



دانشگاه آزاد اسلامی

واحد نجف آباد

دانشکده مهندسی کامپیوتر

شبکه های کامپیوتری

فهرست مطالب

<u>صفحه</u>	<u>عنوان</u>
سه	فهرست مطالب
	فصل اول: آشنایی با شبکه
۱	۱-۱ تاریخچه شبکه:.....
۱	۲-۱ انواع شبکه ها:.....
۳	۳-۱ سه دلیل استفاده از شبکه:.....
	فصل دوم: ساختار شبکه
۵	۱-۲ اجزای اصلی شبکه عبارتند از:.....
۵	۲-۲ End System ها به دو دسته زیر تقسیم می شوند.....
۶	۳-۲ وظایف NIC :.....
۷	۴-۲ IMP (Inter face messaging protocol) ها به ۴ دسته هستند:.....
۸	۵-۲ Media: انتقال داده.....
۸	۱-۵-۲ روش های انتقال داده.....
۹	۲-۵-۲ مزیت های روش نوری:.....
۹	۶-۲ انواع Media.....
۹	۷-۲ انواع Media کابلی به صورت زیر می باشد:.....
۱۰	۲-۷-۱ نوع اول Coaxial:.....
۱۰	۲-۷-۲ نوع دوم Twisted pair:.....
۱۲	۳-۷-۲ نوع سوم Isdn:.....
۱۲	۴-۷-۲ نوع چهارم x25:.....
۱۳	۵-۷-۲ نوع پنجم فیبر نوری:.....
۱۴	۸-۲ انواع مدیا های بی سیم.....
۱۴	۱-۸-۲ نوع اول Wifi:.....
۱۵	۲-۸-۲ نوع دوم Wimax:.....
۱۵	۳-۸-۲ نوع سوم Microwave:.....
۱۶	۴-۸-۲ نوع چهارم Infrared:.....
۱۶	۵-۸-۲ نوع پنجم لیزری ::.....
	فصل سوم: توپولوژی های شبکه
۱۸	۱-۳ توپولوژی:.....
۱۸	۱-۱-۳ Broad cast:.....
۲۰	۲-۱-۳ Point To Point:.....
۲۱	۲-۳ انواع توپولوژی های broadcast:.....
۲۱	۲-۳ Bus۱:.....
۲۱	۲-۳ Ring۲:.....
۲۲	۳-۲-۳ شبکه های ماهواره ای:.....
۲۲	۳-۳ انواع توپولوژی های Point To Point:.....
۲۲	۳-۳-۱ bus:.....

۲۳: star ۲-۳-۳
۲۴: Full mesh ۳-۳-۳ شبکه کامل یا
۲۴: ۴-۳-۳ شبکه درختی :
۲۵: ۴-۳-۳ توپولوژی اینترنت:
	فصل چهارم: معماری شبکه
۲۷: ۱-۴ معماری شبکه:
۲۷: ۲-۴ معماری (Open System Interconnection) OSI
۲۸: ۳-۴ لایه های مدل OSI :
۲۸: ۱-۳-۴ لایه ۷: Application یا کاربرد :
۲۸: ۲-۳-۴ لایه ۶: presentation یا نمایش:
۲۸: ۳-۳-۴ لایه ۵: session یا جلسه :
۲۸: ۴-۴-۴ لایه ۴: Transport یا لایه انتقال :
۲۹: ۵-۴-۴ لایه ۳: Network یا شبکه:
۲۹: ۶-۴-۴ لایه ۲: Data link یا پیوند داده ها :
۲۹: ۷-۴-۴ لایه ۱: physical یا فیزیکی:
۳۰: ۵-۴-۴ جایگاه لایه شبکه :
۳۰: ۶-۴-۴ مدل TCP/IP:
۳۱: ۷-۴-۴ مدل Client/server :
	فصل پنجم: لایه پیوند داده ها
۳۳: ۱-۵ ماهیت خطا:
۳۳: ۲-۵ ویژگی های خطا:
۳۳: ۳-۵ نحوه برخورد با خطا:
۳۳: ۱-۳-۵ Error detection یعنی کشف خطا:
۳۴: ۲-۳-۵ Error Recovery یعنی اصلاح خطا:
۳۴: ۴-۵ روش کشف خطای تکی:
۳۴: ۱-۴-۵ توازن زوج:
۳۴: ۲-۴-۵ روش اصلاح خطای تکی:
۳۶: ۵-۵ چند جمله ای معادل یک پیغام:
۳۶: ۶-۵ عملیات ماژول ۲ :
۳۶: ۷-۵ چند جمله ای : $G(x)$.
۳۶: ۸-۵ روش الگوریتم CRC :
۴۰: ۹-۵ نحوه برخورد با خطای Burst .
۴۰: ۱۰-۵ تشخیص خرابی یا خطا توسط الگوریتم CRC
۴۱: ۱۱-۵ وظایف لایه پیوند داده ها
	فصل ششم: لایه شبکه
۴۴: ۱-۶ وظایف لایه شبکه عبارتند از
۴۴: ۲-۶ ویژگی های الگوریتم مسیریابی عبارتند از :
۴۴: ۳-۶ انواع الگوریتم های مسیریابی :
۴۵: ۴-۶ دسته بندی الگوریتم های مسیریابی
۵۱: ۵-۶ عوامل بروز ازدحام :
۵۲	

۶-۶	روش های جلوگیری از بروز ازدحام :
۷-۶	بن بست به سه روش رخ می دهد	
۵۲	فصل هفتم: لایه حمل..... چهار
۵۶	۱-۷ لایه حمل نحوه انتقال داده ها را به یکی از ۳ روش زیر مشخص می کند :
۵۶	۱-۱-۷ روش Virtual circuit یا روش VC :
۶۰	۲-۱-۷ روش Message Switching :
۶۱	۳-۱-۷ روش Packet switching :
۶۲	۲-۷ نمودار Packet switching , Message switching :
	فصل هشتم : شبکه های محلی	
۶۴	۱-۸ ویژگی های شبکه های محلی :
۶۵	۲-۸ انواع شبکه های محلی از نظر عملکرد :
۶۶	۳-۸ عملکرد CSMA در شبکه های محلی : Carrier Sense Multiple Access.....	
۶۶	۱-۳-۸ Persistent : مصر
۶۶	۲-۳-۸ Non Persistent : غیر مصر
۶۷	۴-۸ Ethernet (CSMA / CD) :
۶۸	۵-۸ Collision Free :
۶۸	۱-۵-۸:Bitmap
۶۹	۲-۵-۸ : BRAP
۶۹	۳-۵-۸ : Multi Level Multi Access : MLMA
۶۹	۶-۸ Limited Contension : رقابت مجدد :
۷۰	۷-۸ Token Ring :
۷۱	۸-۸ Slotted Ring :
۷۱	۹-۸ Shared memory
۷۱	۱-۹-۸ Multi processor :
۷۲	۲-۹-۸ Multi Computer روش
۷۳	۱۰-۸ آدرس های IP :
۷۳	۱۱-۸ کلاس های آدرس IP :
۷۴	۱۲-۸ مفهوم Subnet Mask :
۷۵	۱۳-۸ آدرس های Invalid :
۷۵	۱۴-۸ ارتباط بین شبکه های محلی :
۷۶	۱۵-۸ وظایف Getway عبارتند از
	فصل نهم : شبکه اینترنت	
۷۸	۱-۹ TCP/IP :
۸۰	۲-۹ پسوند صفحات وب :
۸۱	۳-۹ پروتکل های Mail دو دسته اند
۸۱	۱-۳-۹ دسته اول پروتکل های دسترسی به Mail
۸۱	۲-۳-۹ دسته دوم انتقال Mail در شبکه:
۸۳	۴-۹ مراحل ایجاد وب سایت :
۸۳	۵-۹ سرویس VPN :
۸۵	۶-۹ لایه حمل :
۸۵		

..... ۷-۹ لایه حمل به وسیله دو پروتکل کانال ارتباطی را مشخص می کند:

..... TCP ۱-۷-۹ : ۸۵

..... UDP ۲-۷-۹ : ۸۷

..... ۸-۹ لایه شبکه : ۸۷

..... ۹-۹ جدول مسیریابی (Forward Table) : ۸۸

..... RIP : ۱-۹-۹ جدول : ۸۹

..... OSPF: ۲-۹-۹ جدول : ۹۰

فصل اول:

آشنایی با شبکه

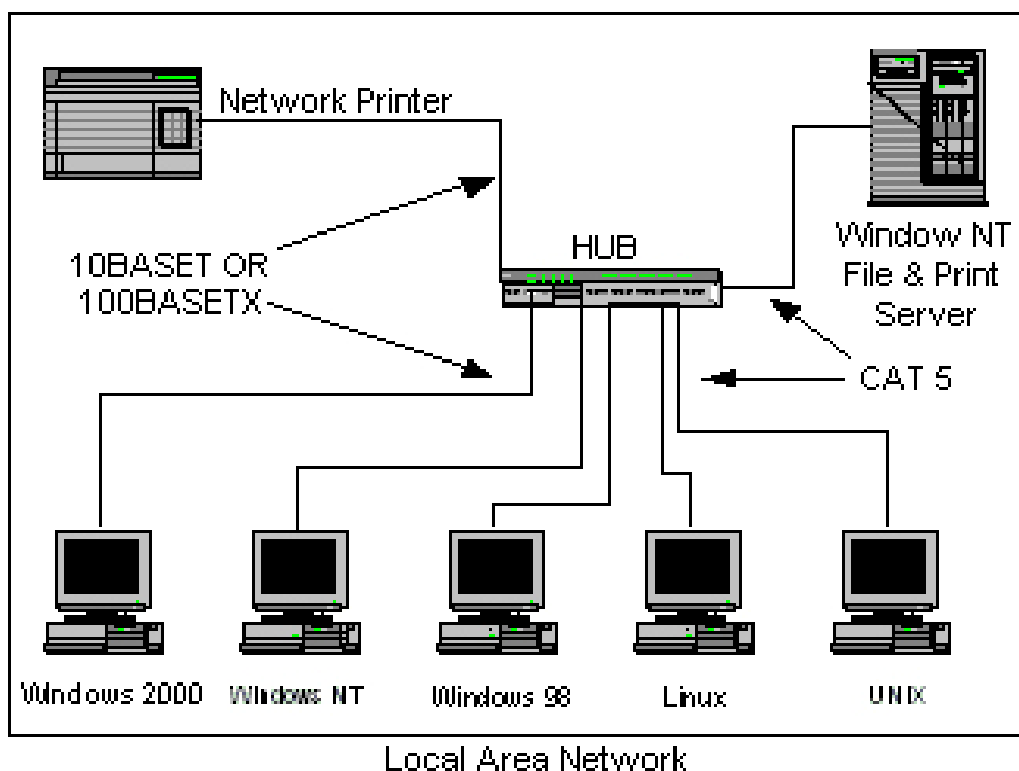
۱-۱ تاریخچه شبکه:

شبکه در سال ۱۹۶۰ متولد شد. و نام اولین شبکه *Arpanet* بود که این شبکه به منظور ایجاد ارتباطات بین پایگاه های نظامی با استفاده از خطوط تلگراف و تلفن که در آن زمان وجود داشت ایجاد شد. در سال ۱۹۷۰ مراکز علمی (مراکز تجاری) به این شبکه پیوستند. نکته قابل توجه این است که شبکه ها تا سال ۱۹۸۰ منسجم نبودند اما از سال ۱۹۸۰ به بعد با هم ادغام شدند و *Internet* را تشکیل دادند. در سال ۱۹۸۹ وب سایت ها به دو منظور ایجاد شدند (۱) انتقال اطلاعات (۲) یکجا و منسجم قرار گرفتن اطلاعات زیرا تا قبل از ۱۹۸۹ اطلاعات یک جا قرار نداشتند و همچنین امکان انتقال اطلاعات نیز وجود نداشت. در سال ۱۹۹۰ اینترنت عمومی شد. و در سال ۲۰۰۰ *VO IP, IP TV* ها به روی کار آمدند.

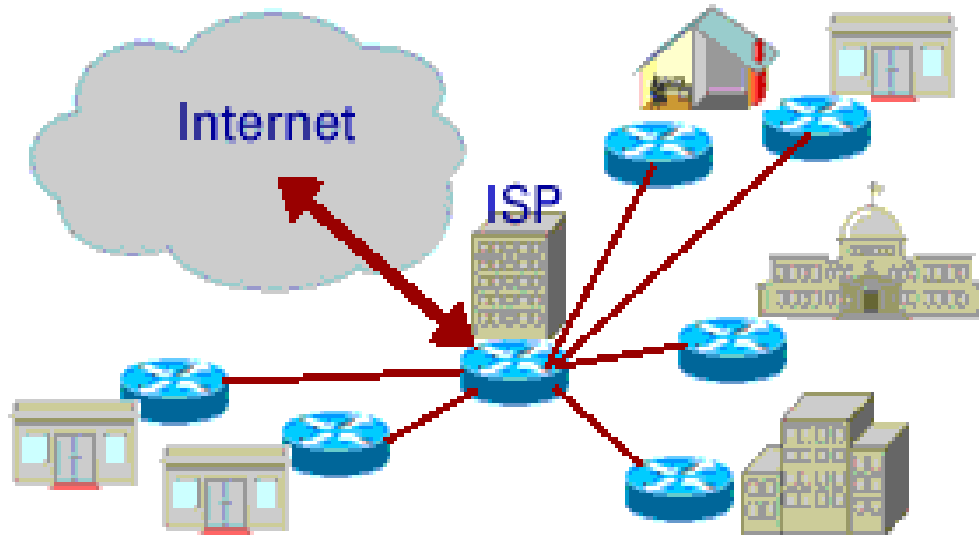
۱-۲ انواع شبکه ها:

شبکه ها را بر اساس فاصله کامپیوتر ها از هم تقسیم بندی می نمایند.

۱. *Lan* (شبکه محلی): فاصله کامپیوتر ها از هم در حد یک ساختمان است. و حدودا تا شعاع ۲۰۰ متر را پوشش می دهد.

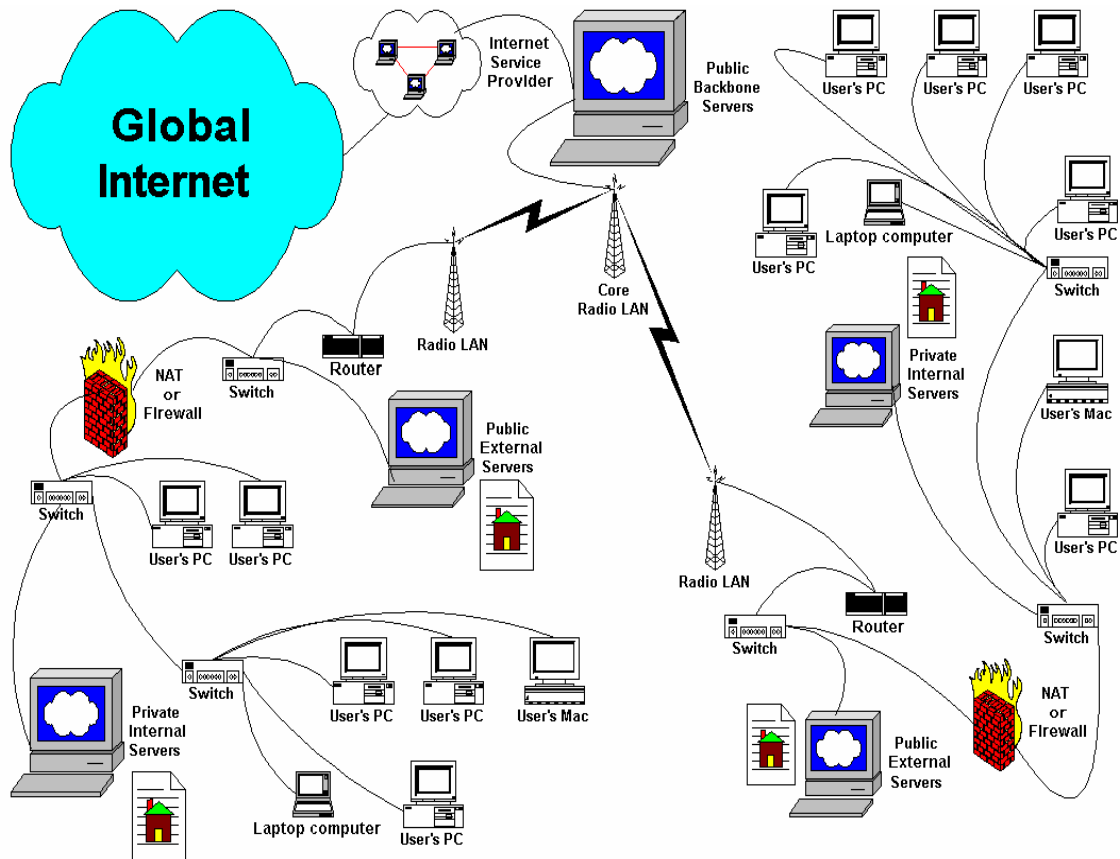


۲. *Man* (شبکه شهری) : فاصله یک شهر را پوشش میدهد . حدوداً ۲۰ کیلومتر



Metropolitan Area Network (MAN)

۳. *Wan* (شبکه های گسترده) : مثل اینترنت محدودیت جغرافیایی ندارد .



Wide Area Network

نکته قابل توجه این است که : میانگین سرعت در شبکه های گسترده کمتر از شهری و هر دو شبکه گسترده و شهری کمتر از شبکه های محلی می باشد پس هر چه فاصله کمتر سرعت بالاتر است .

۱-۳ سه دلیل استفاده از شبکه:

۱. تبادل اطلاعات
۲. استفاده اشتراکی از منابع
۳. انجام محاسبات پیچیده : بعضی از محاسبات توسط یک کامپیوتر قابل انجام نیست و مجبور می شویم از چند کامپیوتر با هم استفاده نماییم که این مساله نیاز ما را به شبکه زیاد می کند .

فصل دوم:

ساختار شبکه

در فصل دوم ابتدا اجزای سخت افزاری شبکه بررسی می شود .

۱-۲ اجزای اصلی شبکه عبارتند از :

(۱) *EndSystem* ها

(۲) *IMP* ها

(۳) *Media*

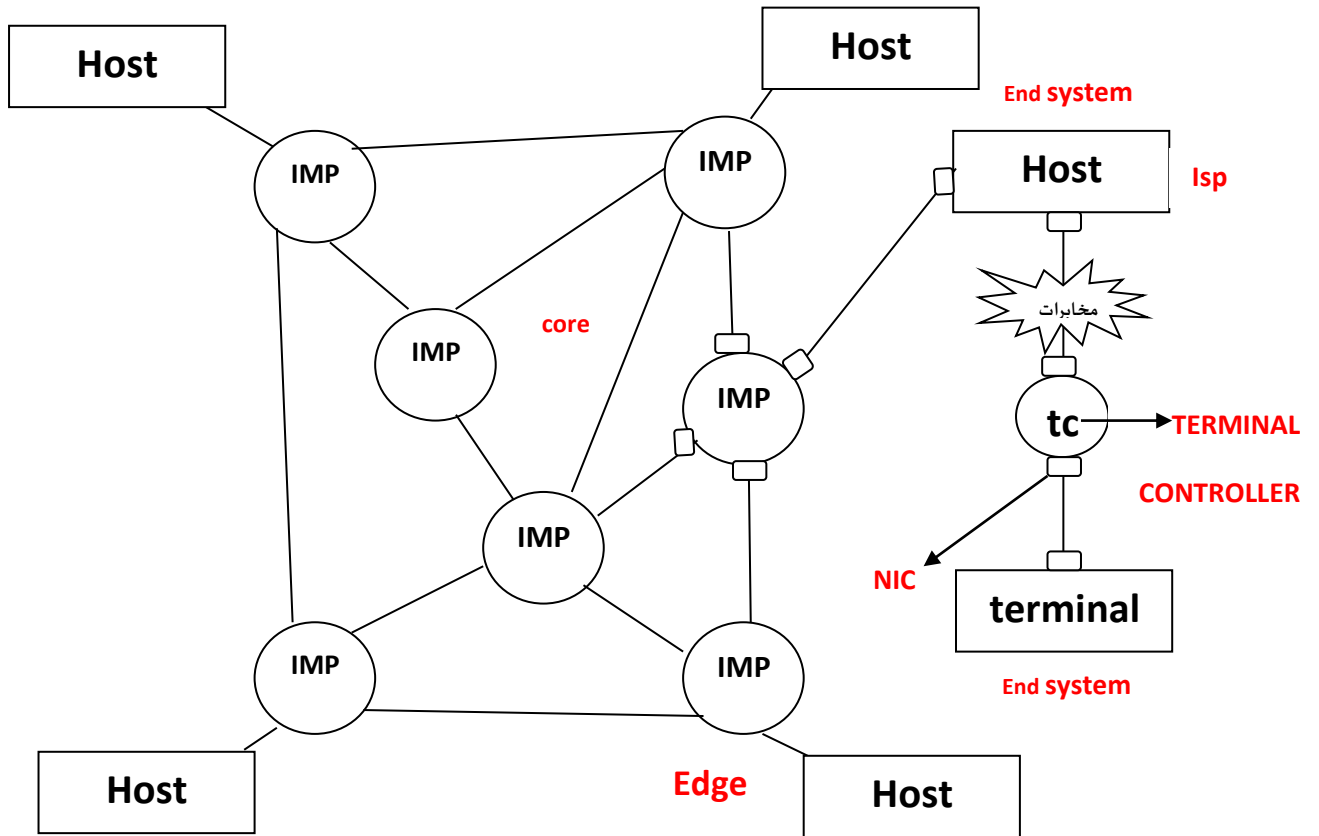
- *EndSystem* : هر ابزاری که بتواند از شبکه استفاده کند یا سرویسی را در شبکه ارائه دهد مثل کامپیوتر- لپ تاپ - چاپگر- تب لت و ... را *EndSystem* گویند. (یعنی کامپیوتر ها یا دستگاه هایی که در انتهای شبکه قرار دارند)
- *IMP* : ابزاری برای ایجاد ارتباطات
- *Media* (رسانه انتقالی) : ابزاری که انتقال داده ها را بر عهده دارد (می تواند کابلی یا بی سیم باشد)

۲-۲ *EndSystem* ها به دو دسته زیر تقسیم می شوند :

۱. *Host* یا *server* : ارائه کننده سرویسی می باشد که قرار است در شبکه در اختیار دیگر کامپیوتر ها (*Terminal*) قرار بگیرد .
۲. *Terminal* یا *Client* : مشتری (سرویس گیرنده)

تفاوت *Host* و *Terminal* : *Host* یک کامپیوتر دائمی است که دارای آدرس ثابت است و مستقیماً به شبکه متصل می شود اما *Terminal* یک کامپیوتر موقت است که دارای آدرس متغیر است و با واسطه *Host* می تواند به شبکه متصل شود .

در شکل زیر ساختار یک شبکه گسترده را مشاهده می نمایید :



Core : Imp هایی که وسط قرار دارند و به بقیه Imp ها متصل هستند را در اصطلاح *core* می گویند .

Edge : Imp هایی که لبه ی شبکه ها هستند و به *End system* ها متصل اند را لبه یا *Edge* گویند.

۳-۲ وظایف NIC :

۱. تبدیل دیجیتال به آنالوگ و بالعکس
۲. تبدیل موازی به سریال و بالعکس : در کامپیوتر انتقال اطلاعات یا ۳۲ یا ۶۴ بیتی همزمان است اما روی رسانه انتقالی باید در یک باند حرکت کنند و موازی را به سریال تبدیل می کند .

۲-۴ IMP (Inter face messging protocol) ها به ۴ دسته هستند :

لایه ۱ (Layer 1) : فقط وظیفه برقراری ارتباط را دارند . و هیچگونه مدیریتی روی اطلاعات انجام نمی دهد .
مانند Hub



لایه ۲ (layer2) : علاوه بر برقراری ارتباط امکان مدیریت داده ها را نیز بر عهده دارد مانند Switch



لایه ۳ (layer3) : امکان ارتباط بین ۲ یا چند شبکه شبکه متفاوت از هم را برقرار می کند. مانند Router این IMP وظیفه مسیریابی دارد.



لایه ۴ (Layer4) : توانایی کنترل ورودی و خروجی های یک شبکه را دارد مانند firewall

نکته : لایه اول و دوم مربوط به شبکه محلی یا *Lan* می باشد و لایه سوم و چهارم مربوط به شبکه گسترده یا *wan*

Hub/Switch : ابزاری است که مزایای هر دو را دارد . هر جا لازم باشد مانند یک سوئیچ رفتار می کند و هر جا نیاز به مدیریت نباشد همچون *Hub* عمل می کند . امتیاز *Hub* نسبت به *Switch* در سرعت بالای آن است .
Switch/Router یا سوئیچ لایه ۳ : هم می تواند سوئیچ باشد و هم می تواند مسیر یاب .

نکته ۱ : مهم این است که *Terminall controller* یک سوئیچ لایه ۳ محسوب می شود زیرا از یک طرف به یک شبکه محلی و از یک طرف به شبکه گسترده متصل است .

نکته ۲ : مسیر یاب باید تمام ورودی هایش از شبکه های مختلف باشند نباید ورودی ها در یک شبکه قرار داشته باشند .

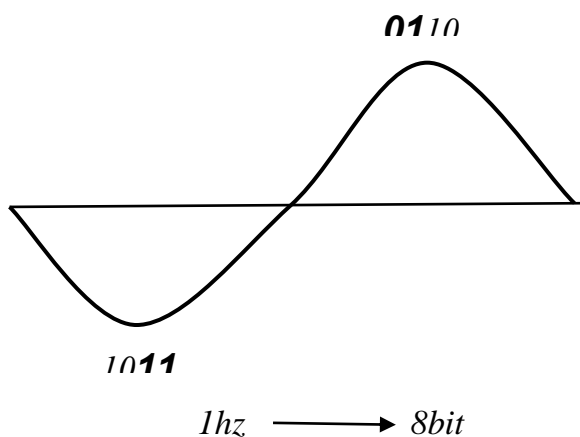
۲-۵ Media : انتقال داده

۲-۵-۱ روش های انتقال داده

۱. موج سینوسی

۲. نوری

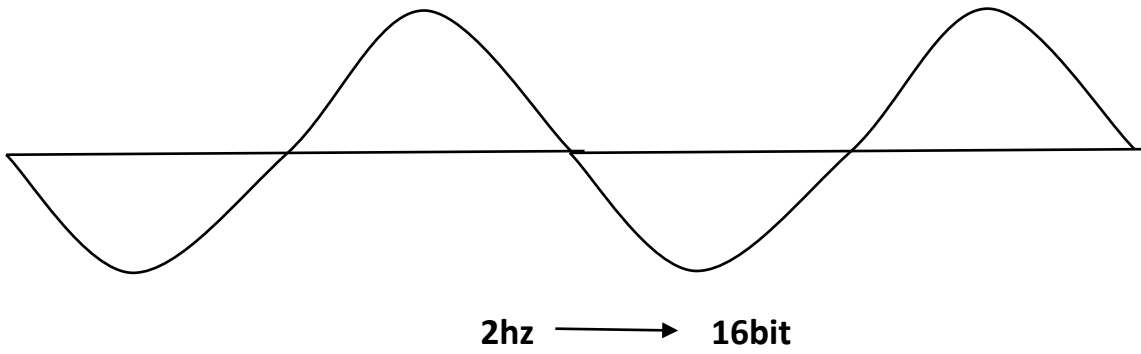
۱- موج سینوسی : امواج می توانند بر روی خود دیتا (صفر و یک) را حمل کنند. در یک سیکل کامل اگر مثلاً در یک ثانیه در نظر بگیریم ۸ بیت قابل انتقال است.



هر چه فرکانس بیشتر سرعت بالا تر است . پس سرعت وابسته به فرکانس است .

ویژگی موج ها در این است که موج های غیر هم نوع که فرکانس هایشان متفاوت باشد روی هم تاثیر نمی گذارند .

۲-نوری: دیتا های (صفر و یک) را به وسیله نور انتقال می دهد.



۲-۵-۲ مزیت های روش نوری :

۱. سرعت بالا
۲. عدم وجود نویز برای نور

۲-۶ انواع *Media*

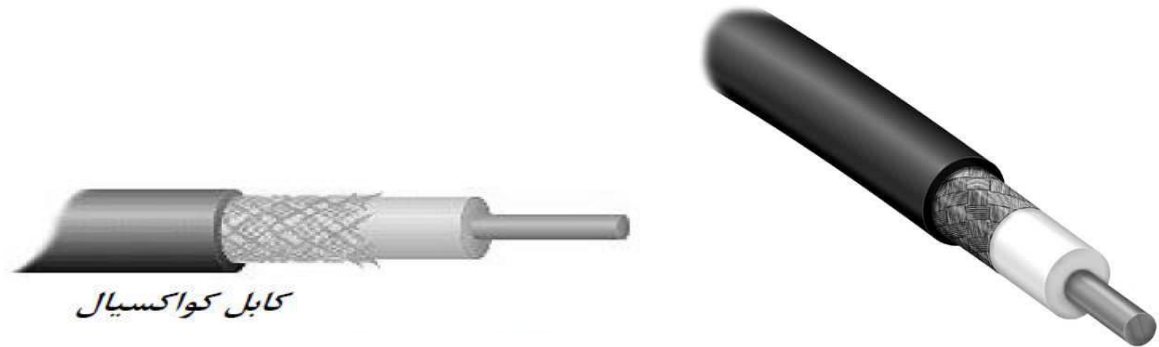
- (۱) کابلی
- (۲) بی سیم

۲-۷ انواع *Media* کابلی به صورت زیر می باشد :

- (۱) *Coaxial*
- (۲) *Twisted pair*
- (۳) *Isdn*
- (۴) *X25*
- (۵) *Fiber iptic*

۲-۷-۱ نوع اول *Coaxial* :

در این نوع کابل هسته مسی کار انتقال اطلاعات را انجام می دهد و روکش آن را *shield* می گویند حدود ۲۰۰ متر داده را می تواند انتقال دهد و سرعت آن *10 mbps* می باشد برای شبکه های محلی استفاده می شود .



کابل کواکسیال

۲-۷-۲ نوع دوم *Twisted pair* :

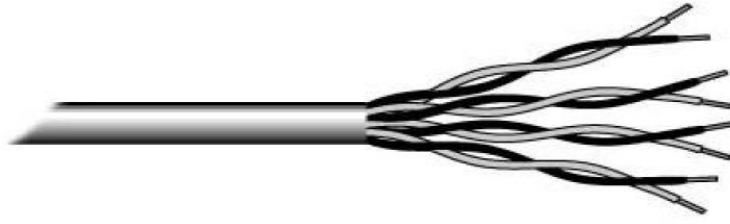
یا زوج سیم به هم تابیده شده که به آن کابل زوجی نیز گویند دارای دسته بندی خاصی است که به آن *category* گویند. این زوج سیم به هفت دسته زیر تقسیم می شود داده را تا فاصله ۲۰-۳۰ کیلومتر انتقال می دهد و سرعت آن *56 kbps* است.

Cat1 یک زوج به هم تابیده شده مثل خط تلفن

-
-
-

Cat5
Cat6
Cat7

} کابل های ۸ رشته ای

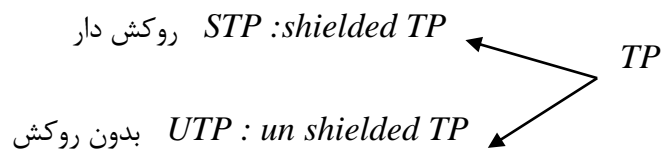


Cat7, Cat6, Cat5 داده ها را تا شعاع ۲۰۰ متر انتقال می دهند و سرعت این سه *cat* برای شبکه های محلی به ترتیب زیر است

<i>Cat5</i>	→	100Mbps
<i>Cat6</i>	→	1Gbps
<i>Cat7</i>	→	10Gbps

در هر سه مورد فوق امکان افزایش فرکانس بدون تداخل وجود دارد.

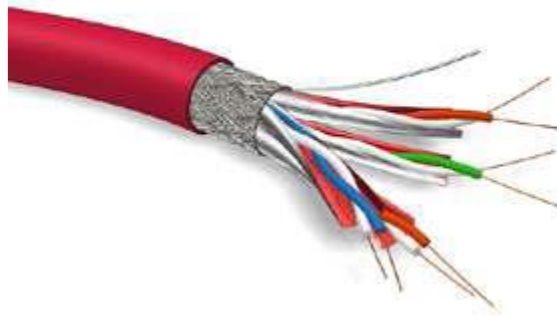
نوع دیگر تقسیم بندی زوج سیم به صورت زیر است.



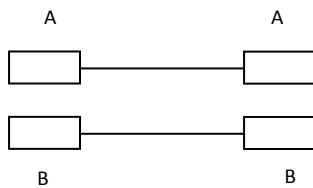
۱. نمونه ای از کابل های *UTP* یا کابل بدون روکش:



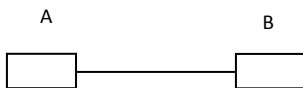
۲. نمونه ای از کابل های STP یا کابل روکش دار



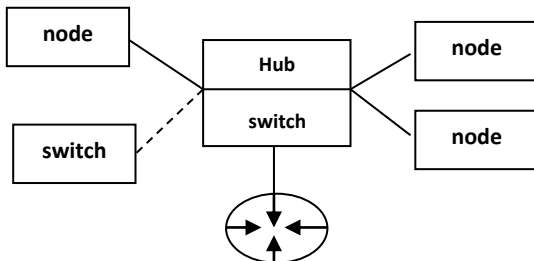
Straight : کابل مستقیم : یک کابل cat 5,6,7 ۸سیم درون خود دارد که نوع رنگ بندی آنها متفاوت است. ترتیب رنگ ها در ۴ زوج سیم به هم پیچیده در *Rj45* مهم است. اگر دوسر سیم از استاندارد (*968 A*) یا دو سر از استاندارد (*968 B*) استفاده کنند. به این سیم *Straight* گوئیم. (ترتیب رنگی دوسر کابل یکسان است)



Croxt : کابل متقاطع : اگر یک سر سیم از استاندارد (*968 A*) و یک سر از استاندارد (*968 B*) استفاده کنند. به این کابل *Croxt* گوئیم.



برای اتصال دودستگاه متفاوت به یکدیگر از کابل *Straight* استفاده می شود و برای اتصال دو دستگاه یکسان به یکدیگر از کابل *Croxt* استفاده میکنند.



۲-۷-۳ نوع سوم *Isdn* :

کاربرد اصلی این کابل در انتقال تصویر است. (مثل تلفن تصویری و تلویزیون کابلی) البته داده را نیز می تواند انتقال دهد. سرعتش 128 kbps و تا شعاع حدود $20 - 30$ کیلو متر داده را انتقال می دهد.

۲-۷-۴ نوع چهارم $x25$:

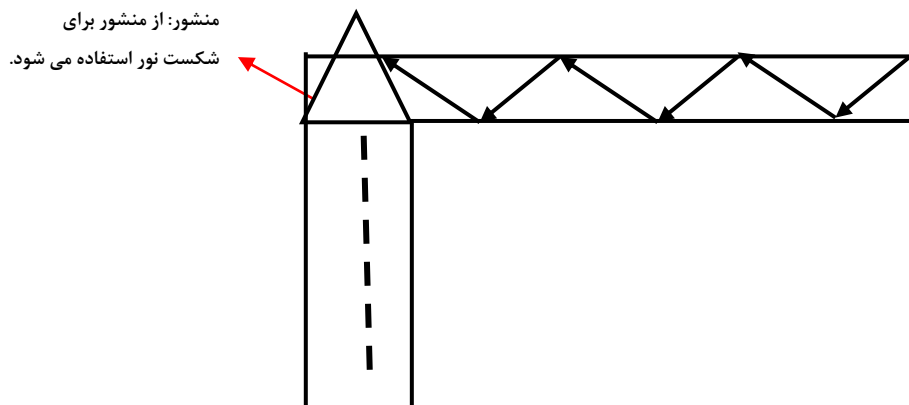
از این کابل برای انتقال فاکس با سرعت بالا استفاده می شود زیرا تلفن برای انتقال فاکس سرعتش کم است. برای انتقال داده تا شعاع $20-30$ کیلومتر استفاده می شود و سرعت این کابل ها 1.5 mbps می باشد.

۲-۷-۵ نوع پنجم فیبر نوری :

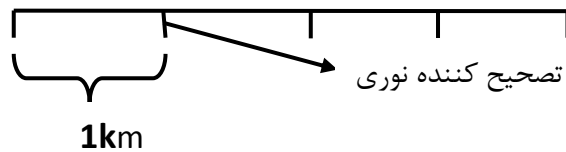
کابلی است که از تعدادی رشته ی شیشه ای تشکیل شده .

نکته ۱: بالاترین سرعت در تمام این روش ها را فیبر نوری دارد .

نکته ۲ : فیبر نوری انعطاف پذیر نیست یعنی قابل خم شدن نیست . نور هم شکسته نمی شود.



نکته: فیبر نوری محدودیت مسافت ندارد ولی هر یک کیلومتر به یک کیلومتر نیاز به یک تصحیح کننده ی نوری دارد.





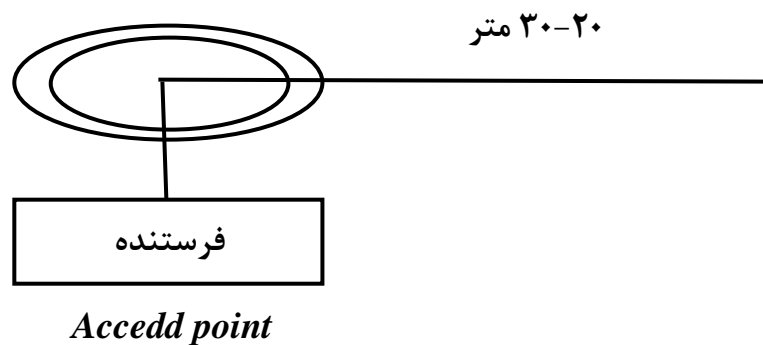
فیبر نوری و انواع مختلفی از آن (4-Core/6-Core/8-core/...)

۲-۸ انواع مدیا های بی سیم

- ۱) Wifi
- ۲) Wimax
- ۳) Microwave
- ۴) Infrared
- ۵) لیزری

۲-۸-۱ نوع اول Wifi :

یکی از روش های انتقال بی سیم است که فرستنده آن تا شعاع ۲۰-۳۰ متری رامی تواند پوشش دهد. سرعت آن $30-50\text{ mbps}$ می باشد نمونه این دستگاه *Access point* یا مودم های بی سیم خانگی است. *wifi* از استاندارد *IEEE 802/11* برای انتقال اطلاعات استفاده می نمایند. مزیت این روش در این است که کابل ندارد و قابل حمل است ولی مزیت سرعت بالا را ندارد.



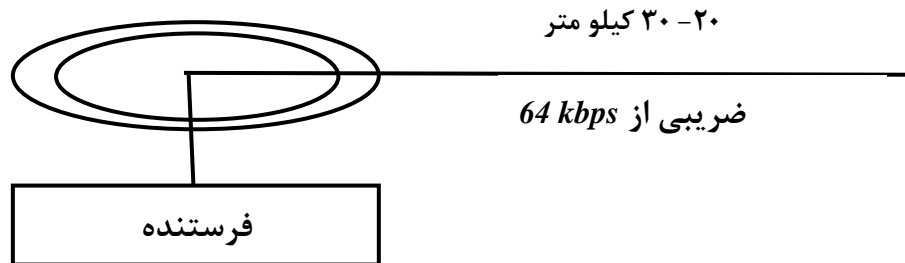
نکته ۱: بلوتوث یک نوع *wifi* ضعیف است و تا فاصله حدودا ۱۰ متر و با سرعت حداکثر ۷۲۰ kbps می تواند داده را انتقال دهد.

نکته ۲: یک نوع *wifi* خیلی ضعیف *NFC* است که در اتوبوس کارت ها رو جلوی آن می گیرند.

نکته ۳: همیشه امواج بی سیم سرعت کمتری نسبت به امواجی که در کابل هستند دارند .

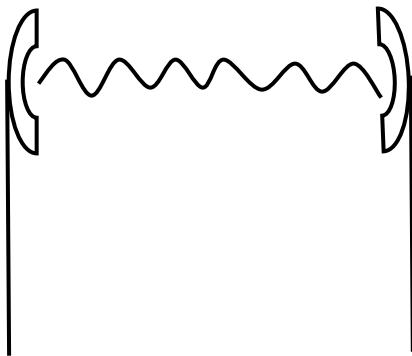
۲-۸-۲ نوع دوم *Wimax*:

فرستنده اش به وسیله امواجی که تولید می کند می تواند یک شهر را پوشش دهد. حداکثر سرعت آن ضریبی از ۶۴ kbps است و تا شعاع $۲۰-۳۰\text{ km}$ را پوشش می دهد



۲-۸-۳ نوع سوم *Microwave*:

متشابه *Wimax* سرعتش ضریبی از ۶۴ kbps است و تا شعاع حدود $۲۰-۳۰$ کیلومتر را پوشش می دهند. تنها تفاوت این دو در خطی بودن *Microwave* است. در ضمن در این روش دکل های آن باید در یک خط مستقیم و روبروی هم باشند



۲-۸-۴ نوع چهارم *Infrared*:

مادون قرمز شعاع کمی حدود ۱۰ متر را پوشش می دهد و در ضمن باید روبروی هم قرار بگیرند. سرعت آن زیر ۱ mbps است و از آن در کنترل تلویزیون استفاده می شود.

۲-۸-۵ نوع پنجم لیزری:

دقیقا مانند فیبر نوری است. با این تفاوت که کابل ندارد. یعنی مبدا و مقصد باید روبروی هم باشند.

فصل سوم:

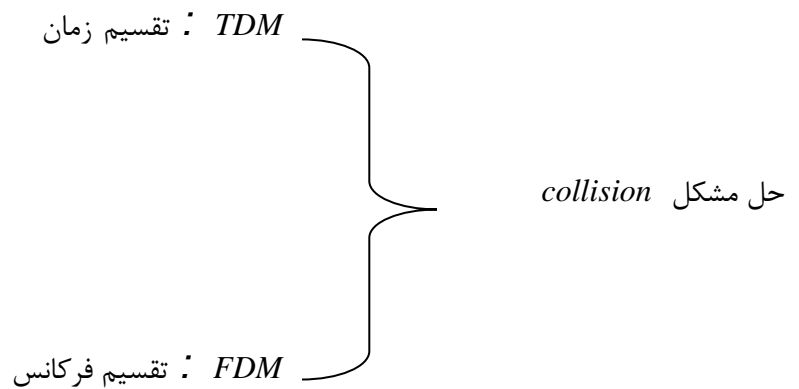
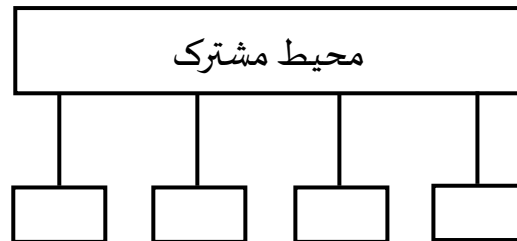
تولودشی های شبکه

۳-۱ توپولوژی :

نحوه اتصالات برای انتقال داده ها که به ۲ دسته اصلی (*Broad cast* و *Point To Point*) تقسیم می شوند.

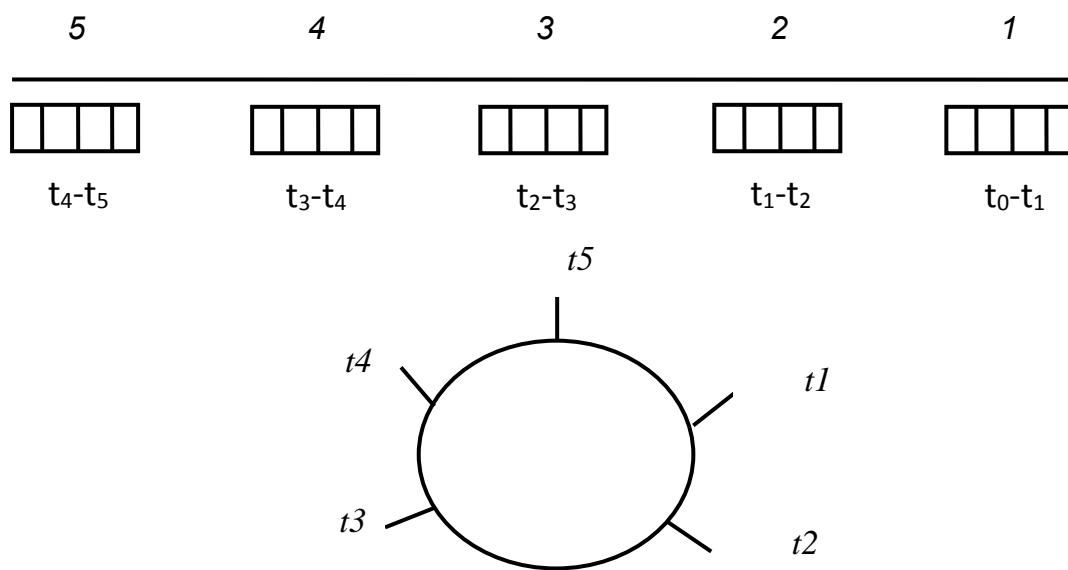
۳-۱-۱ *Broad cast* :

این یک روش پخشی است. یک محیط مشترک برای ارتباطات ایجاد می شود. کامپیوترها به محیط مشترک متصل هستند. فرستنده پیغام خود را بر روی محیط مشترک ارسال می کند. گیرنده یا گیرنده ها پیغام را دریافت می نمایند. در *Broad cast* مسیر یابی معنا ندارد. یک مشکل این روش *collision* یا تصادم است این مشکل زمانی رخ می دهد که دو فرستنده هم زمان پیغامی را ارسال نمایند.



(Time Devision Multiplexing) TDM (1)

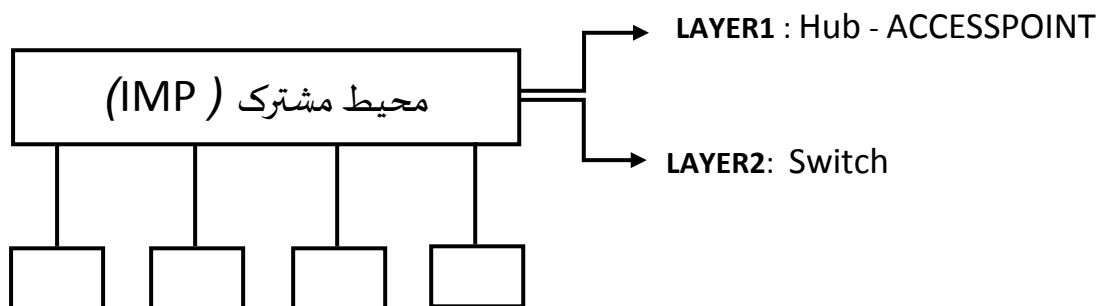
تقسیم زمان برای فرستنده ها : مثلا فرستنده های ۱ تا ۵ می خواهند همزمان پیغام ارسال کنند در این روش پیغام ها به قسمت های کوچکتر تقسیم می شوند سپس فرستنده اول در زمان اول قسمت اول پیغام خود را ارسال می نماید یعنی در زمان (t_0-t_1) . فرستنده دوم قسمت اول پیغام را در زمان (t_1-t_2) ارسال می نماید این روند تا زمانی ادامه پیدا می کند که کلیه ی پیام های هر ۵ فرستنده ارسال شوند.



(Frequency Devision Multiplexing) FDM (2)

در این روش هر فرستنده ها هر کدام یک روش جداگانه برای ارسال پیغام دارند.

نکته محیط اشتراک را IMP ها یی که یا لایه یک یا لایه دو هستند ایجاد می کنند.

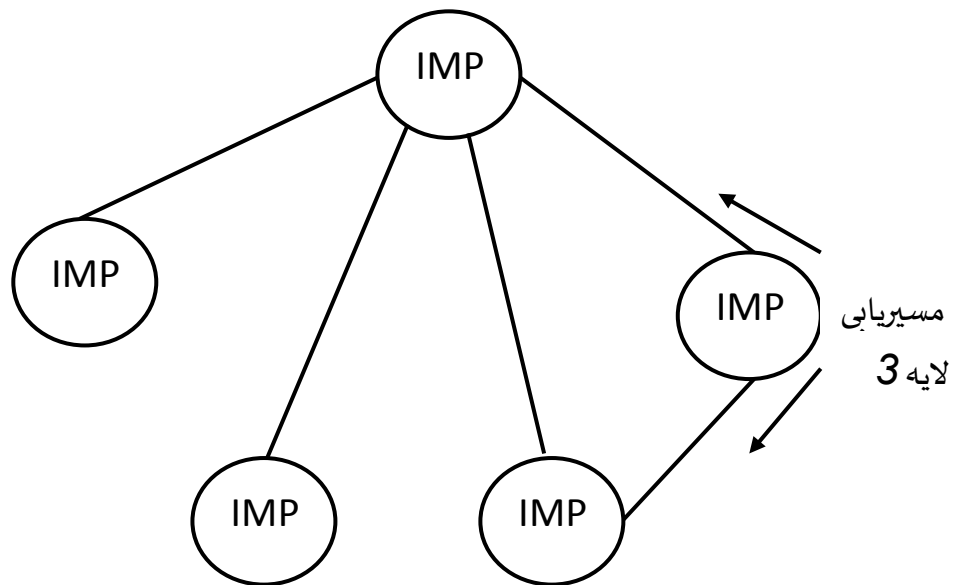


Point To Point 2-1-3 :

در این روش IMP مشترک وجود ندارد و هر IMP به یک یا چند IMP دیگر متصل است. مهم ترین مساله در این توپولوژی مسیریابی است .

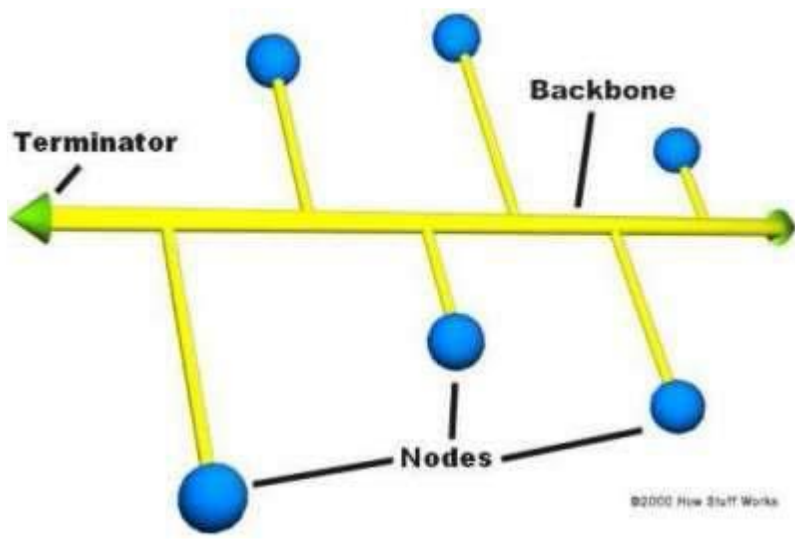
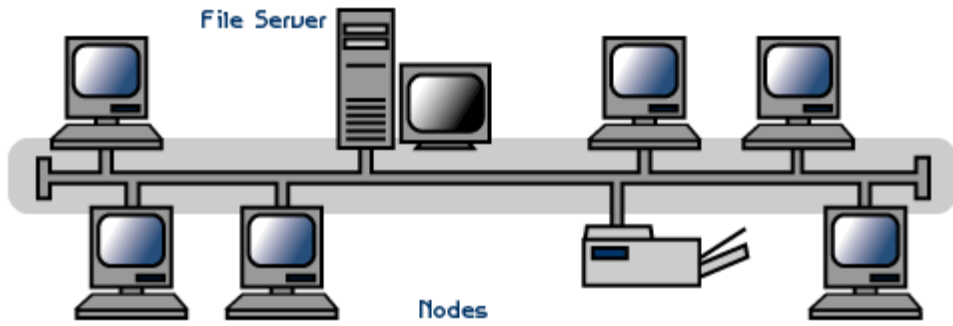
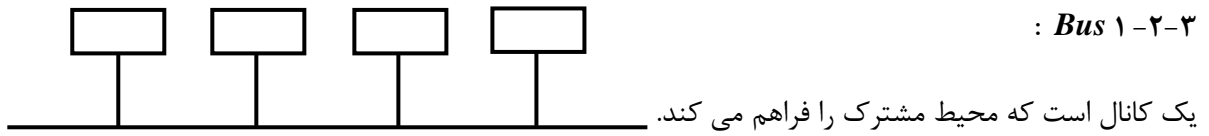


نکته : تمام IMP ها لایه ۳ می باشند زیرا باید مسیریابی انجام دهند.



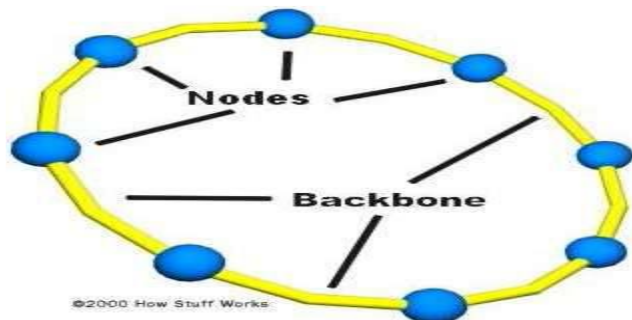
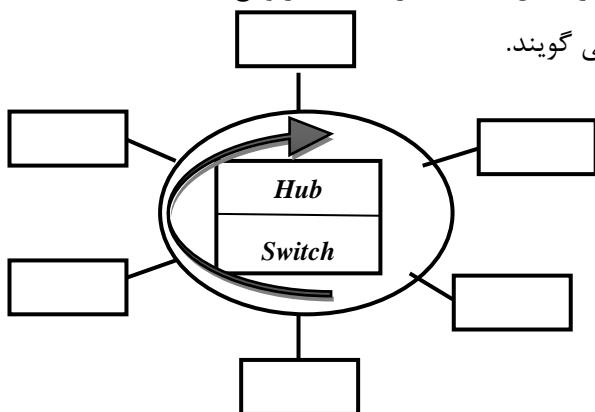
۲-۳ انواع توپولوژی های *broadcast* :

۱-۲-۳ *Bus* :



۲-۲-۳ *Ring* :

یا شبکه های حلقه ای ، داده ها درون یک حلقه حرکت می کنند. در ضمن حلقه مجازی است در واقع یک محیط مشترک توسط Hub و Switch ایجاد می شود که به آن حلقه می گویند.



۳-۲-۳ شبکه های ماهواره ای :

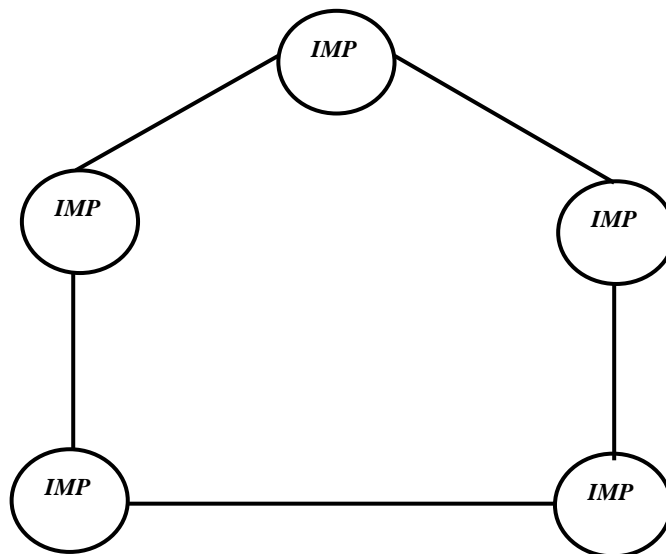
ماهواره محیط مشترک را فراهم می کند.

نکته ۱ : همه شبکه های محلی از توپولوژی *Broadcast* از نوع *Ring* که همان (هاب و سوئیچ) است استفاده می کنند .

نکته ۲: هیچ کدام از شبکه های محلی *Point To Point* نیستند . در شبکه های محلی از مسیر یاب استفاده نمی شود.

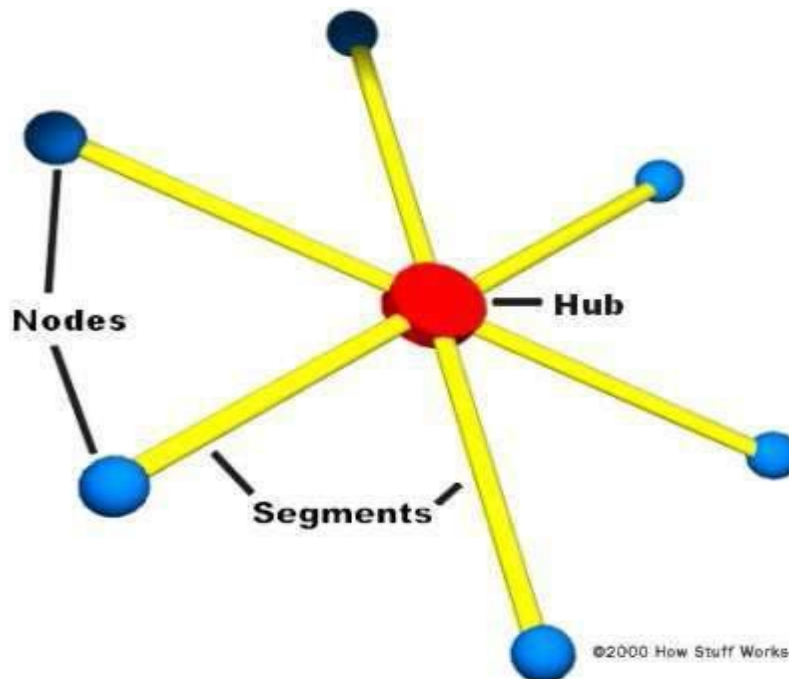
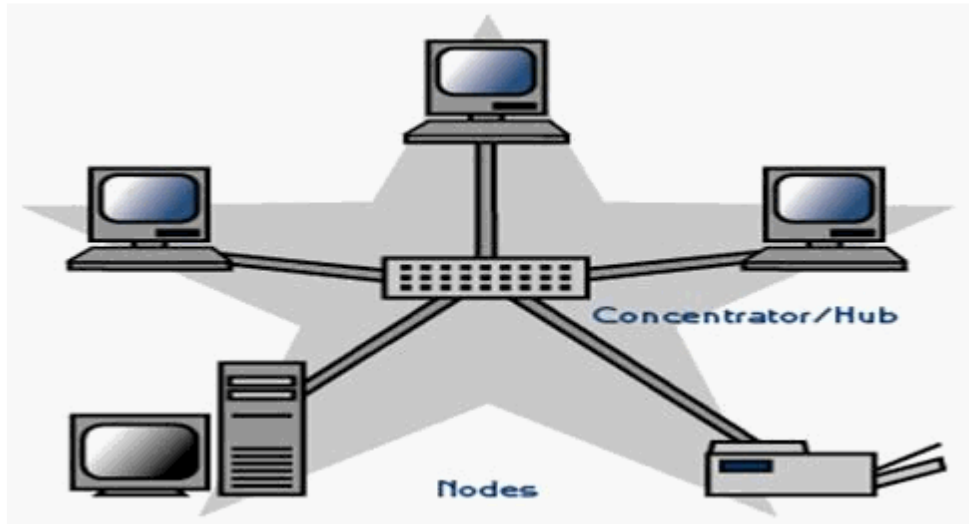
۳-۳ انواع توپولوژی های *Point To Point* :

۳-۳-۱ *IMP* ها دو به دو به هم متصل اند و یک حلقه (*loop*) را تشکیل می دهند. در *loop* تمام *IMP* ها در لایه ۳ قرار دارند.



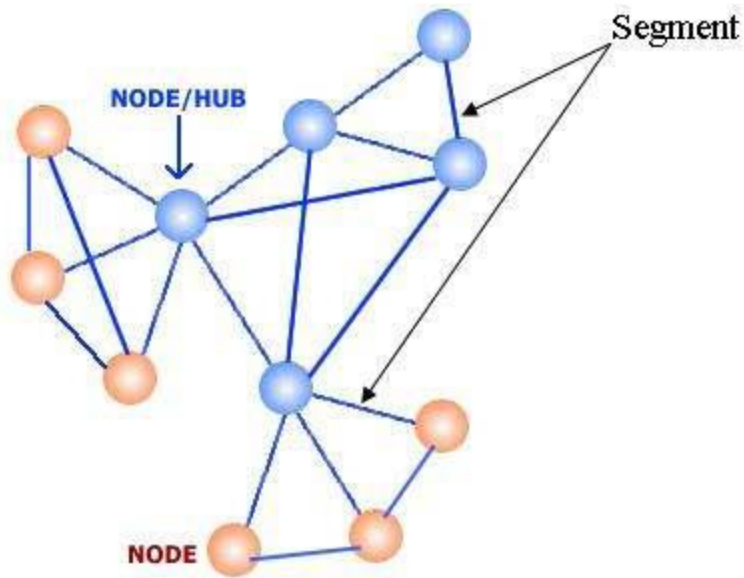
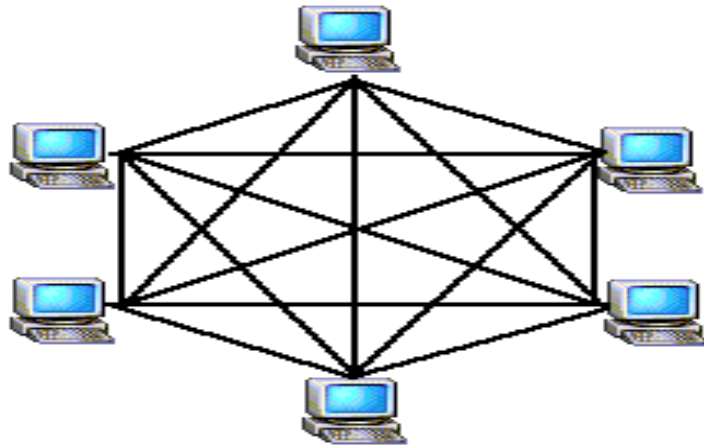
۳-۳-۲ star :

نکته قابل توجه در این بخش این است که IMP وسط حتما باید لایه ۳ باشد زیرا باید مسیریابی را انجام دهد اما IMP های کنارمی توانند لایه ۱ و لایه ۲ نیز باشند.



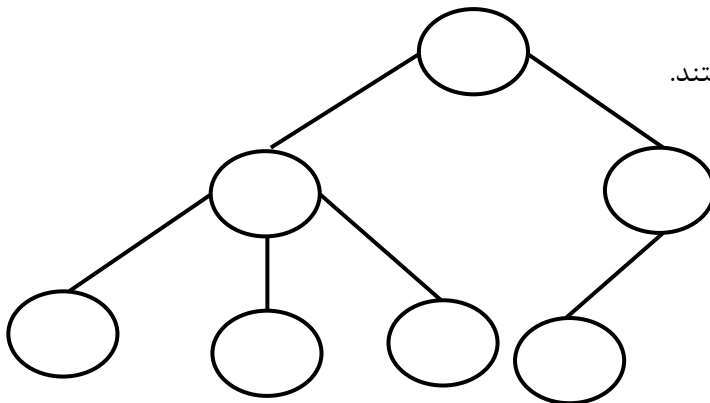
۳-۳-۳ شبکه کامل یا *Full mesh* :

در این نوع توپولوژی تمامی ارتباطات ممکن وجود دارد. هر جا سرعت بالا بود نیاز است این شبکه را استفاده کنیم. تمام *IMP* ها باید در لایه ۳ باشند زیرا همه مسیریابی می کنند.



۴-۳-۳ شبکه درختی :

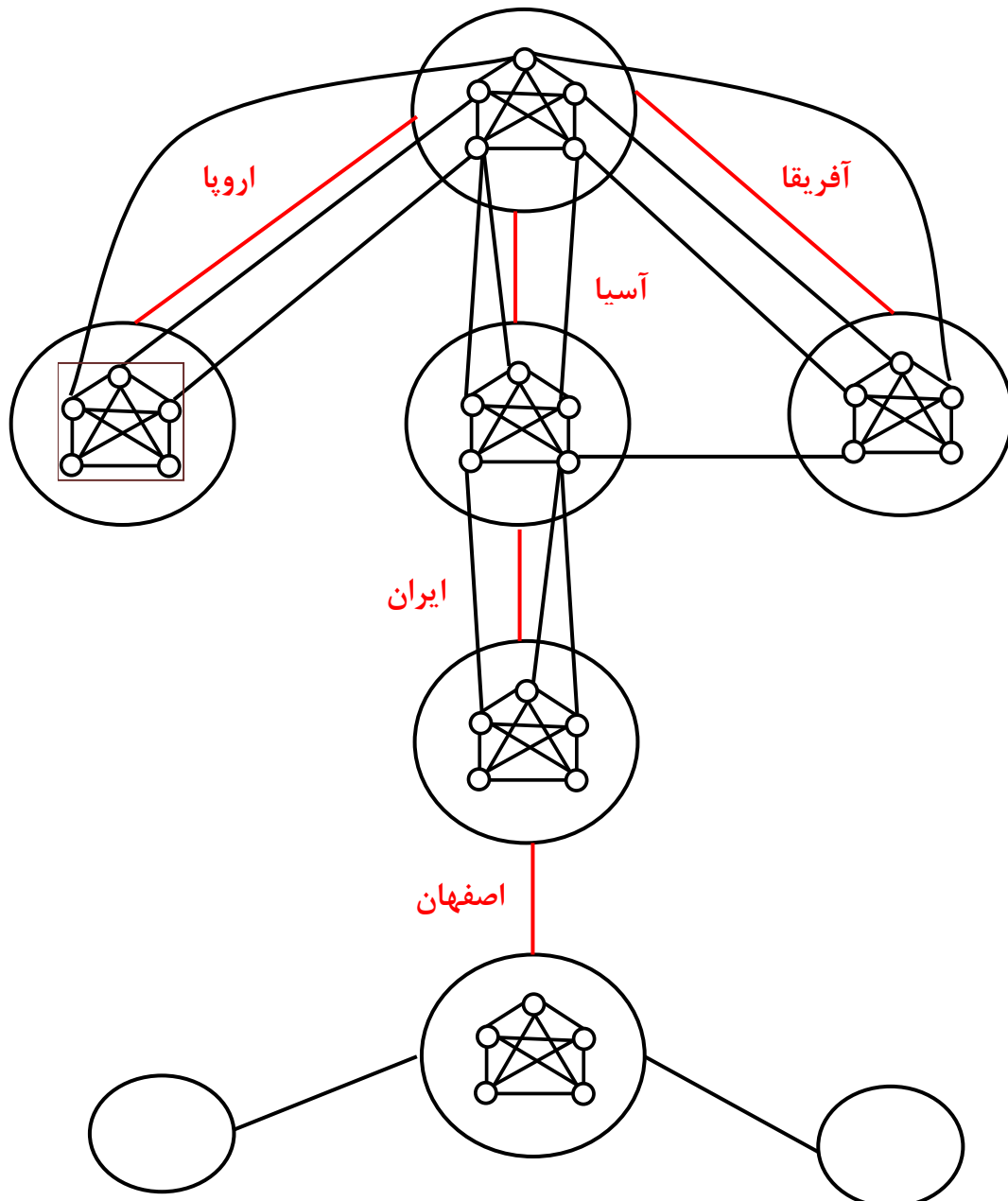
پدرها در این گراف حتما لایه ۳ هستند.



۳-۴ توپولوژی اینترنت:

ساختار اینترنت بیشتر به توپولوژی درختی نزدیک است اما همه توپولوژی های گفته شده در آن کاربرد دارد. روند کار در این توپولوژی بدین شکل است. یک *Node* اصلی داریم که در آمریکا است (۱۶ سرور در آن قرار دارد). این *Node* اصلی دارای چندین فرزند مانند اروپا، آفریقا، آسیا می باشد و هر کدام از فرزند ها هم به چند قسمت تقسیم می شوند. در هر *Node* یک ساختار *full mesh* داریم. اما در کل مثل یک ساختار درختی است. تمام *full mesh* فیبر نوری هستند

نکته: درختی کامل نیست و می تواند به روش های دیگر هم به هم متصل شوند.



فصل چهارم:

معماری شبکه

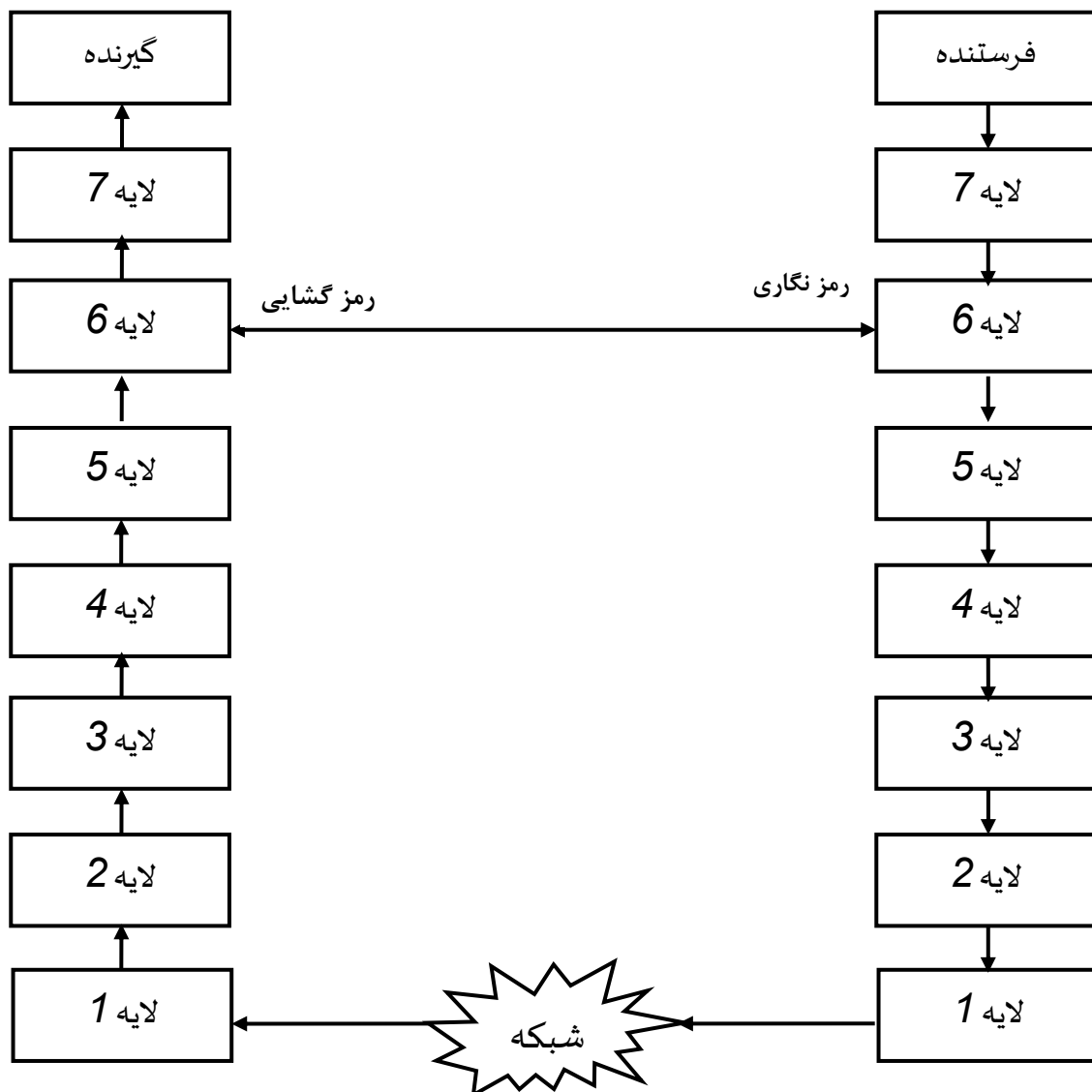
۴-۱ معماری شبکه:

منظور از معماری شبکه مرحله‌ای است که بر روی اطلاعات برای ارسال آنها در شبکه انجام می‌شود.

۴-۲ معماری (Open System Interconnection) OSI :

مدل عمومی است که به عنوان مدل مرجع برای شبکه استفاده می‌شود. یعنی مدل‌های عملی از روی این مدل ساخته می‌شوند.

توضیح شکل زیر: برای فرستادن پیغام هفت لایه کار روی آن انجام می‌شود که به آن معماری لایه‌ای گفته می‌شود. یعنی هر عملیات در یک بخش مستقل است. شاید این سوال به ذهن شما برسد که چرا از معماری لایه‌ای استفاده می‌شود؟ زیرا پیاده‌سازی آن راحت‌تر خواهد بود.



تعریف *protocol*: ارتباطات بین لایه های متناظر بین فرستنده و گیرنده را نشان می دهد.

۳-۴ لایه های مدل *OSI* :

۳-۴-۱ لایه ۷: *Application* یا کاربرد :

اطلاعاتی که از کاربر می گیرد . در این لایه پیغام (*message*) نام دارد. در هر لایه با توجه به عملیاتی که آن انجام می دهد یک *header* به پیغام اصلی اضافه می شود.



۳-۴-۲ لایه ۶: *presentation* یا نمایش:

در این لایه کارهای زیادی انجام می شود که ۳ تای آنها مهم هستند (۱) فشرده سازی (۲) رمز نگاری (۳) تبدیل استاندارد ها (استاندارد های فرستنده و گیرنده را به هم تبدیل می کند) در این لایه پیغام (*message*) نام دارد .

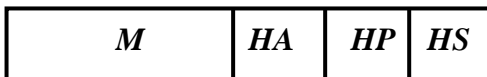
نکته : امنیت در تمام لایه ها می تواند باشد ولی رمز نگاری فقط در لایه ۶ امکان پذیر است.

چرا به این لایه نمایش می گویند؟ چون اطلاعات را قابل نمایش می کند.



۳-۴-۳ لایه ۵: *session* یا جلسه :

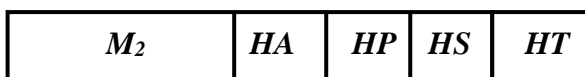
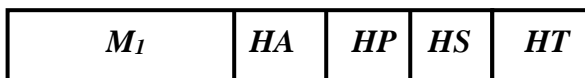
ارتباط شما را با یک مقصد که لایه ۵ است را کنترل می کند.(این کنترل می تواند کنترل زمانی یا کنترل دسترسی باشد) به طور کلی مدیریت ارتباطات یا مدیریت جلسه و کنترل آن بر عهده این لایه می باشد. در این لایه پیغام (*message*) نام دارد .



نکته : لایه های ۵ و ۶ نرم افزاری هستند.

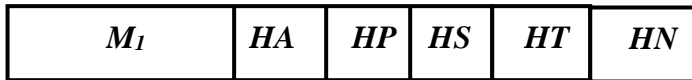
۳-۴-۴ لایه ۴: *Transport* یا لایه انتقال :

لایه انتقال یا حمل دو وظیفه مهم دارد (۱) پیغام را به بسته های کوچکتر تبدیل می کنند جهت انتقال ساده تر (۲) تعیین نوع کانال ارتباطی (دوطرف یا یک طرفه) پیغام شکسته شده در این مرحله را *segment* گویند



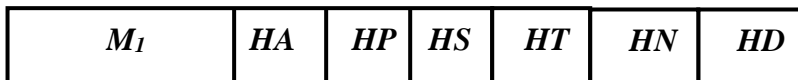
۴-۴-۵ لایه ۳: *Network* یا شبکه:

وظیفه لایه سوم مسیر یابی است. مسیر یابی یعنی انتخاب بهترین مسیر ممکن تا مقصد مورد نظر. دو وظیفه دیگر این لایه (۱) کنترل ازدحام (۲) کنترل بن بست می باشد. در ضمن ساده ترین کار این لایه بر روی پیغام، مشخص نمودن مبدا و مقصد است. در این لایه پیغام (*paket*) نام دارد

۴-۴-۶ لایه ۲: *Data link* یا پیوند داده ها:

وظایف زیادی دارد اما این دو از همه مهم تر هستند: (۱) کنترل خطا که البته این امکان در لایه ۴ نیز وجود دارد (۲) ایجاد فریم

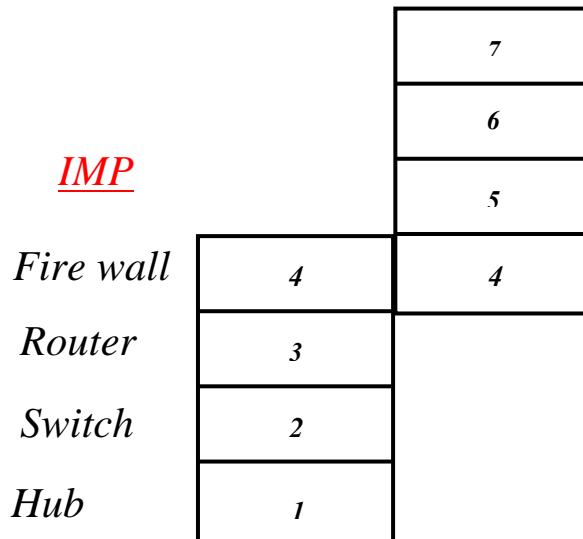
HD یا همان *Header Data Link* حاوی چیست؟ اطلاعاتی که فرستنده بر روی پیغام قرار می دهد که گیرنده به کمک آن مدیریت خطا را انجام دهد یعنی ببیند خطا کجاست و آن را اصلاح کند.

۴-۴-۷ لایه ۱: *physical* یا فیزیکی:

فریم را که دریافت کرد به صورت بیت های پشت سر هم انتقال می دهد.

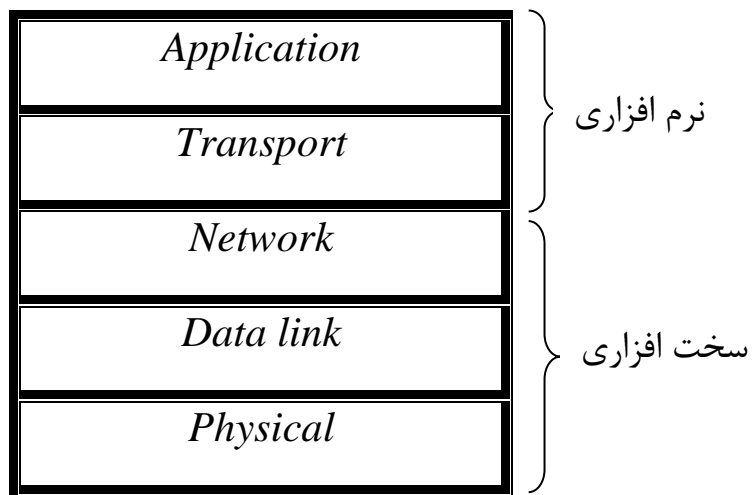
۴-۵ جایگاه لایه شبکه :

در شبکه *End System* ها وجود دارند که لایه ۵ و ۶ و ۷ به کامپیوتر وصل اند . لایه های ۱ و ۲ و ۳ در *IMP* هستند و لایه ۴ بعضی وقت ها در *End System* و بعضی وقت ها در *IMP* قرار می گیرند.



۴-۶ مدل *TCP/IP* :

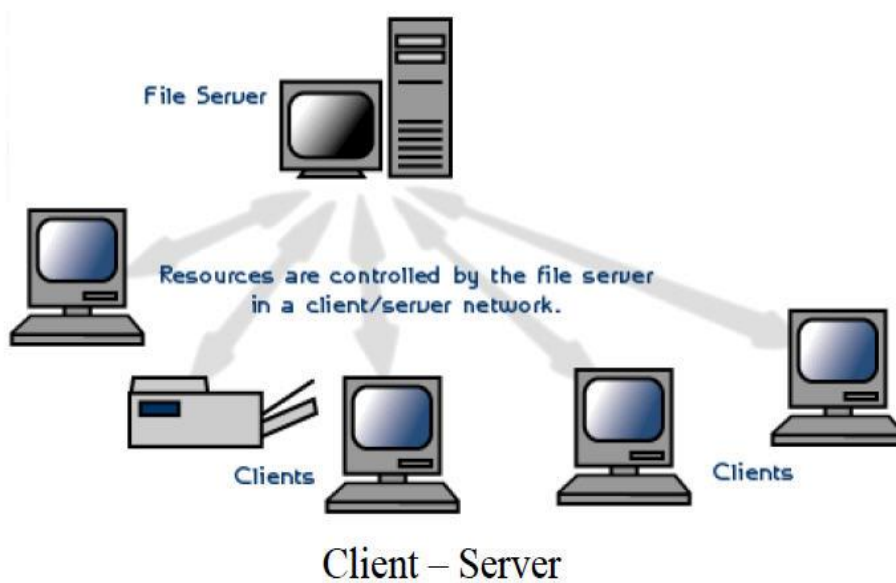
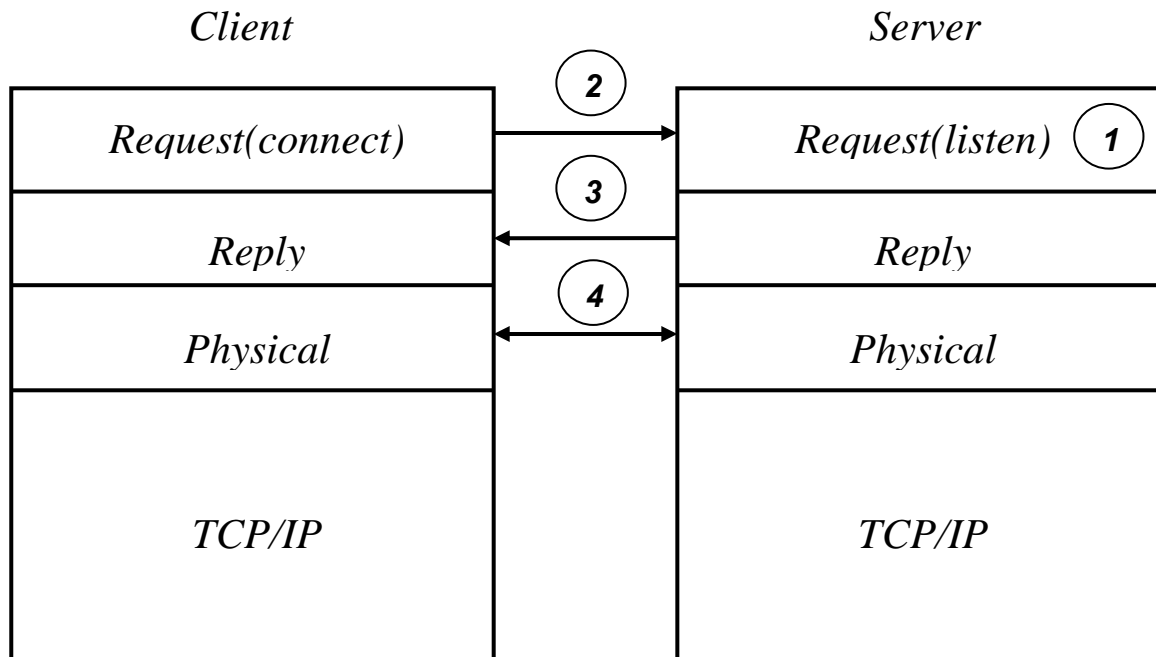
پروتکلی است که امروزه از آن برای ارتباط بین کامپیوتر ها هم در اینترنت و هم در شبکه های محلی استفاده می شود



Network Access

۷-۴ مدل Client/server :

یک مدل کاملا نرم افزاری است. که بر روی یکی از مدل های شبکه اجرا می شود. روند کار بدین صورت است
 مرحله (۱) یک server ابتدا گوش به زنگ دریافت یک اتصال از طرف Client است
 (۲) در خواست Client از server (3 server می تواند بپذیرد یا نپذیرد (۴) در لایه بعد از طریق لایه فیزیکی
 می تواند انتقال اطلاعات داشته باشد.



فصل پنجم:

لایه پیوند داده ها

۵-۱ ماهیت خطا :

وقتی که یک پیغام از طرف فرستنده به صورت M ارسال شود اما به صورت M' به دست گیرنده برسد می گویند خطا رخ داده است

فرستنده $M=11011101$

گیرنده $M'=11000011$

در واقع خطا (E) : به این معنا است که بیت های صفر به یک تبدیل شوند و یا بر عکس. چرا خطا ایجاد می شود؟ بر اثر وجود نویز در مدیا

فرستنده $M = 110110101$

(+)

خطا $E = 000111000$

گیرنده $M' = 110001101$

$E = 000111000$

$M = 110110101$

۵-۲ ویژگی های خطا

(۱) ناگهانی بودن خطا

(۲) انفجاری بودن خطا

(۱) ناگهانی بودن خطا : خطا قابل پیش بینی نیست. یعنی هیچ شبکه بدون خطایی وجود ندارد .

(۲) انفجاری بودن خطا : یعنی چندین بیت پشت سر هم از بین می رود نه یک بیت . خطا ماهیت دقیقی ندارد که سر یک بیت خاص برود و فقط آن را خراب کند.

۵-۳ نحوه برخورد با خطا

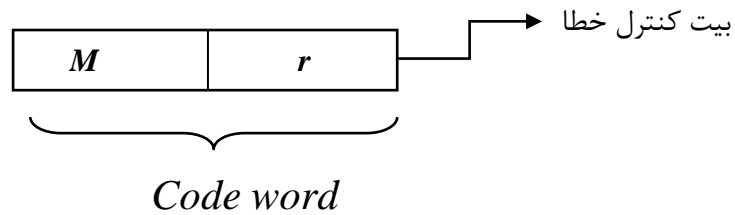
۵-۳-۱ Error detection یعنی کشف خطا:

در این روش اگر گیرنده متوجه شد پیغام خطا دارد به فرستنده اعلام می کند پیغام را مجددا ارسال کند. اما قادر به تصحیح آن نیست.

۵-۳-۲ Error Recovery یعنی اصلاح خطا:

گیرنده علاوه بر تشخیص خطا خود اقدام به اصلاح خطا می نماید.

: Code word



۵-۴ روش کشف خطای تکی:

۵-۴-۱ توازن زوج:

در فرستنده فقط یک بیت اضافه می کند . (در ضمن مقدار این یک بیت را طوری در نظر می گیریم که تعداد یک های آن زوج شود)

	m	r
فرستنده :	۱۱۰۱۱۰۱۱	۰
1 bit		
گیرنده :	۱۱۰۱۰۱۱۰	۰

نکته : در این روش اگر در دو بیت خطا رخ دهد گیرنده قادر به تشخیص آن نخواهد بود زیرا خطاها یکدیگر را

۱۱۰۱۰۱۱۱	۰
----------	---

می پوشانند

۵-۴-۲ روش اصلاح خطای تکی :

تعریف فاصله همینگ : حداقل تعداد بیتی که در یک روش کنترل خطا تغییر کند یا خراب شود ولی گیرنده متوجه خطای آن نشود مثلا فاصله همینگ در توازن زوج حداقل ۲ بیت است (یعنی اگر دو بیت خراب شود متوجه نخواهد شد)

فاصله همینگ: رابطه ی بین فاصله همینگ ، تعداد کشف خطا ، و تعداد اصلاح خطا را بیان می کند.

فاصله همینگ	تعداد کشف خطا	تعداد اصلاح خطا
d	$d - 1$	$d - 1/2$
3	2	$2/2 = 1$

نکته: اگر بخواهید در گیرنده یک بیت را اصلاح کنید باید از روشی استفاده کنید که فاصله همینگ در آن ۳ باشد .

کاربرد فاصله همینگ چیست؟ فاصله همینگ با تعداد بیت های r در ارتباط است . اگر بخواهید فاصله همینگ ۳ شود بیت های r از رابطه زیر به دست می آید

$$M+r+1 \leq 2^r$$

M : تعداد بیت های پیام اصلی.

r : مجهول است در واقع نشان دهنده تعداد بیت هایی است که باید به پیغام فرستنده اضافه شود .

مثال (اگر $M = (1101011011)$ ، r چند است؟

$$M = \underbrace{\begin{array}{|c|c|} \hline 1101011011 & 0 \\ \hline \end{array}}_{\text{Code word}}$$

$$r = ?$$

$$10+r+1 \leq 2$$

$$r = 1 \quad 12 \leq 2$$

$$r = 2 \quad 13 \leq 4$$

$$r = 3 \quad 14 \leq 8$$

$$r = 4 \quad 15 < 16 \quad \checkmark$$

۵-۵ چند جمله ای معادل یک پیغام:

هر جا یک است X را می نویسیم و هر جا صفر است X را نمی نویسیم.

43210
مثال ۱) ۱۱۰۱۰

$$X^4 + X^3 + X$$

321
مثال ۲) ۱۰۰۱

$$X^3 + 1$$

۵-۶ عملیات ماژول ۲:

عملیات ریاضی بر روی اعداد باینری که جمع و تفریق در آن به صورت XOR است.

۵-۷ چند جمله ای $G(x)$:

یک پیغام عمومی است که بین فرستنده و گیرنده توافق می شود. فرستنده توسط این پیغام یعنی $G(x)$ ، $Codeword$ یا همان $T(x)$ را می سازد و گیرنده توسط این پیغام اصلاح خطا را انجام می دهد.

$T(x)$: فرستنده

.

.

.

اصلاح خطای تکی $\xrightarrow{G(x)}$ $T'(x)$ گیرنده

۵-۸ روش الگوریتم CRC :

یکی از الگوریتم های اصلاح خطای تکی الگوریتم CRC است. که ۲ بخش دارد یکی فرستنده (سمت فرستنده) و یکی گیرنده (سمت گیرنده). در طرف فرستنده اولین کار یک $G(x)$ باید بدست می آورید.

I - تعیین $G(x)$: یک پیغام عمومی است که ۲ شرط دارد

I از درجه r باشد. با توجه به پیغام r بدست می آید سپس $G(x)$ را از فرمول $r+I$ محاسبه می شود. }
 ۲) بیت های اول و آخر آن حتما یک هستند ولی بیت های وسط مهم نیستند.

$$M=1101011011$$

$$r = 4$$

$$G(x)=5$$

$$10001$$

یکی توسط فرستنده انتخاب می شود و استاندارد می شود. \rightarrow 10011

$$10101$$

.

.

.

$$11111$$

۲- به تعداد r بیت صفر به سمت راست پیغام اضافه می شود.

$$11010110110000$$

۳- باقی مانده تقسیم مازول ۲ پیغام مرحله ۲ بر $G(x)$ را بدست می آوریم.

11010110110000	10011
10011	111100
10011	
10011	
10110	
10011	
10100	
10011	
1110	

۴- باقی مانده تقسیم را با پیغام مرحله 2 XOR می کنیم. که جواب مرحله ۴ بدست می آید جواب این بخش در

اصل همان *code word* است.

11010110110000	
\oplus	1110
11010110111110	
	} r
} Code word	

نکته مهم: اگر در طرف گیرنده $T(x)$ را بخواد باید به ۱ و ۰ ش، زیر عمل کند.
37

$$T(x) = T'(x) \oplus E(x)$$

$$\frac{T'(x)}{G(x)} = \frac{T(x)}{G(x)} \oplus \frac{E(x)}{G(x)}$$

همیشه صفر است

$$\frac{T'(x)}{G(x)} = \frac{E(x)}{G(x)}$$

نکته: $E(x)$ فقط دارای یک بیت با مقدار یک می باشد البته این بیت همان بیت اول است.

1- ابتدا باید باقی مانده این تقسیم را به دست آورید. $T'(x)$ اگر باقی مانده صفر شد پس خطا وجود نداشته است
G(x)

در غیر اینصورت باید 2 مرحله زیر را انجام دهید.

2- بدست آوردن $E(x)$

$$T(x) = T'(x) + E(x) - 3$$

مثال) $T(x)$ را محاسبه کنید؟

$$T'(x) = 11010111111110$$

$$G(x) = 10011$$

مرحله 1:

11010111111110	10011
<u>10011</u>	1
10011	
<u>10011</u>	
11111	
<u>10011</u>	
11001	
<u>10011</u>	
10101	
<u>10011</u>	
1100	

$$\begin{array}{r}
 \text{مرحله دو : } E(x) \ 1000000 \\
 \underline{10011} \\
 \boxed{1100} \\
 \text{38} \\
 \text{مرحله سه : } 11010111111110 \\
 \text{+ } 1000000 \\
 \hline
 T(x) = 11010110111110
 \end{array}$$

تمرین : مقدار $T(x)$ را به دست آورید.

I - ابتدا باید باقی مانده تقسیم $T'(x)$ را به دست می آوریم.
 $G(x)$

$$T'(x) = 11010110111010$$

$$G(x) = 10011$$

$$\begin{array}{r}
 11010110111010 \\
 \underline{10011} \\
 10011 \\
 \underline{10011} \\
 10011 \\
 \underline{10111} \\
 10011 \\
 \underline{10001} \\
 10011 \\
 \underline{100}
 \end{array}$$

2- مرحله دو به دست آوردن $E(x)$ است اما قبل از آن باید به این نکته توجه داشت که $E(x)$ عددی است که باقی مانده تقسیمش بر $G(x)$ ما را به باقی مانده $T'(x)$ بر $G(x)$ می رساند.

در این جا ما قصد داریم اولین عددی که بر $G(x)$ تقسیم شود و باقی مانده آن 100 شود و فقط یک بیت یک داشته باشد و بقیه بیت های آن صفر باشد را محاسبه کنیم.

$$\begin{array}{r}
 E(x): \ 100 \\
 \underline{0} \\
 \boxed{100}
 \end{array}$$

نکته I : در صورتی که باقی مانده دارای یک بیت با مقدار I باشد خود نشان دهنده $E(x)$ است

نکته 2: در تقسیم باینری مقدار عدد مهم نیست تعداد مهم است که باید یکسان باشند.

$$\begin{array}{r} 100000 \\ 10011 \overline{)10011} \\ \hline \end{array}$$

نکته 3: در صورتی که با قرار دادن تعدادی صفر برای $E(x)$ که بیشتر از تعداد ارقام $T'(x)$ بود به این معنی است که پیغام دریافت شده بیش از یک خطا بوده است.

3- مرحله آخر زمان محاسبه $T(x)$ است

$$\begin{array}{r} 11010110111010 \\ \oplus 100 \\ \hline T(x) 11010110111110 \end{array}$$

5-9 نحوه برخورد با خطای Burst:

در شبکه های کامپیوتری معمولاً بیش از یک بیت خراب می شود در صورتی که روش های بیان شده برای کشف خطای تکی می باشند برای حل این مشکل از روش زیر استفاده می کنیم.

در این روش از هر پیغام فقط یک بیت آن را ارسال می کنند در صورتی که یکی از این پیغام های جدید دچار خرابی شوند و چند بیت آنها خراب شوند از هر پیغام فقط یک بیت خراب شده و توسط الگوریتم CRC قابل اصلاح است.

T_1	1	1	0	1	0	1
T_2	1	0	1	1	0	1
T_3	1	0	0	1	1	1
T_4	0	1	1	0	0	1
T_5	1	0	1	0	0	1

5-10 تشخیص خرابی یا خطا توسط الگوریتم CRC:

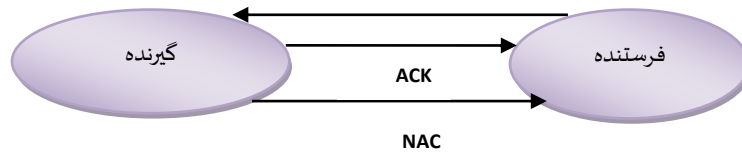
خطا یا در طرف فرستنده قابل تشخیص است یا در طرف گیرنده

1- در طرف گیرنده:

خطا را گیرنده تشخیص می دهد.

وقتی فرستنده یک پیغام را به گیرنده می هد و گیرنده پیغام را دریافت می کند اگر درست بود یا خطا داشت ولی قادر به اصلاح خطا بود به فرستنده پیغام می دهد که پیام را به درستی دریافت کرده است (ACK) اما اگر خطا را تشخیص دهد اما قادر به اصلاح آن نباشد پیغام (NAK) را که همان پیغام دریافت غلط است را ارسال می کند.

چرا گیرنده پیغام (ACK) را ارسال می کند؟ تا فرستنده بسته بعدی پیغام را ارسال کند.



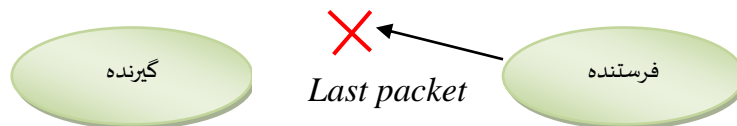
2- در طرف فرستنده:

40

خطا را فرستنده تشخیص می دهد .

اگر فرستنده پیغامی را ارسال کند اما وسط راه گم شود و اصلا به دست گیرنده نرسد گیرنده چون نمی دانست قرار است برای او پیغام ارسال شود جواب هم نمی دهد بنابراین خود فرستنده باید متوجه شود پیغام گم شده است.

هر فرستنده یک تایمر دارد که پاسخ گیرنده باید در زمان مشخص دریافت شود اگر زمان گذشته باشد می گوئیم *Time out* شده و به پیغام گم شده *Last packet* یا پیغام گم شده گوئند.



۵-۱۱ وظایف لایه پیوند داده ها

(۱) کنترل خطا

(۲) ایجاد فریم

ایجاد فریم: وظیفه دوم لایه پیوند داده ها ایجاد فریم است پیغامی که تا به حال ایجاد کردیم $T(x)$ ، سه بخش به آن اضافه می کنیم و آن را ارسال می کنیم. در لایه دوم *Frame* ارسال می شود.



Frame

قسمت های مختلف شکل بالا را به اجمال توضیح می دهیم

الف) *Type*: نوع فریم را مشخص می کند که سه نوع دارد:

(۱) *Data*: فرستنده به گیرنده ارسال می کند

(۲) *ACK*: پاسخ دریافت صحیح

(۳) *NAK*: پاسخ دریافت غلط

ب) FSN : شماره فریم ارسالی

تمام پیغام هایی که ارسال می شوند دارای FSN ، صفر هستند مگر اینکه فرستنده مجبور شود پیغام را دو یا چند بار ارسال کند. دفعه دوم FSN ، یک می شود پس می توان گفت FSN عملاً تعداد تکرار از ارسال یک پیغام را نشان می دهد.

نکته : تعداد حداکثر تکرار یک پیغام بستگی دارد به اینکه طول FSN چند بیت باشد مثلاً اگر طول FSN ، دو باشد تعداد ارسال ها به شرح زیر است.

0	0
---	---

 0

0	1
---	---

 1

1	0
---	---

 2

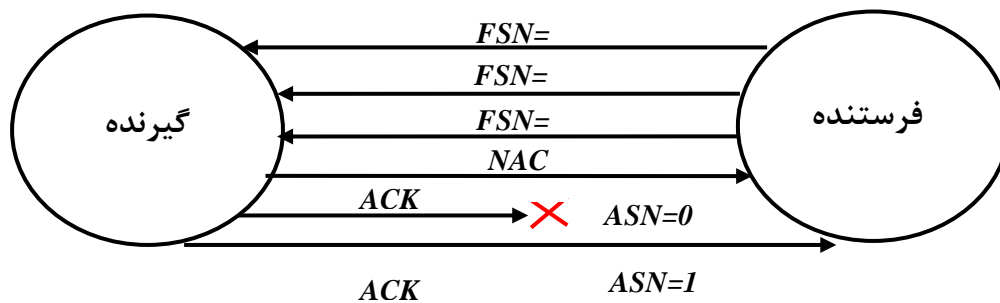
1	1
---	---

 3

ج) ASN : شماره ACK ارسالی

وقتی که یک فرستنده یک پیغام را به یک گیرنده می دهد اما بار اول که ارسال گیرنده دریافت کرد ولی با خطا، بار دوم درست دریافت کرده و پیغام ACK را ارسال می کند اما جوابش به دست فرستنده نمی رسد فرستنده فکر می کند اینبار پیغام وسط راه خراب شده است یکبار دیگر پیغام را ارسال می کند بار سوم گیرنده یکبار دیگر پاسخ ACK خود را با ASN ، یک به فرستنده ارسال می کند

نکته : تمام مواردی که برای FSN گفته شد برای ASN نیز صادق است اما در طرف گیرنده



کاربرد ASN و FSN :

برای جلوگیری از دریافت پیغام های تکراری و همچنین حفظ ترتیب ارسال و دریافت پیغام ها استفاده

فصل هشتم:

لایه شبکه

۶-۱ وظایف لایه شبکه عبارتند از

- ۱) مسیریابی
- ۲) کنترل ازدحام
- ۳) کنترل بن بست

مسیریابی :

برای مسیریابی از یک پیغام استفاده می شود به نام *Call Setup*. وظیفه این پیغام پیدا کردن مسیر مناسب جهت ارسال پیغام اصلی می باشد (قبل از اینکه پیغام اصلی ارسال شود یک *Call Setup* ارسال می شود که مسیر را مشخص می کند تا پیغام اصلی ارسال شود).

چگونه مسیریابی انجام می شود؟ مسیریابی توسط الگوریتم مسیریابی انجام می شود.

۶-۲ ویژگی های الگوریتم مسیریابی عبارتند از :

- ۱- الگوریتم باید ساده باشد : به دلیل اینکه الگوریتم های مسیریابی توسط *IMP* ها پیاده سازی می شوند و *IMP* ها پردازنده های قوی نیستند زیرا *CPU* ندارند بنابراین تا حد امکان باید ساده باشند.
- ۲- الگوریتم باید بهینه باشند: یعنی مسیریاب کوتاه ترین و خلوت ترین مسیر ممکن را انتخاب کند
- ۳- سازگار باشد: یعنی اگر یک مسیر اضافه یا کم شد بتواند مسیریابی را با توجه به مسیرهای جدید انجام دهد یعنی مسیریابی مناسب را ایجاد نماید. الگوریتم بایستی در صورت بروز تغییرات در شبکه بتواند خود را با این تغییرات وفق دهد.
- ۴- سرسخت باشد : یعنی بتواند حداقل یک مسیر تا مقصد پیدا کند.
- ۵- با توجه به نوع پیغام اولویت آن را تعیین نماید: یعنی اگر دو پیغام به *IMP* رسید باید بدانند اولویت کدام پیغام بالا تر است و بر اساس آن مسیریابی درست را انجام دهد.

۶-۳ انواع الگوریتم های مسیریابی :

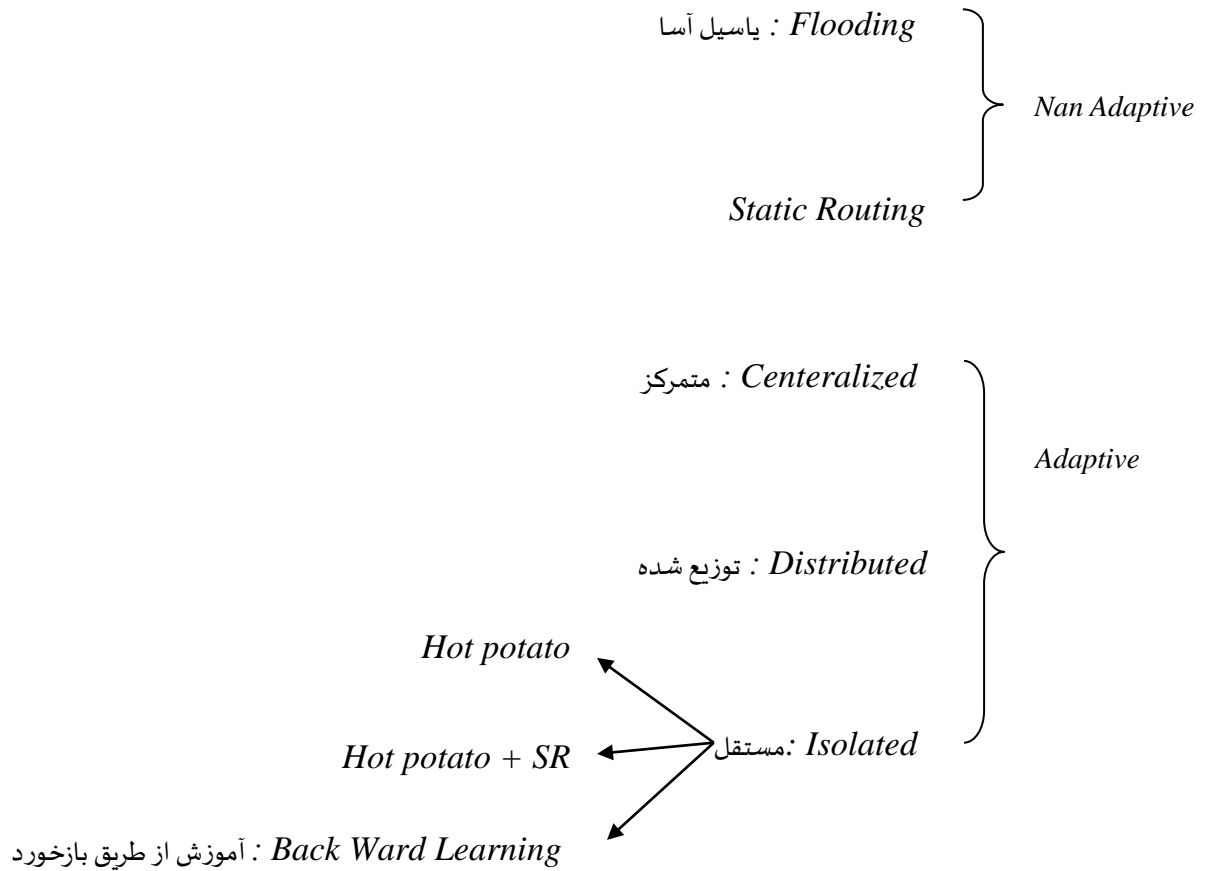
۱) غیر وفقی یا *Nan Adaptive* :

الگوریتم هایی هستند که تغییرات شبکه باید به صورت دستی در آنها اعمال شود یعنی اگر تغییراتی در شبکه صورت بگیرد این تغییرات خودکار اعمال نمی شود.

۲) وفقی یا *Adaptive* :

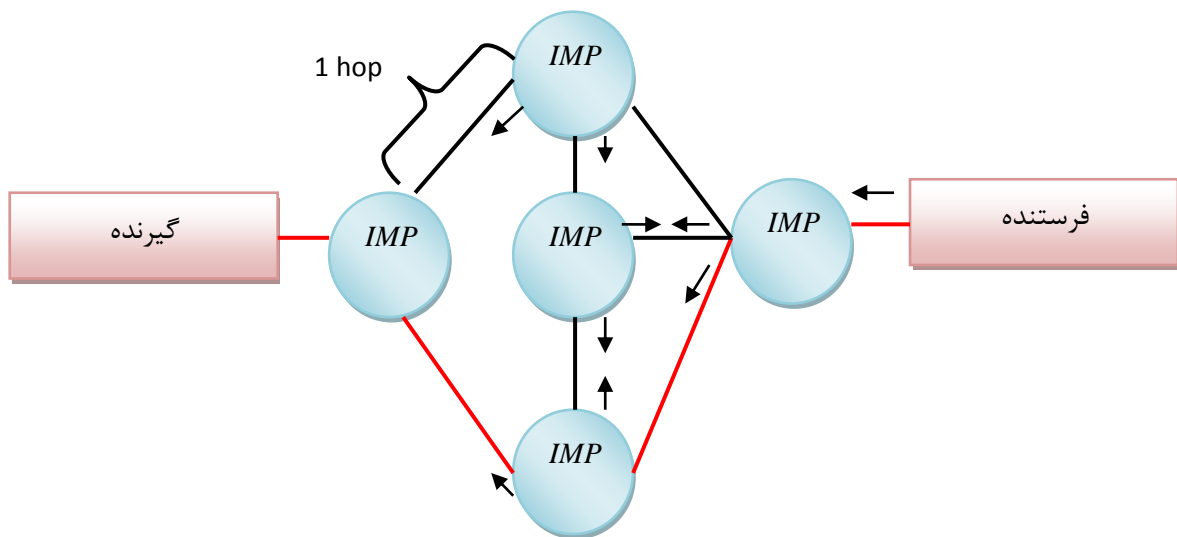
الگوریتم هایی هستند که تغییرات به صورت خودکار در مسیریابی اعمال می شود و دیگر لازم نیست خودمان دستی تغییرات را به الگوریتم بدهیم.

۴-۶ دسته بندی الگوریتم های مسیریابی



۱ - Flooding یاسیل آسا :

در این روش فرستنده تعداد زیادی *Call setup* را به شبکه ارسال می کند تا در مسیرهای مختلف حرکت کنند. *Call setup* یی که زودتر به گیرنده برسد و برگردد نشان دهنده بهترین مسیر انتخاب شده است.



نکته : تمام دایره ها در این فصل مسیریاب لایه ۳ هستند چون در مورد شبکه است و باید بتوانند مسیریابی کنند.

نکته ۲ : مسیریابی فقط مختص به یک پیغام نیست ممکن است هزاران پیغام قرار باشد از این شبکه عبور کنند.

نکته ۳ : حفظ مسیر بر عهده لایه حمل است.

مشکل *Call setup* های سرگردان :

Time To Live: TTL یک عدد است که حداکثر تعداد *hop* هایی که یک پیغام می تواند از آن عبور کند را نشان می دهد. *No-of-hop*.

hop : فاصله بین هر دو *IMP* لایه سه را *hop* می گوئیم.

نکته ۱ : از هر *Router* که رد می شویم یک *hop* محسوب می شود.

نکته ۲ : همه پیغام ها *TTL* دارند.

نکته ۳ : هر بار که یک *Call setup* از یک *IMP* رد می شود عدد *TTL* آن یک واحد کم می شود تا زمانی که به *TTL* به مقدار صفر برسد وقتی صفر شد *IMP* بسته را حذف می کند. در ضمن کار نداریم فاصله *hop* ها از یکدیگر یک متر یا صد کیلو متر باشند.

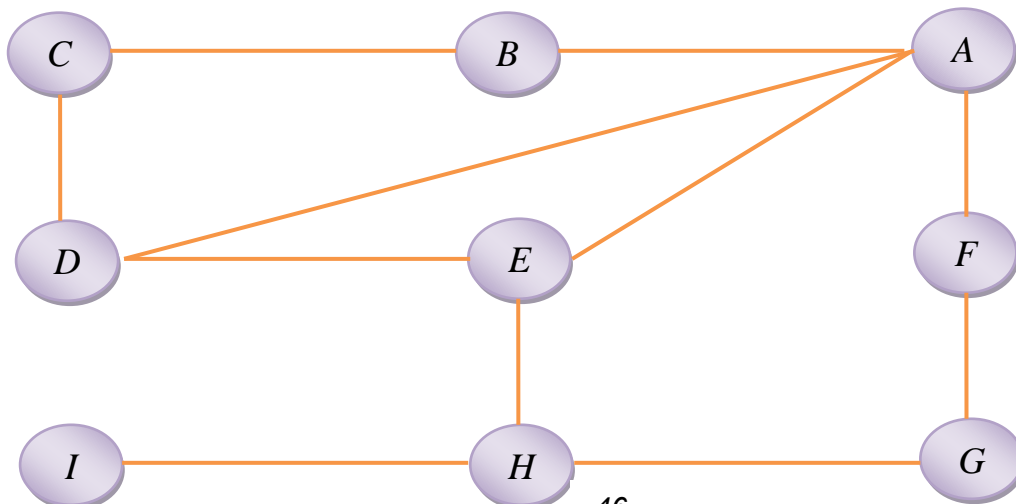
(۲) الگوریتم *Static Routing*

در این روش هر *IMP* دارای یک جدول مسیریابی است که به صورت دستی توسط مدیر شبکه تنظیم می شود.

نکته ۱ : در هر *IMP* جدول هایی داریم که اول از همه باید مقصد های مورد نظر را تعیین کنیم سپس برای هر مقصد چند مسیر تعریف می کنیم .

مسیر: منظور از مسیر *IMP* هایی هستند که در آن مسیر وجود دارند.

مقصد : *IMP* هایی هستند که قرار است بیشتر با آنها سروکار داشته باشیم. یا *IMP* هایی که ارتباطاتمان با آنها نسبت به بقیه بیشتر است در ضمن این موارد را مدیر شبکه به صورت دستی مشخص می کند.



نکته ۲: تعداد مقصد ها محدودیت ندارد و شما می توانید همه IMP ها را به عنوان مقصد معرفی کنید اما نیازی نیست، IMP هایی که بیشتر با آنها ارتباط داریم را به عنوان مقصد در جدول set می کنیم و از طریق آنها به سایر IMP ها دسترسی پیدا می کنیم.

ساختار جدول مسیریابی برای همه IMP ها به فرم زیر است

مثلا جدول مسیریابی IMP ، A را در زیر مشاهده می کنید:

مقصد ها	مسیر ۱	مسیر ۲	مسیر ۳
E	B .9	D .7	H .6
C	B	F	H
D	D	B	H

نکته ۳: به این مسیر ها (مسیر ۳ و ۲ و ۱) در شبکه اصطلاحاً گام بعدی یا $Next\ hop$ می گوئیم.

نکته ۴: کنار هر مسیریاب یک عدد نوشته می شود که نشان دهنده اولویت مسیر است در واقع این عدد نشان می دهد که کدام مسیر خوب و کدام مسیر بد است. به این عدد وزن مسیر می گویند. در ضمن وزن مسیر یک عدد بین صفر و یک است و به دو عامل (طول مسیر - خلوت بودن مسیر) بستگی دارد که این فیلد وزن نیز توسط مدیر و به صورت دستی تعیین می شود نکته ای که باید بدان توجه شود این است که هر چه عدد وزن مسیر به یک نزدیک باشد مسیر بهتر و خلوت تر است.

نکته ۵: عیب این روش در این است که اگر تغییری در شبکه رخ داد باید به صورت دستی در همه جداول مسیریابی اعمال شود.

انواع الگوریتم های وفقی

(۱) $Centralized$: متمرکز

یک روش وفقی است که بر اساس جدول مسیریابی کار می کند فرق این روش با روش های قبل در این است که یکی از IMP های شبکه مسئول ایجاد و به روز رسانی جداول مسیریابی در شبکه می شود این IMP ها هر چند وقت یکبار با ارسال پیغام در شبکه ساختار شبکه را دریافت می کند و توسط آن جداول مسیریابی را ایجاد و

بروزرسانی می کند (برای شبکه هایی که تعداد *IMP* های آن کم می باشد و تغییرات آن زیاد است کاربرد دارد یعنی شبکه اینترنت نمی تواند از آن استفاده کند).

(۲) الگوریتم *Distributed* :

این الگوریتم ها بیشترین استفاده را در شبکه های گسترده دارند منظور شبکه هایی است که تعداد *IMP* های آنها زیاد است. در این الگوریتم ها هر *IMP* برای ایجاد و بروزرسانی جداول فقط نیازمند اطلاعات همسایگان خود می باشد

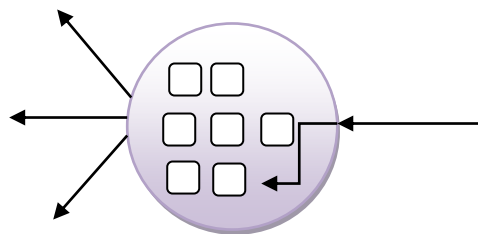
(۳) الگوریتم *Isolated* :

این الگوریتم ها فقط با استفاده از اطلاعات داخلی هر *IMP* مسیریابی را انجام می دهد

این الگوریتم به سه دسته زیر تقسیم می شود :

الف (*Hot potato*) :

طریقه کار این الگوریتم به این صورت است یک پیغام را به درون *IMP* می فرستند تا *IMP* آن را به یکی از ۳ مسیری که دارد هدایت کند. *IMP* پیغام را به مسیری می دهد که صف انتظارش از همه کمتر است. در اصل فقط می خواهد پیغام را عبور دهد تا بار ترافیک شبکه را کاهش دهد.



نکته ۱ : این الگوریتم فقط خلوت ترین مسیر را انتخاب می کند. اما همیشه خلوت ترین مسیر بهترین مسیر نیست زیرا بعضی از مواقع ممکن است این مسیر یک مشکلی داشته باشد که باقی *IMP* ها آن را انتخاب نکرده اند.

نکته ۲ : اشکالی که ممکن است در این روش به وجود بیاید این است که به جای اینکه شبکه خلوت شود بیشتر با شلوغی مواجه می شود چون وقتی یک بسته را سریع عبور می دهد باقی *IMP* ها فکر می کنند یک مسیر خوب است زیرا پیغام ها سریع رد شدند. پس تعداد پیغام های بیشتری به این مسیر می فرستند. پس به جای اینکه مسیر خلوت شود بیشتر شلوغ می شود که به این مشکل *Burning with Hot potato* (سوختن با سیب زمینی داغ) می گویند.

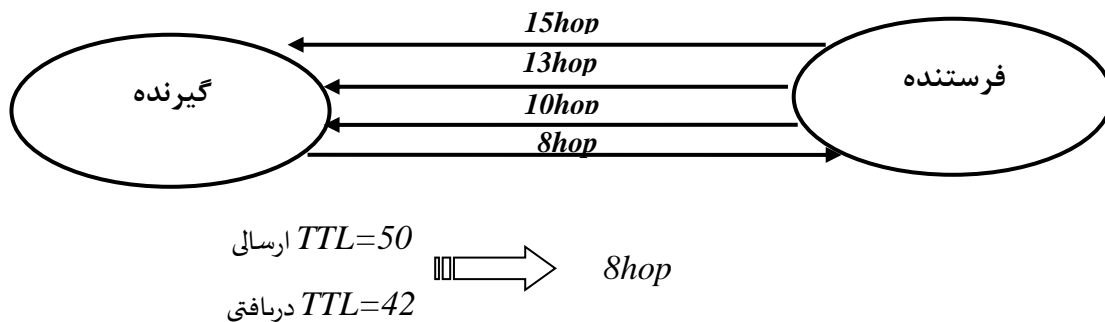
Hot potato + SR (ب)

در این روش علاوه بر خلوت بودن مسیر وزن مسیر را نیز در نظر می‌گیریم. وزن مسیر عددی بین ۱ و ۱۰ است. این الگوریتم یک مسیر را انتخاب می‌کند که هم کوتاه باشد و هم زیاد شلوغ نباشد یعنی وزن مسیر مساعد باشد.

ج) Back Ward Learning: آموزش از طریق بازخورد

یک فرستنده با یک گیرنده پیغام رد و بدل می‌کند. فرستنده مثلاً سه مسیر را می‌شناسد (یعنی در جدول مسیریابی خود تا گیرنده ذکر شده سه مسیر دارد) که به نظر خودش بهترین مسیر هایی بوده که می‌توانسته تا مقصد مورد نظر انتخاب کند. حالا گیرنده یک پاسخ می‌دهد. وقتی فرستنده پاسخ را دریافت می‌کند و مسیر آن را با مسیرهای ذخیره شده در جدول خود مقایسه می‌کند. می‌بیند این مسیر از سه مسیر قبلی کوتاه تر است پس این مسیر جدید را جایگزین یکی از مسیر های خود می‌کند.

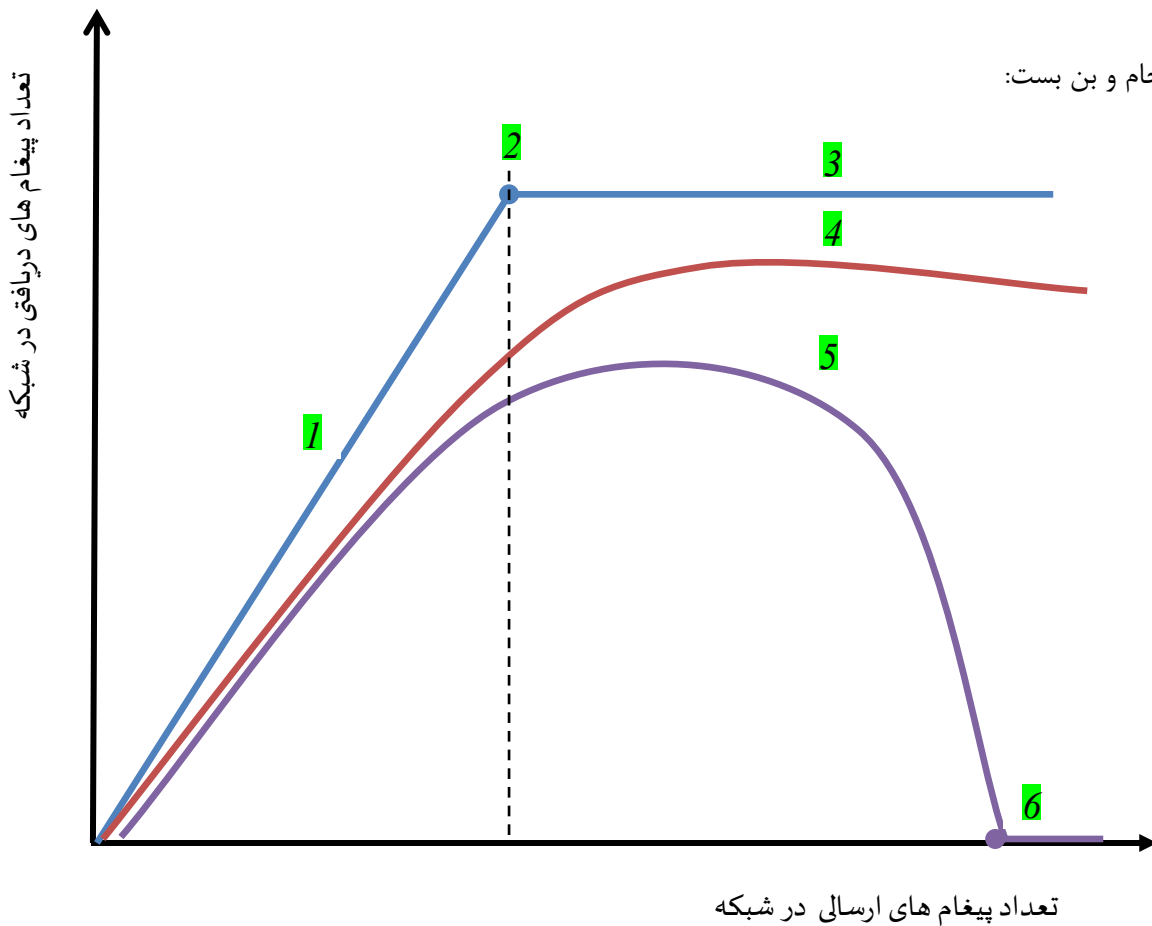
اما فرستنده از کجا متوجه می‌شود مسیر کوتاه تر بوده است؟ فرستنده طول تمام مسیر ها را در جدول دارد یعنی علاوه بر وزن مسیر طول مسیر را هم بر حسب hop دارد. فرستنده با توجه به TTL پیغام دریافتی می‌فهمد که طول مسیر چند hop بوده است. مثلاً اگر $TTL=50$ باشد و وقتی به فرستنده رسید $TTL=42$ شود یعنی طول آن ۸ بوده است.



توضیح خلاصه استاد: فرستنده پس از دریافت پیغام با توجه به TTL آن تعداد hop هایی که از گیرنده تا فرستنده طی کرده را محاسبه می‌کند با توجه به این عدد این مسیر را می‌تواند جایگزین یکی از مسر های قبلی کند.

نکته: آموزش از طریق بازخورد یعنی هنگامی که بسته را دریافت می‌کنیم می‌بینیم از مسیر کوتاه تری آمده یا نه

کنترل ازدحام و بن بست:



بخش ۱: در این بخش از نمودار یک روند رو به رشد را مشاهده می کنید. این بدین معنا است که هر تعداد پیغام ارسال شود همان مقدار دریافت می کند در واقع هیچگونه ازدحامی رخ نداده است. یعنی شبکه ازدحام ندارد در ضمن هیچ پیغامی در صف نمی ماند. میزان رشد این نمودار بستگی به گنجایش شبکه دارد.

بخش ۲: از این نقطه به بعد ازدحام رخ می دهد نقطه ازدحام به گنجایش شبکه و گنجایش شبکه به دو عامل پهنای باند و سرعت *IMP* ها بستگی دارد

بخش ۳: وقتی رشد نمودار تبدیل به خط صاف می شود یعنی همانقدر که ارسال داریم دریافت نداریم.

بخش ۴: حالت عملی: منحنی می شود و یک نقطه ثابت و مشخص به عنوان نقطه ازدحام نداریم.

بخش ۵: حال اگر حالتی پیش بیاید که آنقدر رشد آن کم شود تا دوباره به منحنی برسد به نقطه بن بست می رسیم.

بخش ۶: نقطه بن بست است از این نقطه به بعد هر چه ارسال کنیم چیزی دریافت نمی کنیم.

۵-۶ عوامل بروز ازدحام :

۱- کند بودن IMP ها :

IMPهایی که در وسط است یک سری ورودی و خروجی دارد که اگر کند باشد باعث بروز ازدحام می شود.



وظایف IMP ها :

✓ انتقال داده

✓ کنترل خطا

✓ مسیریابی

۲- ورودی بیش از گنجایش خروجی

اگر مثلاً ورودی $2mbps$ ولی خروجی آن $1mbps$ باشد یعنی اگر ورودی بیش از گنجایش خروجی باشد ازدحام

رخ می دهد.



۳- خود ازدحام باعث تشدید ازدحام در بخش های دیگر می شود.

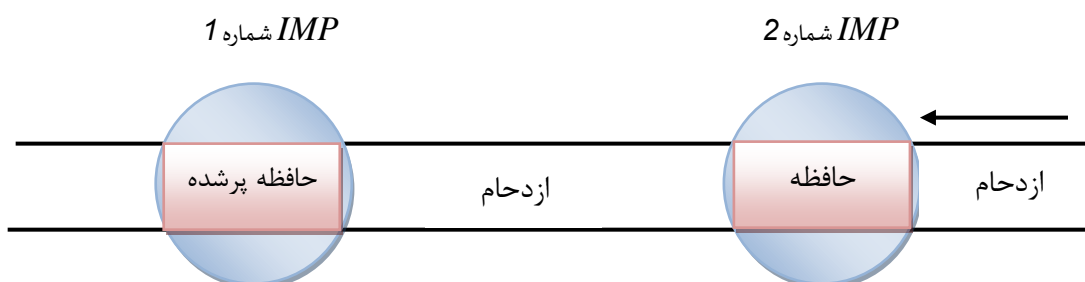
حافظه درون IMP شماره ۱ پر شده است پس IMP شماره ۲ نیز نمی تواند حافظه خود را خالی کند زیرا پیغام

ها که نمی توانند در وسط راه بمانند. در شبکه زمانی که پیغام ارسال می شود باید توسط گیرنده دریافت شود و

هیچ موقع پیام معلق نمی ماند بنابراین IMP شماره ۲ تا زمانی که در IMP شماره ۱ حافظه خالی نباشد پیغامی

را ارسال نمی کند پس ازدحام در IMP شماره ۱ باعث ازدحام در IMP شماره ۲ شد و این روند ازدحام به

صورت زنجیره ای در کل شبکه اتفاق می افتد .



۶-۶ روش های جلوگیری از بروز ازدحام:

(۱) پیش تخصیص بافر ها: یعنی برای اینکه ازدحام به وجود نیاید فرستنده قبل از ارسال پیغام از وجود بافر خالی در گیرنده مطمئن شود

(۲) حذف اختیاری *packet* ها: هر جا که ازدحام به وجود آمد به تعداد از پیغام هایی که در *IMP* هستند را حذف می کند. اما شاید این سوال به ذهن شما بیاید که از بین بسته ها کدام یک باید حذف شود؟ اولاً پیغام های *ACK* را حذف نمی کنیم زیرا حذف آن کمکی به کم کردن ازدحام نمی کند زیرا فرستنده در صورت عدم دریافت پاسخ *ACK*، پیغام را مجدداً ارسال می کند. از بین پیغام ها پیغامی اولویتش حذفش بیشتر است که *TTL* بیشتر دارد. *TTL* بزرگتر یعنی مسیری که باید طی کند بیشتر است پس باید آن را حذف کنید.

نکته: پیغامی که *TTL* بزرگتری دارد یعنی به مبدا نزدیک تر است پس مقرون به صرفه است آن را حذف کنیم.

(۳) کنترل جریان یا *Flow control*: این روش میگوید فرستنده بیش از گنجایش شبکه پیغام ارسال نکند.

بن بست به چند طریق رخ می دهد؟

۶-۷ بن بست به سه روش رخ می دهد ابتدا انواع آنها و سپس روش های کنترل آنها ذکر می شود:

۱- *Direct store & forward*

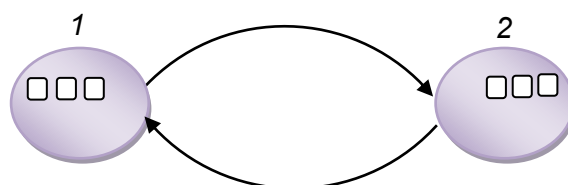
۲- *In Direct store & forward*

۳- بن بست بر اثر تقسیم پیغام

روش اول *Direct store & forward*:

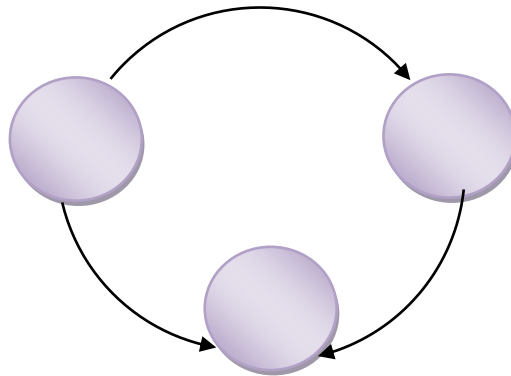
روش اول بین دو *IMP* مجاور اتفاق می افتد مثلاً *IMP* اول حافظه اش پر شده یعنی هر ۳ جای خالی اش پر شده و می خواهد داده ها را برای *IMP* دوم ارسال کند تا حافظه اش خالی شود. پس اولی منتظر است حافظه دومی خالی شود و دومی منتظر است حافظه اولی خالی شود و هیچ موقع هم ارسال و دریافت پیغام انجام نمی دهند بنابراین بن بست رخ داده است.

نکته: ما باید کاری کنیم که بن بست رخ ندهد زیرا معمولاً بن بست راه حل ندارد



روش دوم *In Direct store & forward* :

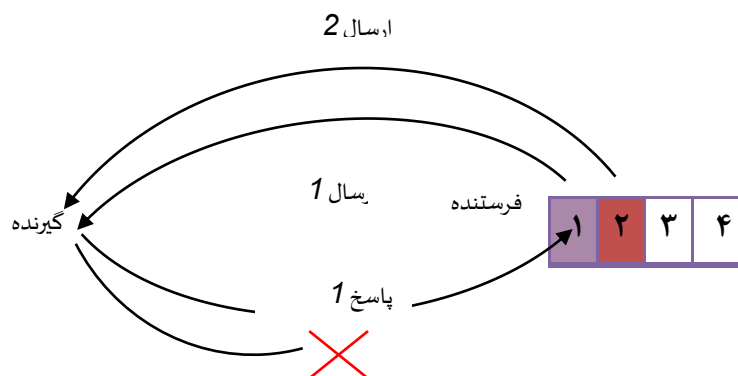
IMP اولی منتظر دومی ، دومی منتظر سومی و سومی منتظر اولی خواهد بود (حالت چرخش دارند). یعنی به صورت غیر مستقیم دچار بن بست شده اند اصطلاحاً به این ها گراف انتظار می گویند یعنی می توان برای آنها گراف انتظار تشکیل داد و در ضمن این *IMP* ها با هم تشکیل *LOOP* می دهند .



روش سوم بن بست بر اثر تقسیم پیغام :

زمانی رخ می دهد که فرستنده مثلاً پیغام را به چند بخش تقسیم کرده تا به گیرنده ارسال کند یک بخش از پیغام را که ارسال کرد باید پاسخ دهد و همینطور تا آخر حال اگر یکی از پاسخ ها به فرستنده نرسد چه اتفاقی می افتد؟ اگر فرستنده تایمر نداشته باشد گیرنده می گوید من جواب ۲ را دادم و منتظر پیغام ۳ است . فرستنده هم می گوید هنوز جواب ۲ را نگرفته ام و منتظر جواب می ماند و بن بست به وجود می آید یعنی این دو منتظرهم می مانند.

البته این مشکل را می توان با یک تایمر در طرف فرستنده حل کرد که مثلاً اگر در یک زمان مشخص پاسخ خود را نگرفت مجدداً اطلاعات قبلی را ارسال کند.

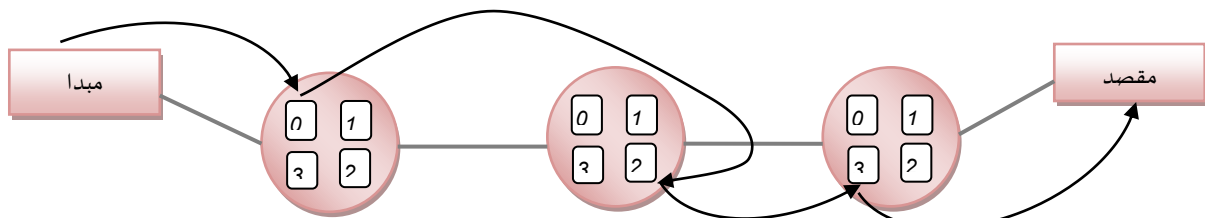


برای حل مشکل بن بست در روش های ۱ و ۲ به صورت زیر عمل می کنیم :

در این روش فرستنده پیغام خود را به اولین *IMP* در بافر شماره ۰ ی صفر که با آن در ارتباط است ارسال می کند شماره صفر در *IMP* فقط مربوط به *Host* خودش است اگر صفر اشغال بود باید صبر کند تا آزاد شود هر *IMP* می تواند پیغام را به بافر شماره بزرگتر (در صورت وجود) ارسال کند یعنی *IMP* ای که بسته را دریافت کرد می تواند به بافر ۱ و ۲ و ۳ بدهد.

نکته : مثلا اگر یک قبلا اشغال شده باشد می تواند به ۲ و اگر ۲ اشغال شده باشد به ۳ بدهد ولی اگر ۳ اشغال بود با وجود اینکه ۱ و ۲ خالی باشند بسته را حذف می کند اما در این جا ۳ پر نبوده

نکته ۲ : در صورتی که بافری با شماره بزرگتر وجود نداشته باشد پیغام را حذف می کند حذف به این دلیل انجام می شود که بن بست رخ ندهد.



فصل هفتم:

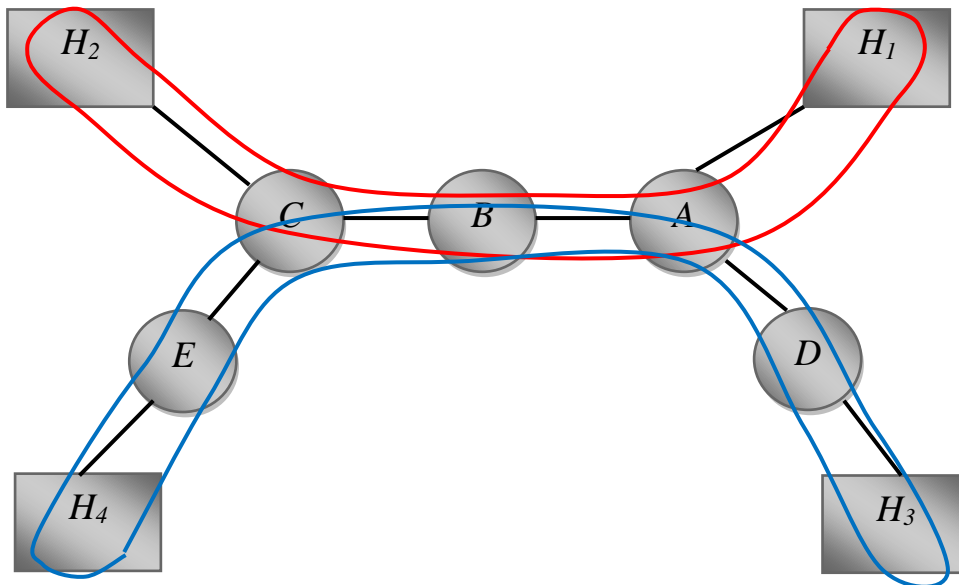
لایه حمل

۷-۱ لایه حمل نحوه انتقال داده ها را به یکی از ۳ روش زیر مشخص می کند :

- ۱) مدار مجازی *Virtual circuit*: کانال دوطرفه و پیغام را تقسیم می کند .
- ۲) سوئیچینگ پیغام *Message switching*: کانال یکطرفه و پیغام تقسیم نمی شود.
- ۳) سوئیچینگ بسته *Packet switchings*: کانال یکطرفه و پیغام تقسیم می شود .

۷-۱-۱ روش *Virtual circuit* یا روش *VC* :

از روشی به نام *Circuit switching* گرفته شده و درمخابرات استفاده می شود. بر روی سوئیچینگ مداری یکسری اصلاحات انجام می شود و روش *VC* به وجود می آید.



در روش *Circuit switching* کاری که انجام می شود یک مسیر دوطرفه اختصاصی بین مبدا و مقصد در نظر گرفته می شود اختصاصی یعنی H_3 و H_4 اگر خواستند با هم ارتباط داشته باشند باید صبر کند کانال خالی شود. عیب این روش این است که استفاده اشتراکی از این روش امکانپذیر نیست این روش در شبکه قابل پیاده سازی نیست.

اصلاح این روش : میگوید با اینکه کانال رزرو شده می توان زمان سکوت را به H های دیگر داد پس مسیر A, B مسیر اشتراکی می شود.

در روش های VC کانال ها کاملا اختصاصی نیستند. می توانند به صورت اشتراکی استفاده شوند

چرا روش VC نیازمند نگهداری مسیر است؟ چون مسیر دوطرفه است پیغام که می رود از همان مسیر هم پاسخش باید برگردد و برای پیغام های بعدی نیز باید از همان مسیر استفاده شود. چه کسی این مسیرها را نگهداری می کند؟ بر عهده ی IMP های لایه ۴ است که این کار را به کمک جدول های VC انجام می دهد جدول VC درون IMP ها قرار دارد. هر مسیری که تغییر کند جدول تغییر می کند.

نکته: بنابراین وظیفه جدول VC نگهداری مسیر ها است.

جدول ها VC شکل قبل:

C		
H ₂	15	B

B			
15	C	10	A
40	E	30	A

A			
10	B	--	H ₁
30	B	20	D

D		
H ₄	40	B

D			
20	A	--	H ₃

نکته مهم در B است: زیرا باید بفهمد که کدام بسته را به C و کدام را به E تحویل دهد لذا برای تشخیص مسیرها از عددی استفاده می کند که به آن برچسب می گویند.

نکته ۱: برای A (ورودی) برچسب نمی خواهد ولی لحظه ای که می گوید خروجی را به B تحویل بده یک عدد به عنوان Label به آن می چسباند که این عدد Random است.

نکته ۲: شماره خروجی قبلی با ورودی بعدی باید یکی باشد.

نکته ۳: در هر IMP در هر ستون نباید اعداد تکراری باشد.

نکته ۴: IMP ها در روش VC پیچیده ترند. IMP های لایه ۴ این توانایی را دارند که جداول VC را نگهداری کنند اما لایه ۳ ندارند.

ویژگی های روش VC :

(۱) قابلیت اطمینان:

قابلیت اطمینان یکی از ویژگی های بارز این روش است چون در این روش همه IMP ها دارای پاسخ اند (از همان مسیر دوطرفه) یعنی فرستنده مطمئن می شود پیغام به دست گیرنده رسیده است نام دیگر این روش $Connection oriented$ یا همان اتصال گراست یعنی پیغام ها باید حتما پاسخ داشته باشند. پس قابلیت اطمینان بالا است.

عیب آنها این است که

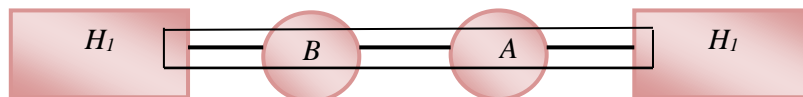
(۱) ترافیک بیشتر می شود.

(۲) IMP هایش پیچیده ترند.

نکته: همه IMP هایی که در اینترنت هستند لایه ۴ نیستند چون IMP های لایه ۴ گران هستند.

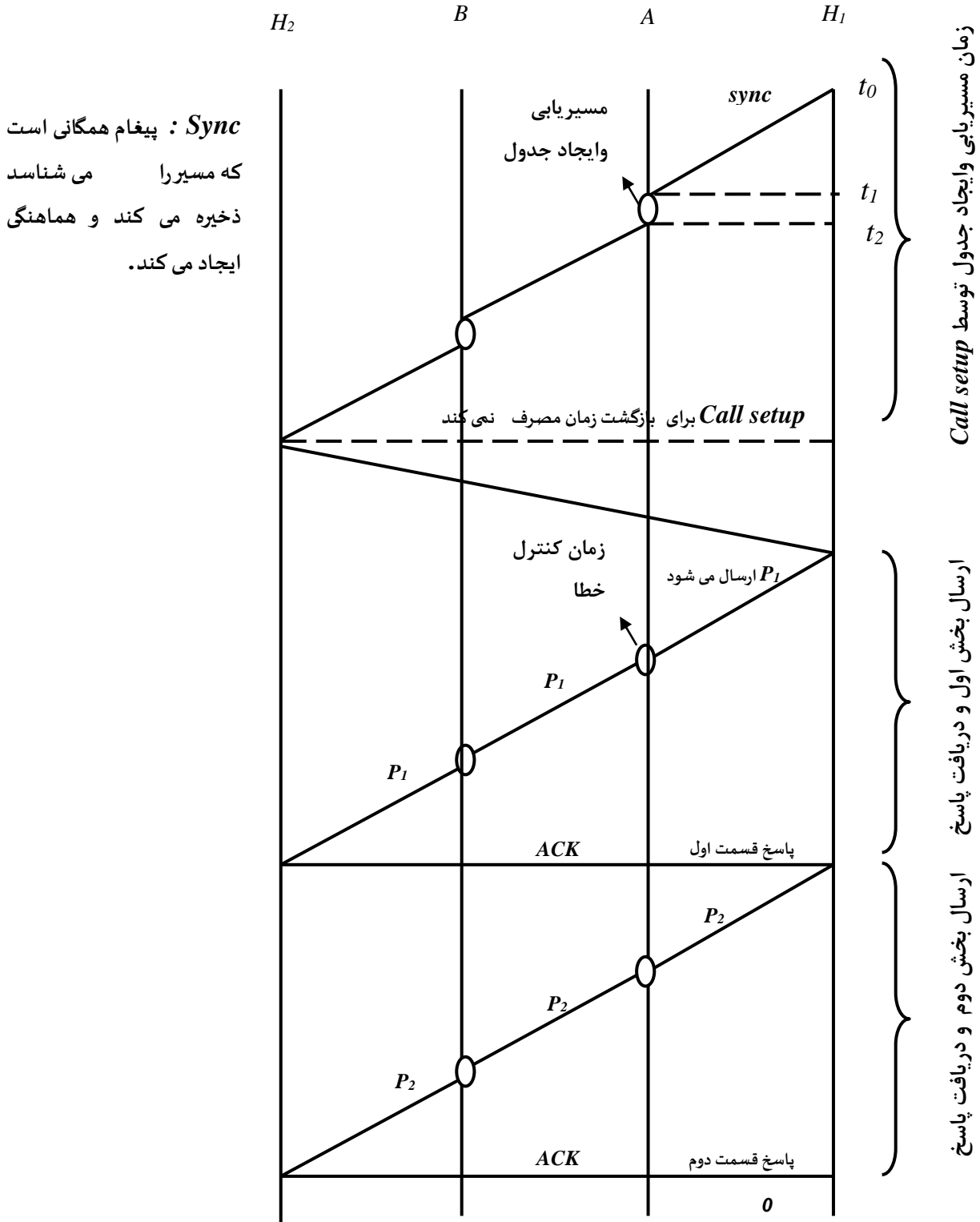
نمودار زمانی روش VC :

در روش VC اولین کاری که انجام می دهد این است که $Call setup$ مسیر را می رود و بر می گردد بعد این مسیر رزرو می شود حال پیغام ها یکی یکی می توانند ارسال شوند.



مراحل اتصال در روش VC:

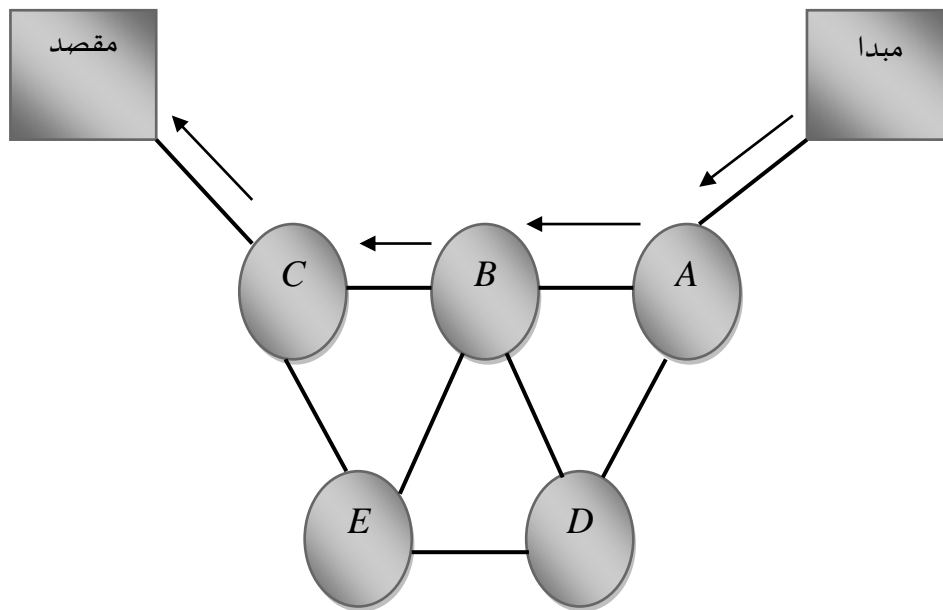
اول پیغام *Call setup* به *A* می رسد به زمانی برای مسیریابی و ایجاد جدول مصرف می شود و بعد مسیر را طی می کند و به *H2* می رسد



Sync: پیغام همگانی است که مسیر را می شناسد ذخیره می کند و هماهنگی ایجاد می کند.

۷-۱-۲ روش *Message Switching* :

پیغام تقسیم می شود - کانال یکطرفه است . فرستنده کل پیغام خود را به اولین *IMP* ارسال می کند این *IMP* پیغام را از یک مسیر آزاد به *IMP* بعدی ارسال کرده و ارتباط قطع نمی شود این کار تا زمان رسیدن پیغام به مقصد ادامه پیدا می کند . در این روش مسیر نگهداری نمی شود . (چون برگشتی ندارد پس نیازی به نگهداری مسیر نیست) .

*Packet*

نکته : روش های *VC* لایه ۴ را به صورت سخت افزاری پیاده سازی می کنند . اما روش *Message switching, switching* به صورت نرم افزاری.

ویژگی های *Message switching* :

محاسن:

- (۱) *IMP* ها ساده هستند. چون نیاز به ساختن جدول *VC* ندارند
- (۲) ترافیک کمتر . چون هیچ مسیر رزرو شده نیست: معمولا پیغام ها در این روش پاسخ ندارند (*Connection Less*) یا بدون اتصالند چون نیاز به پاسخ ندارند و اگر هم داشتند مطمئنا از مسیر رفت

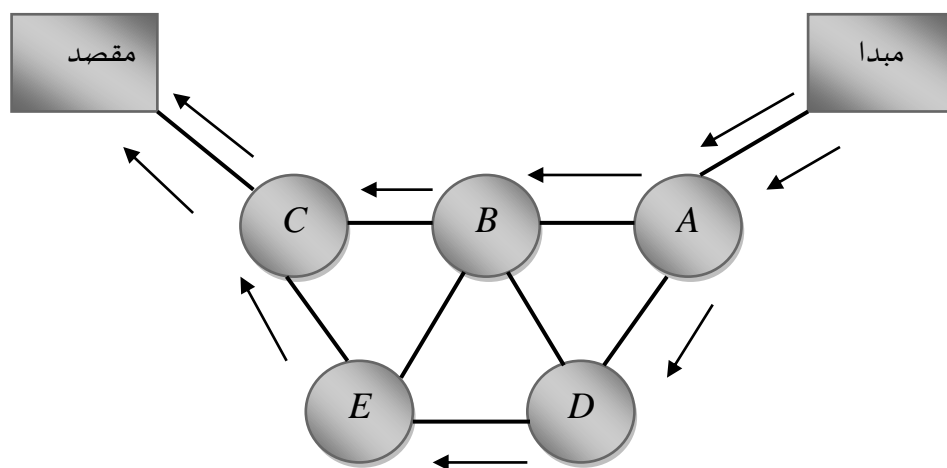
نیست و از مسیر دیگری پاسخ می دهند . پاسخ و دریافت به صورت هم زمان نیست چون مسیر یکطرفه است.

عیب :

پیغام را تقسیم نمی کند . وقتی پیغام تقسیم نشود *IMP* ها باید بافر بزرگی برای نگهداری داشته باشند این عیب در *Packet switching* رفع شد

۳-۱-۷ روش *Packet switching* :

همه ویژگی های روش قبل را دارد فقط در مبدا پیغام تکه تکه می شود و سپس هر قطعه از یک مسیر می رود تا به مقصد برسد و در مقصد تمام تکه ها با هم یکی شده و پیغام اصلی را حاصل می کنند.



۲-۷ نمودار *Packet switching*, *Message switching*

۶۱

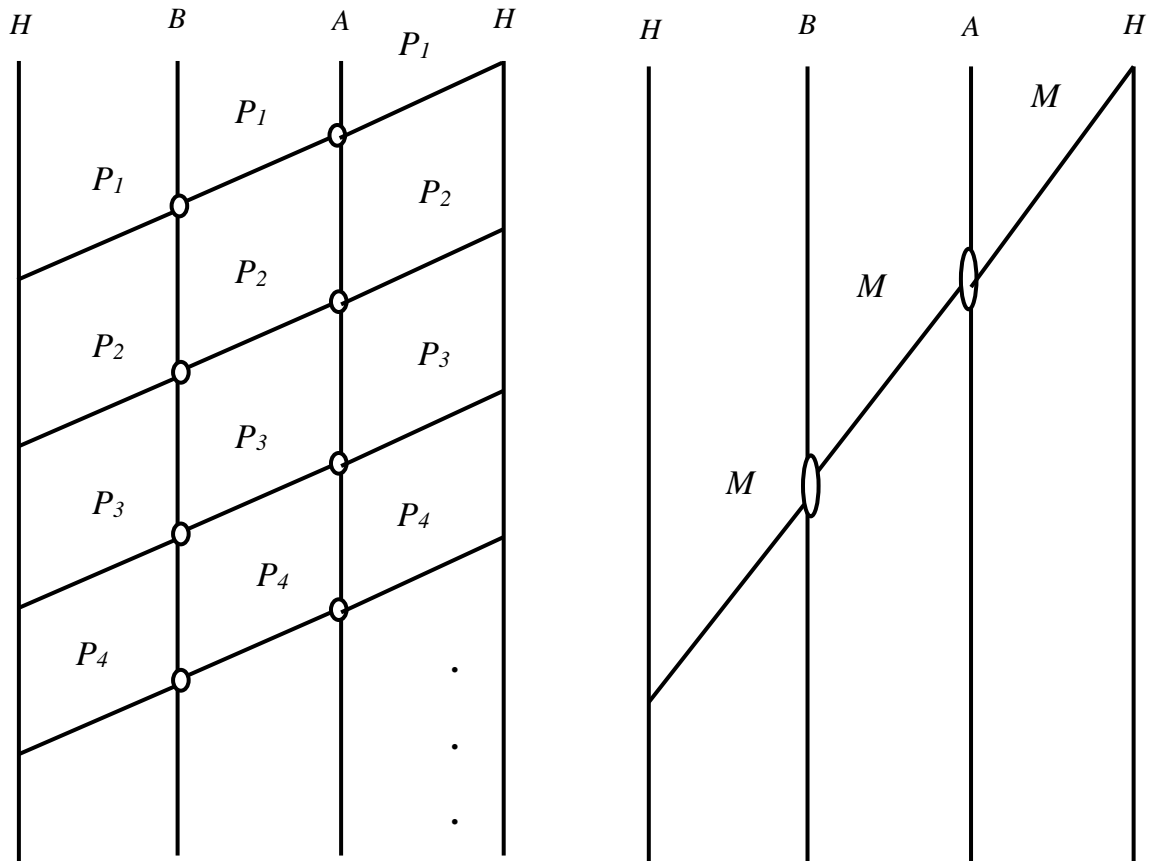
نکته: *Packet switching*: پیغام ها به تکه های

کوچکتر تقسیم می شوند آنگاه ابتدا قسمت اول را

ارسال می کند و نیاز به پاسخ ندارد پس قسمت دوم

نکته: *Message switching*: *Call Setup*

ارسال نمی کند از همان ابتدا پیغام را ارسال می کند



نکته ۱: در روش *Packet switching* چون مسیر ها ذخیره نمی شوند هر بار مسیر یابی می شود.

نکته ۲: روش های *Message Switching* و *Packet switching* برای پیغام هایی استفاده می شوند که نیاز به

پاسخ ندارند مثلا پیغام های *Broad cast* این روش ها قابلیت اطمینان ندارند. یعنی اصلا نمی توان یقین پیدا کرد

که بسته به دست گیرنده رسیده است یا خیر

نکته ۳: در روش *TCP/IP* از روش *Packet switching* استفاده می شود اینترنت یا (مدل *TCP/IP*) فاقد

اطمینان در سطح *Router* ها می باشد (قابلیت اطمینان در مدل *TCP/IP* به صورت نرم افزاری توسط پروتکلی

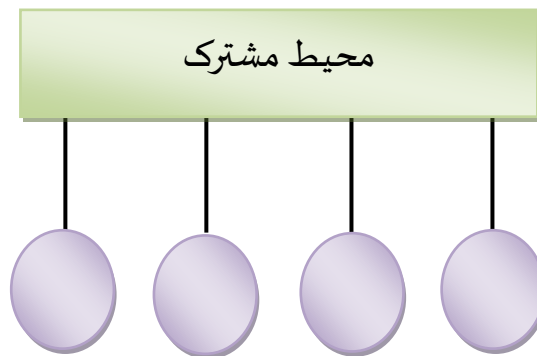
به نام *TCP* فراهم می شود)

فصل هشتم:

شبکه های محلی

۸-۱ ویژگی های شبکه های محلی :

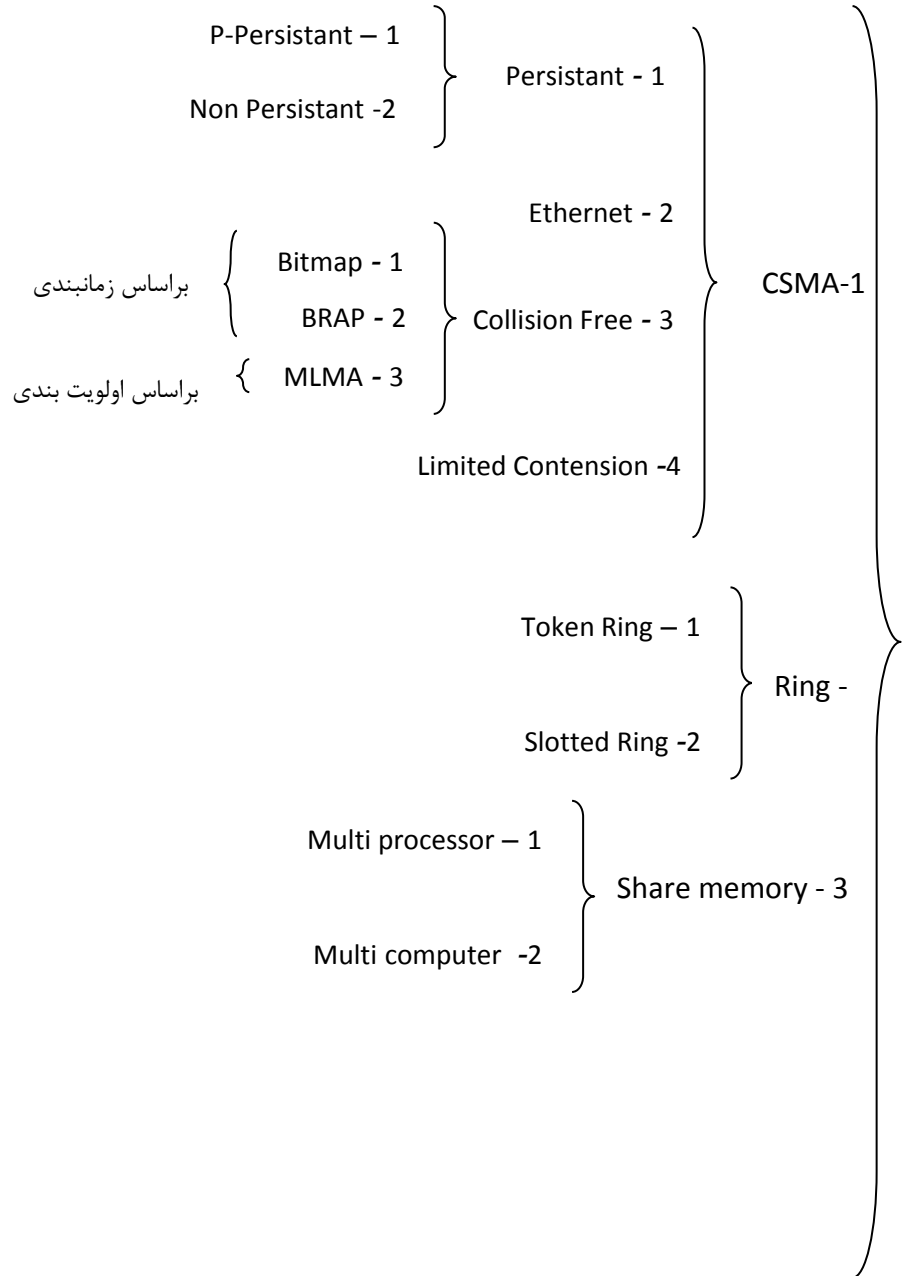
- ۱- توپولوژی *Broad cast* دارند
- ۲- مسیریابی ندارند
- ۳- *ACK* ندارند (نیاز به *ACK* نیست چون حتی اگر بخواهد مطمئن شود توسط روش دیگری به نام محیط مشترک (*Hub/Switch*) می تواند بفهمد



- ۴- مالکیت خصوصی : یعنی می توان از استاندارد های مخصوص یک شبکه استفاده کرد استانداردهایی که در شبکه های دیگر وجود ندارد

۸-۲ انواع شبکه های محلی از نظر عملکرد :

شبکه های محلی به ۳ شکل بر اساس نوع عملکردشان تقسیم می شوند:

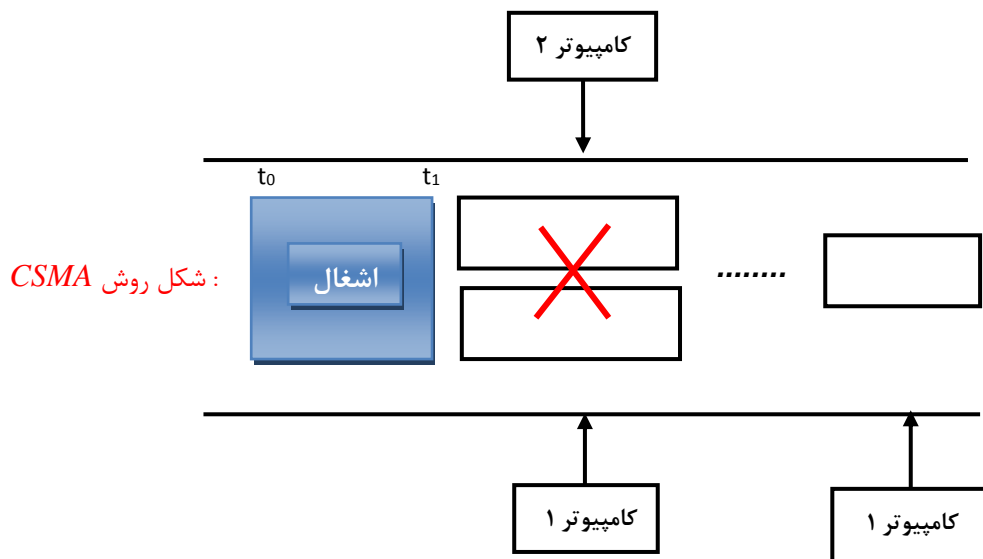


۳-۸ عملکرد CSMA در شبکه های محلی : *Carrier Sense Multiple Access*

در این روش فرستنده ها (کامپیوترها) منتظر آزاد شدن کانال می مانند به محض آزاد شدن ، هر فرستنده پیغام خود را ارسال می کند فرستنده ای که زودتر پیغام خود را ارسال کرده باشد کانال را در اختیار می گیرد. اگر هم زمان دو یا چند فرستنده با هم شروع به ارسال کنند پیغام همه از بین می رود و در این حالت ، یک زمان تصادفی صبر کرده دوباره شروع به ارسال می کند.

نمودار زمانی کانال مشترک :

عیب این روش این است که برای تشخیص تصادم باید کل پیغام را ارسال کند.



۱-۳-۸ *Persistent* : مصر

در صورتی که فرستنده در هنگام ارسال پیغام با اشغالی خط مواجه شود مرتباً گوش به زنگ می ماند تا به محض آزاد شدن خط پیغام خود را ارسال کند.

۲-۳-۸ *Non Persistent* : غیر مصر

در این روش فرستنده اصراری برای گرفتن خط بلافاصله پس از آزاد شدن آن ندارد یعنی در صورت اشغال شدن خط یک زمان تصادفی صبر کرده مجدداً شروع به ارسال پیغام می کند.

$$0 < p < 1$$

p : احتمال مصر بودن فرستنده ها : هرچه مقدار آن به یک نزدیک تر باشد یعنی بیشتر مصر است . اگر برابر یک شود یعنی همان *Persistent* و اگر برابر صفر شود ، می شود *Non Persistent* .

نکته : در صورت اشغالی شبکه (خط) بهتر است که p به صفر نزدیک تر باشد یعنی همه مصر نباشد کمتر مصر باشد.

۴-۸ Ethernet (CSMA / CD) :

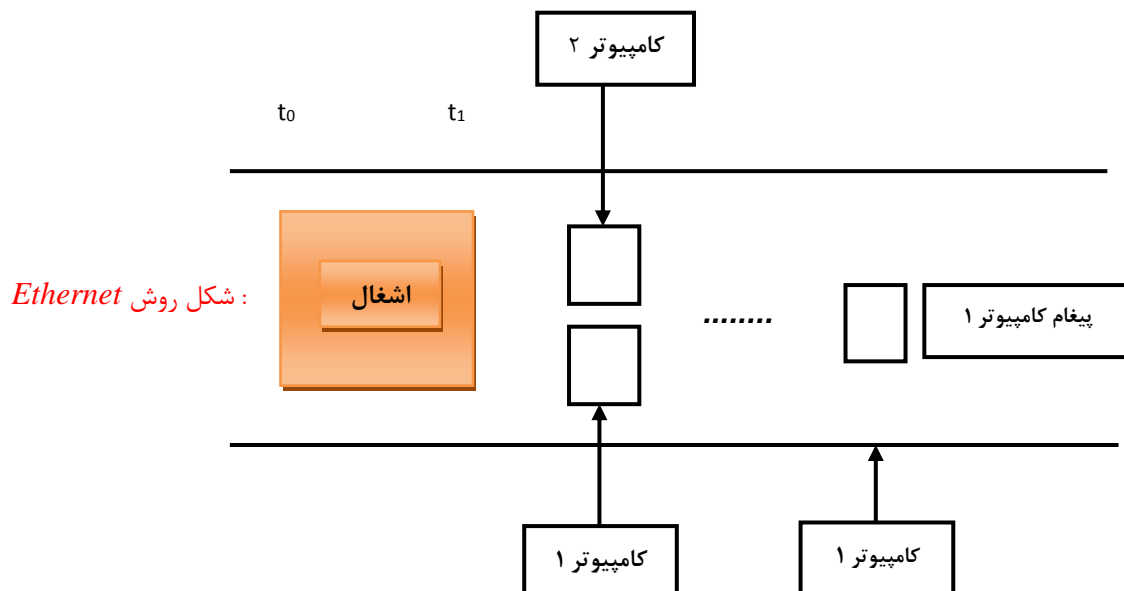
در روش *Ethernet* قبل از ارسال کل پیغام یک پیغام کوچک ارسال می شود تا تشخیص خطا دهد.

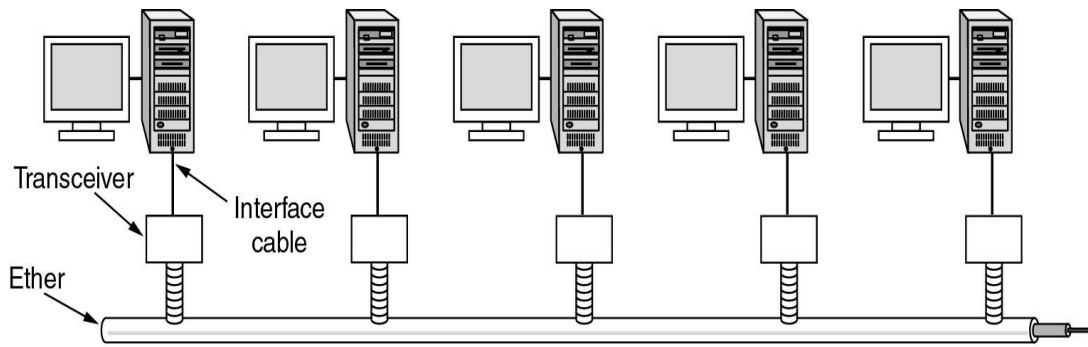
نکته : از زمان t_0 تا t_1 کانال اشغال و فرستنده ها گوش به زنگ هستند یعنی منتظر خالی شدن خط هستند.

نکته مهم : تفاوت روش *CSMA* و *Ethernet* در این می باشد که روش دوم یک پیغام کوچک جهت تشخیص خطا اول ارسال می کند.

Ethernet : در این روش فرستنده ها ابتدا یک پیغام جهت تشخیص بر خورد در شبکه ارسال می کنند بقیه ی مراحل شبیه *CSMA* است .

Ethernet پر کاربرد ترین روش در شبکه های محلی است .





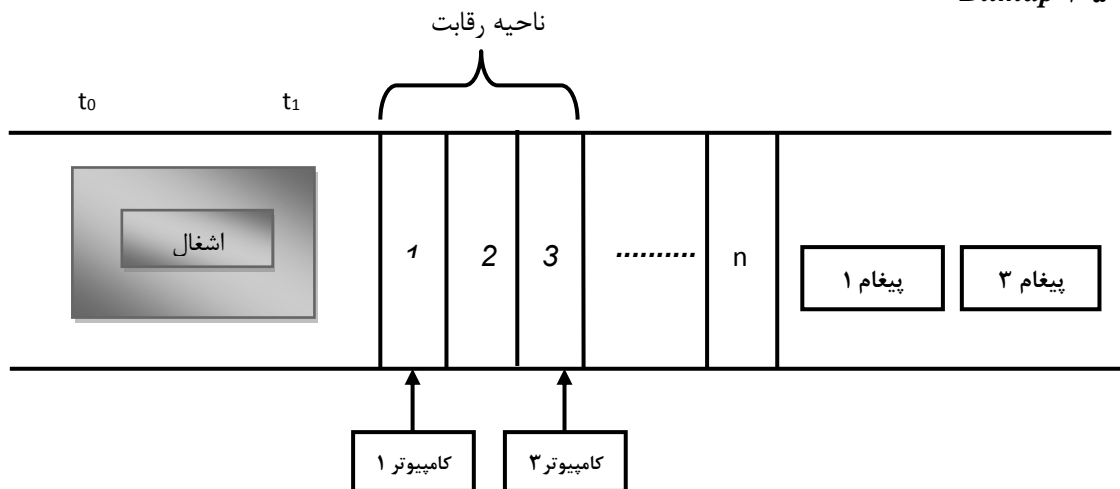
Architecture of the original Ethernet.

۸-۵ Collision Free: (روش بدون برخورد):

روش های که برخورد در آن نیست .

کاری می کنیم که اصلا برخورد پیش نیاید به وسیله ی زمان بندی و اولویت بندی می توان کاری کرد که Collision رخ ندهد.

۸-۵-۱ Bitmap



ناحیه رقابت : تقسیم بندی بین کامپیوترها یا فرستنده ها است . جایی که مشخص می شود کدام فرستنده باید کانال را در اختیار بگیرد.

n : تعداد کل کامپیوترهای شبکه .

در این روش هر فرستنده یک زمان مشخص جهت ارسال درخواست خود دارد. درخواست خودش را در ناحیه خودش ارسال می کند و در ضمن در این روش ابتدا تمام درخواست ها را دریافت می کند.

:BRAP ۲-۵-۸

در این روش به هر کامپیوتر بلافاصله پس از ارائه درخواست کانال تخصیص داده می شود.

نکته : در روش قبل همه باید صبر می کردند تا در خواست ها بیایند بعد پیغام بدهند اما در این روش بلافاصله بعد از ارسال درخواست همان موقع پاسخ داده می شود (البته اگر در زمان خودش باشد).

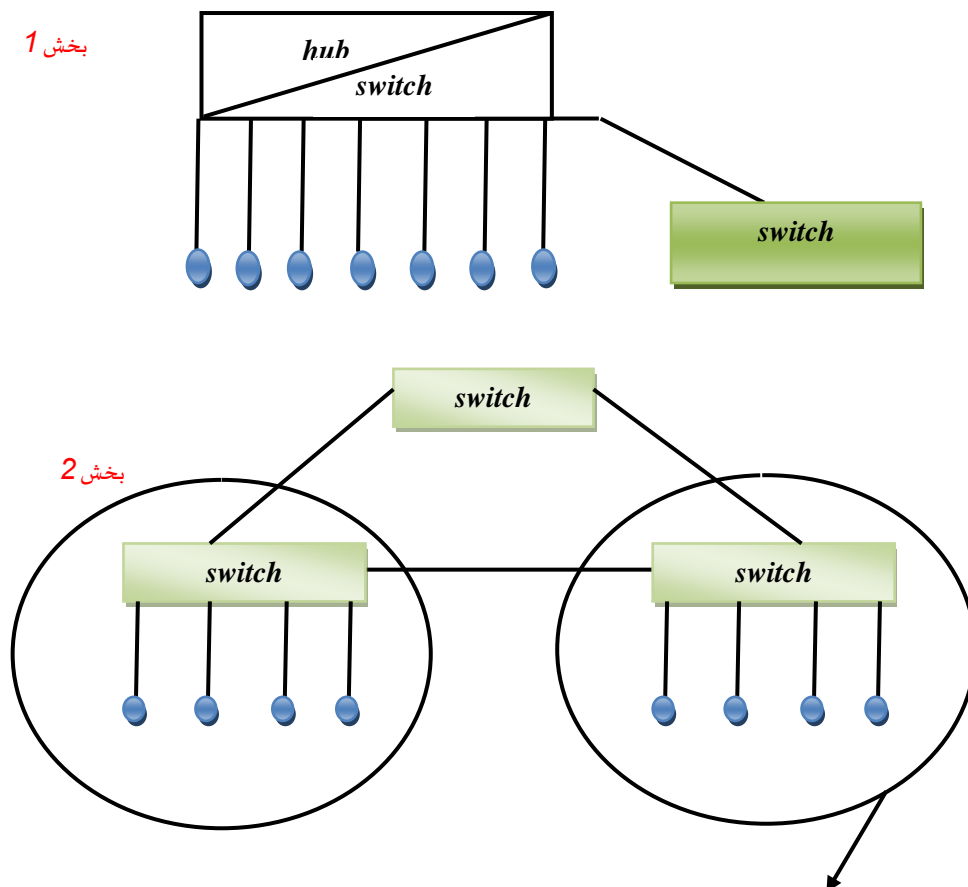
: Multi Level Multi Access : MLMA ۳-۵-۸

در این روش به هر فرستنده یک عدد اولویت داده می شود، در صورتی که پیغامی ارسال کند و با پیغام دیگری همزمان شود با توجه به اولویت فرستنده ها کانال اختصاص می یابد .

نکته : این سه روش را روش های بدون برخورد یا *Collision Free* یا *CSMA/CA* می گویند.

: Limited Contension : رقابت مجدد : ۶-۸

در این روش تعداد سوئیچ ها نشان دهنده تعداد ناحیه های برخورد متفاوت می باشد که رقابت بر سر گرفتن یک محیط مشترک بین کامپیوترها تقسیم می شود .



به هر یک از مسیرها یک collision domain می گوئیم .

نکته : یک شبکه به دو محیط مشترک تقسیم شده که به آن تقسیم رقابت یا محدود کردن رقابت می گوئیم .

در بخش ۲ نمی توان از *hub* استفاده کرد و حتما باید سوئیچ باشد و در ضمن اگر دو *hub* بگذاریم یا یک *hub* مرکزی قرار دهیم فرقی نمی کند .

۷-۸ *Token Ring* :

در این روش یک پیغام به اسم *token* در شبکه در حال چرخش است . فرستنده پس از دریافت آن پیغام خود به همراه *token* تغییر شکل یافته یا همان (*connector*) را ارسال می کند .

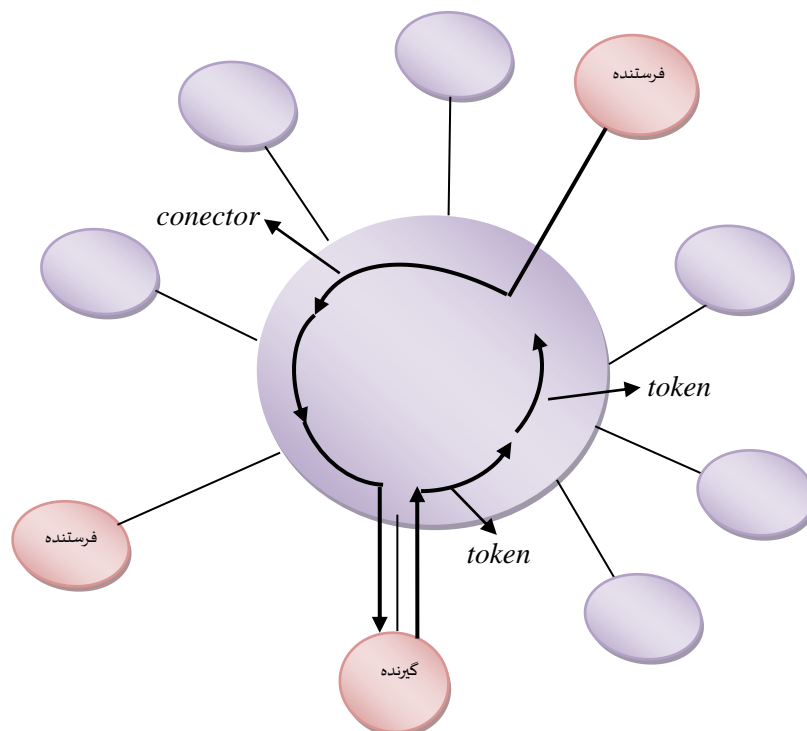
نکات مهم :

۱- مساله برخورد پیش نمی آید.

۲- در شبکه ی محلی زیاد کاربرد ندارد اما در شبکه ی شهری و به کمک فیبر نوری گزینه ی خوبی است .

۳- حلقه ها را *hub / switch* ایجاد می کنند .

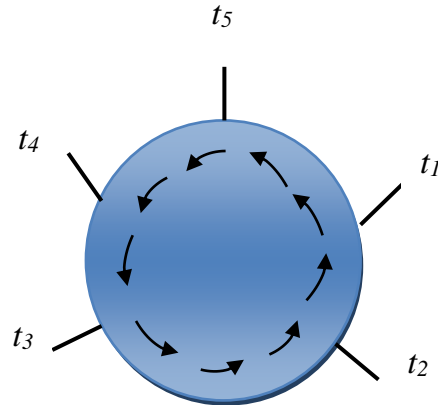
روند کار *token Ring* : محیط مشترک یک حالت چرخشی دارد یک پیغام به اسم *token* در آن، در حال حرکت است . اگر یک کامپیوتر(فرستنده) *token* را دریافت کرد پیغام خود را با استفاده از *token* می که عوض کرده و تحت عنوان *connector* قرار داده است در شبکه ارسال می کند . حال اگر یک فرستنده ی دیگر قصد ارسال پیغام را داشته باشد و حالت پیغام *connector* باشد مجاز به ارسال نیست تا زمانی که بسته به مقصد برسد.گیرنده بسته ی پیغام را دریافت و *connector* را به *token* تغییر می دهد و ارسال می کند .



۸-۸ *Slotted Ring*:

در این روش حلقه به تعدادی بخش زمانی تقسیم می شود. در هر بخش زمانی، یک پیغام قابل انتقال می باشد.

تقریباً این روش معادل روش *TDM* برای شبکه های محلی است.



تفاوت روش *Slotted Ring* با *TDM*: در روش *TDM* در هر زمان فقط یک بخش از پیغام ارسال می شود اما در روش *Slotted Ring* لزوماً بخشی از پیغام نیست ممکن است کل پیغام باشد.

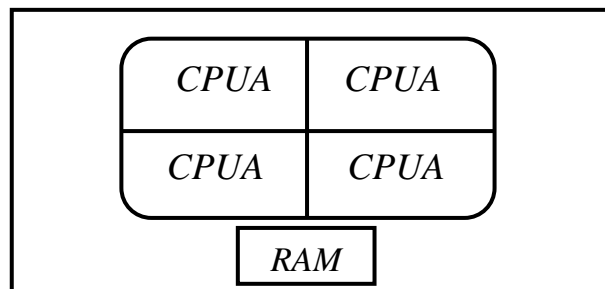
نکته مهم: یکی از دلایل استفاده از *Token* این است که *Collision* ندارد.

۸-۹ *Shared memory*: برای افزایش سرعت در اجرای برنامه ها استفاده می شود.

۸-۹-۱ *Multi processor*:

اشکالات روش *Multi processor*:

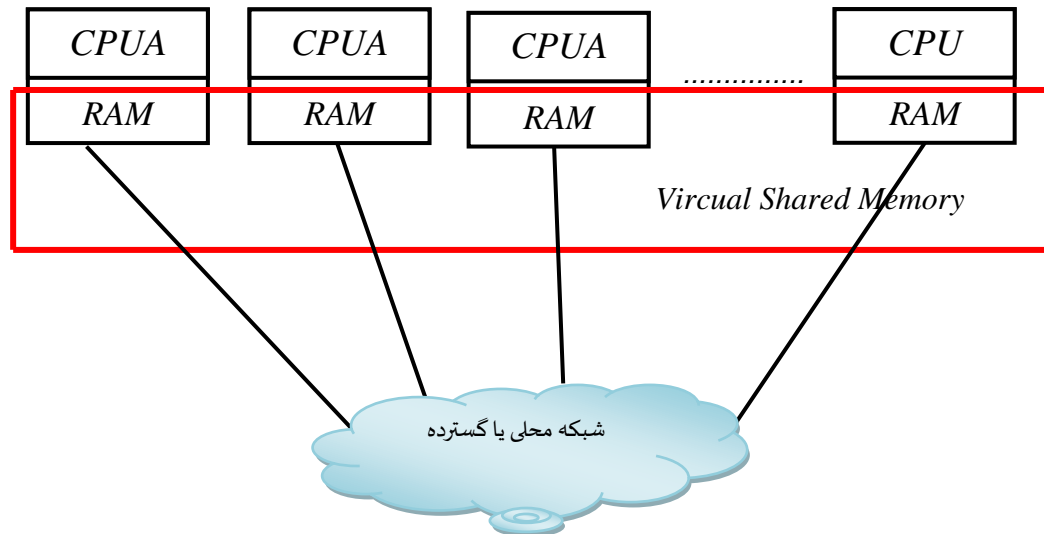
- (۱) قیمت بالا
- (۲) محدودیت در تعداد *CPU*
- (۳) پیچیدگی سخت افزار



پردازش موازی بالاترین سرعت

مشکلات روش فوق به کمک روش *Multi Computer* حل می شود.

۸-۹-۲ روش *Multi Computer*:



محاسن روش *Multi Computer*:

- (۱) قیمت پایین
- (۲) تعداد *cpu* نامحدود
- (۳) طراحی ساده

عیب روش *Multi Computer*:

نیاز به مدیریت نرم افزاری پیچیده ای دارد. یعنی باید کل *RAM* ها در یک برنامه اجرا شوند که به آن *Virtual Shared Memory* (حافظه مشترک مجازی) می گویند که درست کردن آن نیازمند نرم افزار پیچیده ای است.

نکته: به جای ساختن کامپیوتر با تعداد *CPU* بالا می توان از کامپیوترهای موجود در شبکه برای پردازش موازی استفاده کرد.

۸-۱۰ آدرس های *IP* :

مهم ترین بخش در شبکه است

محدوده آدرس های *IP* :

آدرس های *IP* ۳۲ بیت یا ۸ بایت است.

0-255 0 0-255 0 0-255 0 0-255

آدرس های *IP* به ۵ دسته تقسیم می شوند که به آنها کلاس های *IP* می گویند.

۸-۱۱ کلاس های آدرس *IP* :

1-126.X.X.X

(۱) کلاس A

SM:255.0.0.0

آدرس خصوصی: 10.X.X.X

تعداد کامپیوتر: 254^3

(۲) کلاس B

128-191.X.X.X

SM:255.25.0.0

آدرس خصوصی: 172.16-32.X.X

تعداد کامپیوتر: 254^2

(۳) کلاس C

192-223.X.X.X

SM:255.25.255.0

آدرس خصوصی: 192.168.X.X

تعداد کامپیوتر: 254

کلاس D (۴)

224-239.X.X.X

کلاس E (۵)

240-255.X.X.X

نکته ۱: آدرس IP باید منحصر به فرد باشد.

نکته ۲: آدرس ها بر اساس اعداد اول تقسیم بندی می شوند .

نکته ۳: کلاس های A,B,C برای آدرس دهی هر End system و هر IMP لایه ۳ به بالا می باشند .

نکته ۴: کلاس D,E برای Test ,Broadcast استفاده می شوند.

۸-۱۲ مفهوم Subnet Mask :

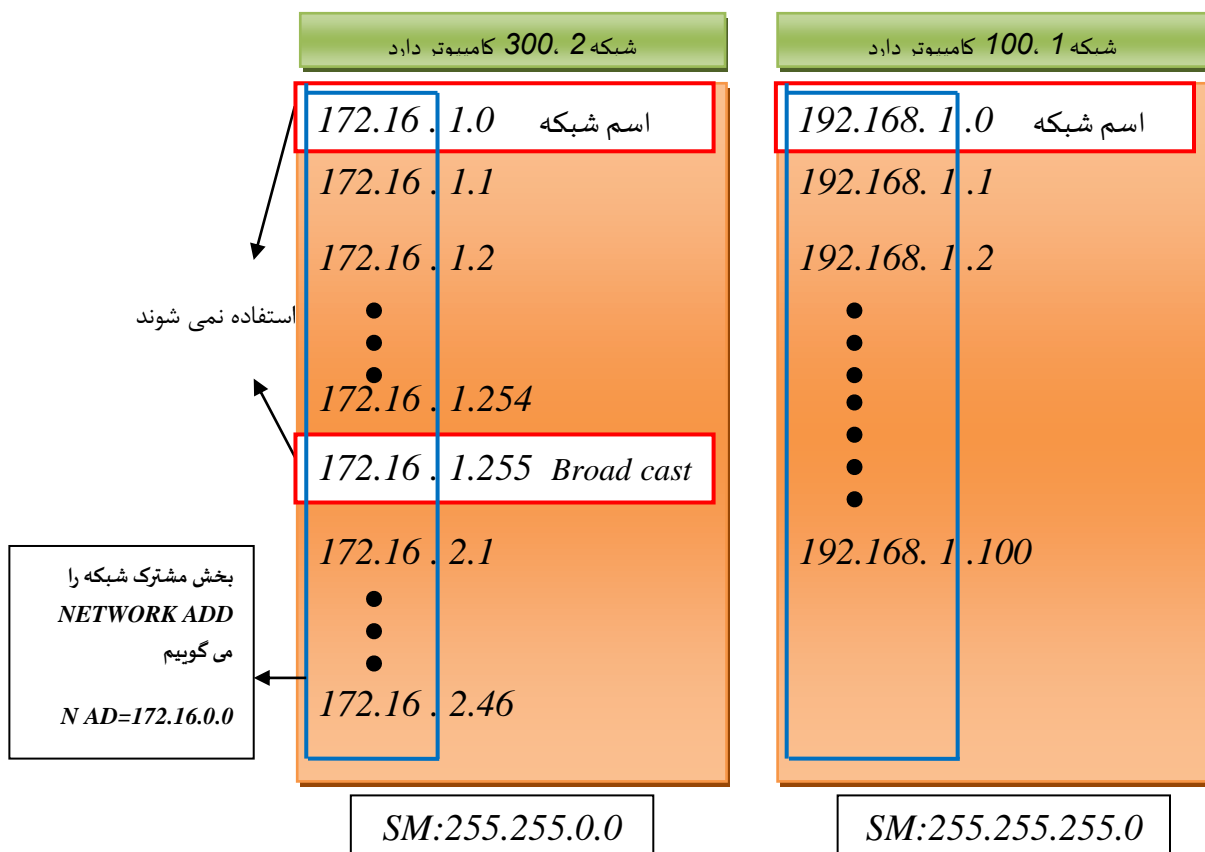
مکانیزمی است که نشان می دهد کدام قسمت از آدرس IP ثابت و کدام بخش متغیر است

نکته ۱: برای اینکه آدرس شبکه با سایر آدرس ها یکسان شود آخر آن صفر می گذاریم.

نکته ۲: در آدرس دهی کامپیوتر ها حتما آدرس ها پشت سر هم هستند.

نکته ۳: Network Address بخشی از آدرس است که در همه مشترک است.

برای درک بهتر مفهوم Subnet mask به مثال زیر دقت کنید:



۸-۱۳ آدرس های *Invalid*:

در هر کلاس تعدادی از آدرس ها مشخص می شود و قرار داد می شود در شبکه اینترنت استفاده نشوند (محدوده آنها در بخش کلاس های *IP* توضیح داده شدند)

از آدرس های *Invalid* در شبکه های محلی استفاده می شود دلیل آن این است که اگر این شبکه محلی به اینترنت وصل شد آدرس های آن با آدرسی در شبکه اینترنت تداخل نداشته باشد

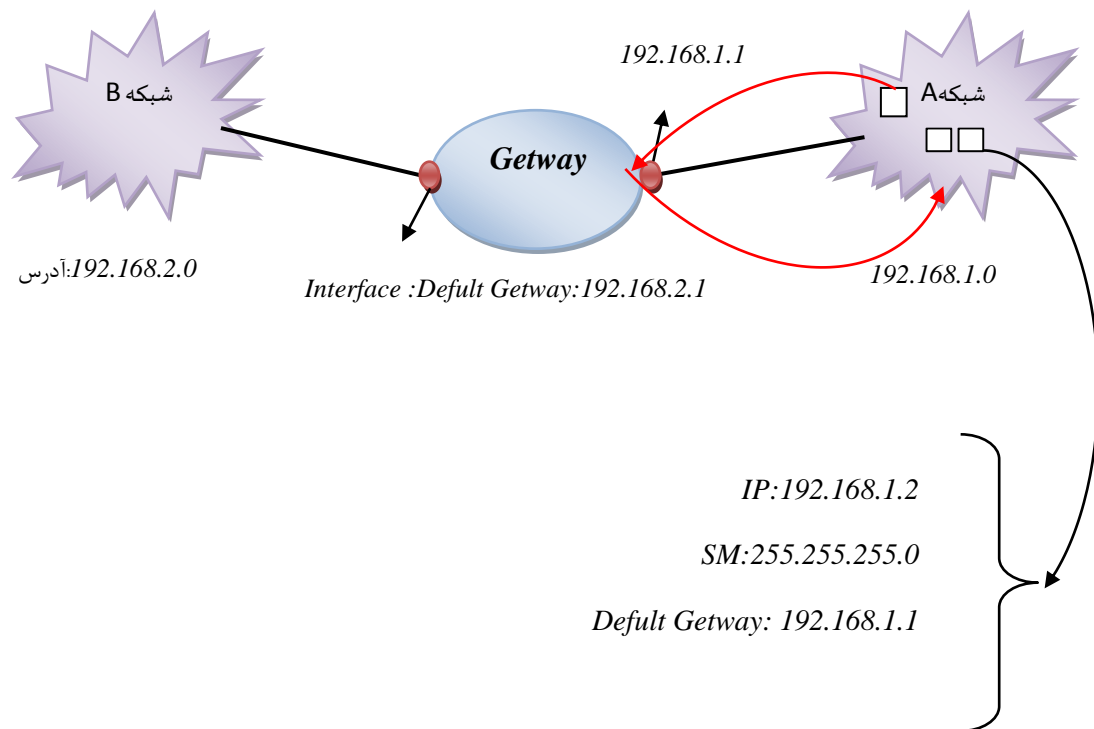
۸-۱۴ ارتباط بین شبکه های محلی:

دو شبکه جداگانه اگر داشته باشیم مثلا شبکه *A, B* با دستگاهی به نام *Gateway* (یا دروازه ارتباطی) به هم وصل می شوند *Gateway* می تواند یک *IMP* لایه ۳ باشد مثل *Router* یا *switch* لایه ۳ یا یک کامپیوتری که ۲ تا کارت شبکه در دو سر خود دارد که به آن اصطلاحا *Multi home* گفته می شود.

آدرس *IP*، *Gateway* چطور است؟

نکته ۱: *Gateway* دو تا *Interface* دارد

نکته ۲: معمولا اولین آدرس *IP* مربوط به شبکه را به *Gateway* می دهیم.



۸-۱۵ وظایف *Gateway* عبارتند از :

۱- ایجاد ارتباط فیزیکی

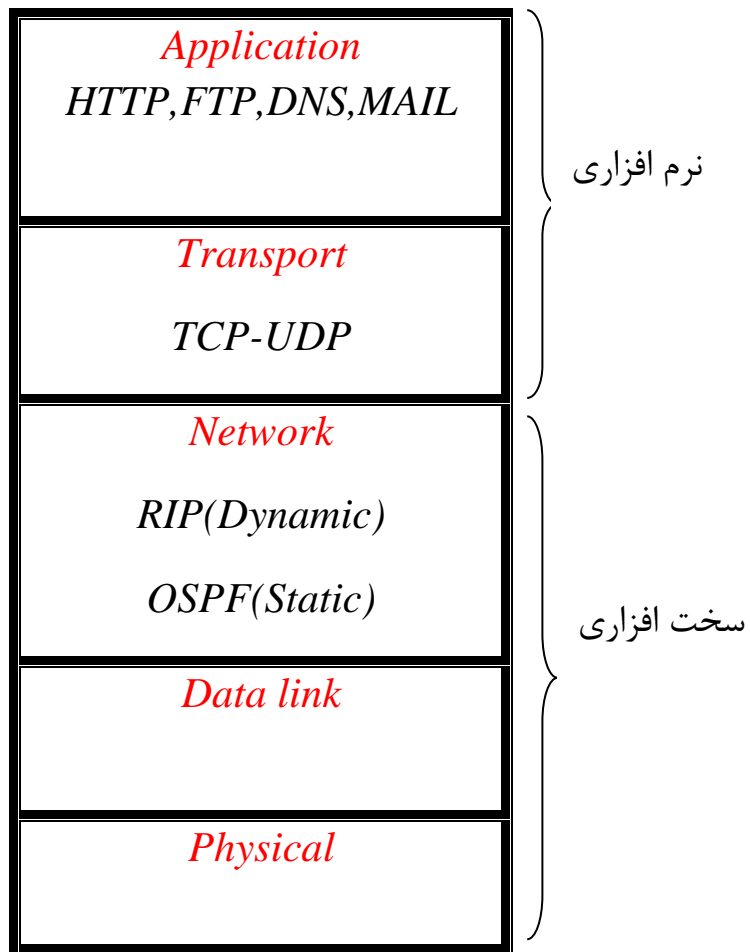
۲- مسیریابی

۳- فیلترینگ ترافیک شبکه : یعنی اگر مبدا و مقصد یک پیغام در یک شبکه باشند *Gateway* اجازه خروج به آن پیغام نمی دهد (یعنی پیغام از شبکه خارج نمی شود)

فصل نهم:

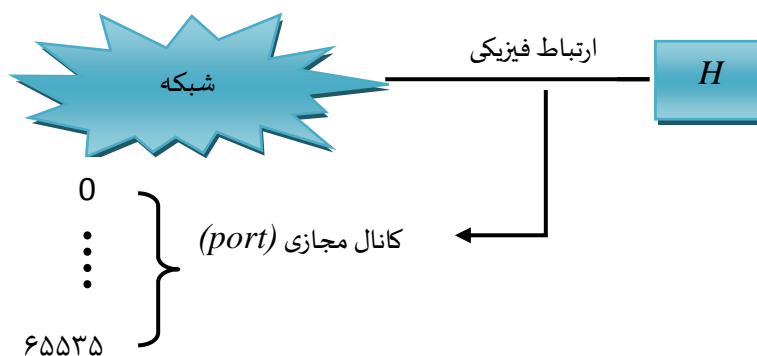
شبکه اینترنت

: TCP/IP1 -9



- پروتکل *HTTP*: وظیفه انتقال صفحات وب را بر عهده دارد هر پروتکل از یک پورت جداگانه جهت انتقال داده استفاده می کند

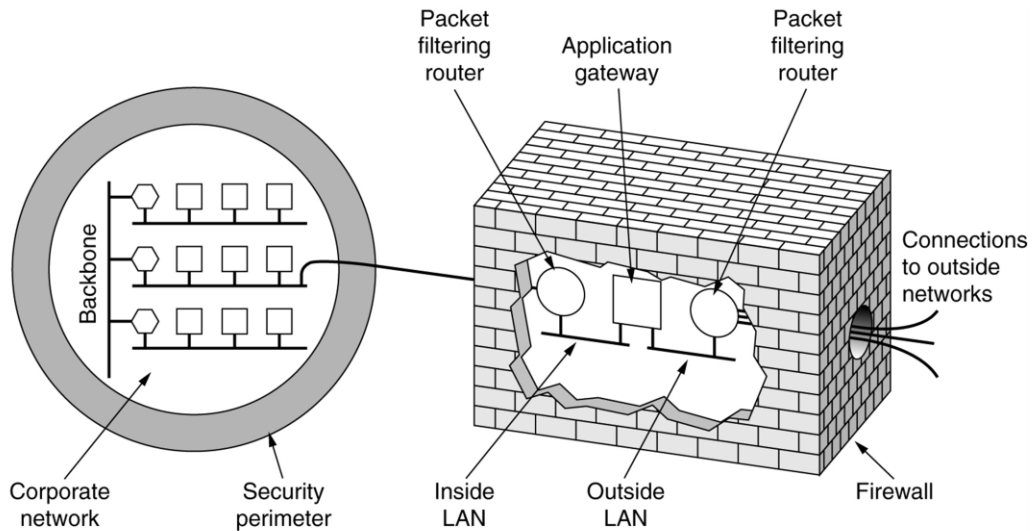
نکته ۱: مفهوم *port* (کانال مجازی) : هر *host* به شبکه با یک ارتباط فیزیکی وصل است ارتباط فیزیکی به تعدادی کانال مجازی تقسیم می شود به این کانال های مجازی اصطلاحاً *port* گفته می شود که دارای شماره از ۰ تا ۶۵۵۳۵ است



نکته ۲: از ۰ تا ۱۰۲۳ پورت های استاندارد و مربوط به پروتکل های استاندارد هستند صفحات وب از پورت ۸۰ استفاده می کنند.

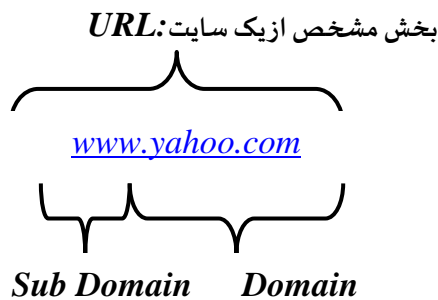
Firewall: همه پورت های بدون استفاده را می بندد به جز آنهایی که نیاز است باز باشند.

Firewalls



A firewall consisting of two packet filters and an application gateway.

نکته: در پروتکل *HTTP* با وب سایت ها سر و کار داریم



محیط برنامه نویسی صفحات وب *HTML* است که به تنهایی قوی نیست و باید با زبانهای *Script* همراه شوند

سمت *Client* :

HTML+ Script

java Script

VB Script



سمت *Server* :

HTML+ASP

C#

VB.NET

J#



استاندارد *CGI* : امکان استفاده از برنامه های اجرایی را در داخل *HTML* می دهد یعنی امکان اجرای فایل *EXE* را در داخل *HTML* می دهد.

۲-۹ پسوند صفحات وب :

.html

.HTM

.Asp

.ASPX

.PHP

هر وب سایت یکی از صفحات وب خود را به عنوان صفحه اصلی قرار می دهد و از طریق آن صفحه می توان به صفحات دیگر *link* زد نام صفحه اصلی می تواند یکی از موارد زیر باشد :

۱- *default*

۲- *Index*

۳- *Home*

- پروتکل *HTTPS* : جاهایی که قرار است در صفحات وب اطلاعات مهم منتقل شود از این پروتکل استفاده می کنیم مثل بانک ها ایمیل ها و

نکته *HTTP* فایل ها را به صورت رمزنگاری نشده انتقال می دهد اما در *HTTPS* به صورت رمز شده منتقل می شوند.

- پروتکل *FTP* : دانلود و آپلود فایل ها توسط این پروتکل انجام می شود.
- پروتکل *FTPS* : دانلود و آپلود فایل ها به صورت رمز شده.

۹-۳ پروتکل های *Mail* دو دسته اند

(۱) دسترسی به *Mail*

(۲) انتقال *Mail*

۹-۳-۱ دسته اول پروتکل های دسترسی به *Mail*

- ۱- *POP3* : برای دسترسی به ایمیل جهت خواندن - نوشتن - یا ارسال ایمیل برای کاربر
- ۲- *IMAP* : همان کارهای *POP3* را انجام می دهد فقط محدودیت در میزان *Attachment* ندارد
- ۳- *MIME* : امکان استفاده از مولتی مدیا در داخل ایمیل

۹-۳-۲ دسته دوم انتقال *Mail* در شبکه:

SMTP : برای انتقال بین کامپیوترها در شبکه استفاده می شود.

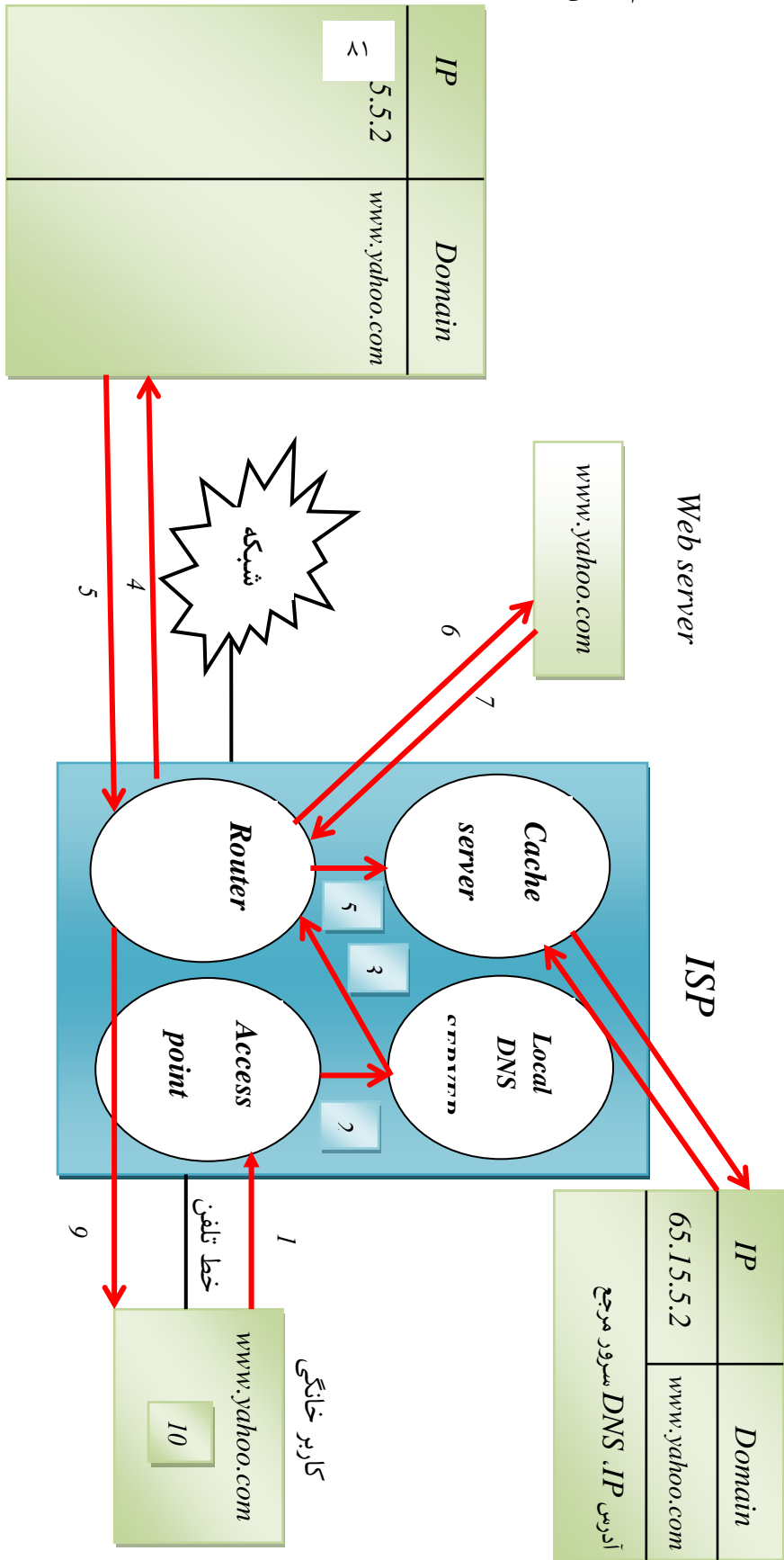
پروتکل *DNS : Domain Name System*

سرویس است که نام الفبایی را به عددی و بالعکس تبدیل می کند.

۱- نام الفبایی : *Domain Name*

هر وب سایت

۲- نام عددی : IP



مرحله ۱: درخواست کاربر شامل نام وب سایه: ۸۲ IP ارسال می شود

مرحله ۲: این نام در DNS محلی بررسی شده در صورت عدم وجود به یک DNS سرور مرجع از طریق روتر ارجاع می شود (مرحله ۴ و ۳)

مرحله ۵: آدرس IP معادل سایت به Router داده می شود

مرحله ۶: از طریق این آدرس IP (Web server) و درخواست دریافت فایل های وب ارسال می شود.

مرحله ۷: فایل های وب سایت دریافت شده

مرحله ۸: یک کپی از آن داخل Cache server قرار می گیرد

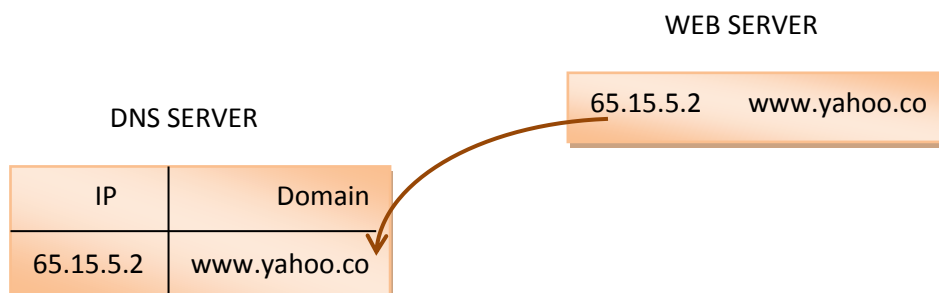
مرحله ۹: به درخواست کننده جهت اجرای این وب سایت در مرحله ۱۰ داده می شود

۴-۹ مراحل ایجاد وب سایت :

(۱) اول برنامه نویسی سایت

(۲) تعیین نام و ثبت آن

(۳) تعیین یک سرور برای نگهداری سایت به آن Web hosting یا میزبانی صفحات وب می گوئیم

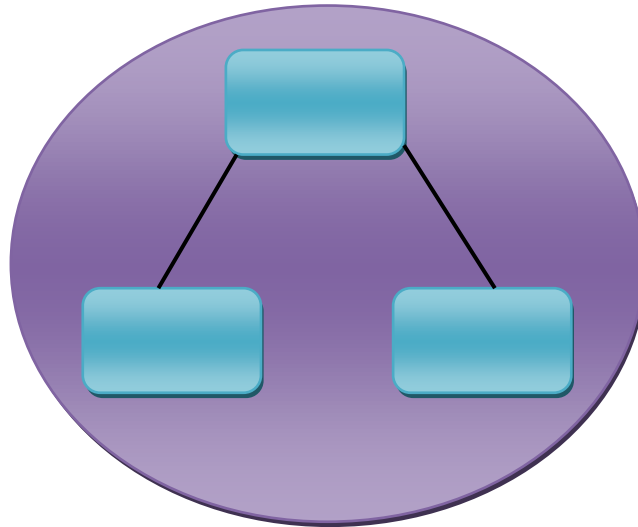


یک وب سرور پیدا می کنیم و سایت خودمان را روی آن قرار می دهیم این وب سرور یک آدرس IP به ما اختصاص می دهد مثلا آدرس 65.15.5.2 و این آدرس IP را در DNS سرور نیز قرار می دهد و ثبت می کند .

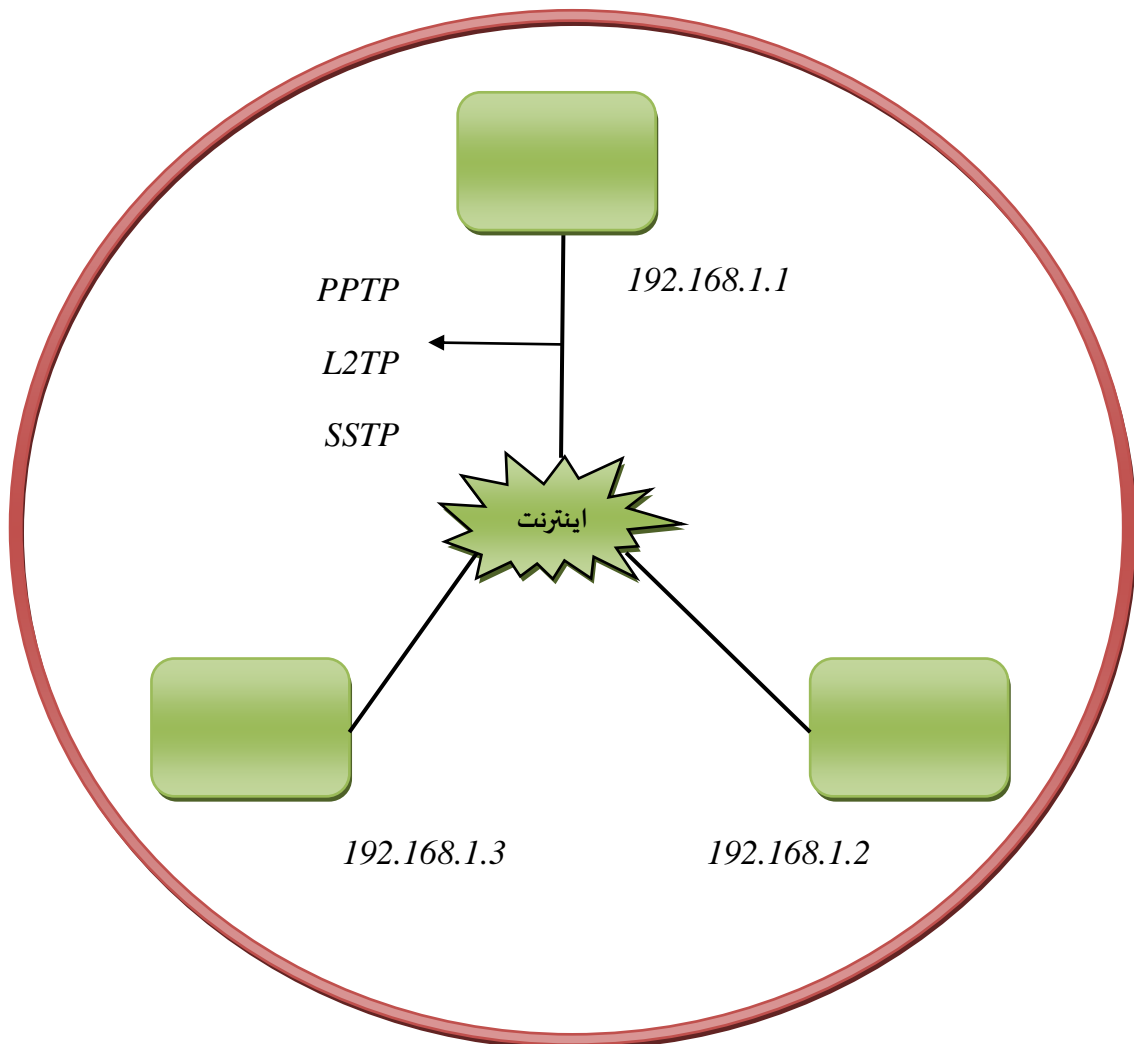
۵-۹ سرویس VPN :

فرض کنید ۳ کامپیوتر در خانه دارید که با یک شبکه محلی به هم وصل هستند و امنیت دارند حالا می خواهیم این شبکه را گسترش دهیم یعنی به جای اینکه ۳ کامپیوتر در یک خانه باشند ۳ کامپیوتر را در ۳ جای مختلف کشور در

نظر می گیریم . می خواهیم از بستر اینترنت استفاده کنیم برای اینکه ارتباطات این ۳ کامپیوتر در ۳ نقطه مختلف را برقرار کنیم ولی باز هم این ۳ کامپیوتر همدیگر را مثل یک شبکه محلی ببینند (با وجود اینکه در شبکه اینترنت هستند اما کانال های ارتباطی وجود داشته باشد که بخواهیم اینها را در یک شبکه ثابت حساب کنیم)



Private network



Virtual private network

در شکل قبل بین دو یا چند کامپیوتر ارتباطی از طریق شبکه گسترده فراهم می کنیم که بقیه از وجود چنین ارتباطی مطلع نباشند و پیغامی که رد و بدل می شود را هیچکس متوجه نمی شود. پروتکلی که این کانال را ایجاد می کند پروتکلی به نام *PPTP* می باشد.

Point to Point Tunnel Protocol : PPTP

پروتکل ایجاد *VPN* است (یعنی پروتکل تونل زنی) در این روش اطلاعات رمز نیستند ولی می توان آنها را رمز کرد این کار به کمک *SSTP, L2TP* صورت می گیرد. *L2TP* امنیت در لایه ۲ را تضمین می کند.

نکته: هرچه *Security* در لایه پایین تری باشد امنیتش بیشتر است.

۹-۶ لایه حمل :

وظیفه لایه حمل تعیین نوع کانال ارتباطی می باشد. که در مدل *TCP/IP* به صورت نرم افزاری تعیین می شود. در مدل *TCP/IP* وظیفه انتقال اطلاعات و تعیین کانال ارتباطی بر عهده *End System* ها است نه *IMP* ها پس *IMP* های مدل *TCP/IP* ساده هستند.

۹-۷ لایه حمل به وسیله دو پروتکل کانال ارتباطی را مشخص می کند:

۱- *TCP*

۲- *UDP*

۹-۷-۱ *TCP : Transfer Control Protocol* (پروتکل کنترل انتقال): یعنی انتقال اطلاعات را کنترل می کند چون دو طرفه است پس به آن کنترلی می گویند.

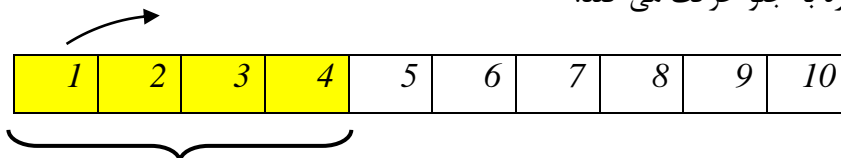
ویژگی های *TCP* :

(۱) کانال دو طرفه : در این روش مسیر ارتباطی ذخیره نمی شود یعنی در *IMP* ها جدول *vc* نداریم و دو طرفه بودن کانال به این معنی است که پیغام ها حتما دارای پاسخ هستند و در ضمن پاسخ ها لزوما از همان مسیر ارسال شده دریافت نمی شود

(۲) *Connection Oriented* : اتصال گرا است یعنی همه پیغام ها حتما پاسخ دارند پس فرستنده مطمئن می شود پیغام به دست فرستنده رسیده یا نه

۳) قابلیت اطمینان : قابلیت اطمینان دارد زیرا می تواند مطمئن شود پیغام به دست گیرنده مورد نظر رسیده است.

۴) کنترل ازدحام : در مدل *OSI* وظیفه کنترل ازدحام بر عهده لایه سوم است اما در روش *TCP/IP* کنترل ازدحام به صورت نرم افزاری توسط لایه ۴ انجام می شود. طریقه کار بدین شکل است که یک پروتکل داریم به نام *Sliding Window* یا پنجره ۸۵ که فرستنده به کمک آن به جای ارسال یک پیغام چندین پیغام (*packet*) هم زمان با هم ارسال می کند تا راندمان شبکه افزایش یابد پس از اینکه پاسخ این پیغام ها دریافت شد پنجره به جلو حرکت می کند.



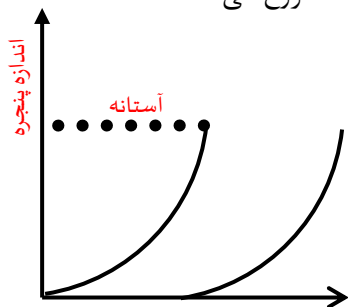
اندازه پنجره که در اینجا 4 است اندازه آن باید *Window size*:

طوری انتخاب شود که ازدحام رخ ندهد

کنترل ازدحام به وسیله تعیین طول پنجره یا *Window size* به یکی از دو روش زیر انجام می شود :

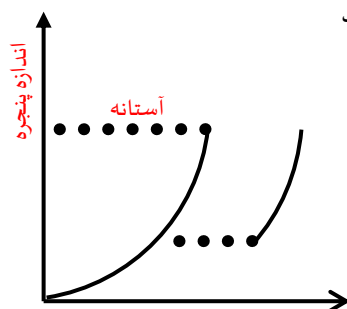
۱- *TCP Tahoe* : ترافیک سنگین : نرخ رشد تعداد *lost packet* ها را مشخص می کند هرچه نرخ بیشتر

ازدحام بیشتر است در این روش به محض رسیدن به آستانه مجدداً از یک شروع می کند



۲- *TCP Reno* : ترافیک سبک : وقتی به آستانه رسید به جای شروع از یک ، از نصف قبلی شروع می کند

(نصف آستانه) در ضمن آستانه تعداد اندازه پنجره را مشخص می کند



در این دو روش اندازه پنجره لرزان مرتباً اضافه می شود تا به یک آستانه برسد (آستانه تعداد *packet* هایی که حذف شده اند) در روش اول به محض رسیدن به آستانه مجدداً از یک شروع می شود ولی در روش دوم از نصف طول پنجره قبل

۵) کاربرد *TCP* : در اکثر پروتکل های شبکه مثل *Mail* , *FTP* , *HTTP* و.... که رسیدن پیام به مقصد مهم است از *TCP* استفاده می شود

۹-۷-۲ *User Datagram Protocol : UDP* ۸۶

ویژگی های *UDP*

- ۱) کانال یکطرفه است
- ۲) *Connection Less* است یعنی پیام ها بدون پاسخ می باشند.
- ۳) قابلیت اطمینان ندارند
- ۴) کنترل ازدحام ندارند : چون مسیر دو طرفه نیست و پیام از یک مسیر آزاد ارسال می شود.
- ۵) کاربرد: کاربرد آن در تمام پیام های *Broadcast* است همچنین پیام هایی که رسیدن کل پیام به گیرنده مهم نباشد مثل مولتی مدیا .

نکته ۱: *TCP* از پروتکلی استفاده می کند به نام *ARQ*

پروتکل *ARQ* : *Automatic Repeat request*

در این روش کنترل خطا به عهده ی فرستنده می باشد یعنی در صورتی که پیام به گیرنده نرسد فرستنده مجدداً آن را ارسال می کند

نکته ۲: *UDP* از پروتکلی به نام *FEC* استفاده می کند

پروتکل *FEC* : *Forward Error Control* :

در این روش کنترل خطا بر عهده گیرنده است یعنی اگر پیغامی به دست گیرنده نرسد یا گیرنده نتواند خطای آن را برطرف کند فرستنده مجدداً آن را ارسال نمی کند.

8-9 لایه شبکه :

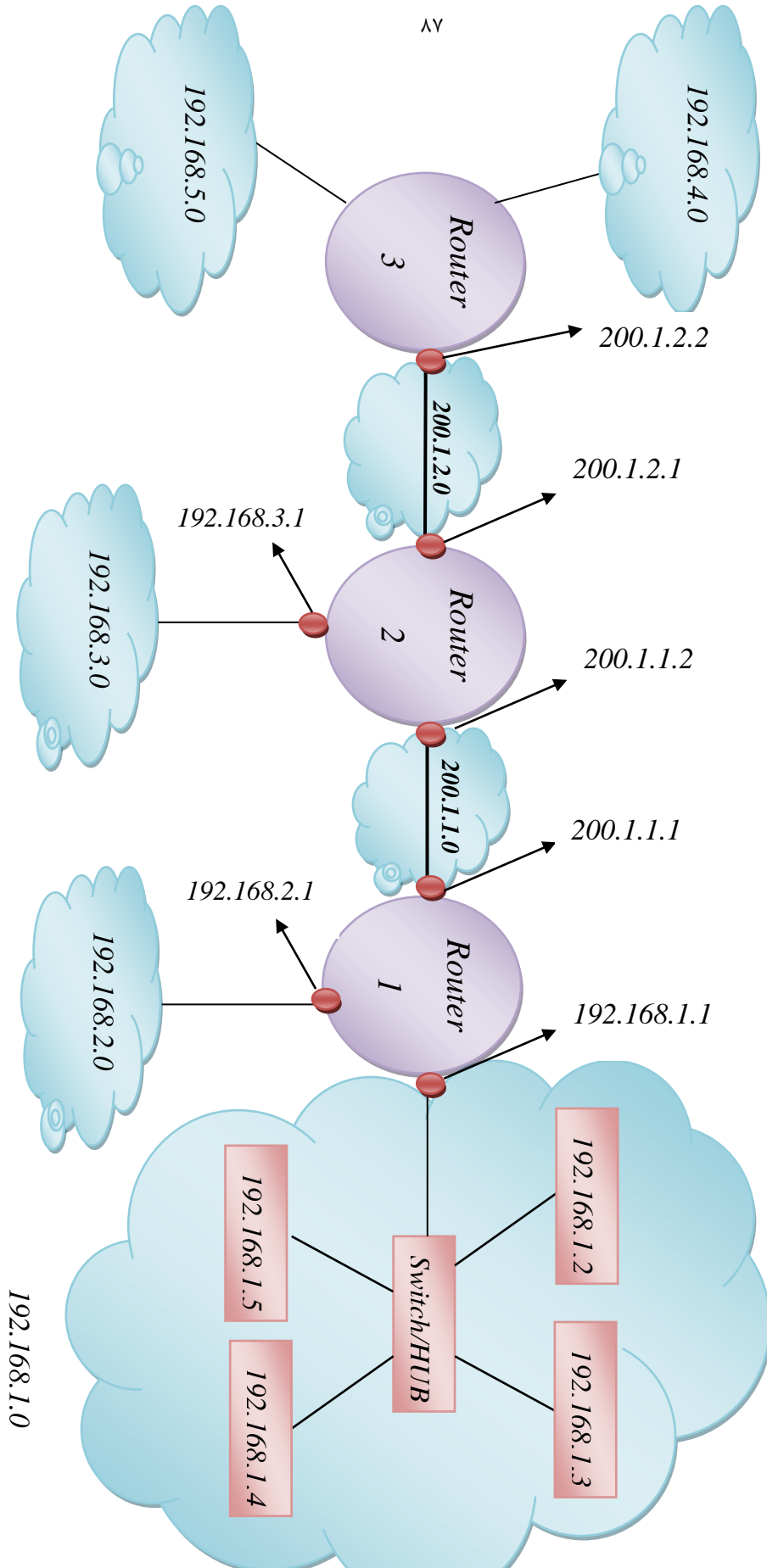
وظیفه لایه شبکه فقط مسیر یابی است.

مسیریابی را توسط دو الگوریتم انجام می دهد :

- ۱) *Static* یا *link state* : نمونه ای از این الگوریتم *OSPF* می باشد. در روش *Static* یک پیام به نام *LSP* به کل شبکه ارسال می شود تا ساختار شبکه را به دست آورد. *Static* به درد شبکه هایی می خورد که تغییراتشان کم است.

(۲) *Dynamic* یا *Distance vector* : نمونه ای از این الگوریتم *RIP* است. هر *IMP* فقط به کمک اطلاعات همسایگانش جدول مسیریابی را پر می کند.

۹-۹ جدول مسیریابی (*Forward Table*)



۹-۹-۱ جدول *RIP*:

فقط آدرس شبکه هایی که به آن متصل هستند را می نویسیم و ترتیب هم مهم نیست

۸۸

: Router1

جدول *RIP*

192.168.1.0
192.168.2.0
200.1.1.0

: Router2

جدول *RIP*

200.1.1.0
200.1.2.0
192.168.3.0

: Router3

جدول *RIP*

192.168.4.0
192.168.5.0
200.1.2.0

سوال: روش *RIP* را توضیح دهید؟ روشی است که با اطلاعات همسایگان کار می کند .

OSPF: ۲-۹-۹

به هر *Router* آدرس تمامی شبکه هایی که به آن وصل نیست را در جدول آن اضافه می کنیم

نکته *Next hop* : آدرس اولین *Inter Face* یی که در مسیر رسیدن به یک شبکه مقصد وجود دارد.

: Router1

جدول OSPF

Next hop

192.168.3.0/200.1.1.2
200.1.2.0/200.1.1.2
192.168.4.0/200.1.1.2
192.168.5.0/200.1.1.2

: Router2

جدول OSPF

192.168.1.0/200.1.1.1
192.168.2.0/200.1.1.1
192.168.4.0/200.1.2.2
192.168.5.0/200.1.2.2

: Router3

جدول OSPF

200.1.1.0/200.1.2.1
192.168.3.0/200.1.1.1
192.168.2.0/200.1.2.2
192.168.1.0/200.1.2.2