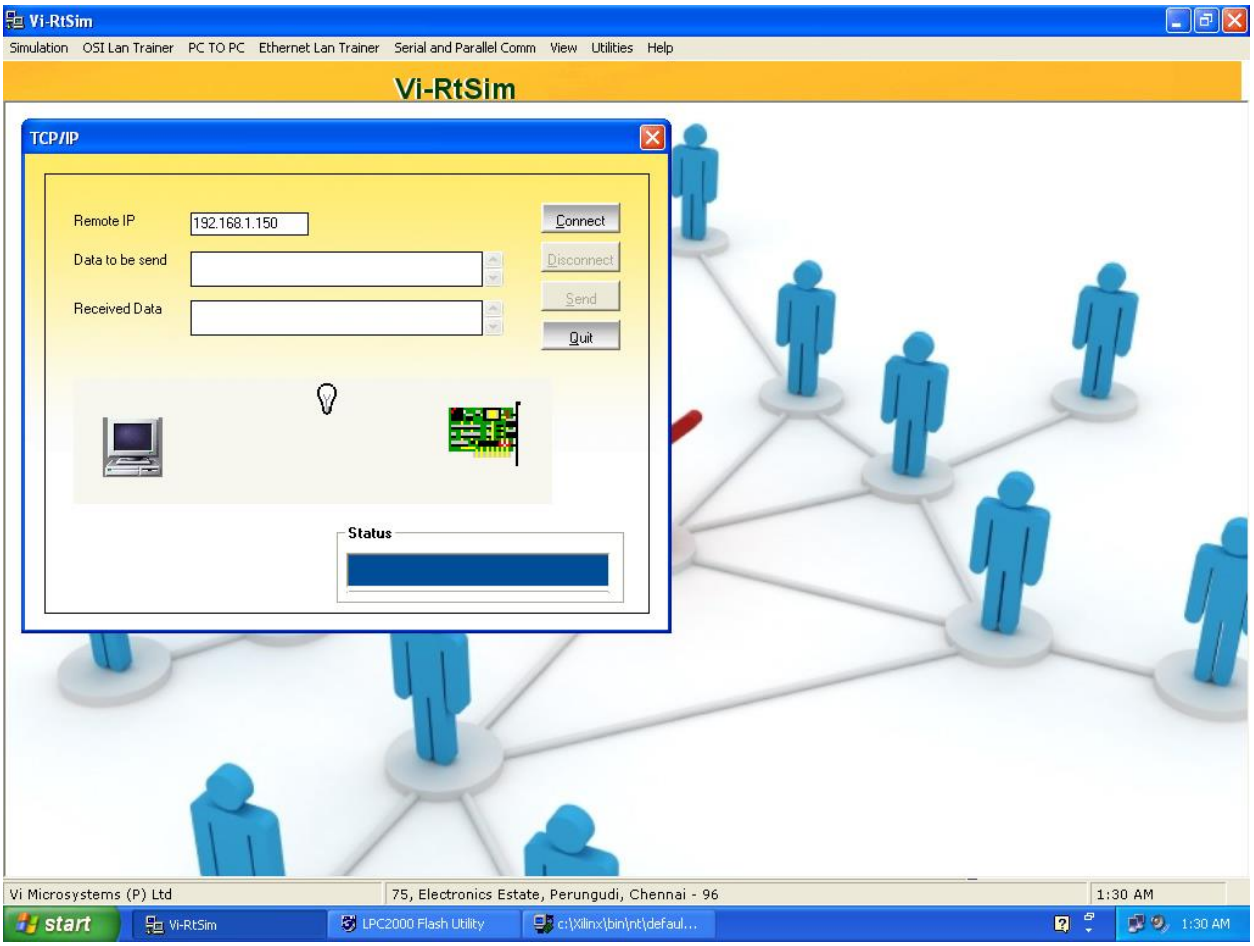
- In ARM, a secondary may initiate a transmission without permission from the primary whenever the channel is idle.
- ARM does not alter the primary and secondary relationship in any other way

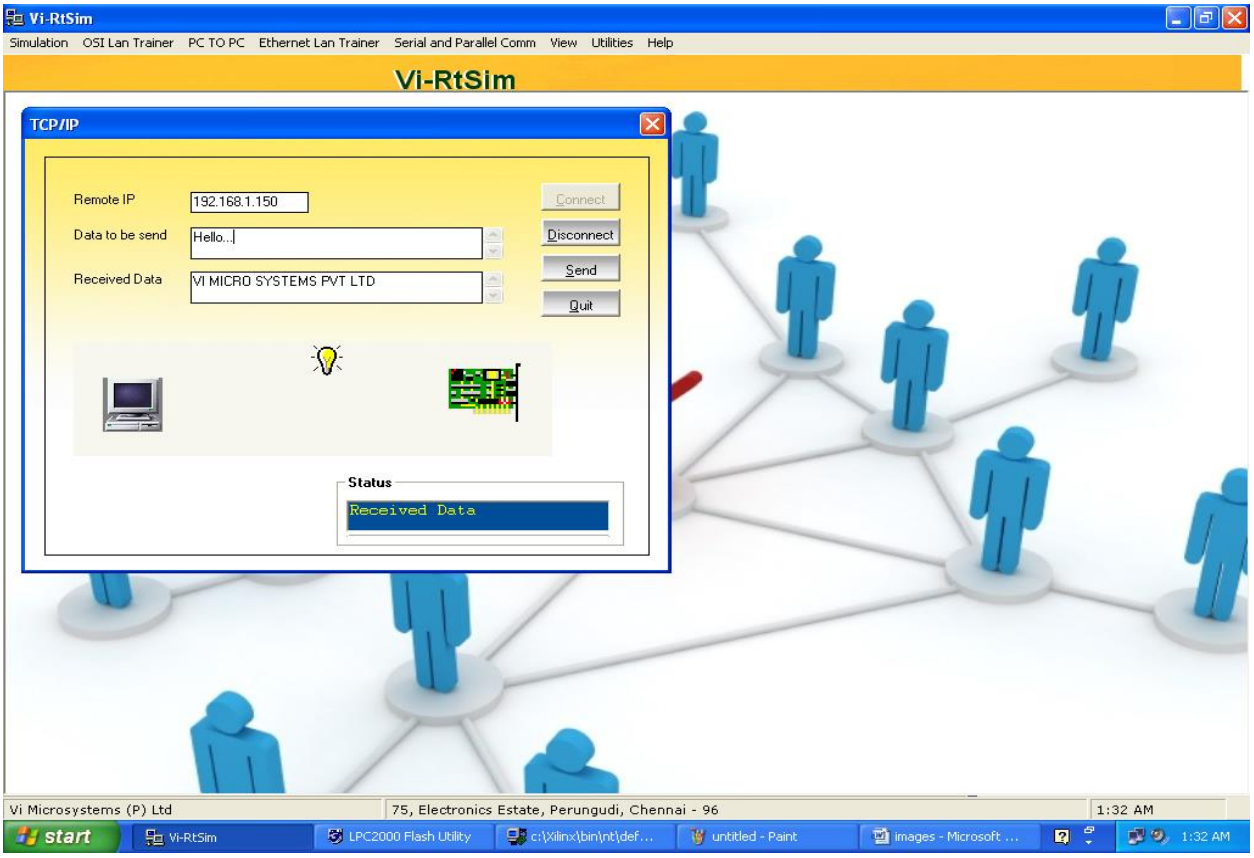
- All transmission from a secondary must still be made to the primary for relay to a final destination.

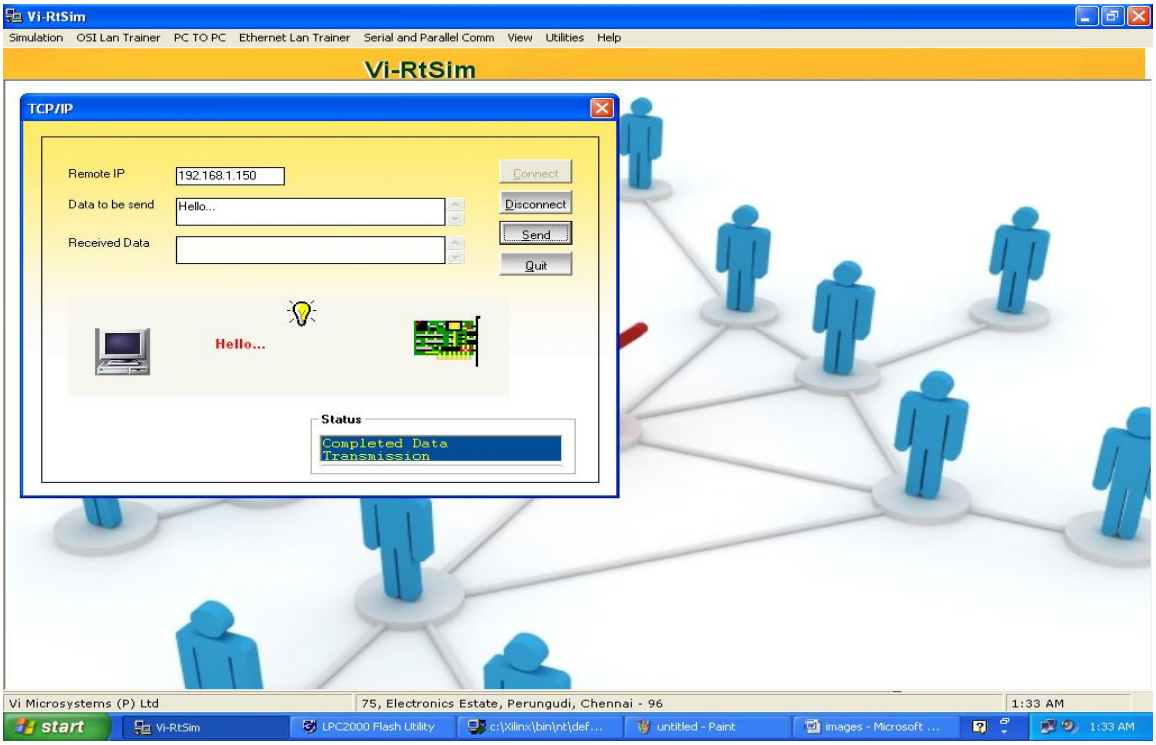
Asynchronous Balanced mode (ABM):

- In ABM, all stations are equal and therefore only combined stations connection in point to point are used.
- Either combined station may initiate with the other combined station without permission

	(NRM)	**(ARM)**	**(ABM)**
Station type	Primary and secondary	Primary and secondary	combined
Initiator	Primary	either	any




Vi-RtSim
Simulation OSI Lan Trainer PC TO PC Ethernet Lan Trainer Serial and Parallel Comm View Utilities Help
Vi-RtSim
TCP/IP
Remote IP
192.168.1.150
Data to be send
Hello...
Received Data
VI MICRO SYSTEMS PVT LTD
Connect
Disconnect
Send
Quit
Status
Received Data
Vi Microsystems (P) Ltd
75, Electronics Estate, Perungudi, Chennai - 96
1:32 AM
start
Vi-RtSim
LPC2000 Flash Utility
c:\Xilinx\bin\nt\def...
untitled - Paint
images - Microsoft ...
1:32 AM


Vi-RtSim
Simulation OSI Lan Trainer PC TO PC Ethernet Lan Trainer Serial and Parallel Comm View Utilities Help
Vi-RtSim
TCP/IP
Remote IP
192.168.1.150
Data to be send
Hello...
Received Data
Connect
Disconnect
Send
Quit
Hello...
Status
Completed Data Transmission
Vi Microsystems (P) Ltd
75, Electronics Estate, Perungudi, Chennai - 96
1:33 AM
start
Vi-RtSim
LPC2000 Flash Utility
c:\Xilinx\bin\nt\def...
untitled - Paint
images - Microsoft ...
1:33 AM

Procedure:

- Start the process
- Write a program to free up all packet queue of HDLC channel
- Creat a HDLC channel
- Destroy the creat HDLC channel
- Reset the HDLC packet encoder function
- Write the HDLC packet decoder function
- Stop the process

Result:

- Thus the high data link control has been implemented successfully

Ex.No: 5 Study of Socket Programming and Client – Server model

Aim:

To study about the socket programming and client server model.

Theory:

Socket Programming:

Socket provides the communication mechanism between two computer using TCP.A client program creates a socket on its end of the communication attempts to connect that socket to a server.

When the computer connection is made, the server creates a socket object on the end of the communication.The client and server can move communicate by writing to and reading from the socket.

The JAVA net socket class provides a socket and the JAVA net server socket class provides a mechanism for the server program to listen for client and establish connection with them.

The following stebs occur when establishing a TCP connection between two computers using sockets.

1. The server instruction a server socket object, denotes which port number communication is to our on.
2. The server involves, the accept () method of the screen socket class.This method wait until a client connect to the server on the given port.
3. After the server is waiting,actent instantiar a socket object specifying a server name and port number to connect
4. The constructor of the socket class attempts to connect the client to the specified server and port number.
5. .On the server side,the accepts() method returned a reference to a new socket on the server then is connected to the client sockets.

Server socket class method:

The JAVA net server socket class is used by server application to obtain a port and listen for client request.It has four instructions.

S.no	Methods with Description
1	Public server socket throws IOE exception (inport) attempts to create a server socket bound to be specified port.
2	Public server socket (intport,int block log) throws IOE exception. Similar to the previous construction the back log parameter specifies how many incoming client to core in a wait queue.
3	Public server socket (intport, int back log, int adder) throws IOE exception. Similar to the previous constructors, the int adder parameter specifies the local adder to the int address is used for server trah.may have the server that may allowing the server of its IP address to accept client request on.
4	Public server socket() throws IOE exception: Creates on unbound server socket when using their constructor,in the bind() method when open are ready to bind server socket.

S.no	Description
1	Public int get local port() Return the port that server socket is listening on.
2	Public socket accept() throws IOE exception Waits for an incoming client.This method blocks until either a client connect to the server on the specified port or the socket time out() method.
3	Public void set so time out (int time out) Set the time out value for the long server socket wait for a client deciding the accept().
4	Public void bind(socket address host,intbatchlay) Binds the socket to the specified server port in the socket address object.

Inet address class methods:

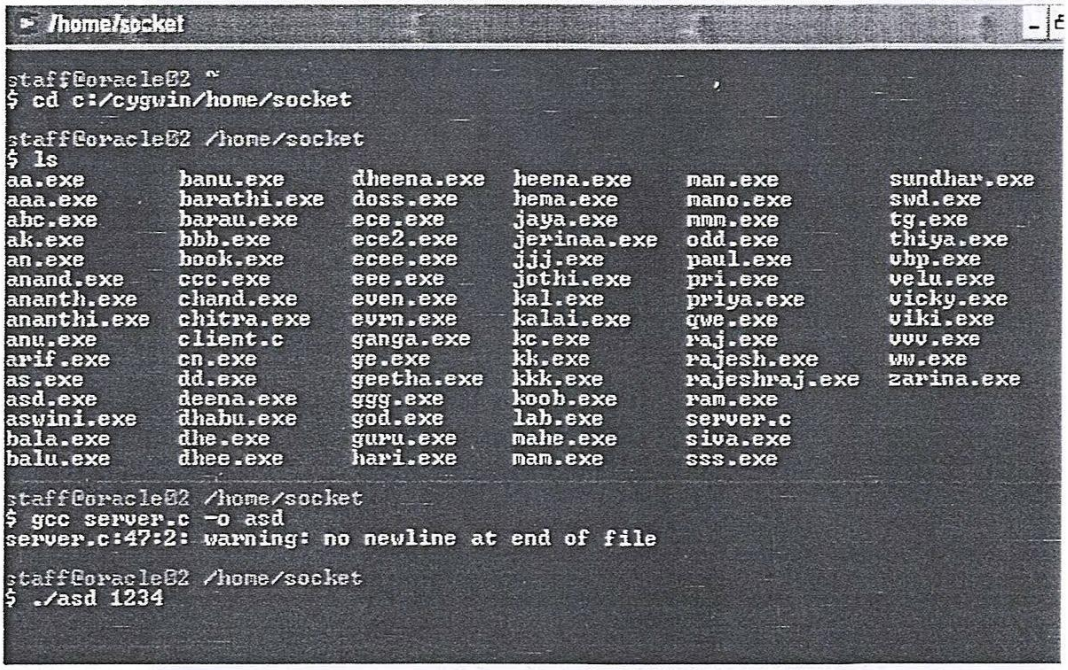
This class represent an Internet Protocol () address.There are following useful methods which you would need while socket programming.

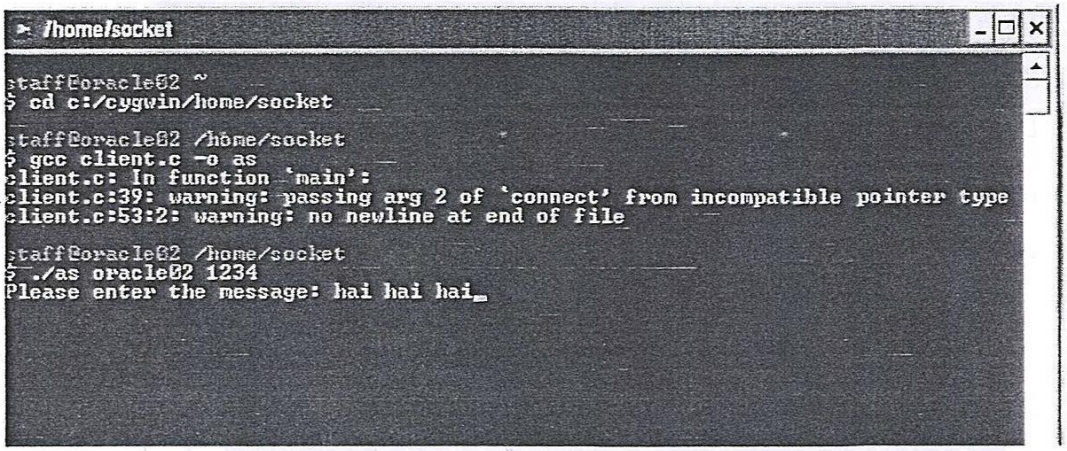
1	Static inet address get by address(byte/address) Return as intel address object given the input address.
2	Static int address get by address (string, host, byte (). Create an input address based on the provided host have input address.
3	Static inet address get by name (string host) determine the input address of a host, given the hosts name.
4	String get host address() Returns the input address string in textual regulation.
5	String get host name() Gets the host name for this input address.
6	Static inct address get local host() Return the local host.
7	String to String Convert this IP address to a string.

Result:

Thus the study of socket programming and client server model has been completed successfully.

Transfer of file from PC to PC using wireless-Socket Processing





Ex No: 6. Write a socket Program for Echo/Ping/Talk commands using wireless-Socket Processing

AIM:

To write a program for transfer the data from one PC to PC using Wireless-Socket processing

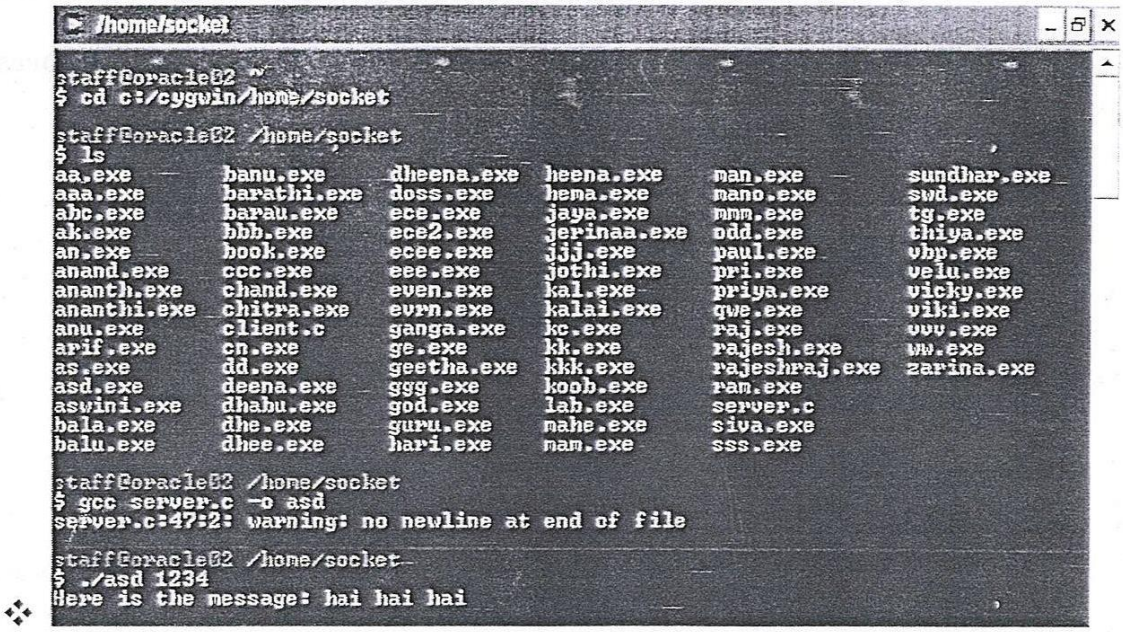
APPARATUS REQUIRED:

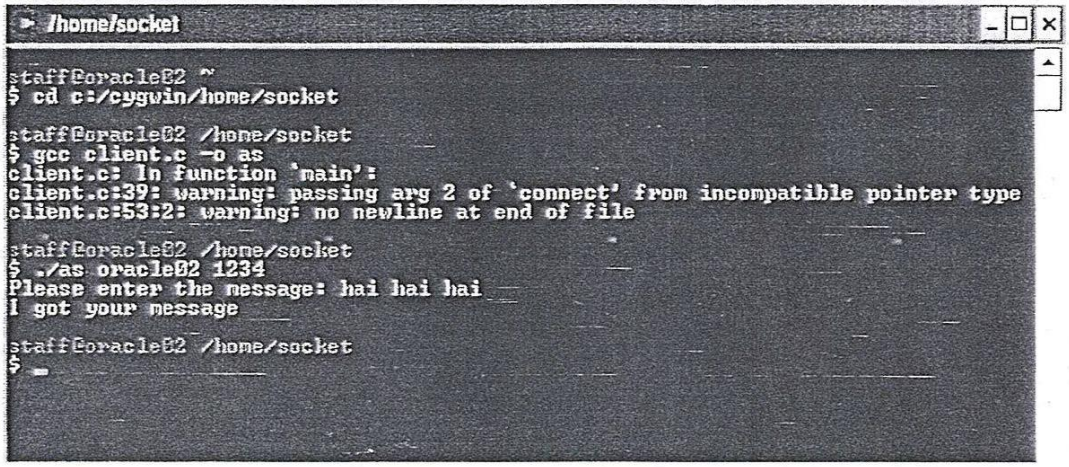
1. Personal computer.
2. Network simulator software.

THEORY:

Socket address:

- Process to process delivery needs to identify the IP address and the port address at each end to make a connection.
- The combination of an IP address and a port number is called a socket address.
- The client socket address defines the client process uniquely just as the server socket address defines the server process uniquely.
- A transport layer protocol needs a pair of socket address "the client socket address and the server socket address".
- These are past of the header and the transport layer protocol header. The IP header contains IP address. The TCP header contains the port number.
- The MAC sub layer protocol is quite different from that of Ethernet due to inherent complexity of the wireless environment compared to that of a wired system.
- With Ethernet, a station just waits until the either goes silent or starts transmitting.
- If it does not received, a noise burst back within the first 64 bytes, the frame has almost assuredly been delivered correctly.


/home/socket
staff@oracle02 ~
$ cd c:/cygwin/home/socket
staff@oracle02 /home/socket
$ ls
aa.exe banu.exe dheena.exe heena.exe man.exe sundhar.exe
aaa.exe barathi.exe doss.exe hema.exe mano.exe swd.exe
abc.exe barau.exe ece.exe jaya.exe mmm.exe tg.exe
ak.exe bbb.exe ece2.exe jerinaa.exe odd.exe thiya.exe
an.exe book.exe ecee.exe jjj.exe paul.exe vbp.exe
anand.exe ccc.exe eee.exe jothi.exe pri.exe velu.exe
ananth.exe chand.exe even.exe kal.exe priya.exe vicky.exe
ananthi.exe chitra.exe evrn.exe kalai.exe qwe.exe viki.exe
anu.exe client.c ganga.exe kc.exe raj.exe vvv.exe
arif.exe cn.exe ge.exe kk.exe rajesh.exe ww.exe
as.exe dd.exe geetha.exe kkk.exe rajeshraj.exe zarina.exe
asd.exe deena.exe ggg.exe koob.exe ram.exe
aswini.exe dhabu.exe god.exe lab.exe server.c
bala.exe dhe.exe guru.exe mahe.exe siva.exe
balu.exe dhee.exe hari.exe nam.exe sss.exe
staff@oracle02 /home/socket
$ gcc server.c -o asd
server.c:47:2: warning: no newline at end of file
staff@oracle02 /home/socket
$ ./asd 1234
Here is the message: hai hai hai


/home/socket
staff@oracle02 ~
$ cd c:/cygwin/home/socket
staff@oracle02 /home/socket
$ gcc client.c -o as
client.c: In function 'main':
client.c:39: warning: passing arg 2 of 'connect' from incompatible pointer type
client.c:53:2: warning: no newline at end of file
staff@oracle02 /home/socket
$ ./as oracle02 1234
Please enter the message: hai hai hai
I got your message
staff@oracle02 /home/socket
$

PROCEDURE

1. Create a new file in the notepad.
2. Type the message and save the file either in “C” or “D” drove
3. Open VI-RT SIM software from desktop.
4. Select a process to act as server using following steps.

   cd c:/cygwin/socket/home

   gcc server.c-o file 1(note file 1 should be file already present in folder

   c:/cygwin/socket/home)

   ./file 1 1234
5. Select a process to act as client using following steps.

   cd c:/cygwin/socket/home

   gcc client.c-o file2 (note file2 should be file already present in folder c:/cygwin/socket/home)

   ./file2 oracle02 1234 (note oracle02 is name given to personal computer by administrator)
6. Now client process will ask “please enter the message”
7. The message typed will be received by server process.
8. Now the server will acknowledge by sending “I got your message”.
9. The file is transferred successfully from one PC to another.

RESULT:

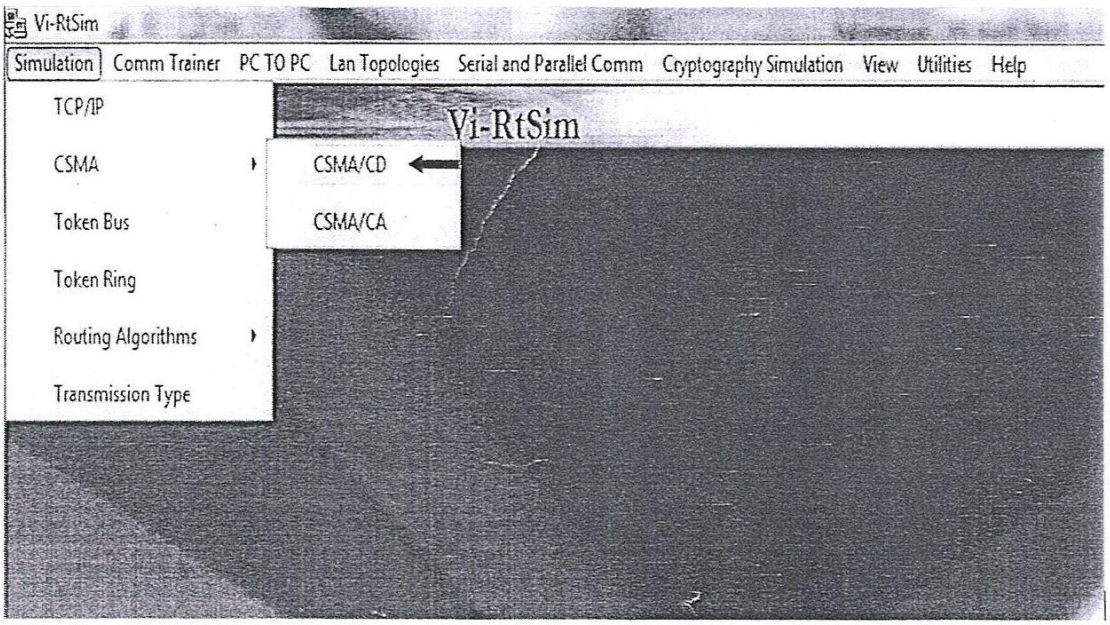
Thus the transfer of data from one PC to PC using windows [wireless] socket processing has been performed.

STUDY OF CSMA/CD USING SIMULATION

Program:

```
Include <protocol.h>
void main()
{
        Frame  X,Y;
   X="data1";
   Y="data2";

  CSMACD_INIT();
  CSMACD_START();

              CSMA_SEND(B,A,X);
              CSMA_SEND(A,C,Y);

   R=COLLISION_OCCUR();
             if(R)
            {
                WAIT(1000);

                RETRANSMIT(B,A);
                RETRANSMIT(A,C);
            }
}
```

MENU BAR



Ex. No: 7(a). Study of CSMA/CD using Simulation

AIM:

To Simulation and study the performance of CSMA/CD

APPARATUS REQUIRED:

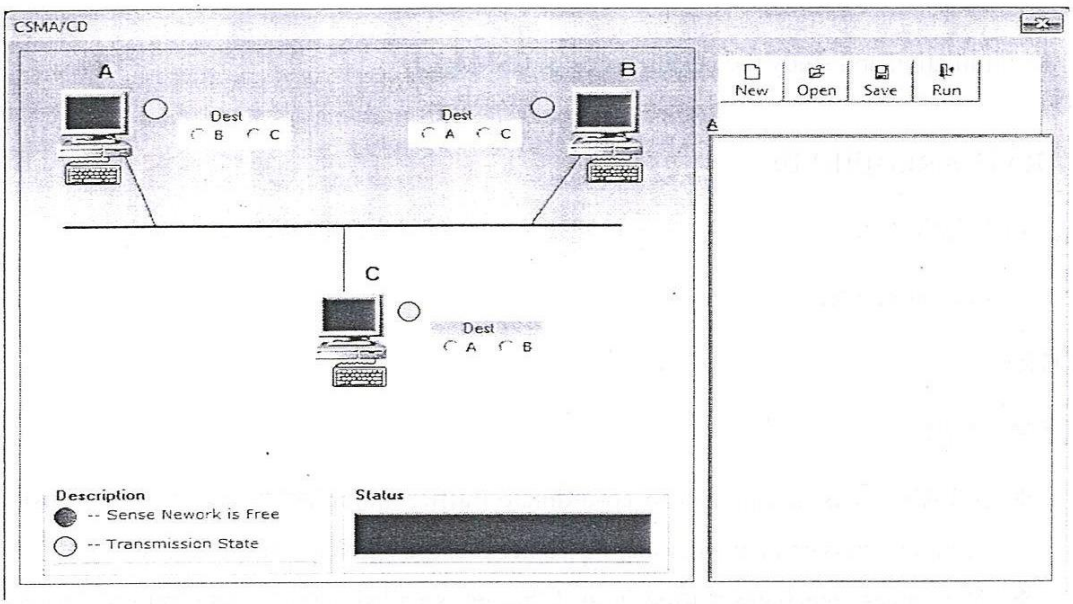
1. VI-RTSIM software.
2. Personal computer

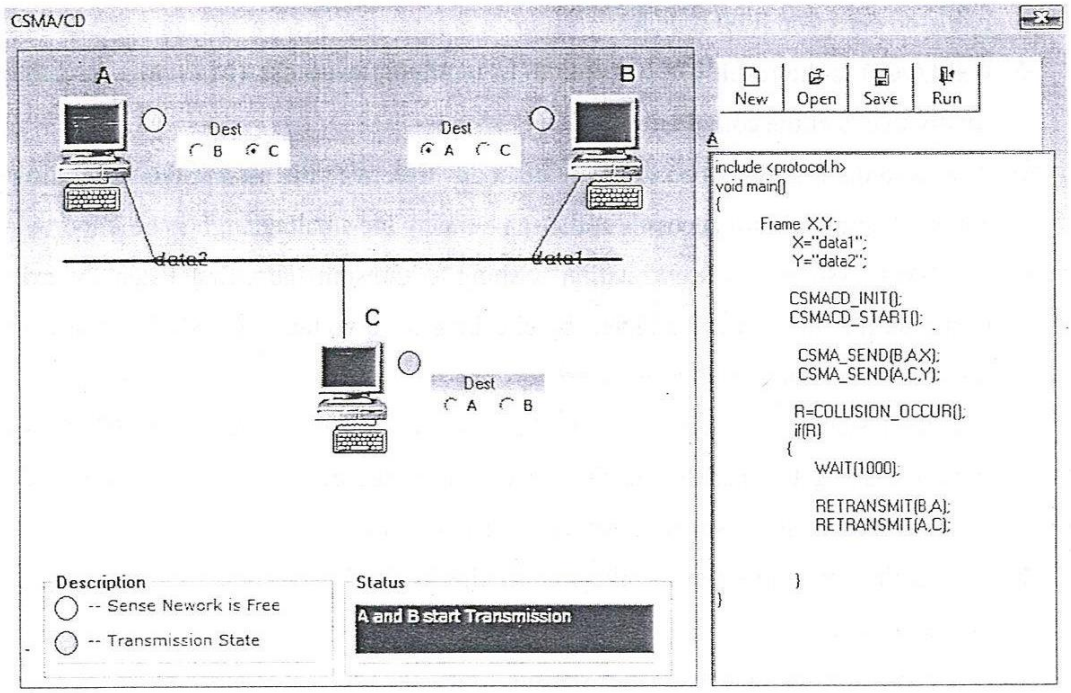
THEORY

CSMA/CD:

- A LAN needs mechanism to co-ordinate traffic, minimize the number of collision that occur and maximize the number of frames that are delivered successfully.
- The access mechanism used in an Ethernet is called carrier sense multiple access with collision detection [CSMA/CE]
- CSMA/CD is the result of evolution from multiple access (MA) to sense the carrier and finally to detect the collision.
- Access to the link was open to any node at any time with the assumption that adds of two devices completely for accesses at the same time were small enough to be important.
- In CSMA system, any work station wishing to transmit must first listen for exiting to traffic on the line. A device listens by checking for a voltage. A CSMA cut down on the no of collision does not eliminate them.
- The final step in the addition of CSMA/CD is collision detection. In CSMA/CD the station wishing to transmit the first listener to make certain time to transmits the link which is free, then transmits its data, then listen again.
- During the data transmission, the station checks the line for external high voltage that indicates collision.
- If a collision is detected, the station quits the current transmission and waits a predetermined amount of time for the to clear, then sends the data again.

PROGRAMMING ENVIRONMENT

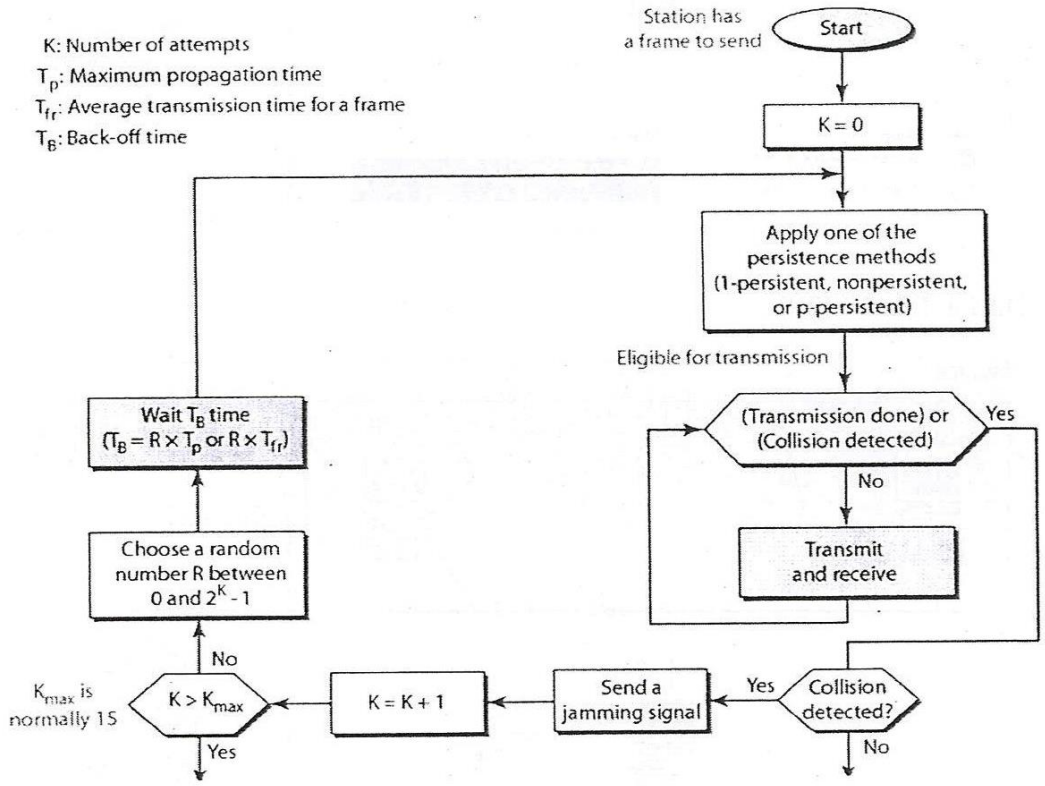




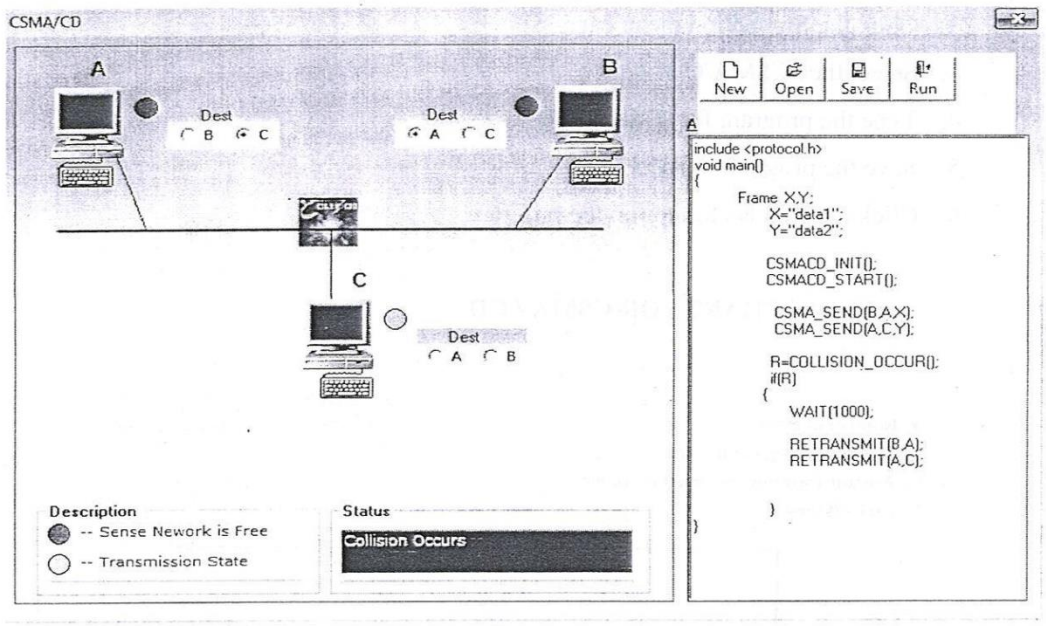
PROCEDURE

1. Open VI-RTSIM software from desktop
2. Click the Simulation menu bar
3. Select the "CSMA/CD" option
4. Type the program for "CSMA/CD"
5. Save the program to RUN.
6. Click the Run Button to run the program.

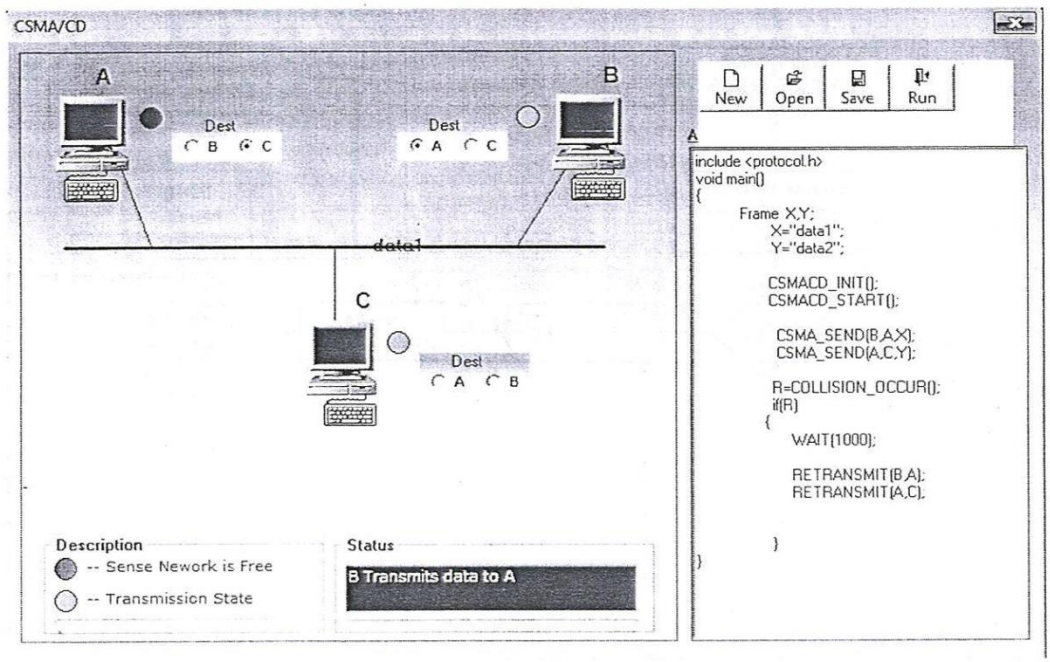
FLOW CHART FOR CSMA / CD

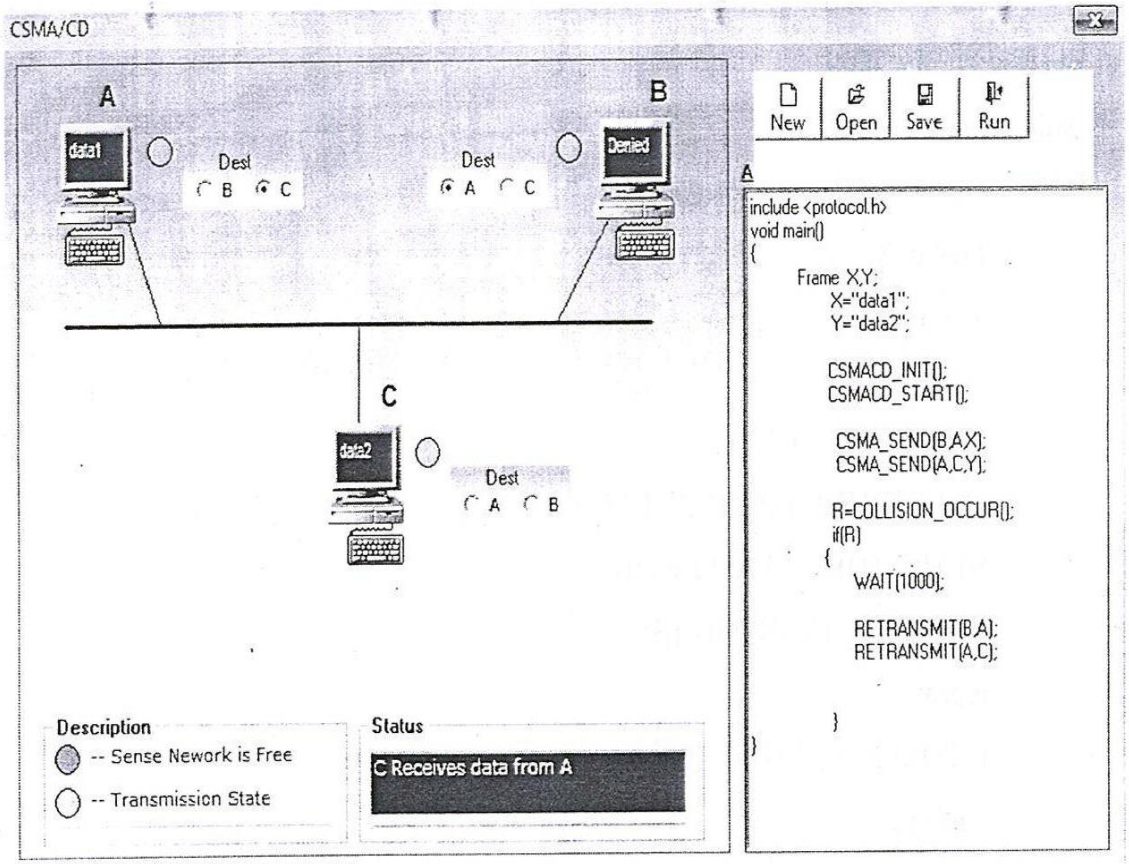


COLLISION OCCURANCE



DATA TRANSFER





Result:

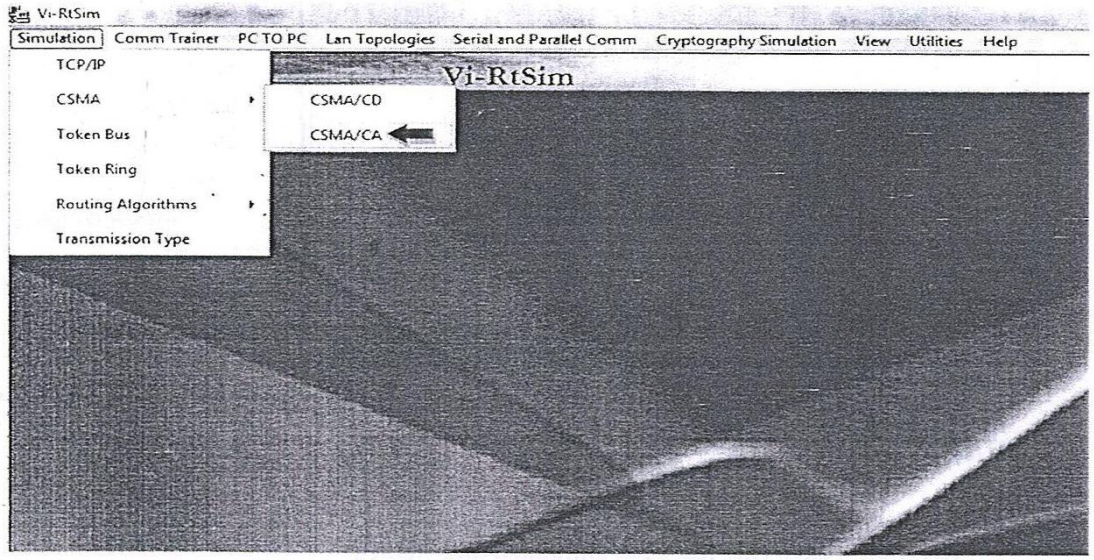
Thus, the CSMA/CD has been simulated and studied.

Study of CSMA/CA using Simulation

Program:

```
include <protocol.h>
void main()
{
        Frame  X;

        X="data1";

        CSMACA_INIT();

        CSMACA_START();

        NODE_LISTEN();

        REQUESTTO_SEND(A,B);

        CLEARTO_SEND(B,A);

        DATATO_SEND(A,B,X);

        ACKNOWLEDGE(B,A);


}
```

MENU BAR



Ex. No: 7(b).Study of CSMA/CA using Simulation

AIM:

To Simulate and study the performance of CSMA/CA protocol.

APPARATUS REQUIRED:

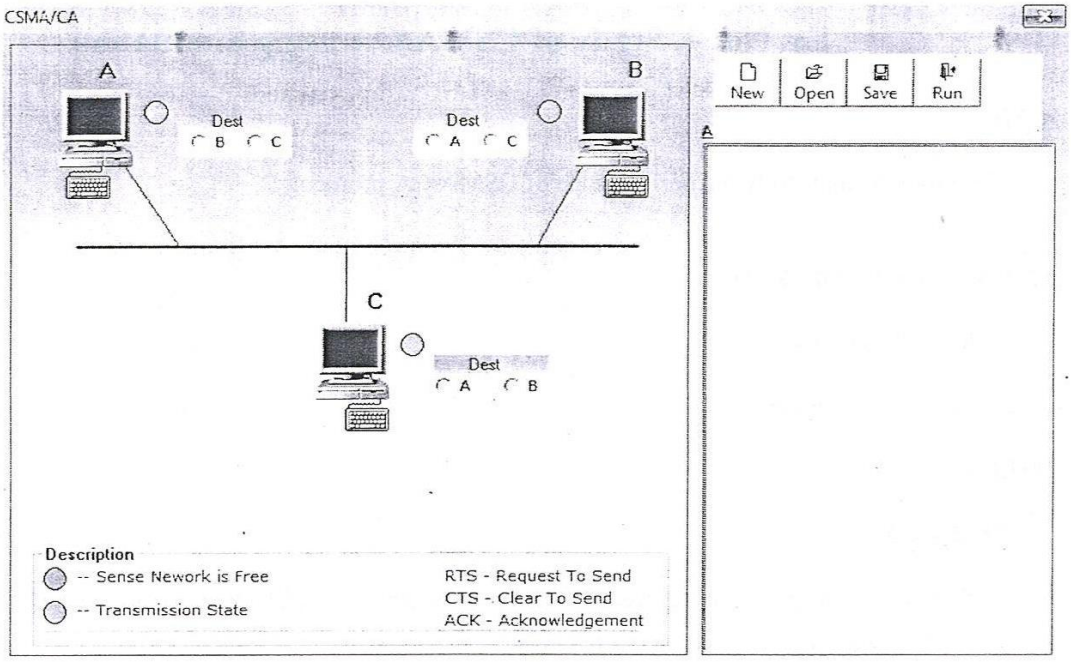
1. VI-RTSIM software.
2. Personal computer.

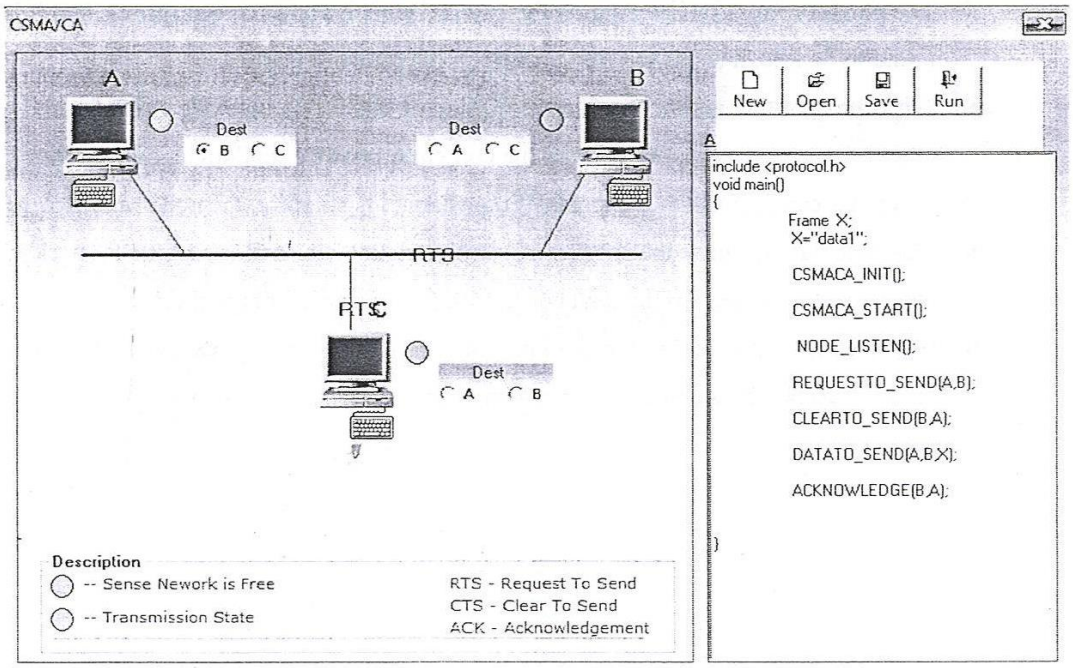
THEORY

CSMA/CA:

- Carrier sense Multiple Access/Collision Avoidance is the channel access mechanism used by most wireless LAN in the ISM bands.
- A channel access mechanism is the part of the protocol which specifies how the node uses the medium; when to listen, when to transmit.
- The protocol starts by listening on the channel (this is called carrier sense), and if it is found to be idle, it sends the first packets in the transmit queue.
- If it is busy (either another node transmission or interface), the node waits the end of the current transmission and then starts the contention (wait a random amount of time).
- When it's contention timer express, if the channel is still idle, the node sends the packet.
- The node having chose the shortest contention delay wins and transmits it's packet. The other nodes just wait for the next contention (at the end of this packet}
- Because the contention is a random number and done for every packets, each node is given an equal chance to access the channel (on average-it is statistic)

PROGRAMMING ENVIRONMENT

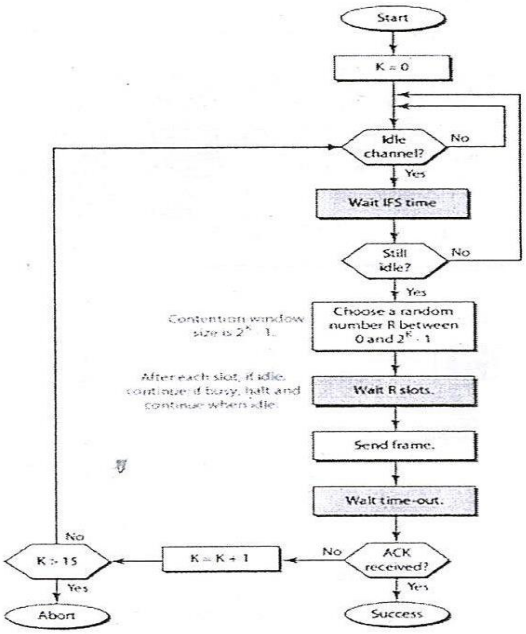


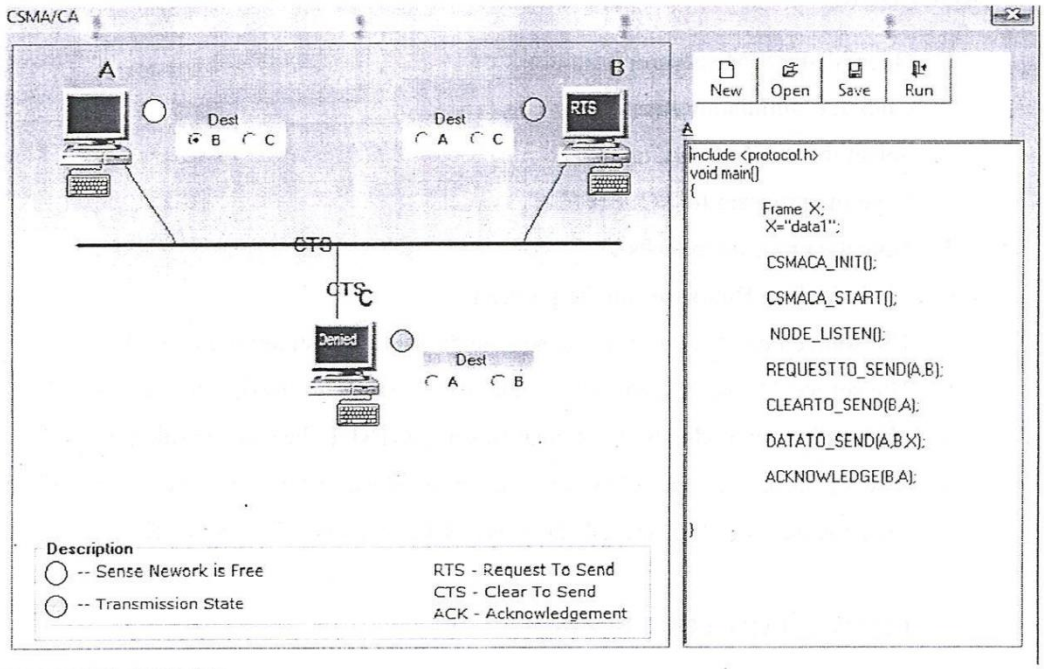


PROCEDURE

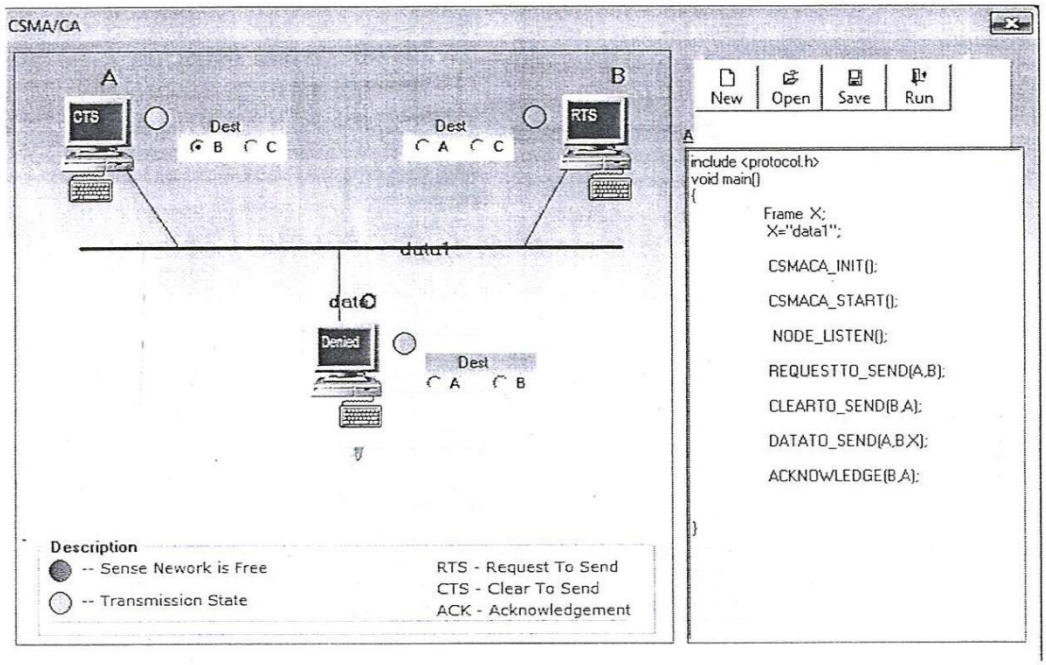
1. Open VI-RTSIM software from desktop
2. Click the Simulation menu bar
3. Select the "CSMA/CA" option
4. Type the program for "CSMA/CA"
5. Save the program to RUN.
6. Click the Run Button to run the program.
7. The source node "A" wants to sends the data to the destination node "B".
8. The source "A" send, Request To Send (RTS) signal to the destination node "B".
9. The destination node "B" send clear to send (CTS) to the source node "A".
10. Now the destination node "B" sends acknowledged "ACK" to source node "A".
11. Thus the data was transmitted successfully from node "A" to node "B"

FLOW CHART FOR CSMA/CA

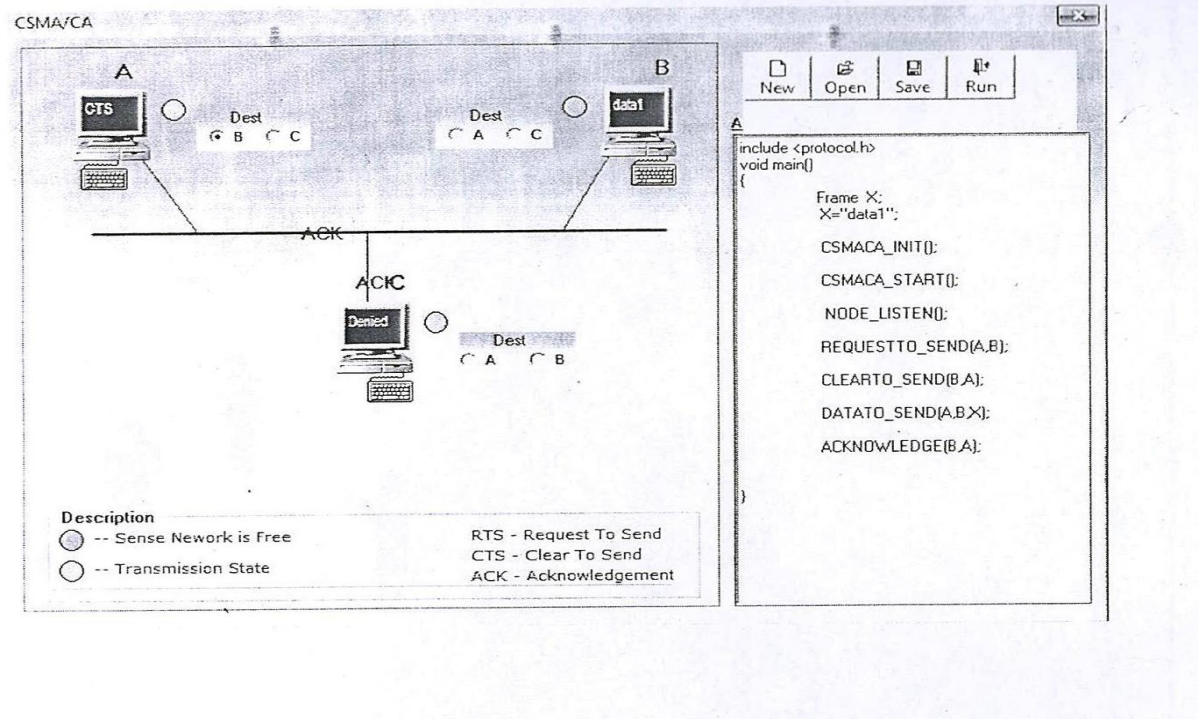




DATA TRANSFER

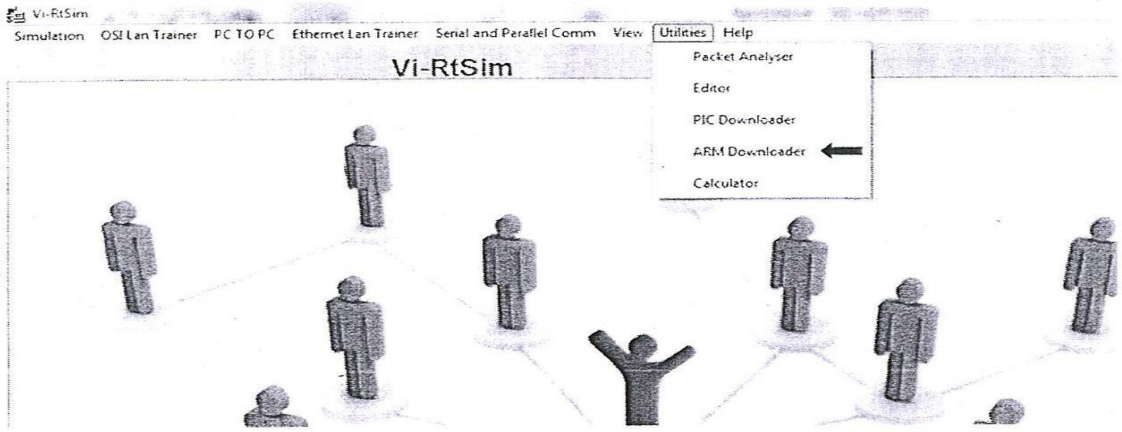


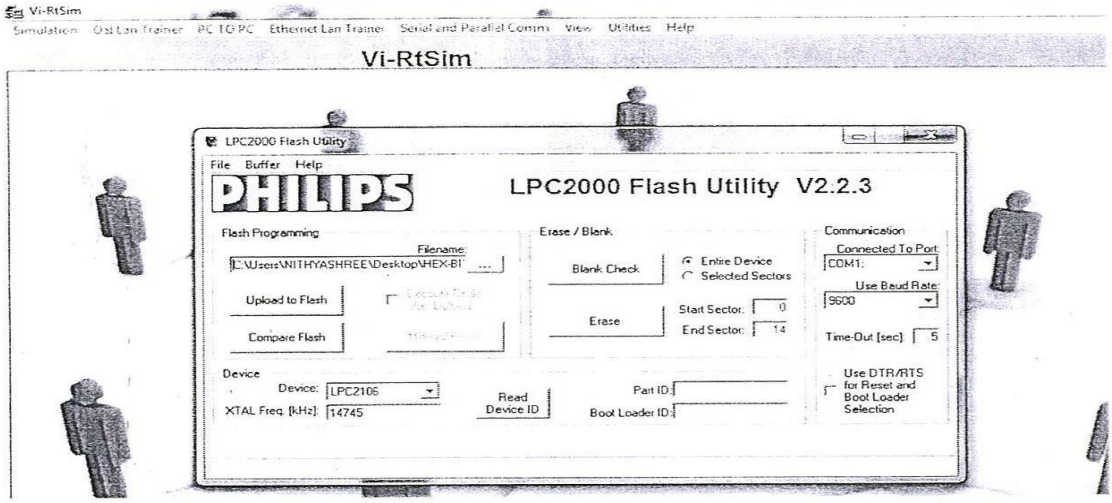
ACKNOWLEDGEMENT



Result:

Thus, the CSMA/CA has been simulated and studied


Vi-RtSim
Simulation OSI Lan Trainer PC TO PC Ethernet Lan Trainer Serial and Parallel Comm View Utilities Help
Vi-RtSim
Packet Analyser
Editor
PIC Downloader
ARM Downloader
Calculator


Vi-RtSim
Simulation OSI Lan Trainer PC TO PC Ethernet Lan Trainer Serial and Parallel Comm View Utilities Help
Vi-RtSim
LPC2000 Flash Utility
File Buffer Help
PHILIPS
LPC2000 Flash Utility V2.2.3
Flash Programming
Filename:
C:\Users\NITHYASHREE\Desktop\HEX-BI
Upload to Flash
Compare Flash
Erase / Blank
Blank Check
Entire Device
Selected Sectors
Erase
Start Sector: 0
End Sector: 14
Communication
Connected To Port:
COM1:
Use Baud Rate:
9600
Time-Out [sec]: 5
Device
Device: LPC2106
XTAL Freq. [kHz]: 14745
Read Device ID
Part ID:
Boot Loader ID:
Use DTR/RTS for Reset and Boot Loader Selection

Ex No: 8. Network Topology - Star, Bus, Ring

AIM:

To implement the network topologies star, bus and ring using Vi-Rt sim.

APPARATUS REQUIRED:

1. VI-RTSIM software.
2. Personal Computer.
3. LAN connectivity cable.
4. LAN trainer kit.

THEORY:

NetworkTopologies

Topology refers to the way a network is laid out either physically or logically. Two or more devices connect to alink; two or more links form a topology. It is the geographical representation of the relationship of all the links and linking devices to each other.

1. Mesh
2. Star
3. Tree
4. Bus
5. Ring
6. Hybrid

STARTOPOLOGY:

Here each device has a dedicated link to the central _hub'. There is no direct traffic between devices. The transmission are occurred only through the central controller namely hub.

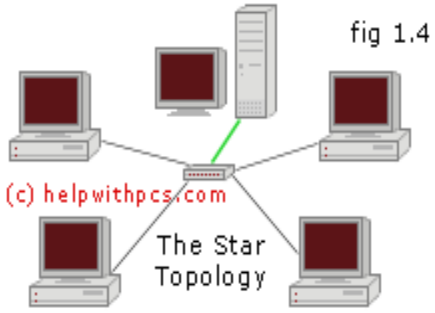


fig 1.4

Advantages:

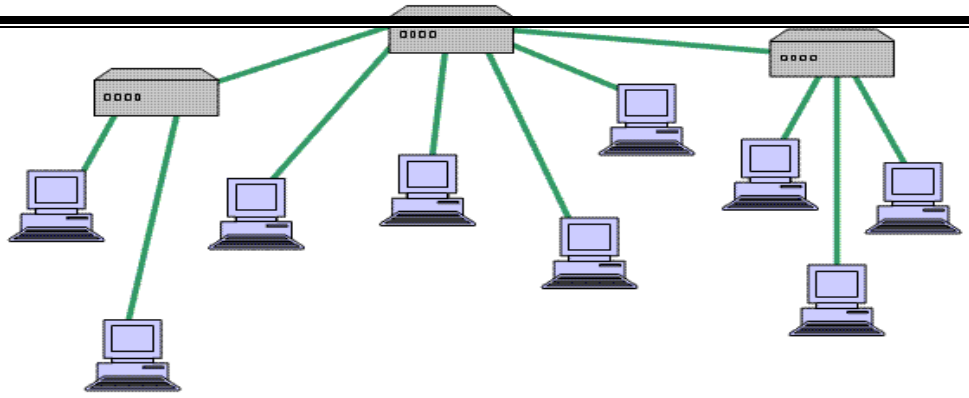
1.Less expensive then mesh since each device is connected only to the hub.
2.Installation and configuration are easy.
3.Less cabling is need then mesh.
4. Robustness.
5. Easy to fault identification &isolation.

Disadvantages:

1. Even it requires less cabling then mesh when compared with other topologies it still large.

TREE TOPOLOGY:

It is a variation of star .Instead of all devices connected to a central hub heremost of the devices are connected to a secondary hub that in turn connected with central hub. The central hub is an active hub . An active hub contains a repeater, which regenerate the received bit pattern before sending.



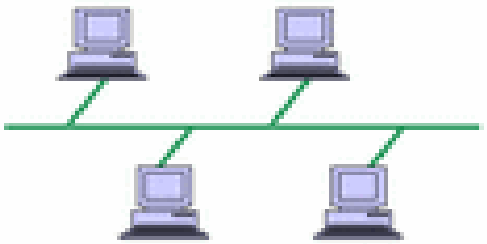
These condary hub maybe active or passive. A passive hub means it just precedes a physical connection only.

Advantages:

1. Can connect morethan star.
2. The distance can be increased.
3. Can isolate and prioritize communication between different computers.

BUSTOPOLOGY:

A bus topology is multipoint. Hereone long cable is act as a backbone to link all the devices are connected to the back bone by drop lines and taps. A drop line is the connection between the devices and the cable. A tap is the splice into the main cable or puncture the sheathing.



Advantages:

1. Ease of installation.
2. Less cabling.

Disadvantages:

1. Difficult reconfiguration and fault isolation.
2. Difficult to add new devices.
3. Signal reflection at top can degradation in quality
4. If any fault in backbone can stops all transmission

Ring topology

Each node is connected to exactly two other nodes, forming a ring. Can be visualized as a circular configuration. Requires at least three nodes

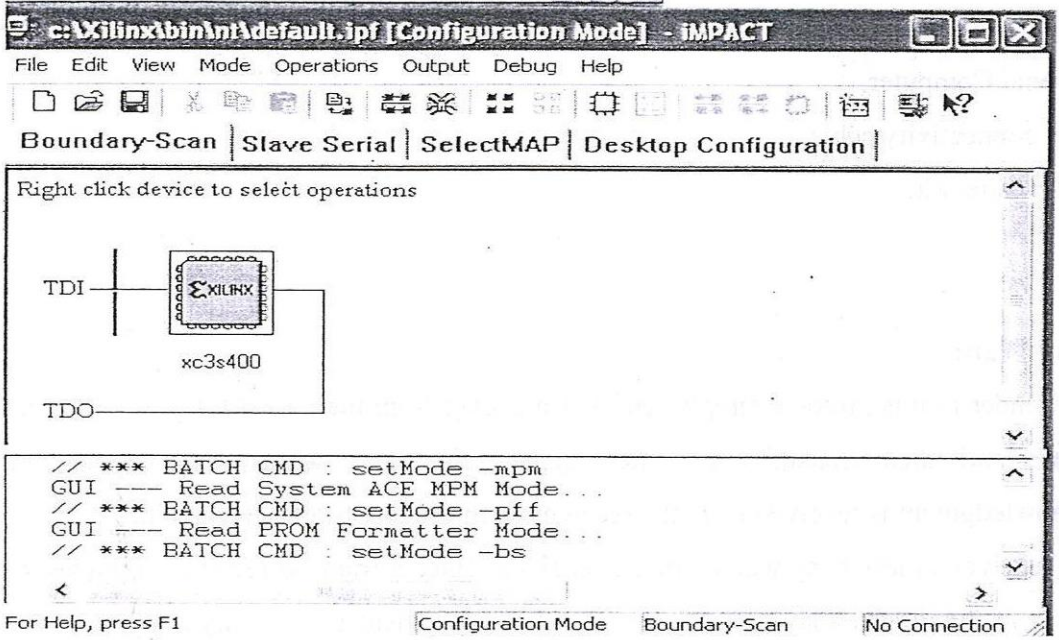
Advantages:

1. Easy to install.
2. Easy to reconfigure.
3. Fault identification is easy.

Disadvantages:

1. Unidirectional traffic.
2. Break in a single ring can break entire network.


XILINX®
DESIGN SOLUTIONS
iMPACT
Release Version: 7.1.02i
Application Version: H.40
Registration ID: UNKNOWN
Copyright (c) 1995-2005 Xilinx, Inc.
All rights reserved.


c:\Xilinx\bin\nt\default.ipf [Configuration Mode] - iMPACT
File Edit View Mode Operations Output Debug Help
Boundary-Scan Slave Serial SelectMAP Desktop Configuration
Right click device to select operations
TDI
XILINX
xc3s400
TDO
// *** BATCH CMD : setMode -mpm
GUI --- Read System ACE MPM Mode...
// *** BATCH CMD : setMode -pff
GUI --- Read PROM Formatter Mode...
// *** BATCH CMD : setMode -bs
For Help, press F1
Configuration Mode
Boundary-Scan
No Connection

PROCEDURE

Downloading "ARM"

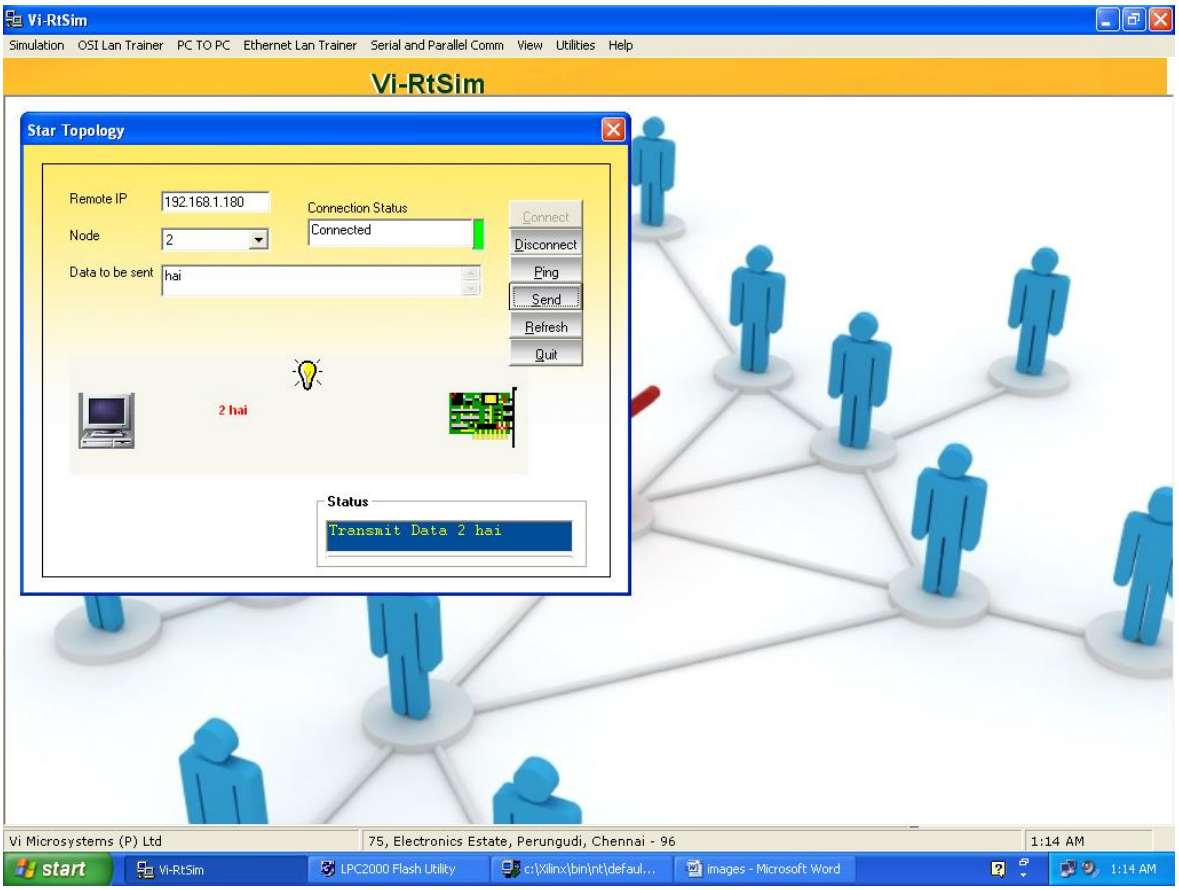
- Open VI-RTSIM software from desktop
- Select the ARM Downloader from UTILITIES menu bar to download the ARM.
- Click Read device ID in the LPC 2000 flash UTILITIES to read the LAN trainer and reset the board at the programming mode and then click OK.
- Select the TOKENBUS.HEX ARM program to be downloaded to the trainer kit.
- Click "Upload to flash" button to upload the program to the LAN trainer.
- After downloading the program we can get the "Program uploaded successfully" message.

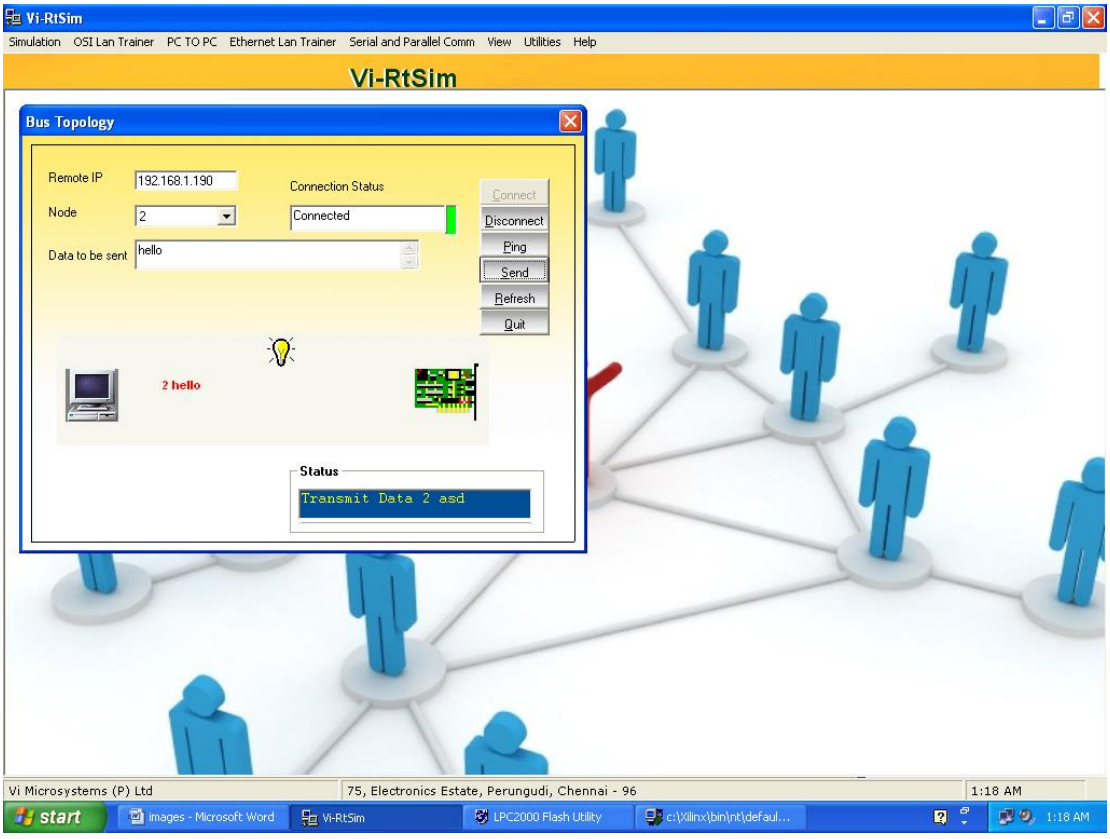
Downloading "FPGA"

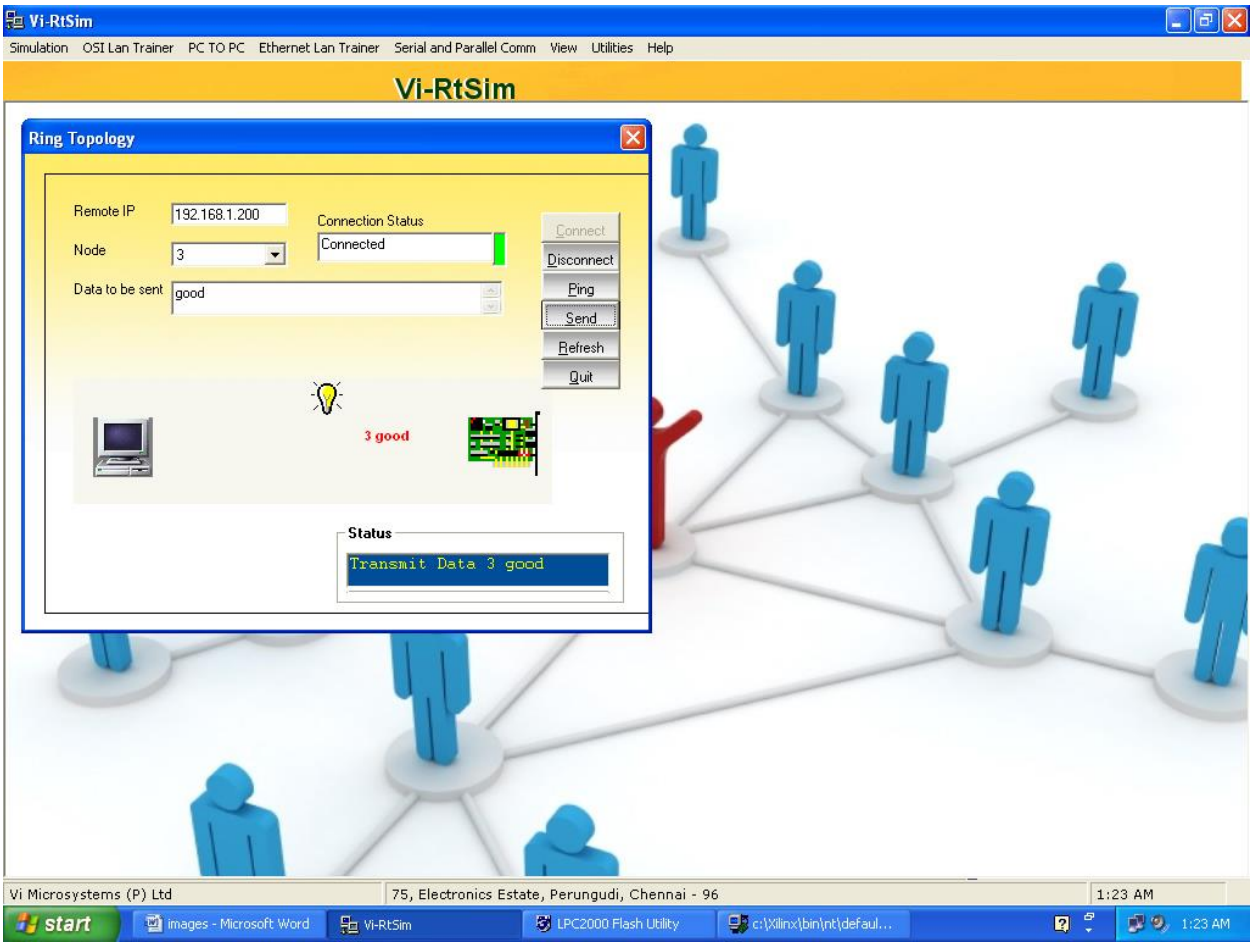
- Now open the software "Impact" to download the FPGA.]
- Select "Create a new project" and clock on "OK".
- Select "configure devices using boundary scan (J-TAG) and click "finish" button.
- Select the TOKENBUS BIT FPGA program to be downloaded to the trainer kit.
- Right click on the device and "program" to download FPGA program.
- After downloading the program we can get the "program successfully" message.

Execution of network topologies program:

- Open VI-RTSIM software from desktop.
- Select network topologies such as star,bus and ring from OSI LAN Trainer menu bar.
- Enter the Parameter.
- At default setting press connect button to conform whether the connection is established or not. After connection is successfully established press start button.
- Enter the data on the field.
- After data field is entered, Sender send the data to receiver.


Vi-RtSim
Simulation OSI Lan Trainer PC TO PC Ethernet Lan Trainer Serial and Parallel Comm View Utilities Help
Vi-RtSim
Star Topology
Remote IP 192.168.1.180
Node 2
Data to be sent hai
Connection Status
Connected
Connect
Disconnect
Ping
Send
Refresh
Quit
2 hai
Status
Transmit Data 2 hai
Vi Microsystems (P) Ltd
75, Electronics Estate, Perungudi, Chennai - 96
1:14 AM
start
Vi-RtSim
LPC2000 Flash Utility
c:\Xilinx\bin\nt\defaul...
images - Microsoft Word
1:14 AM

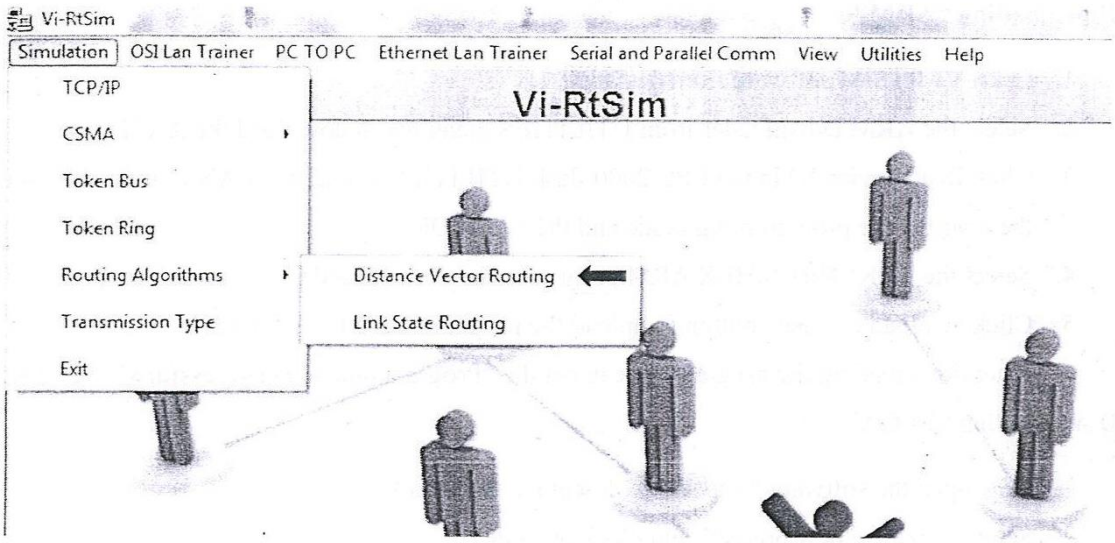

Vi-RtSim
Simulation OSI Lan Trainer PC TO PC Ethernet Lan Trainer Serial and Parallel Comm View Utilities Help
Vi-RtSim
Bus Topology
Remote IP 192.168.1.190
Node 2
Data to be sent hello
Connection Status
Connected
Connect
Disconnect
Ping
Send
Refresh
Quit
2 hello
Status
Transmit Data 2 asd
Vi Microsystems (P) Ltd
75, Electronics Estate, Perungudi, Chennai - 96
1:18 AM
start
images - Microsoft Word
Vi-RtSim
LPC2000 Flash Utility
c:\Xilinx\bin\nt\defaul...
1:18 AM

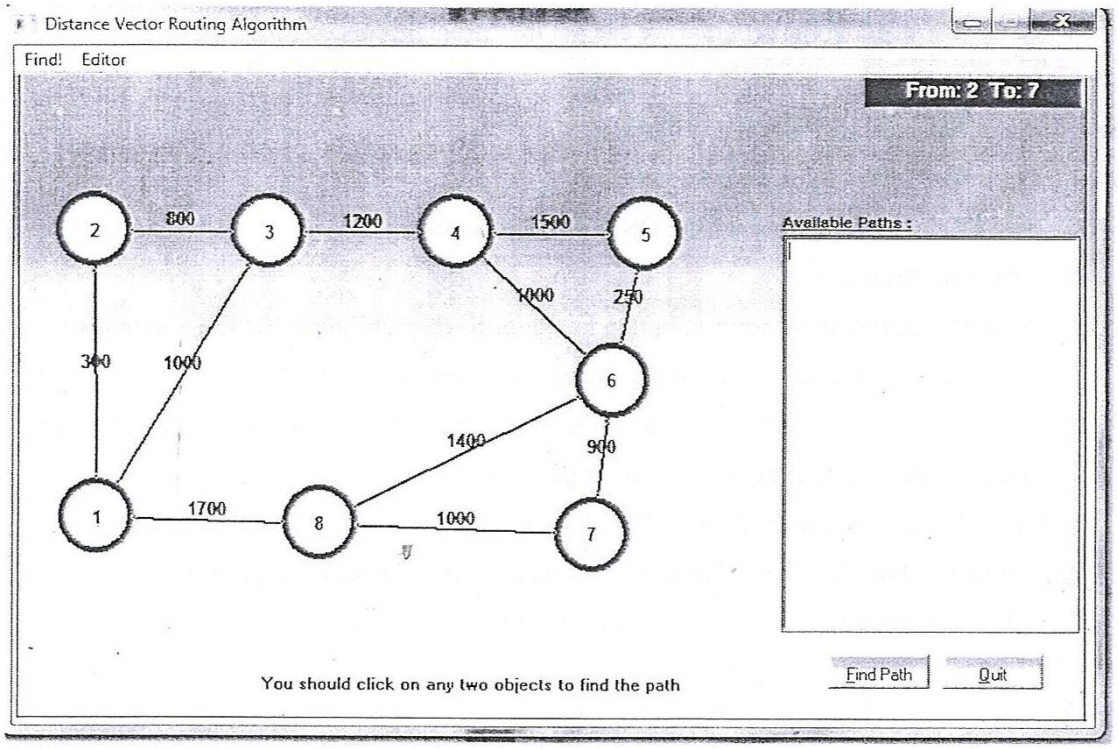


RESULT:

Thus the network topologies such as star, bus and ring have been implemented and its performance is analyzed.

IMPLEMENTATION OF DISTANCE VECTOR ROUTING ALGORITHM





Ex No: 9. Implementation of Distance – Vector Routing Algorithm

AIM:

To implement the Distance – Vector Routing Algorithm

APPARATUS REQUIRED:

1. VI-RTSIM software.
2. Personal computer.

THEORY:

Distance Vector Algorithm:

- A Distance vector routing, each router periodically share its knowledge about the entire network with it's neighbors.
- The three keys to under this algorithm are
    1. Knowledge about the whole network.
    2. Routing only to neighbor.
    3. Information sharing at regular intervals.

Knowledge about the whole work:

- Each router shares its knowledge about entire network. It sends all of its collected knowledge about the network to its neighbors.

Routing only to neighbor:

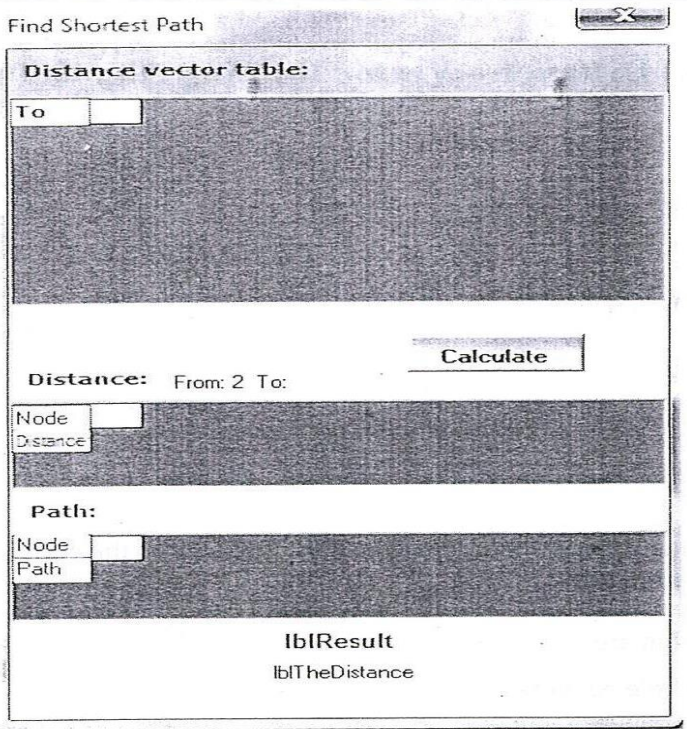
- Each router periodically sends its knowledge about the network only to those routers to which it has direct links. It sends whatever knowledge it has.

Information sharing at regular intervals:

- The every 30 seconds, each router sends its information about the whole network to its neighbors.

Sharing Information:

- LAN's are connected by router, represented by the assuming A, B, C, D, E and F.
- Distance vector routing simplifies the routing process by assuming a lost of one unit for every link.
- The efficiency of transmission is a function only of the number of links required to reach a destination. In this, the cost on hop count.



Find Shortest Path

Distance vector table:

To	1	2	3	4	5	6	7	8
1	0	300	1000	0	0	0	0	1700
2	300	0	800	0	0	0	0	0
3	1000	800	0	1200	0	0	0	0
4	0	0	1200	0	1500	1000	0	0
5	0	0	0	1500	0	250	0	0
6	0	0	0	1000	250	0	900	1400

Calculate

Distance: From: 2 To:

Node	1	2	3	4	5	6	7	8
Distance	300	0	800	2000	3250	3000	3000	2000

Path:

Node	1	2	3	4	5	6	7	8
Path	2	0	2	3	6	4	8	1

The path is: 7 <- 8 <- 1 <- 2

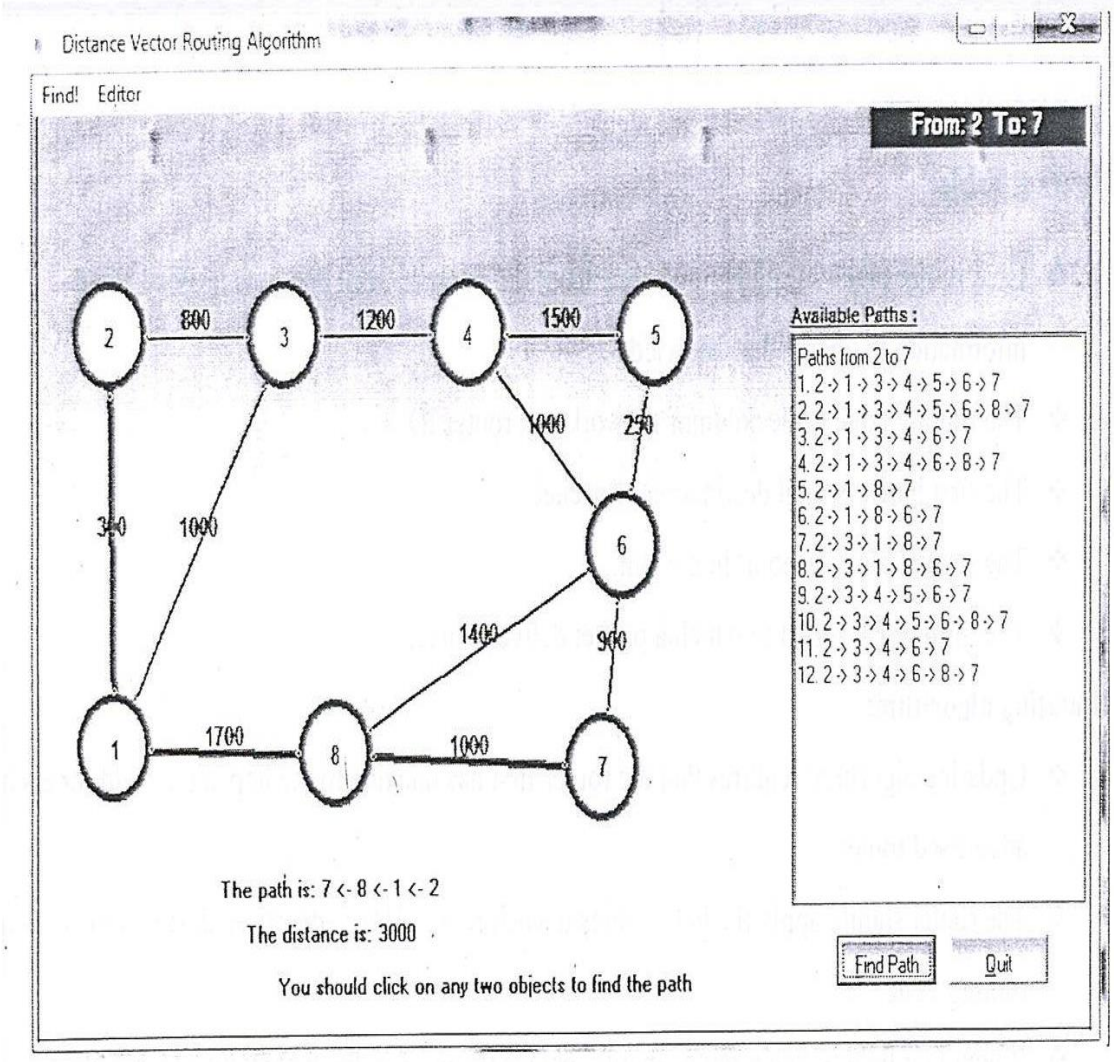
The distance is: 3000

Routing Table:

- Each router gets its initial knowledge about the internet work and how it uses shared information to update that knowledge.
- The routing table has e columns network lost router ID.
- The first block is final destination of packet.
- The second block is no of hop count.
- The third block is that to which a packet delivers must.

Updating algorithm:

- Updating algorithm requires that the router first has one hop to the hop count field for each advertised router.
- The router should apply the below rules to each router, if the advertised destination is not in routing table
- If next hop field is same, router should replace the entry in the table with advertised one.
- If next hop field is same, router should replace the entry in the table with advertised one.
- .If next hop field is not the same, advertised hop count is smaller than the one in the table, the router should replace the entry in the table with new one.
- IF advertised hop count is not smaller, the router should do no routing.


Distance Vector Routing Algorithm
Find! Editor
From: 2 To: 7
2
3
4
5
6
1
8
7
800
1200
1500
1000
250
1000
1400
900
1700
1000
Available Paths :
Paths from 2 to 7
1. 2 -> 1 -> 3 -> 4 -> 5 -> 6 -> 7
2. 2 -> 1 -> 3 -> 4 -> 5 -> 6 -> 8 -> 7
3. 2 -> 1 -> 3 -> 4 -> 6 -> 7
4. 2 -> 1 -> 3 -> 4 -> 6 -> 8 -> 7
5. 2 -> 1 -> 8 -> 7
6. 2 -> 1 -> 8 -> 6 -> 7
7. 2 -> 3 -> 1 -> 8 -> 7
8. 2 -> 3 -> 1 -> 8 -> 6 -> 7
9. 2 -> 3 -> 4 -> 5 -> 6 -> 7
10. 2 -> 3 -> 4 -> 5 -> 6 -> 8 -> 7
11. 2 -> 3 -> 4 -> 6 -> 7
12. 2 -> 3 -> 4 -> 6 -> 8 -> 7
The path is: 7 <- 8 <- 1 <- 2
The distance is: 3000
You should click on any two objects to find the path
Find Path
Quit

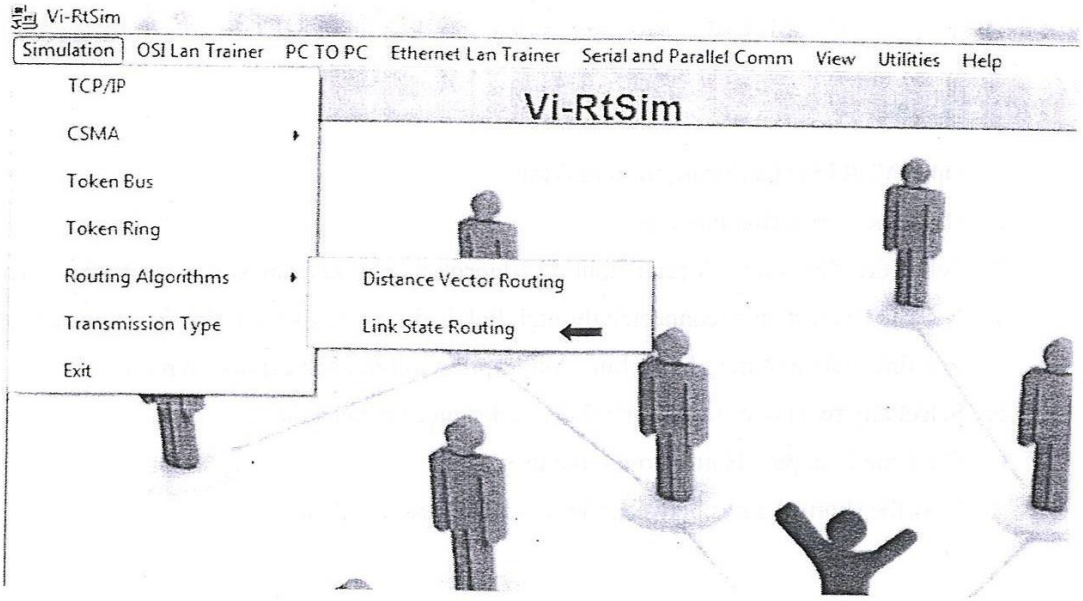
PROCEDURE

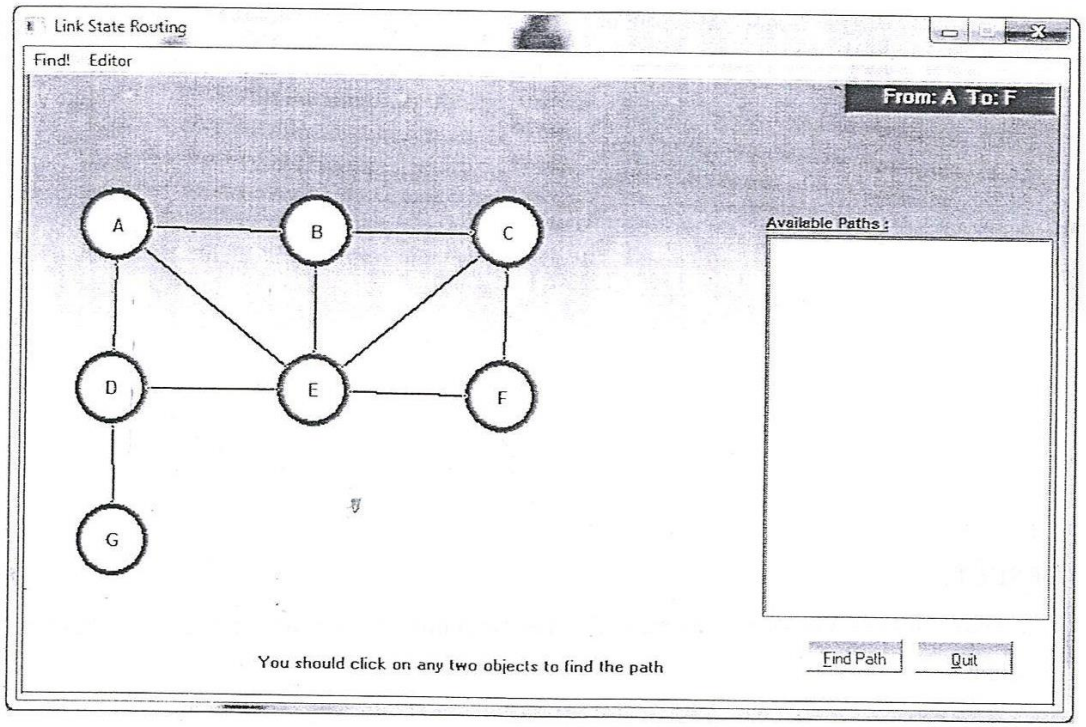
1. Open VI-RTSIM software from desktop
2. Click the Simulation menu bar
3. Select the “Distance – Vector Routing Algorithm” option from Routing algorithm menu bar.
4. Network with routers connected through link is drawn by using option in editor(add router, join link, delete router, delete link, Add caption to link, add caption to router)
5. Select any two nodes to find the shortest distance between them.
6. Click the Find path Button to run the program.
7. Now the shortest paths between the two nodes are calculated.

RESULT:

Thus Distance Vector routing algorithm has been implemented and shortest-path has been circulated.

IMPLEMENTATION OF LINK STATE ROUTING ALGORITHM





Ex No: 10. Implementation of Link State Routing Algorithm

AIM:

To implement the Link State Routing Algorithm

APPARATUS REQUIRED:

1. VI-RTSIM software.
2. Personal computer.

THEORY:

LinkState Vector Algorithm:

- In Link state routing, each router share its information of its neighbors with every other router in the inter-network.

Knowledge about the neighborhood:

- Instead of sending its entire routing table, a router sends information about its neighborhood only.

To all router:

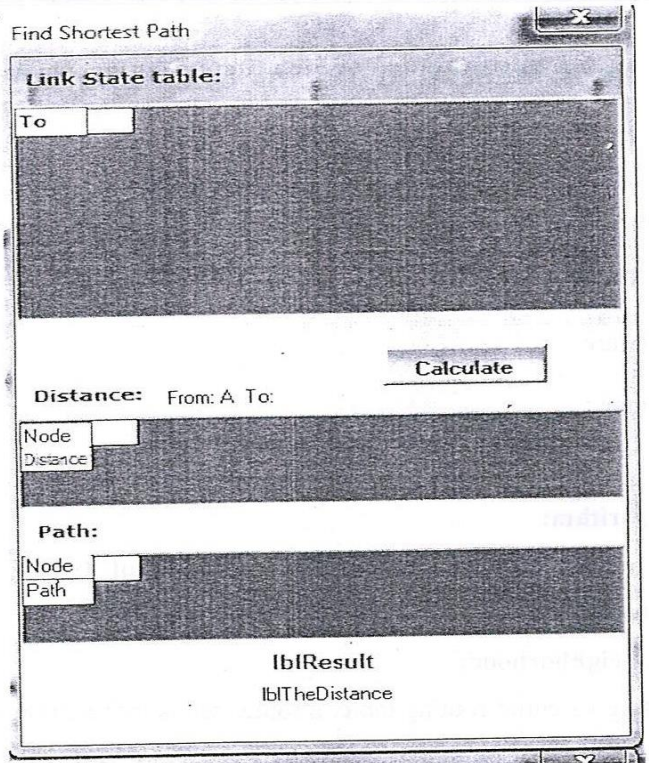
- Each router send this information to every other router on the internetworking, not just to its neighbors.
- If s does so by a process called “flooding” it means that a router sends its information.

Information sharing when there is a Change:

- Each router sends out information about the neighbors when there is a change.

Information sharing:

- Link state routing process use the same internet work as distance vector algorithm.
- Here each other sends its knowledge about is neighbors to every other router in the internet work.
- Cost is applied only by routers and not by any other station on a network, if cost was added by every station, instead of by routers alone, it would accumulate unpredictably.
- Cost is applied as a packet leaves the router rather then as if enters. Most networks are broadcast networks. When a packet is in network every station, including the router, can pick it up, we cannot assign any cost to a packet.



Find Shortest Path

Link State table:

To	A	B	C	D	E	F	G
A	0	1	0	1	1	0	0
B	1	0	1	0	1	0	0
C	0	1	0	0	1	1	0
D	1	0	0	0	1	0	1
E	1	1	1	1	0	1	0
F	0	0	1	0	1	0	0
G	0	0	0	1	0	0	0

Calculate

Distance: From: A To:

Node	A	B	C	D	E	F	G
Distance	0	1	2	1	1	2	2

Path:

Node	A	B	C	D	E	F	G
Path	0	1	2	1	1	5	4

The path is: F <- E <- A

The distance is: 2

Link state packet:

- When a router floods the network with information about its neighborhood, it is said to be advertising. The basis of this advertising is a short packet called a link state packet (LSP).

Advertiser	Network	Cost	Neighbor

Getting information about neighbors:

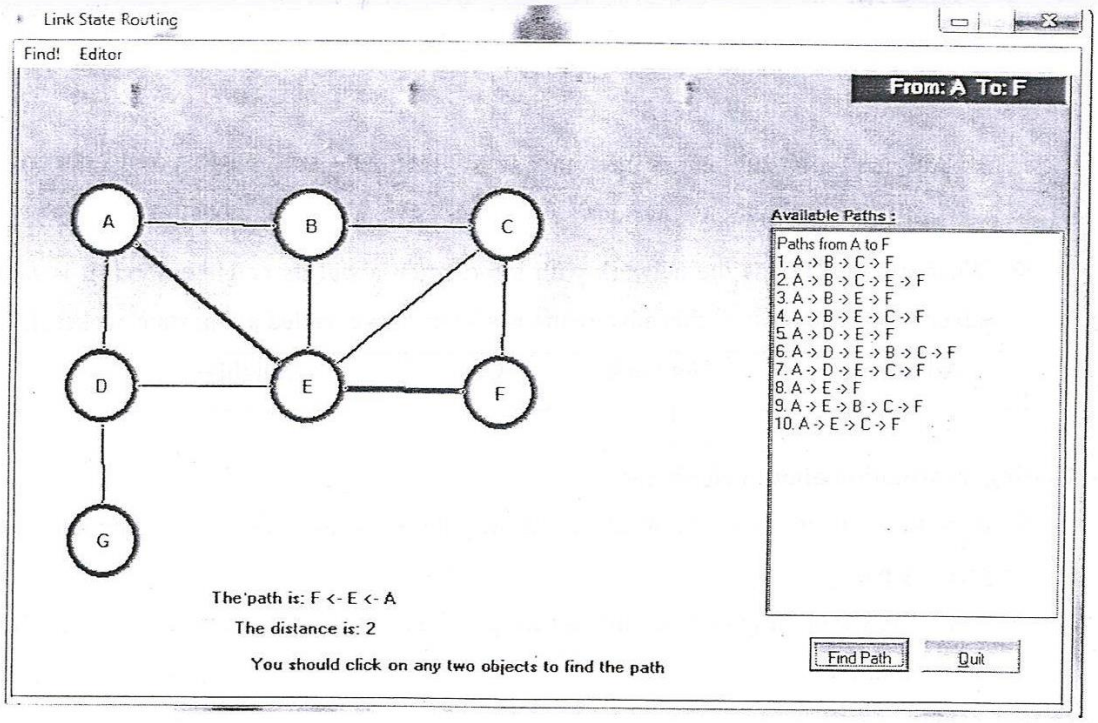
- A router gets its information about its neighbors by periodically sending them a short greeting packet.
- If the neighbor responds to the greeting as expected, it is assumed to be alive and functioning.

Initialization:

- Imagine that all routers in our sample internet work come up at the same time.
- Each router sends a greeting packet to its neighbors to find out the state of each link.

Link – State Database:

- Every router every LSP and puts the information into a link-state database.
- Because every router receives the same LSPs every router builds the same database.
- It stores this database on its disk and uses it to calculate its routing table. If a router is added to be deleted from the system, the whole database must be shared for fast updating.


Link State Routing
Find! Editor
From: A To: F
A
B
C
D
E
F
G
Available Paths :
Paths from A to F
1. A -> B -> C -> F
2. A -> B -> C -> E -> F
3. A -> B -> E -> F
4. A -> B -> E -> C -> F
5. A -> D -> E -> F
6. A -> D -> E -> B -> C -> F
7. A -> D -> E -> C -> F
8. A -> E -> F
9. A -> E -> B -> C -> F
10. A -> E -> C -> F
The path is: F <- E <- A
The distance is: 2
You should click on any two objects to find the path
Find Path
Quit

PROCEDURE

1. Open VI-RTSIM software from desktop
2. Click the Simulation menu bar
3. Select the “Link State Routing Algorithm” option from Routing algorithm menu bar.
4. Network with routers connected through link is drawn by using option in editor(add router, join link, delete router, delete link, Add caption to link, add caption to router)
5. Select any two nodes to find the shortest distance between them.
6. Click the Find path Button to run the program.
7. Now the shortest paths between the two nodes using link state routing algorithm was calculated.

RESULT:

Thus Link-State routing algorithm has been implemented and shortest-path has been circulated

Ex.No:11 Study of network simulation (NS) and Simulation of congestion control algorithm using NS

Aim:

To study about the network simulation NS OPNET and to configure and analyze performance of the congestion control algorithm in TCP using OPNET.

Apparatus Required:

- Personal Computer
- Unio
- OPNET software

Theory:

OPNET is a higher level client based network level simulation at separate packet level originally built for simulation for fixed network .OPNET contain a huge lib of accurate model of commercially available fixed network hardware and protocol how a day.The possibilities for wireless network simulation are also very circle accurate radio transmission pipeline stage for modelling of physical layer.Simulate has a lot of potential but there is exists typically a lock of recent wireless system much.

Structure of OPNET:

OPNET consists of high level user interface which is constructed from C++ source code blocks with a huge liberly of OPNET specific.Network and sub networks ,network topologies geographical co-ordinates mobility.

Various tools of OPNET:

- Network model editor
- Node modeleditor
- Process model editor
- Packet format editor
- OPNET Simulation virvers

Simulation of congestion control algorithm procedure create a project

- ->start reversed module academics editions choose new from file menu.
- ->select project and click of name project top scenario no drop click ok
- ->close pallete
- ->rename objects you added as shown and then save your project

Configure the west socket:

Double click on west subnet mode you get an empty workspace indicating that subnet contain object.

->open object parameter & make source that internet toolbar item selected from ball down.

->Add following item to subnet workspace one scheme serves one internet 4 ships at 100 least.

->Close the pallete to rename object as shown.

->Server west ->route west

->right click on server west node -> edit attributes.

Convert to subnet of IP closed:

->Open object palleteusing two PPP-DS3 bidirectional links connect to last subnet IP & west subnet IP word.

->close pallete.

View the Reset:

Switch to the drop no fast scenario& click

Various menu result in view menu.

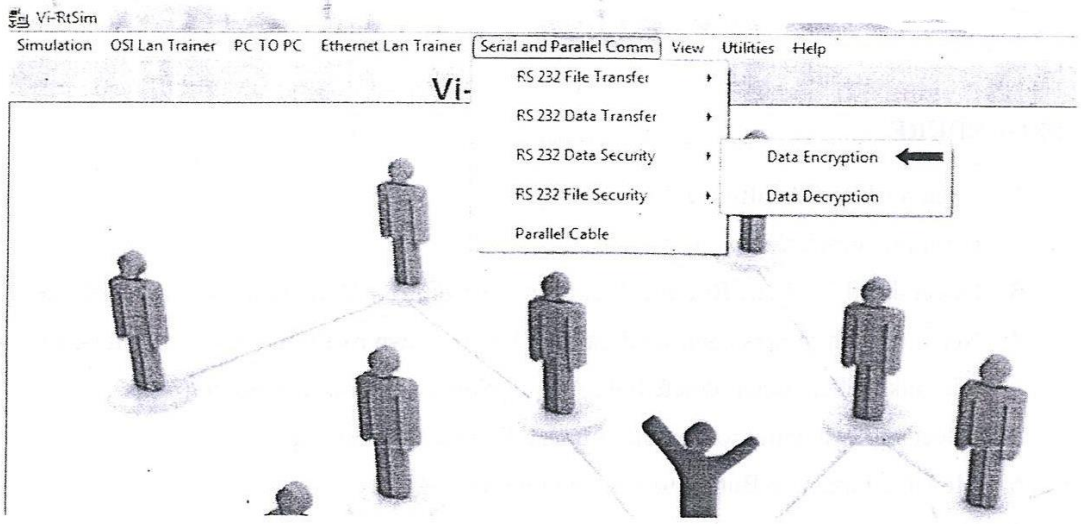
To zoom in an details in graph click and drag on name.

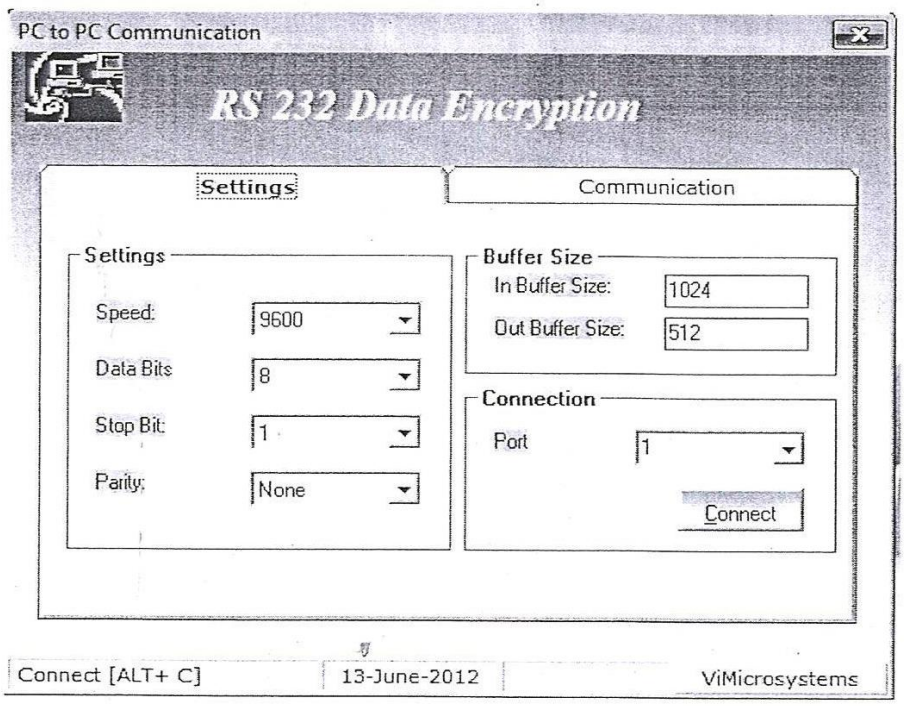
Graph should noted down to resemble fral with every drop close view result dialog box.

Result:

Thus the congestion algorithm in top has been simulated successfully using OPNET.

Implementation of Data Encryption and Decryption.





Ex No: 12. Implementation of Data Encryption and Decryption.

AIM:

To encrypt and decrypt the file while transferring file from one PC to another Pc serially using RS232 cable.

APPARATUS REQUIRED:

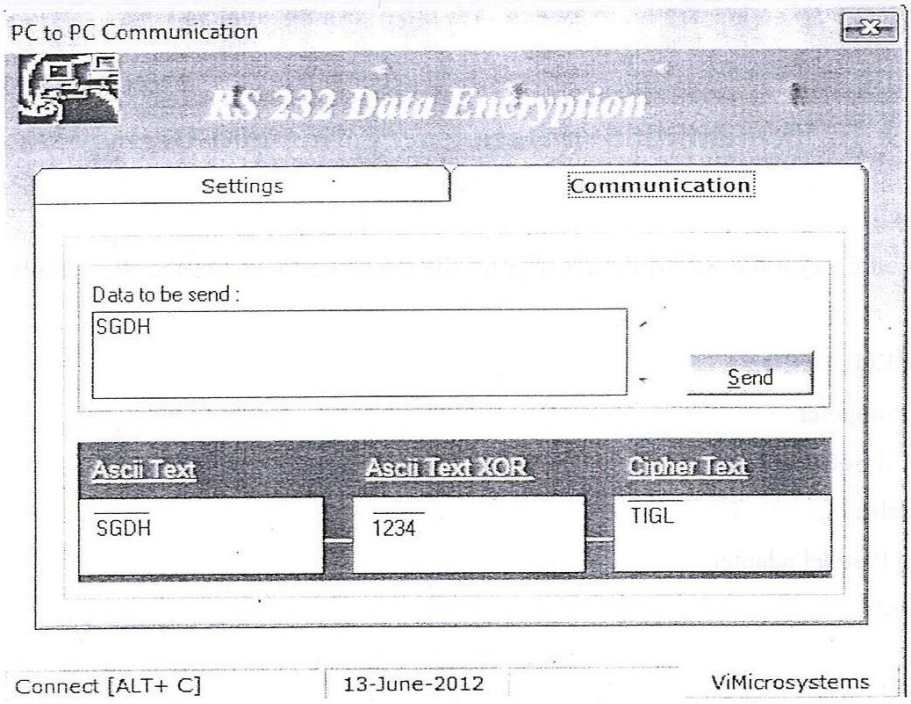
1. Personal computer
2. VI-RT SIM Software
3. RS 232 Cable
4. Serial and Parallel adapter
5. Patch chords.

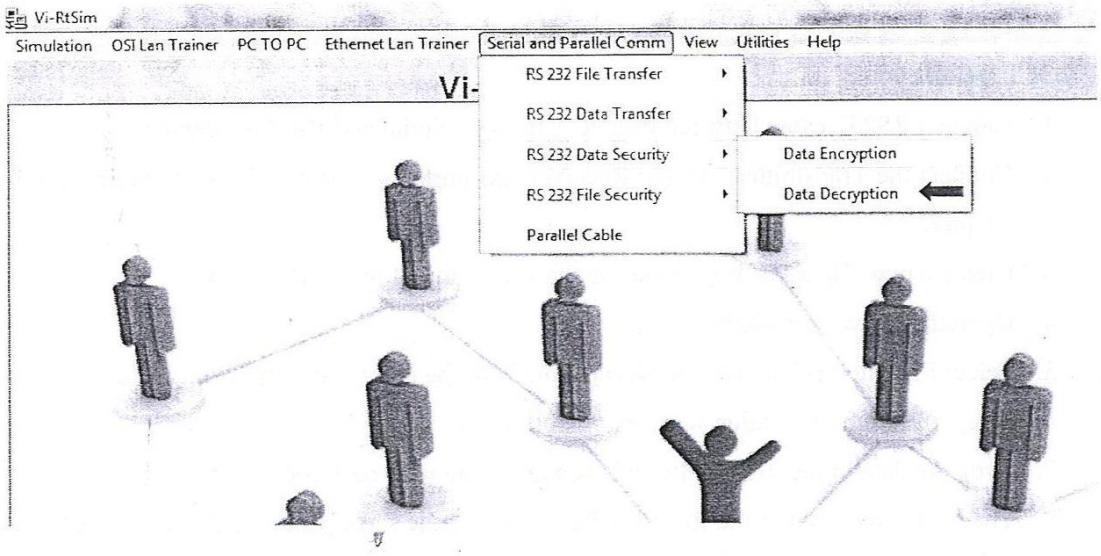
THEORY:

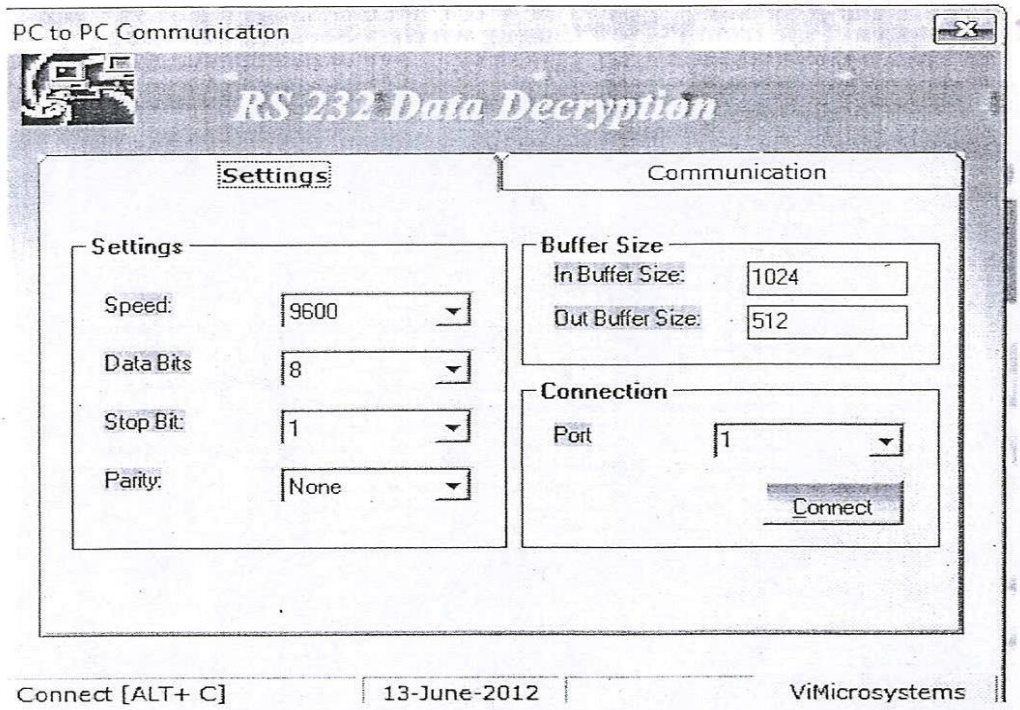
The data are sending from transmitter to the receiver normally in communication. For security purpose the data are converted into any other codings (Encryption) and sent, then at the receiver end the data will be retrieved by decoding the received data (Decryption).

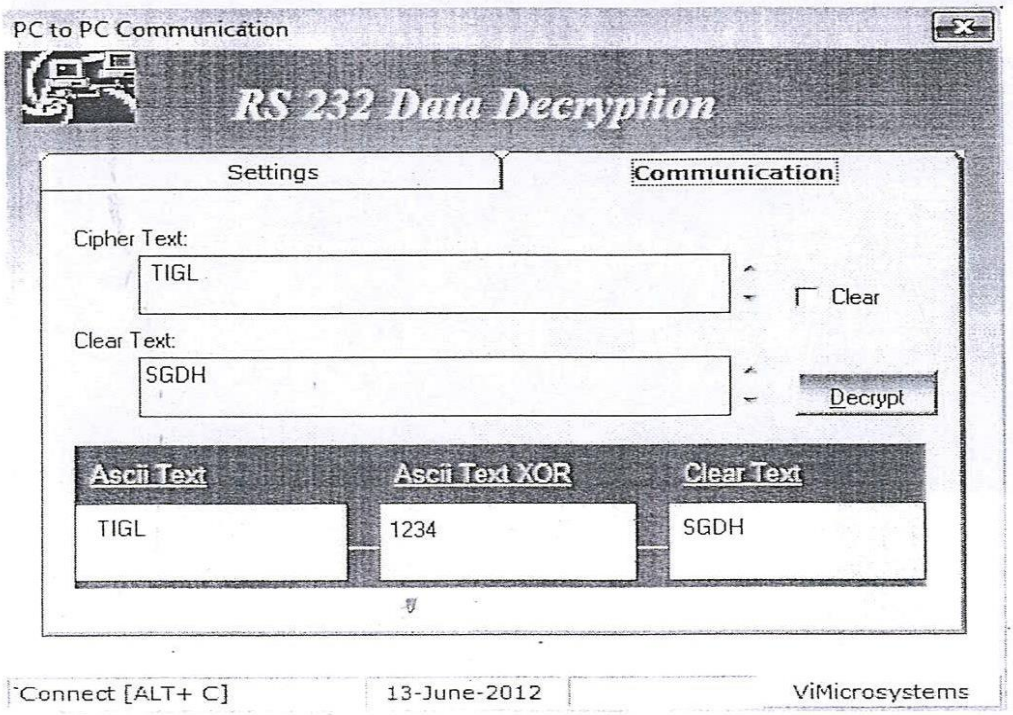
PROCEDURE:

1. Connect RS232 cable between two PC's through Serial and Parallel adapter.
2. Connect the Transmitter Txr and Receiver Rxr ports using patch chords in Serial and Parallel adapter.
3. Create a new file in the note pad save the file in any of the C, D, E drives.
4. Open the software Vi-RtSim.
5. Select File encryption and File Decryption from the menu bar in the both PC's.
6. Select the Communication window on both PC's
7. Type the data in the Transmitter PC, send the data with pass code.
8. The data is received in the Receiver PC and using the same passcode the data is decrypted.


PC to PC Communication
RS 232 Data Encryption
Settings
Communication
Data to be send :
SGDH
Send
Ascii Text
Ascii Text XOR
Cipher Text
SGDH
1234
TIGL
Connect [ALT+ C]
13-June-2012
ViMicrosystems


Vi-RtSim
Simulation
OSI Lan Trainer
PC TO PC
Ethernet Lan Trainer
Serial and Parallel Comm
View
Utilities
Help
RS 232 File Transfer
RS 232 Data Transfer
RS 232 Data Security
RS 232 File Security
Parallel Cable
Data Encryption
Data Decryption



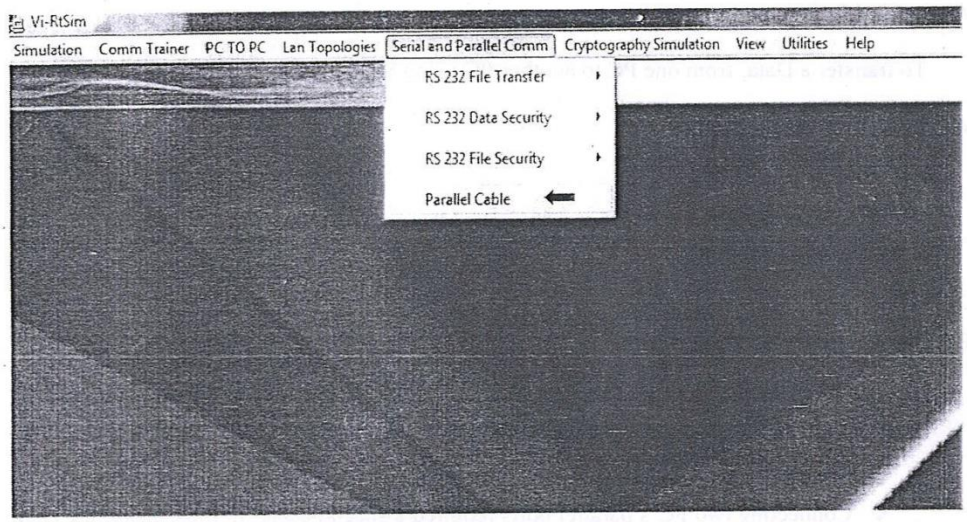


Result:

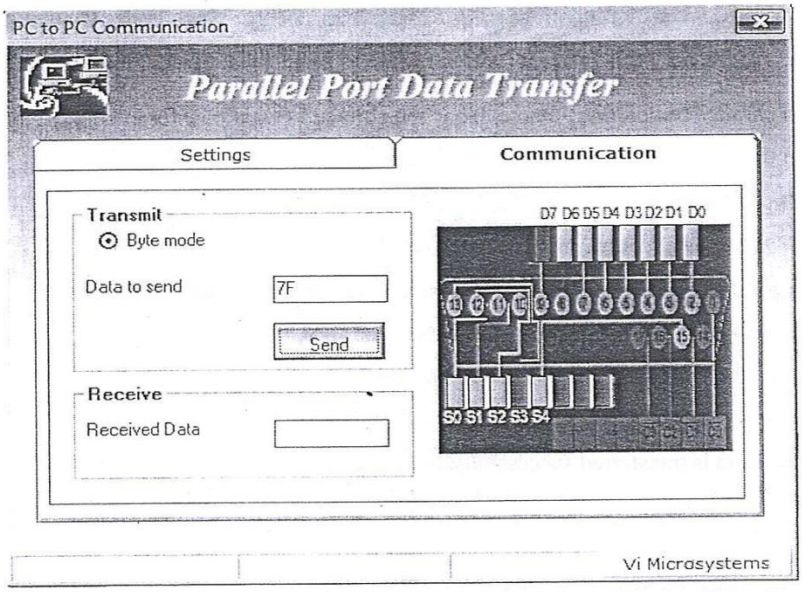
Thus the encryption and decryption of the data while transferring file from one PC to another PC serially using RS232 cable is been performed successfully

. PC to PC **parallel** Communication Using 8-Bit Parallel Cable

MENU BAR



TRANSMITTER SIDE



Ex.No:13 PC to Pc Parallel Communication Using 8-Bit Parallel Cable

AIM:

To transfer a Data, from one PC to another PC using 8-Bit Parallel Cable.

APPARATUS REQUIRED:

1. VI-RTSIM software
2. Personal Computer
3. 8-Bit Parallel Cable

THEORY

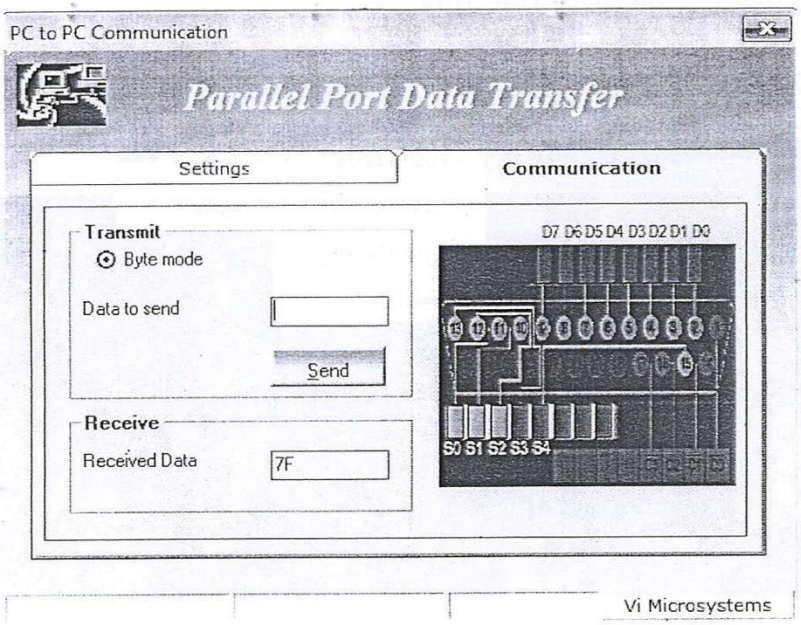
Parallel Communication:

- Binary data consisting of 1's and 0's may be organized into groups of 'n' bits each. A computer produced and consumes data in group of bits much as we conceive of and we spoke language in the form of words than letters.
- Connecting two PC's parallel ports required a special cable. In most cases, the cable will have a 25 pin Male D-sub connector on each end. An ordinary male-to-male D-sub cable won't do, though because the wires on a PC-to-PC parallel cable don't connector straight across, pin for pin.
- In a Pc-to-Parallel link, the host control outputs connect to input on the peripheral, and the host status inputs connect to output on the peripheral. If you connect the parallel ports two PS's straight together, you end up with inputs, and outputs connected to outputs.

PROCEDURE

1. Open VI-RTSIM software from desktop
2. In parallel communication menu bar select parallel Data transfer.
3. In the $T_X$window select "communication".
4. Click the "connect" to connect two PC's.
5. Type the data to Transmit and send the data
6. The Data is transferred successfully from one PC to another.

RECEIVER SIDE

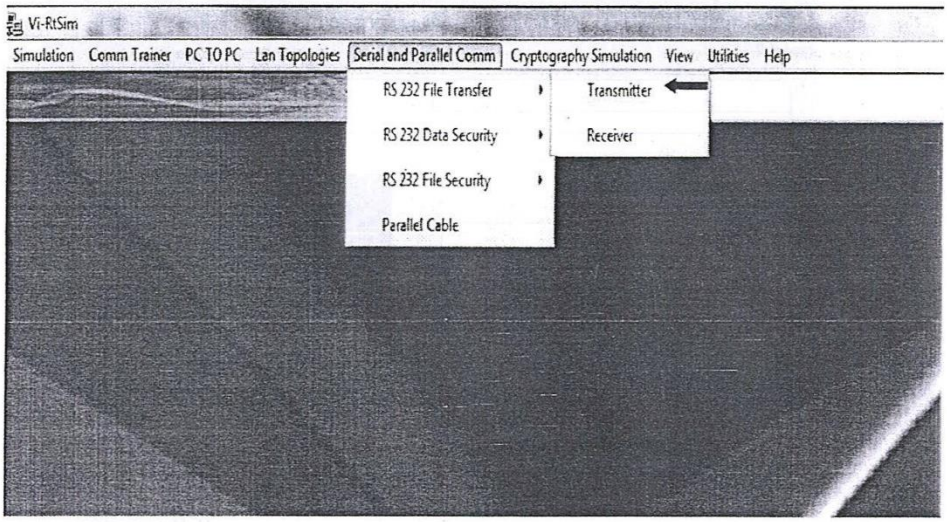


RESULT:

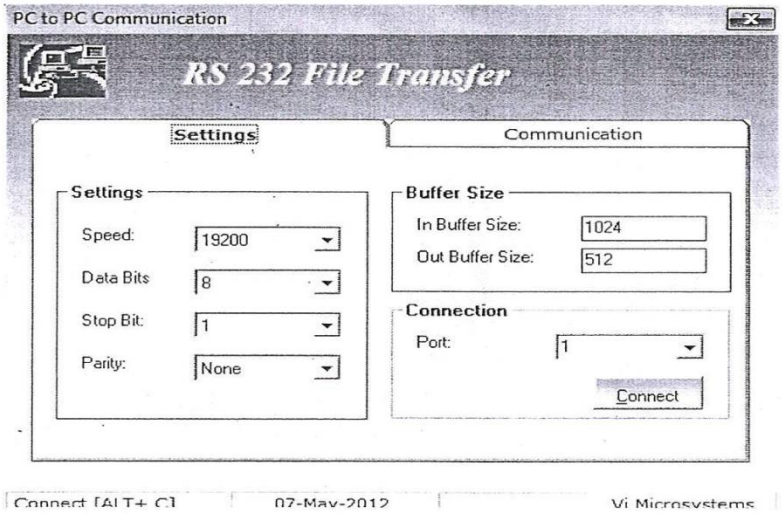
Thus, data is transfer from one PC to another PC by using 8-bit parallel cable communication.

PC to PC Serial Communication Using RS – 232 cables

MENU BAR (TRANSMITTER SIDE)



TRANSMITTER SIDE



Ex. No:14 PC to PC Serial Communication Using RS – 232 cables

AIM:

To transfer a Data, File from one PC to another PC using RS – 232 cable.

APPARATUS REQUIRED:

1. VI-RTSIM software
2. Personal Computer
3. RS – 232 Cable
4. Two Patch Cords

THEORY:

Serial Communication:

- In serial communication, one bit follows another, So we need only one communication channel rather than to transmit data between communication devices.
- In advantages of serial communication over parallel is that with only one communication channel, Serial communication reduces the cost of transmission over parallel by roughly a factor on "n"
- The transmission occurs in two ways and they are Synchronous and Asynchronous transmission.

Asynchronous:

- In this we send one start bit (0) at the beginning and one stop bit (1) at the end of each bit and a gap between each byte.

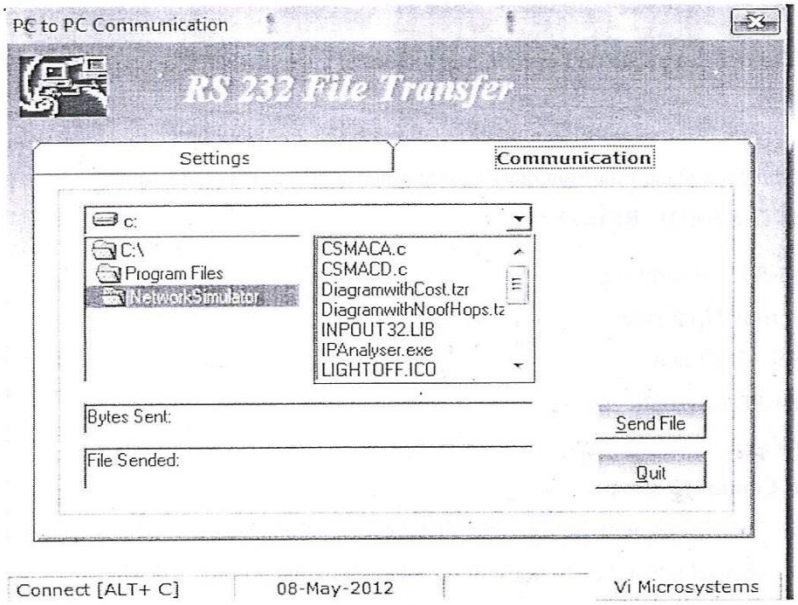
Synchronous:

- In this, we send bit one after other without start and stop bits or gaps. It is the responsibility of receiver to group the bits.

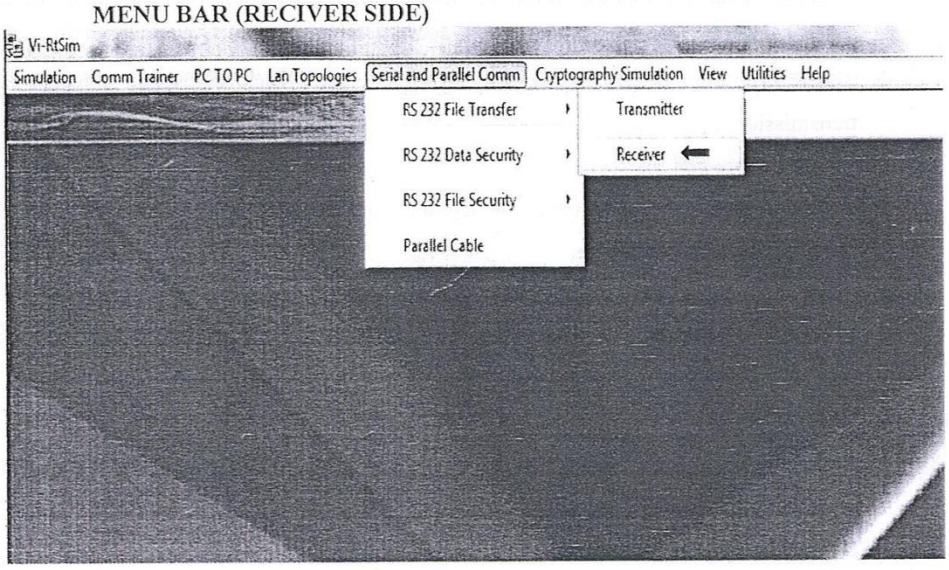
RS-232 Cable:

- Connecting two devices using Rs-232 cable is simplex, usually inputs are connected to input and outputs are connected to output.
- DTE: Data Terminal Equipment.
- DCE: Data Communication Equipment.
- This card contains two 9 pin D – Type connectors for serial communication between tow PC's TXD and RXD lines are terminated at PS/2 connector. CTS and RTS lines are internally connected.

SELECTING A FILE TO TRANSFER



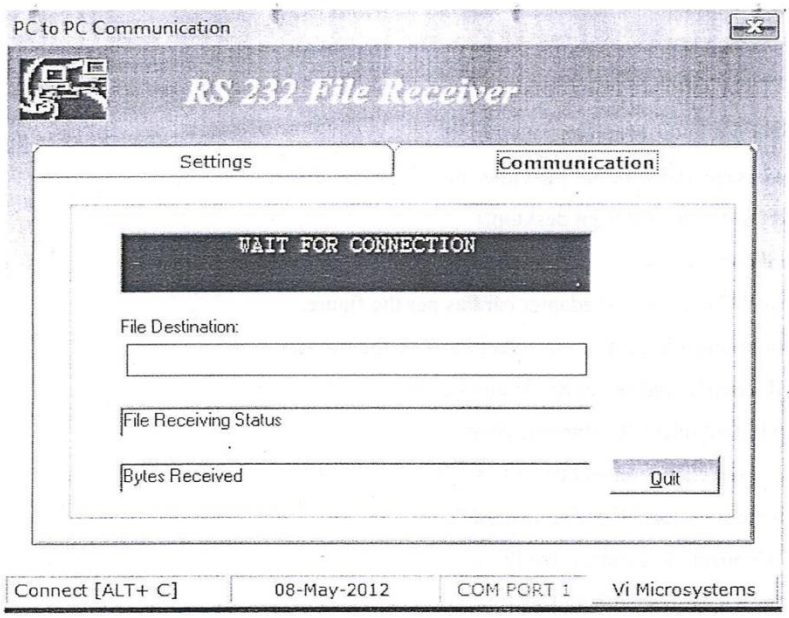
MENU BAR (RECIVER SIDE)



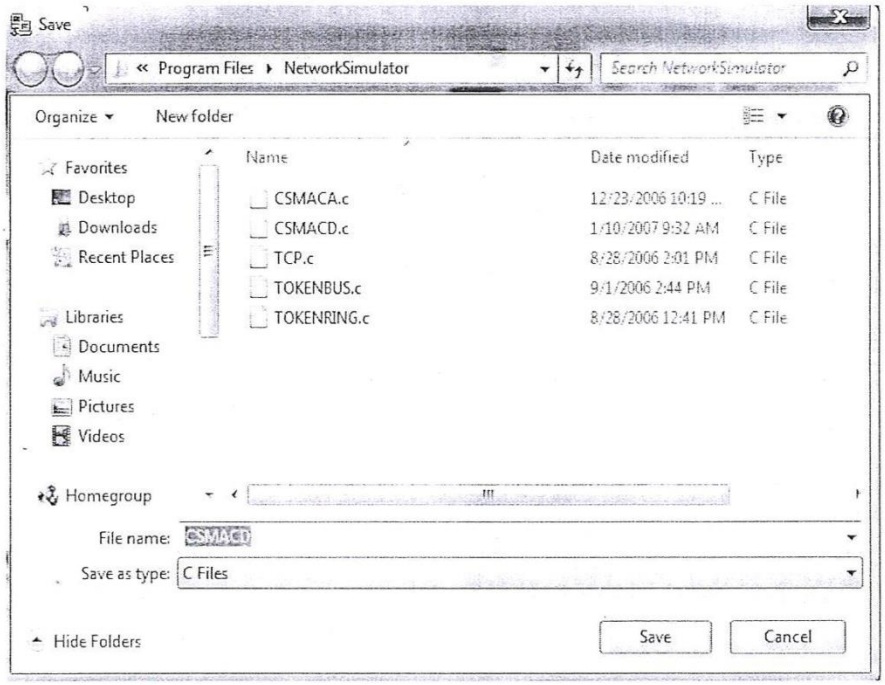
PROCEDURE – FILE TRANSFER

1. Create a new file in the notepad.
2. Type the message and save the file either in “C” or “D” drive.
3. Open VI-RTSM software from desktop.
4. Select serial communication from menu-bar.
5. Connect two PC using serial adapter card as per the figure.
6. In serial communication menu bar select RS-232 file transfer.
7. Select T_X or one PC and select R_X on another Pc.
8. In the Rxwindow select “Communication”.
9. Click the “Connect” to connect two PC’s
10. In the Rxwindow select “Communication”.
11. Click the “Connect” to connect two PC’s
12. Selecting the file to Transmit and send file.
13. The file will be received at the receiver PV, give the name for received file and save it in “C” or “D” drive.
14. The file is transferred successfully from one PC to another.

RECEIVER SIDE



FILE TRANSFERRED AND SAVED AT RECEIVER SIDE



RESULT:

Thus the file and data is transferred from one PC to another PC by serial communication using Rs-232 Cable.