**EXPERIMENT-1**

Microsoft Windows [Version 10.0.22000.493]

(c) Microsoft Corporation. All rights reserved.

C:\Users\91893>hostname

LAPTOP-A3T7UCMD

C:\Users\91893>ipconfig

Windows IP Configuration

Wireless LAN adapter Local Area Connection\* 11:

Media State . . . . . . . . . . . : Media disconnected

Connection-specific DNS Suffix . :

Wireless LAN adapter Local Area Connection\* 12:

Media State . . . . . . . . . . . : Media disconnected

Connection-specific DNS Suffix . :

Wireless LAN adapter Wi-Fi:

Connection-specific DNS Suffix . :

IPv6 Address. . . . . . . . . . . : 2401:4900:1c64:85b3:999a:bfb5:fae9:8e81

Temporary IPv6 Address. . . . . . : 2401:4900:1c64:85b3:bdf7:b9c3:7af4:2efd

Link-local IPv6 Address . . . . . : fe80::999a:bfb5:fae9:8e81%6

IPv4 Address. . . . . . . . . . . : 192.168.1.3

Subnet Mask . . . . . . . . . . . : 255.255.255.0

Default Gateway . . . . . . . . . : fe80::1%6

192.168.1.1

Ethernet adapter Bluetooth Network Connection:

Media State . . . . . . . . . . . : Media disconnected

Connection-specific DNS Suffix . :

C:\Users\91893>color help

Sets the default console foreground and background colors.

COLOR [attr]

attr Specifies color attribute of console output

Color attributes are specified by TWO hex digits -- the first

corresponds to the background; the second the foreground. Each digit

can be any of the following values:

0 = Black 8 = Gray

1 = Blue 9 = Light Blue

2 = Green A = Light Green

3 = Aqua B = Light Aqua

4 = Red C = Light Red

5 = Purple D = Light Purple

6 = Yellow E = Light Yellow

7 = White F = Bright White

If no argument is given, this command restores the color to what it was

when CMD.EXE started. This value either comes from the current console

window, the /T command line switch or from the DefaultColor registry

value.

The COLOR command sets ERRORLEVEL to 1 if an attempt is made to execute

the COLOR command with a foreground and background color that are the

same.

Example: "COLOR fc" produces light red on bright white

C:\Users\91893>ipconfig/all

Windows IP Configuration

Host Name . . . . . . . . . . . . : LAPTOP-A3T7UCMD

Primary Dns Suffix . . . . . . . :

Node Type . . . . . . . . . . . . : Hybrid

IP Routing Enabled. . . . . . . . : No

WINS Proxy Enabled. . . . . . . . : No

Wireless LAN adapter Local Area Connection\* 11:

Media State . . . . . . . . . . . : Media disconnected

Connection-specific DNS Suffix . :

Description . . . . . . . . . . . : Microsoft Wi-Fi Direct Virtual Adapter #3

Physical Address. . . . . . . . . : D8-3B-BF-5D-D1-81

DHCP Enabled. . . . . . . . . . . : Yes

Autoconfiguration Enabled . . . . : Yes

Wireless LAN adapter Local Area Connection\* 12:

Media State . . . . . . . . . . . : Media disconnected

Connection-specific DNS Suffix . :

Description . . . . . . . . . . . : Microsoft Wi-Fi Direct Virtual Adapter #4

Physical Address. . . . . . . . . : DA-3B-BF-5D-D1-80

DHCP Enabled. . . . . . . . . . . : Yes

Autoconfiguration Enabled . . . . : Yes

Wireless LAN adapter Wi-Fi:

Connection-specific DNS Suffix . :

Description . . . . . . . . . . . : Intel(R) Wireless-AC 9560 160MHz

Physical Address. . . . . . . . . : D8-3B-BF-5D-D1-80

DHCP Enabled. . . . . . . . . . . : Yes

Autoconfiguration Enabled . . . . : Yes

IPv6 Address. . . . . . . . . . . : 2401:4900:1c64:85b3:999a:bfb5:fae9:8e81(Preferred)

Temporary IPv6 Address. . . . . . : 2401:4900:1c64:85b3:bdf7:b9c3:7af4:2efd(Preferred)

Link-local IPv6 Address . . . . . : fe80::999a:bfb5:fae9:8e81%6(Preferred)

IPv4 Address. . . . . . . . . . . : 192.168.1.3(Preferred)

Subnet Mask . . . . . . . . . . . : 255.255.255.0

Lease Obtained. . . . . . . . . . : 27 February 2022 19:49:56

Lease Expires . . . . . . . . . . : 28 February 2022 22:30:14

Default Gateway . . . . . . . . . : fe80::1%6

192.168.1.1

DHCP Server . . . . . . . . . . . : 192.168.1.1

DHCPv6 IAID . . . . . . . . . . . : 282606527

DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-28-3D-B9-55-D8-3B-BF-5D-D1-80

DNS Servers . . . . . . . . . . . : 2401:4900:50:9::2a0

2401:4900:50:9::8

fe80::1%6

192.168.1.1

NetBIOS over Tcpip. . . . . . . . : Enabled

Ethernet adapter Bluetooth Network Connection:

Media State . . . . . . . . . . . : Media disconnected

Connection-specific DNS Suffix . :

Description . . . . . . . . . . . : Bluetooth Device (Personal Area Network)

Physical Address. . . . . . . . . : D8-3B-BF-5D-D1-84

DHCP Enabled. . . . . . . . . . . : Yes

Autoconfiguration Enabled . . . . : Yes

C:\Users\91893>nslookup

Default Server: UnKnown

Address: 2401:4900:50:9::2a0

> www.github.com

Server: UnKnown

Address: 2401:4900:50:9::2a0

Non-authoritative answer:

Name: github.com

Address: 13.234.176.102

Aliases: www.github.com

> www.google.com

Server: UnKnown

Address: 2401:4900:50:9::2a0

Non-authoritative answer:

Name: www.google.com

Addresses: 2404:6800:4007:81b::2004

142.250.182.68

> www.sarkariresult.com

Server: UnKnown

Address: 2401:4900:50:9::2a0

DNS request timed out.

timeout was 2 seconds.

Name: www.sarkariresult.com

Addresses: 2606:4700::6812:1385

2606:4700::6812:1285

> www.facebook.com

Server: UnKnown

Address: 2401:4900:50:9::2a0

Non-authoritative answer:

Name: star-mini.c10r.facebook.com

Addresses: 2a03:2880:f144:181:face:b00c:0:25de

157.240.239.35

Aliases: www.facebook.com

> www.twitter.com

Server: UnKnown

Address: 2401:4900:50:9::2a0

Non-authoritative answer:

Name: twitter.com

Address: 104.244.42.65

Aliases: www.twitter.com

C:\Users\91893>tracert

Usage: tracert [-d] [-h maximum\_hops] [-j host-list] [-w timeout]

[-R] [-S srcaddr] [-4] [-6] target\_name

Options:

-d Do not resolve addresses to hostnames.

-h maximum\_hops Maximum number of hops to search for target.

-j host-list Loose source route along host-list (IPv4-only).

-w timeout Wait timeout milliseconds for each reply.

-R Trace round-trip path (IPv6-only).

-S srcaddr Source address to use (IPv6-only).

-4 Force using IPv4.

-6 Force using IPv6.

C:\Users\91893>tracert www.google.com

Tracing route to www.google.com [2404:6800:4007:82c::2004]

over a maximum of 30 hops:

1 6 ms 1 ms 1 ms 2401:4900:1c64:85b3:1210:81ff:fefe:9186

2 47 ms 9 ms 8 ms 2401:4900:1c22:8fff::1

3 4 ms 6 ms 4 ms 2404:a800:1a00:801::109

4 8 ms 24 ms \* 2001:4860:1:1::1944

5 12 ms 13 ms 13 ms 2404:6800:8129::1

6 24 ms 19 ms 15 ms 2001:4860:0:1::53a8

7 16 ms 18 ms 9 ms 2001:4860:0:11dd::c

8 54 ms 54 ms 53 ms 2001:4860::9:4000:e50c

9 47 ms 50 ms 44 ms 2001:4860:0:1340::1

10 47 ms 43 ms 45 ms 2001:4860:0:1::5599

11 57 ms 51 ms 50 ms maa03s47-in-x04.1e100.net [2404:6800:4007:82c::2004]

Trace complete.

C:\Users\91893>ping www.google.com

Pinging www.google.com [2404:6800:4007:82c::2004] with 32 bytes of data:

Reply from 2404:6800:4007:82c::2004: time=54ms

Reply from 2404:6800:4007:82c::2004: time=52ms

Reply from 2404:6800:4007:82c::2004: time=69ms

Reply from 2404:6800:4007:82c::2004: time=50ms

Ping statistics for 2404:6800:4007:82c::2004:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 50ms, Maximum = 69ms, Average = 56ms

C:\Users\91893>netstat

Active Connections

Proto Local Address Foreign Address State

TCP 192.168.1.3:54171 20.197.71.89:https ESTABLISHED

TCP 192.168.1.3:54443 ec2-52-26-203-43:https ESTABLISHED

TCP 192.168.1.3:54467 52.109.124.51:https TIME\_WAIT

TCP 192.168.1.3:54469 52.109.124.51:https TIME\_WAIT

TCP [2401:4900:1c64:85b3:bdf7:b9c3:7af4:2efd]:54167 sd-in-f188:https ESTABLISHED

TCP [2401:4900:1c64:85b3:bdf7:b9c3:7af4:2efd]:54436 whatsapp-cdn6-shv-02-del1:https ESTABLISHED

C:\Users\91893>systeminfo

Host Name: LAPTOP-A3T7UCMD

OS Name: Microsoft Windows 11 Home Single Language

OS Version: 10.0.22000 N/A Build 22000

OS Manufacturer: Microsoft Corporation

OS Configuration: Standalone Workstation

OS Build Type: Multiprocessor Free

Registered Owner: 918937949478

Registered Organization: N/A

Product ID: 00327-35874-88930-AAOEM

Original Install Date: 29/10/2021, 01:39:52

System Boot Time: 12/02/2022, 15:26:04

System Manufacturer: LENOVO

System Model: 81VV

System Type: x64-based PC

Processor(s): 1 Processor(s) Installed.

[01]: Intel64 Family 6 Model 126 Stepping 5 GenuineIntel ~1190 Mhz

BIOS Version: LENOVO CUCN27WW(V1.16), 15/06/2021

Windows Directory: C:\WINDOWS

System Directory: C:\WINDOWS\system32

Boot Device: \Device\HarddiskVolume1

System Locale: en-us;English (United States)

Input Locale: 00004009

Time Zone: (UTC+05:30) Chennai, Kolkata, Mumbai, New Delhi

Total Physical Memory: 7.968 MB

Available Physical Memory: 1.643 MB

Virtual Memory: Max Size: 14.394 MB

Virtual Memory: Available: 2.278 MB

Virtual Memory: In Use: 12.116 MB

Page File Location(s): C:\pagefile.sys

Domain: WORKGROUP

Logon Server: \\LAPTOP-A3T7UCMD

Hotfix(s): 5 Hotfix(s) Installed.

[01]: KB5009469

[02]: KB5004567

[03]: KB5008295

[04]: KB5010386

[05]: KB5009641

Network Card(s): 2 NIC(s) Installed.

[01]: Bluetooth Device (Personal Area Network)

Connection Name: Bluetooth Network Connection

Status: Media disconnected

[02]: Intel(R) Wireless-AC 9560 160MHz

Connection Name: Wi-Fi

DHCP Enabled: Yes

DHCP Server: 192.168.1.1

IP address(es)

[01]: 192.168.1.3

[02]: fe80::999a:bfb5:fae9:8e81

[03]: 2401:4900:1c64:85b3:bdf7:b9c3:7af4:2efd

[04]: 2401:4900:1c64:85b3:999a:bfb5:fae9:8e81

Hyper-V Requirements: VM Monitor Mode Extensions: Yes

Virtualization Enabled In Firmware: Yes

Second Level Address Translation: Yes

Data Execution Prevention Available: Yes

C:\Users\91893>getmac

Physical Address Transport Name

=================== ==========================================================

D8-3B-BF-5D-D1-84 Media disconnected

D8-3B-BF-5D-D1-80 \Device\Tcpip\_{2657868B-5651-44B6-A1F7-D271BE0F2CCC}

C:\Users\91893>

**EXPERIMENT - 2**

**Aim:**Study and Practical implementation of Cross-Wired and Straight Through network cables.

**Objective:** To know the procedure of making straight and cross over network cables which are used for wired network communication.

**Conceptual Background:**

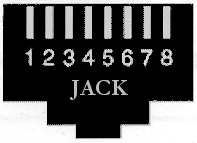
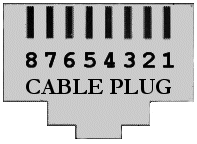
10/100BaseT Ethernet wiring is based upon 8-pin modular jacks.  Two wiring guidelines are offered in Electronic Industry Association (EIA)/Telecommunications Industry Association (TIA) standards:

* EIA/TIA 568A
* EIA/TIA 568B

A standard Ethernet NIC card (DTE) has the following RJ-45 modular pinout:

|  |  |
| --- | --- |
| **Pin** | **Function** |
| 1 | Tx+ |
| 2 | Tx- |
| 3 | Rc+ |
| 6 | Rc- |

The RJ-45 modular jack has 8 conductors/pins.  Viewed from the front:

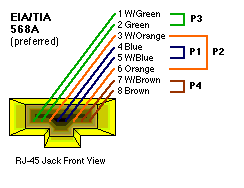
 

***What is 568?***

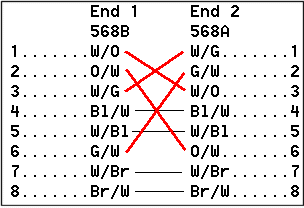
In the world of structured cabling systems the cryptic number 568 refers to the order in which the individual wires inside a CAT 5 cable are terminated. The termination could come at either the user’s end socket, the patch panel or termination frame or even the individual leads that connect a computer to the wall socket. There are currently two different specifications with respect to the order these cables should be terminated contained in the international standards document (ISO/IEC 11801:1995) as there is no indication as to which of these standards is preferred.

**TIA-568A**

In TIA-568A wiring schemes, the White/Orange and White/Green pairs are transposed.  An Ethernet "Crossover" cable can be made by using TIA-568B at one end of the cable and wiring to TIA-568A at the other end of the cable.  These "Crossover" cables can be used to interconnect NIC cards directly, without the need for a hub.  "Crossover" cables are usually "**RED**" in color:

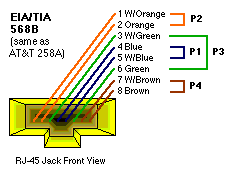
****

**FIGURE 1.0 SHOWS HOW THE TIA/EIA 568A STANDARD IS TO BE TERMINATED. NOTE THE POSITION OF THE GREEN/WHITE GREEN AND THE ORANGE/WHITE ORANGE PAIRS**.

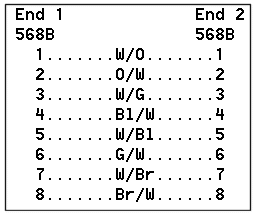


**TIA-568B**

TIA-568B is **the most common wiring scheme** used in 10/100BaseT configurations.  The standard Ethernet "Patch" cable that connects a hub port to a NIC card uses TIA-568 wiring at both ends of a "straight-through" cable:



**IN FIGURE 1.1 YOU CAN SEE THE TIA/EIA 568B STANDARD. ONCE AGAIN NOTE THE POSITION OF THE GREEN/WHITE GREEN AND ORANGE/WHITE ORANGE PAIRS.**



**COMPARING 568A AND 568B**

By looking at the first two specifications we see that the only difference is that the green and orange pairs are terminated to different pins, there is no difference as to what signal is used on what pin, only what color wire is terminated onto it. So technically the standards are the same, they operate in the same manner and neither one is technically superior to another when used in Ethernet applications.

It is when an Ethernet system and a phone system are combined that the difference really becomes apparent.

**Procedure:**

**PATCH CABLE ASSEMBLY INSTRUCTIONS**

1. Skin the cable jacket back about an inch, possibly more.
2. Un-twist each pair, and straighten each wire between the fingers.
3. There are two common wiring standards. Before this step be sure which type you need.

*If you need T568B then place the wires in this order: White/Orange, Orange, White/Green, Blue, White/Blue, Green, White/Brown, Brown*

*However if you need T568A then use this order; White/Green, Green. Whit/Orange, Blue, White/Blue, Orange, White/Brown, Brown. Bring all of the wires together, until they touch.*

1. Recheck the wiring sequence.
2. Hold the grouped (and sorted) wires together tightly, between the thumb, and the forefinger.
3. Cut all of the wires at a perfect 90 degree angle from the cable at 1/2" from the end of the cable jacket. If the wires are not cut straight, they may not all make contact.
4. Insert the wires into the connector (pins facing up).
5. Place the connector into a crimp tool, and squeeze hard so that the handle reaches it's full swing.
6. Repeat the process on the other end. For a straight through cable, use the same wiring.
7. Use a cable tester to test for proper continuity.

**Experiment-3**

Aim: Write a program to implement stop and wait protocol.

#include<iostream>

#include <time.h>

#include <cstdlib>

#include<ctime>

#include <unistd.h>

using namespace std;

class timer {

private:

unsigned long begTime;

public:

void start() {

begTime = clock();

}

unsigned long elapsedTime() {

return ((unsigned long) clock() - begTime) / CLOCKS\_PER\_SEC;

}

bool isTimeout(unsigned long seconds) {

return seconds >= elapsedTime();

}

};

int main()

{

int frames[] = {1,2,3,4,5,6,7,8,9,10};

unsigned long seconds = 5;

srand(time(NULL));

timer t;

cout<<"Sender has to send frames : ";

for(int i=0;i<10;i++)

cout<<frames[i]<<" ";

cout<<endl;

int count = 0;

bool delay = false;

cout<<endl<<"Sender\t\t\t\t\tReceiver"<<endl;

do

{

bool timeout = false;

cout<<"Sending Frame : "<<frames[count];

cout.flush();

cout<<"\t\t";

t.start();

if(rand()%2)

{

int to = 24600 + rand()%(64000 - 24600) + 1;

for(int i=0;i<64000;i++)

for(int j=0;j<to;j++) {}

}

if(t.elapsedTime() <= seconds)

{

cout<<"Received Frame : "<<frames[count]<<" ";

if(delay)

{

cout<<"Duplicate";

delay = false;

}

cout<<endl;

count++;

}

else

{

cout<<"---"<<endl;

cout<<"Timeout"<<endl;

timeout = true;

}

t.start();

if(rand()%2 || !timeout)

{

int to = 24600 + rand()%(64000 - 24600) + 1;

for(int i=0;i<64000;i++)

for(int j=0;j<to;j++) {}

if(t.elapsedTime() > seconds )

{

cout<<"Delayed Ack"<<endl;

count--;

delay = true;

}

else if(!timeout)

cout<<"Acknowledgement : "<<frames[count]-1<<endl;

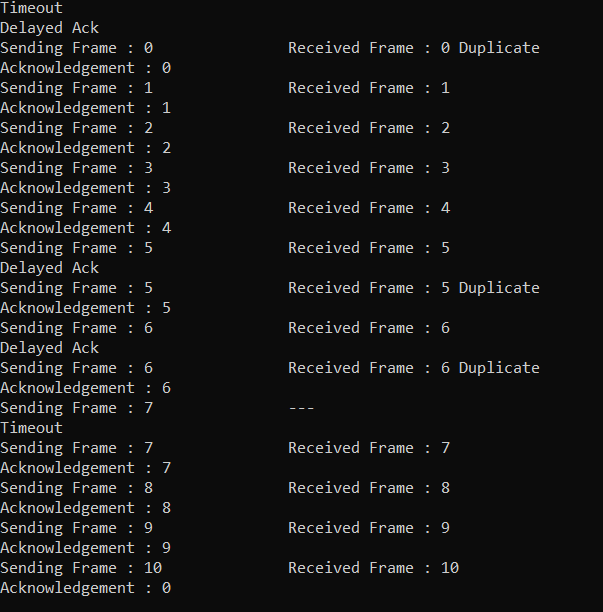
}

}while(count!=10);

return 0;

}

**OUTPUT –**

****