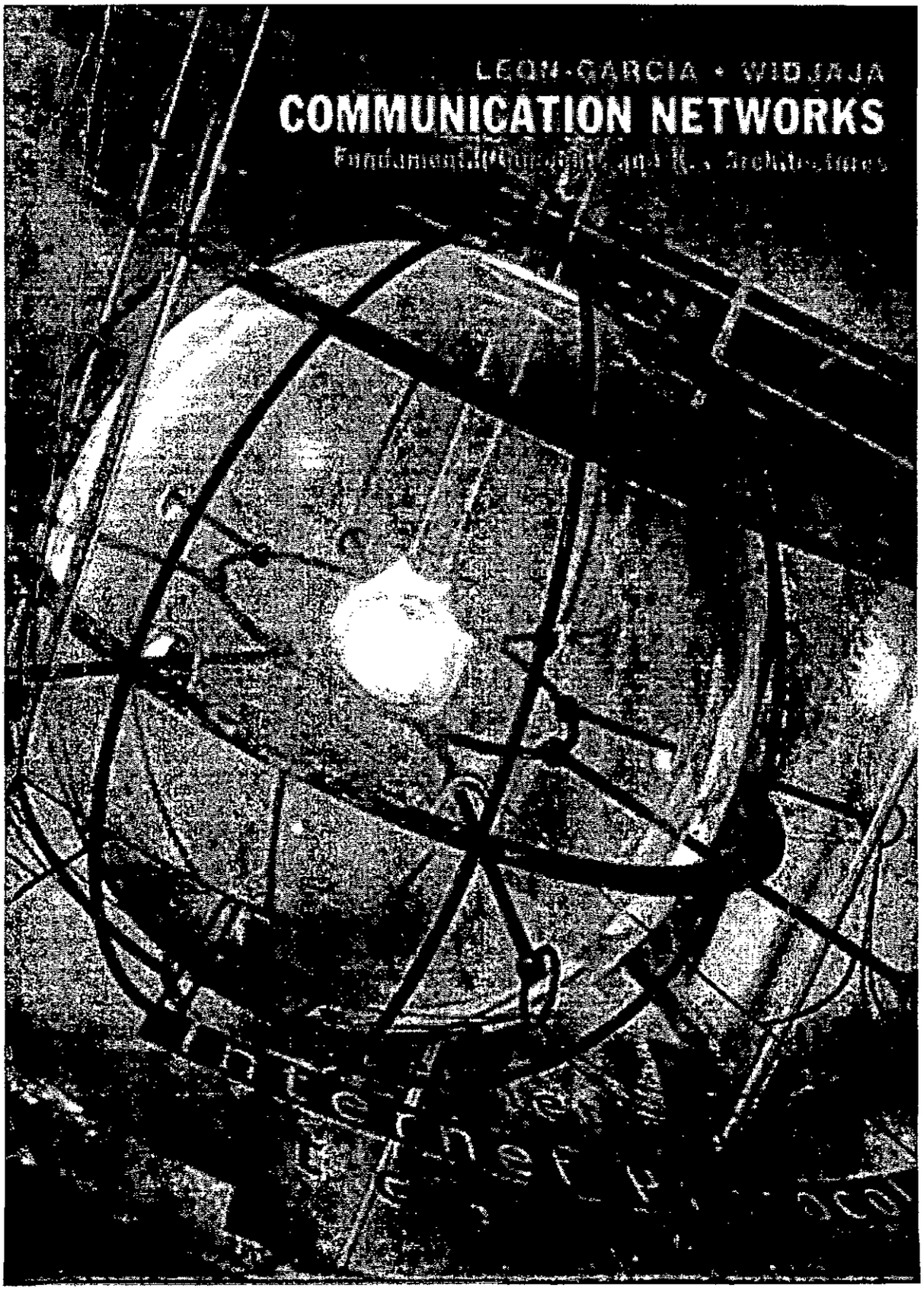
شبکه‌های کامپیوتری (1)

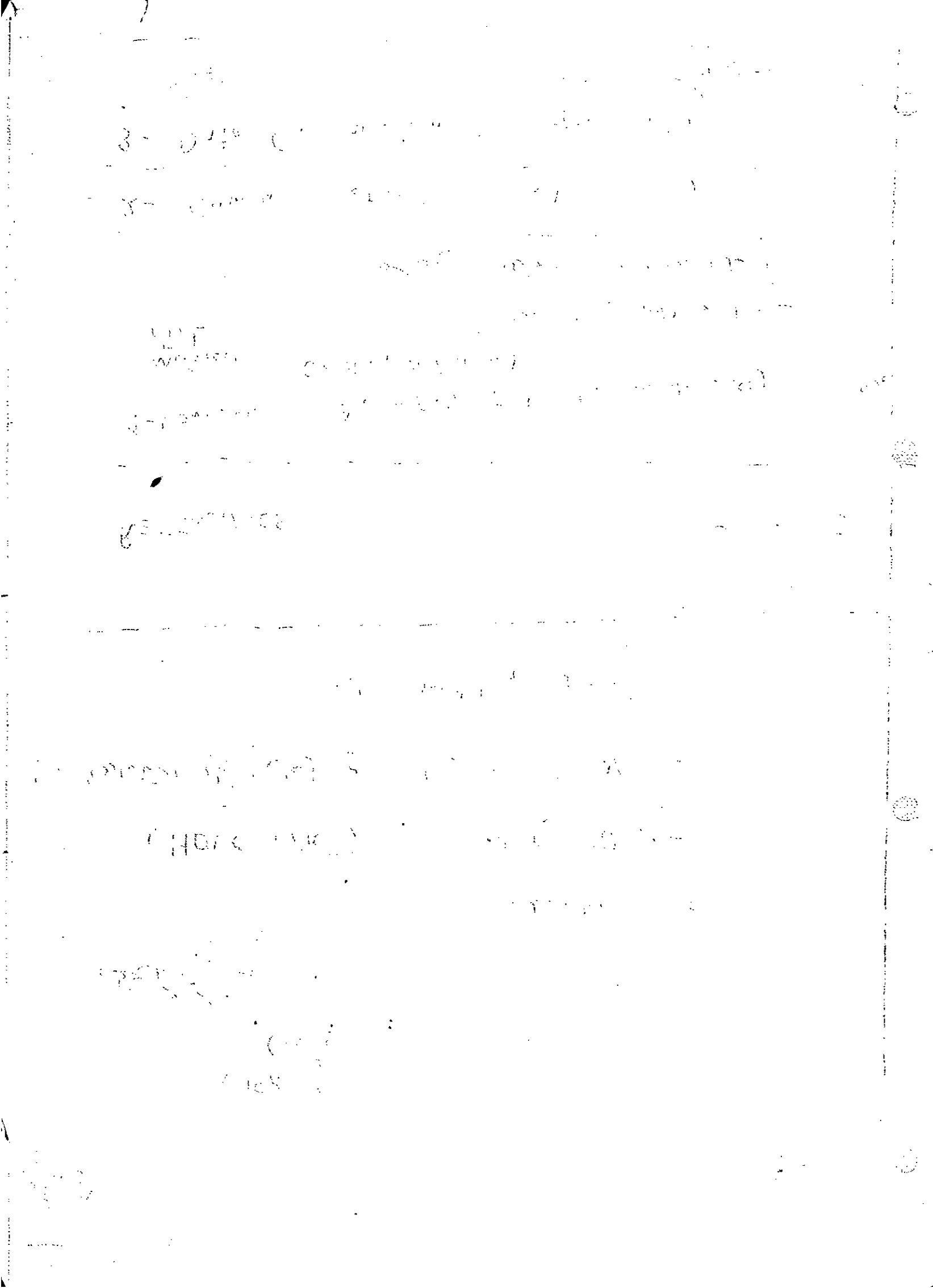
Communication Networks -1-



CEIT :: Amir Kabir University of Technology

« Dr. Sadeghian » [2005 - 2005]

« Nader Hari Ghanbari 7831004 »



SA

نادر حاجی قنبر
7831904

⊕ سرفصل مطالب :

۱- سرویس‌ها و شبکه‌های کامپیوتری

۲- معماری لایه‌ای (مدل OSI و مدل TCP/IP)
(دولایه یا بین هر مدل مطرح می‌گردد.)

۳- انتقال دیجیتال. (مشخصات کانال، مدولاسیون، کدینگ خط، .....)

۴- مالتی پلکسینگ

۵- پروتکل‌های peer-to-peer (ARQ ، HDLC)

۶- LAN ، شبکه‌های محلی، (پروتکل‌های دسترسی، استانداردها، .....)

۷* - شبکه‌های Packet Switching

اسلایدها با نیاز کاربران Network

⊕ مراجع درس :

REFERENCES :

1) Data and Computer Communication by W. Stallings

2) Data Communication and networking by Behrouz Forouzan

3- COMMUNICATION NETWORKS: Fundamental Concepts & Key Architecture (Garcia, Widjadja) McGraw HiLL

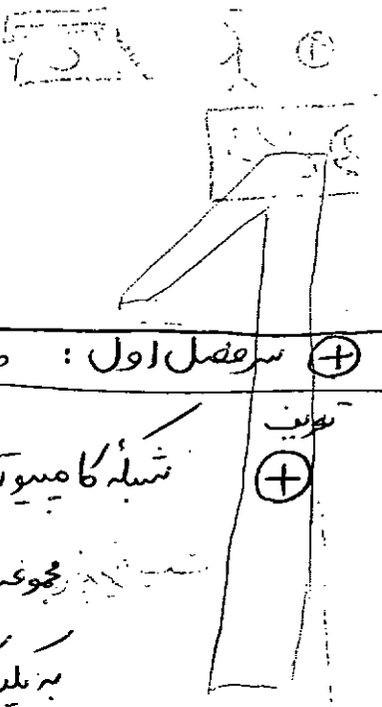
فصول ۱ تا ۶ (لایه‌های اول و دوم)
بقیه فصول در درس شبکه‌های کامپیوتری ۲ مطرح می‌شود.

2) Computer Networks (A. Tannbaum)

3- Data Communication & Computer Networks (W. Stallings)

ارزیابی:

۱- میان ترم	دوشنبه ۵/اردی بهشت / ۸۳	35%
۲- پایان ترم		50%
۳- کوییز		5%
۴- تکالیف		10%



⊕ سرفصل اول: مقدمات، سرویس های کامپیوتری، کلیات شبکه های کامپیوتری

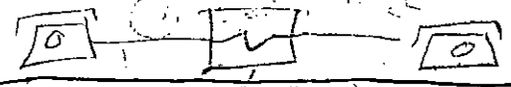
تعریف ⊕

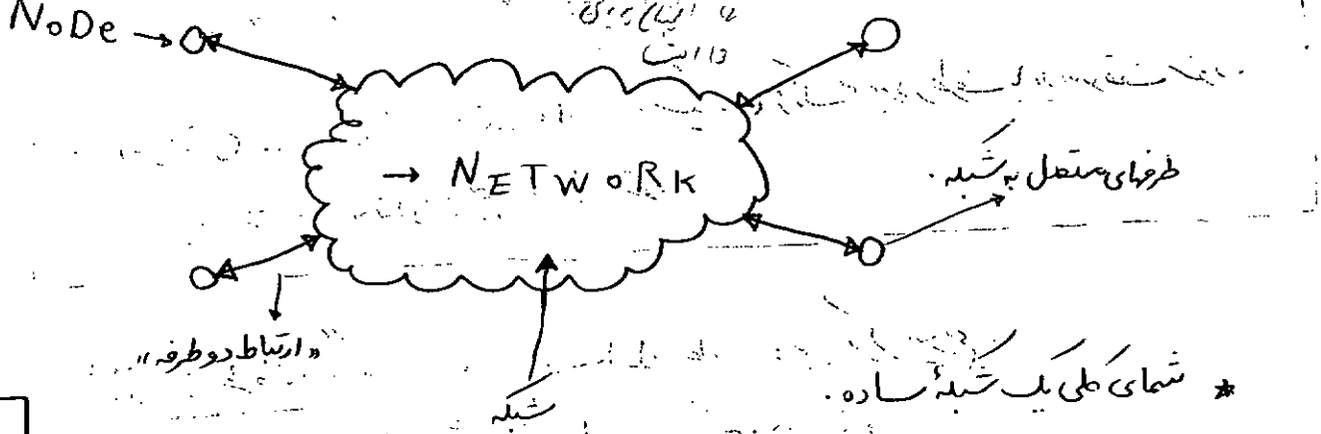
شبکه کامپیوتری: مجموعه‌ی است از کامپیوترهای مستقل که از طریق یک شبکه کامپیوتری به یکدیگر متصلند و یک سرویس را (انتقال فایل، صوت، ...) می دهند.

★ مستقل: کاملاً مستقل از هم و نه مانند اجزای یک سیستم کامپیوتری که از یک پالس ساعت یکسان تغذیه می شوند.

★ شبکه: کامپیوترها هم متصلند و می توانند با یکدیگر ارتباط برقرار کنند.

★ سرویس: مانند پست الکترونیک، .....





★ شمای کلی یک شبکه ساده.

شکل ①

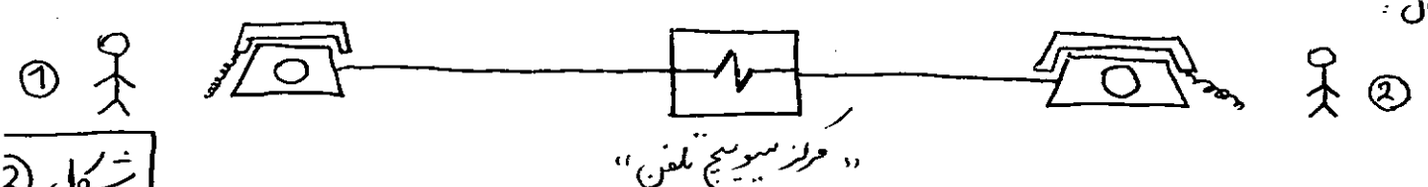
★ برای ادامه بحث باید به تعریفی از اصطلاح پروتکل دست پیدا کنیم.

★ PROTOCOL در لغت به معنای قرارداد و معاهده است.

★ با ارائه مثالی از شبکه ساده تلفنی به بررسی مفهوم پروتکل می پردازیم.

شبکه ساده تلفن

مثال (—)



شکل (2)

عملیات مورد نظر تماس طرف اول با طرف دوم است.

Connection _ oriented

① گوشی را بردار.

② گوش کن.

③ اگر صدایی به جز بوق آزاد شنیدی: آنگاه گوشی را بگذار و مکالمه امکان پذیر نیست.

④ شماره گیری کن. (ارسال شماره به مرکز سویچ)

⑤ بررسی خط شخص مورد نظر توسط مرکز سویچ.

⑥ اگر خط اشغال بود، سیگنال بوق اشغال را برای شخص 1 بفرست.

⑦ اگر خط آزاد بود، مرکز سویچ سیگنال زنگ را برای هر دو طرف می فرستد.

★ سیگنال زنگ برای شخص 1 به معنای موفق بودن تماس مرکز سویچ است.

★ " " " " 2 " " تقاضای مکالمه است.

⑧ اگر شخص 2 گوشی را برداشت، ارسال سیگنال زنگ به هر دو طرف باید متوقف شود و سیم دو طرف به گونه‌ای به هم متصل شود.

★ در این مثال ۳ طرف درگیر بودند، طرف اول، طرف دوم و مرکز سویچ.

★ هر ۳ طرف در این عملیات از روشها و قراردادها آگاهی داشتند.

★ پروتکل در واقع همین قراردادهای مشخص و معینی است که بین این طرفین قرارداشت.

★ برای عملیات پایان مکالمه، مرد اول گوشی را می گذارد که سیگنالی را به مرکز سویچ مشتقل می کند که پس از آن مرکز سویچ ارتباط دو طرف را قطع می کند (و سیگنال بوق اشغال را به طرف دوم می فرستد)

تعریف:

⊕ پروتکل: PROTOCOL:

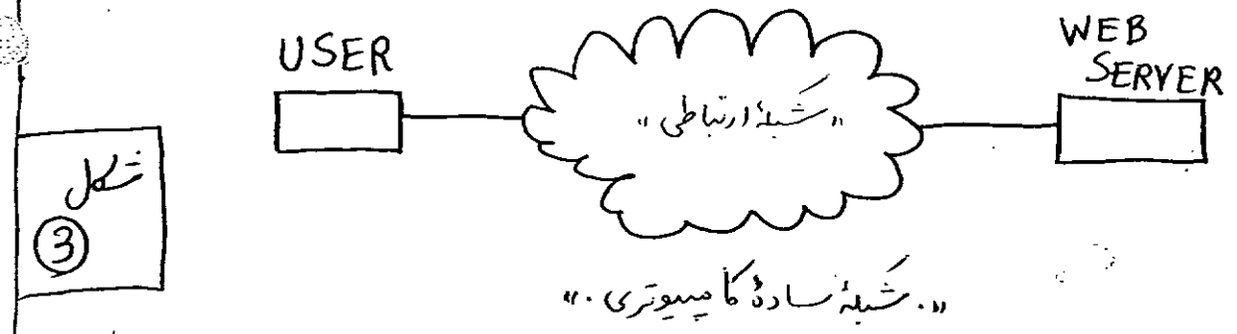
+ یک رشته از عملیات مشخص که برای انجام کاری بین طرفین قرارداد می‌شود.

+ طرفین باید از عملیات آگاهی داشته باشند. HTTP, FTP, SMTP, Simple

+ طرفین باید همگی بر سر قرارداد، توافق داشته باشند.

+ COMPLETENESS: در هر شرایطی باید وظیفه هر طرف کاملاً مشخص باشد.

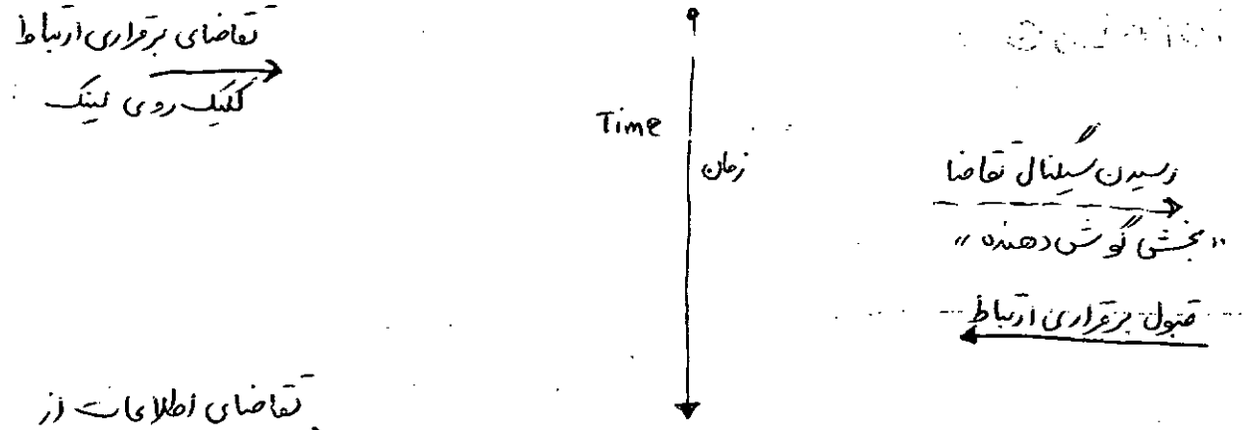
⊖ مثال:



شکل ③

«شبکه ساده کامپیوتری»

★ سرور کسی است که دائماً به خط ارتباطی اش گوش می‌دهد.



«قطع ارتباط»

★ سرویس مورد نظر در اینجا ارسال یا انتقال اطلاعات است.

★ پروتکل در اینجا متأثر از پارامترهای بسیار زیادی است.

برخی از پارامترهای تأثیرگذار در روی سرویسهای مختلف : (پروتکلهای مختلف)

① ـ تأخیر DELAY

درمثال تلفن پس از برقراری ارتباط، ارتباط Real Time است.
ولی درمثال WEB این طور نیست.

② ـ دقت PRECISION

درمثال تلفن اگر نویز باشد قابل قبول است ولی درمثال WEB
یک صفحه درهم ریخته قابل قبول نیست.

③ ـ RELIABILITY

④ ـ حجم و نوع اطلاعات (صوت، فایل ....)

⑤ ـ هزینه و مناسب بودن COST & CONVENIENCE

International Standard Organization
ITU Union
IEEE
پروتکل ها توسط سازمان های استاندارد می کنند

Network TOPOLOGY

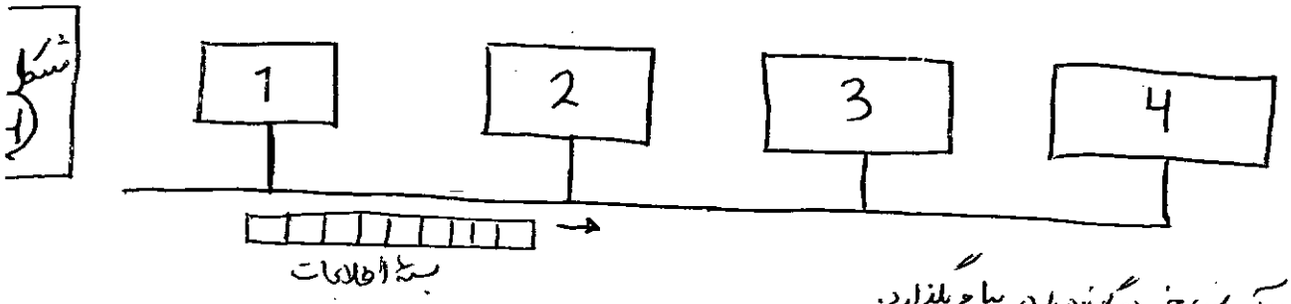
توپولوژی شبکه :

★ توپولوژی در لغت به معنای هندسه اتصال است.
★ از مهمترین عوامل اثرگذار در روی پروتکل ها است.

⊕ انواع توپولوژی های شبکه :

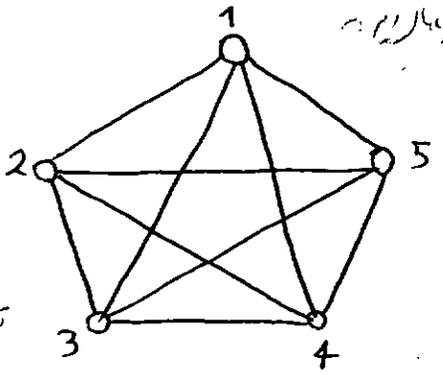
① BUS Topology :

⊛ یک خط دارد.



★ فرستنده باید آدرس خود و گیرنده را در پیام بگذارد.
★ همه باید روی خط به گوش باشند و اگر پیام مال آنها نبود آن را برندارند.

② شبکهٔ Fully Connected Mesh



⊕ شکل ⑤

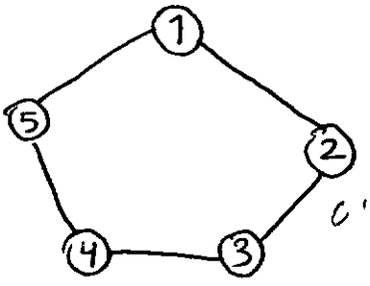
* $\frac{n(n-1)}{2}$ سیم.

رابطهٔ ①

* در اینجا دیگر لازم نیست هر گوشی بدهند و گیرنده پیام بگذارد.
* لازم نیست فرستنده آدرس خودش و گیرنده را روی خط بگذارد ولی فقط باید خط مورد نظر را انتخاب کند.
* در اینجا برای اتصال ۵ نود ۱۰ خط بکار رفته و هزینه بالایی می‌طلبد ولی توپولوژی Bus یک خط دارد.
* در توپولوژی Bus اگر نقطه‌ای از خط قطع شود ارتباط اینی از شبکه با بقیه دیگر قطع می‌شود ولی در اینجا فقط با قطع شدن یک نقطه، ارتباط دو نود با هم قطع می‌شود و بقیهٔ ارتباطات دچار مشکل نمی‌شوند.

---

③ شبکهٔ حلقه‌ای Ring



⊕ شکل ⑥

* n سیم مورد نیاز است.

رابطهٔ ②

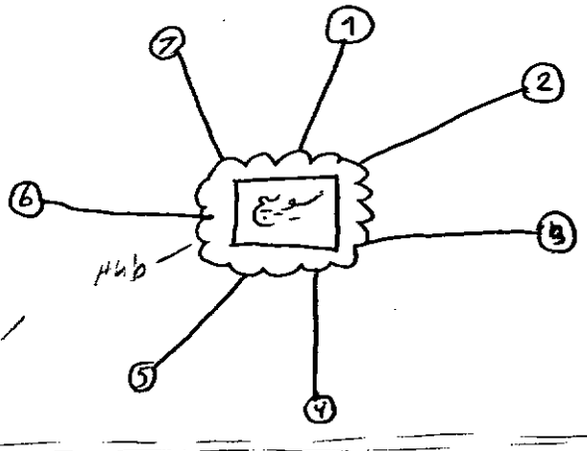
* در اینجا فرستنده آدرس خود و گیرنده را روی پیام نوشته و به نود بعدی می‌فرستد.
* اگر پیامی به نودی رسید، نود باید اگر پیام مال او بود آنرا دریافت کند و اگر مال او نبود به بعدی بفرستد.
* در توپولوژی Bus مشکل همزمانی پیش می‌آید و باید قراری بین هم باشد که چه کسی از خط استفاده کند.
* در توپولوژی Ring مشکل همزمانی به روش فرستادن بسته‌ای به نام Token به نود بعدی حل می‌شود.
* در توپولوژی Bus: هر نودی باید به خط گوش دهد، اگر خط مشغول بود منتظر می‌ماند ولی اگر آزاد بود بسته‌اش را می‌فرستد و باز گوش می‌کند که آیا پیامش با پیام دیگری مخلوط شده یا نه و اگر مخلوط شده بود آنرا پس از تأخیری به مقدار تصادفی مجدداً روی خط بفرستد.

SA

* در شبکه حلقه‌ای اگر ارتباط ۵ با ۱ قطع شود، اولاً هیچ کسی به قبلی‌هایش نمی‌تواند پیام بفرستد ثانیاً راه حل Token دچار مشکل می‌گردد.
* در مورد روش Token بعداً مفصلاً بحث خواهد شد.

④ شبکه ستاره‌ای STAR

شکل ⑦



* n سیم مورد نیاز است.
رابطه ③

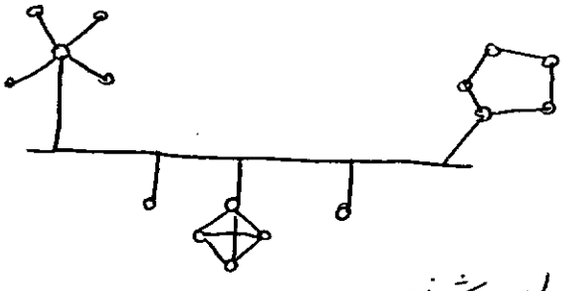
کم مزیت ترافیک

* ارتباط می‌تواند تقریباً REAL-TIME باشد.
* در این جا بحث هرج‌ومرج نداریم ولی بحث اشغال بودن خط مطرح می‌شود (مانند ارتباطات تلفنی)
* اگر مرکز سویچ خراب شود کل شبکه از بین می‌رود (عیب بزرگ)
* در این جا فازهای برقراری ارتباط با سویچ مطرح می‌شود که در توپولوژی های دیگر مطرح نیست (مانند شبکه تلفن)

⑤ شبکه‌های دورگه Hybrid

* تلفیقی از توپولوژی های مختلف

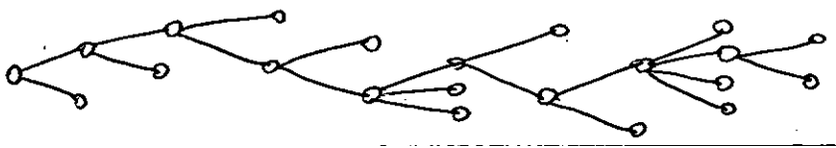
شکل ⑧



* مخلوطی از مسایل توپولوژی های دیگر هگی باهم مطرح می‌شود.

⑥ شبکه های سلسله مراتبی HIERARCHICAL

شکل ⑨



تکنولوژی انتقال : TRANSMETION TECHNOLOGY :

A - روش BROADCAST

B - روش point-to-point

یا صدا آنالوگی شود - تبدیل تصویر زنگ و آنالوگ دیجیتال

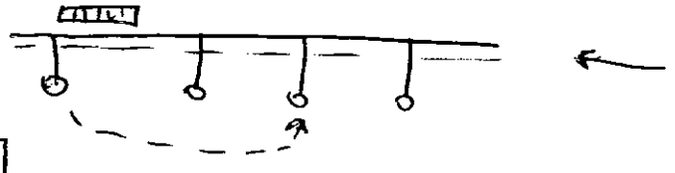
انتقال

شکل (10)

(A) روش Broad cast :

رسانه‌ای وجود دارد که همه نودها پیام را به آن می‌دهند و همه آن را می‌توانند ببینند.

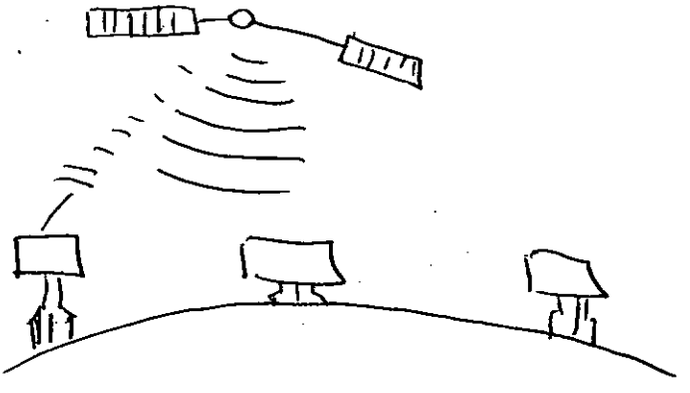
مثل توپولوژی Bus یا تکنولوژی اترنت‌های هاب‌دار (رسانه: Bus یا اترنت‌های هاب‌دار)



(*) پیام در کل Bus پخش می‌شود.

شکل (11)

* روش Broadcast در توپولوژی Bus



(*) پیام در کل فضا منتشر می‌شود.

شکل (12)

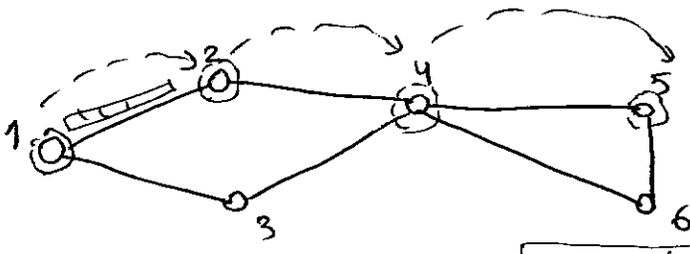
* روش Broadcast برای اترنت‌های زمینی و یک اترنت ماهواره‌ای

(*) به دلیل انتشار سیگنال در کل فضا، به مسایل امنیتی برخورد می‌کنیم.

Mo

(B) روش Point-to-point :

- ✱ پیام از هر نقطه مستقیماً به نقطه دیگری رود، یا از طریق نقاط میانی بین این دو قرار دارند.
- ✱ در اینجا دیگر انتشار در کل فضا نداریم، هرچند نودهای میانی داریم.



Path:
1-2-4-5

1 to 5

شکل (13)

✱ نودهای 3 و 6 دیگر نمی‌توانند پیام منتشر شده‌ای ببینند

تعریف ⊕ آدرس دهی : Routing

انتخاب مسیر (نقاط میانی) برای ارسال پیام از یک نقطه به نقطه‌ای دیگر در روش انتقال نقطه به نقطه.

⊕ شبکه محلی : Local Area Network : LAN

شبکه‌های کوچک محلی، مثلاً در حد سایت یک دانشگاه، و از این قبیل

⊕ شبکه‌های شهری: Metropoliton Area Network : MAN

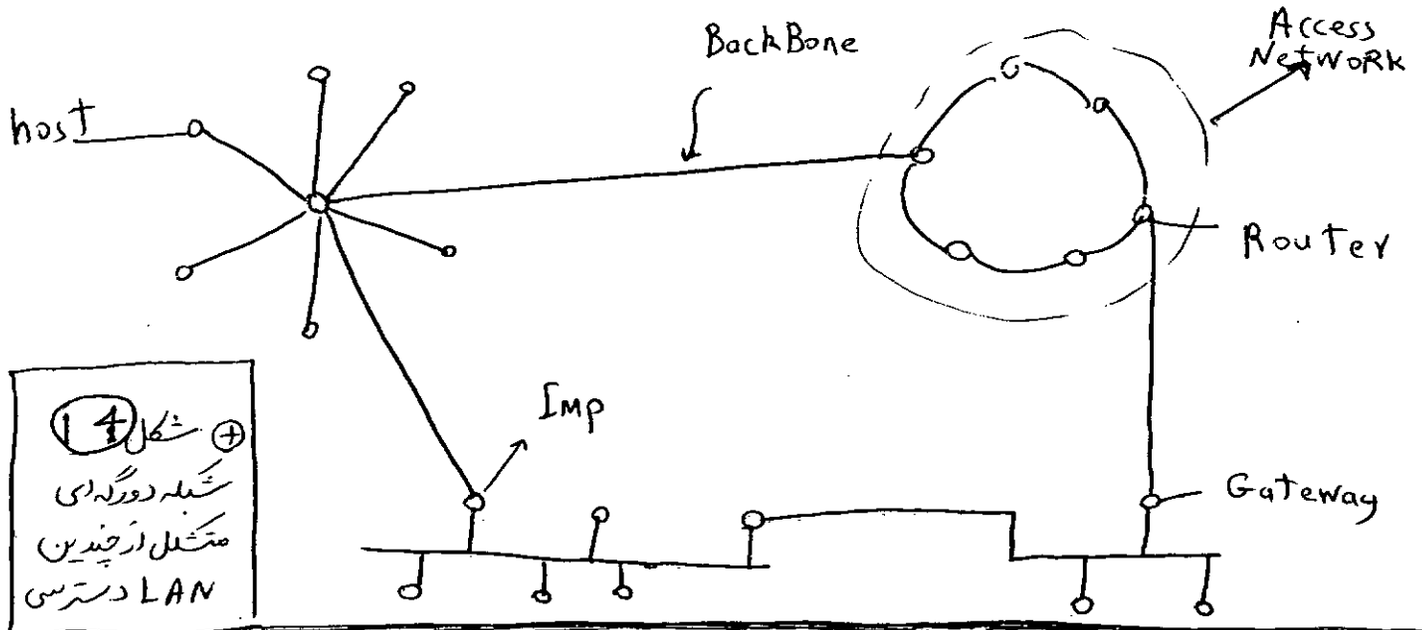
تعدادی از شبکه‌های محلی هم پیوسته.

⊕ شبکه‌های وسیع: Wide Area Network : WAN

شبکه‌های متفاوت هم متصل در سطح یک منطقه بسیار وسیع، مثل یک کشور یا یک قاره.

✱✱ تکنولوژی انتقال در انتخاب پروتکل مربوطه آن شبکه نقش دارد.

✱ مثلاً وقتی در روش Broadcast گیرنده‌ای می‌خواهد پیام بگیرد بقیه هم باید مراقب باشند که آن پیام را نگیرند.



شکل ۱.۴ شبکه دوگانه‌ای متشکل از چندین LAN دسترسی

اصطلاحات : Termology

* host_1 : کامپیوترهایی در هر شبکه‌ی کوچک که میزبان آن شبکه است.

* Imp_2 : INTERMEDIATE MESSAGE PROCESSOR
یا GATEWay ها و
یا Router ها.
کامپیوترهای متصل کننده شبکه‌های میانی.

* Access Network_3 : هر شبکه‌ی محلی کوچک را یک شبکه دسترسی گوییم.

* Back Bone_4 : شبکه‌ی انتقال دهنده دو شبکه‌ی دسترسی

• SCABALETY : امکان گسترش شبکه و افزودن شبکه‌های دسترسی به شبکه‌ی موجود.

• RELIABILITY : ثبات سیستم شبکه.

PARAMETERS

* دو عامل با لا نقش مهمی در انتخاب پروتکل شبکه دارند.

⊕ آدرس دهی : ADDRESSING

وقتی فرستنده‌ای می‌خواهد پیامی به گیرنده بفرستد باید آدرس گیرنده را دقیقاً تعیین کند.
یعنی هم شماره یا آدرس شبکهٔ دسترسی و هم آدرس کامپیوتر مورد نظر را در آن شبکه مشخص کند.

⊗ روشهای آدرس دهی : ADDRESSING METHODS :

Ⓐ روش Flat :

در شبکه‌های محلی کاربرد دارد،
هر نود یک شماره دارد. صلماً در شبکه‌های بسیار بزرگ کاربرد ندارد.

Ⓑ* روش سلسله مراتبی یا Hierarchical

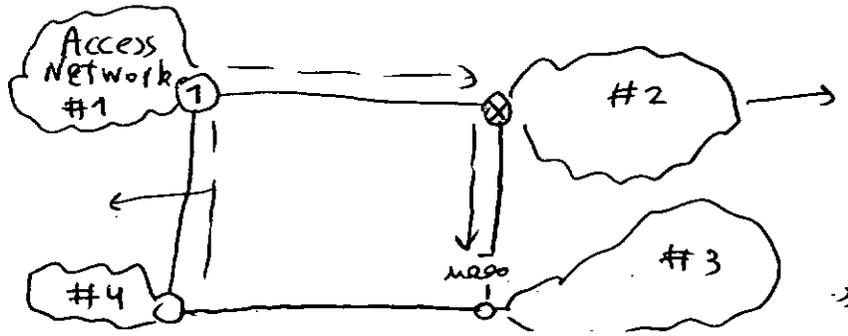
شماره شبکه ← 118 - 27 - 378 ← شماره مقصد
شماره شبکه دسترسی ↑

هر کامپیوتر یک شماره دارد،
هر شبکه هم یک شماره دارد و آدرس دهی مانند آدرس دهی پستی است (سلسله مراتبی)

⊕ کنترل ترافیک : TRAFFIC CONTROL

: کنترل ترافیک به طوری که ترافیک در کل شبکه بطور صحیح انجام شود.

* Congestion : اتباشتگی و پر شدن بافر در یک نود،
در شبکه‌های بزرگ معمولاً اتباشتگی یک نود IMP می‌تواند باعث افزایش اتباشتگی نودها در شبکه‌های دسترسی دیگر شود، که یک راه حل ممکن، استفاده از مسیر دیگری در شبکه است.



⊛ اگر این شبکه‌ای Congestion شد بهتر است مسیر دیگری از طریق نود ۴ برای ارسال پیام استفاده شود.

مشکل (15)

QOS : QUALITY OF SERVICE ⊕

سرویس‌های شبکه همواره و به طرز قابل قبولی جوابگوی نیازهای کاربران و مدیران شبکه باشد،
مثلاً خرابی آن هر روز نباشد و مثلاً چند ماه یکبار خرابی داشته باشد.

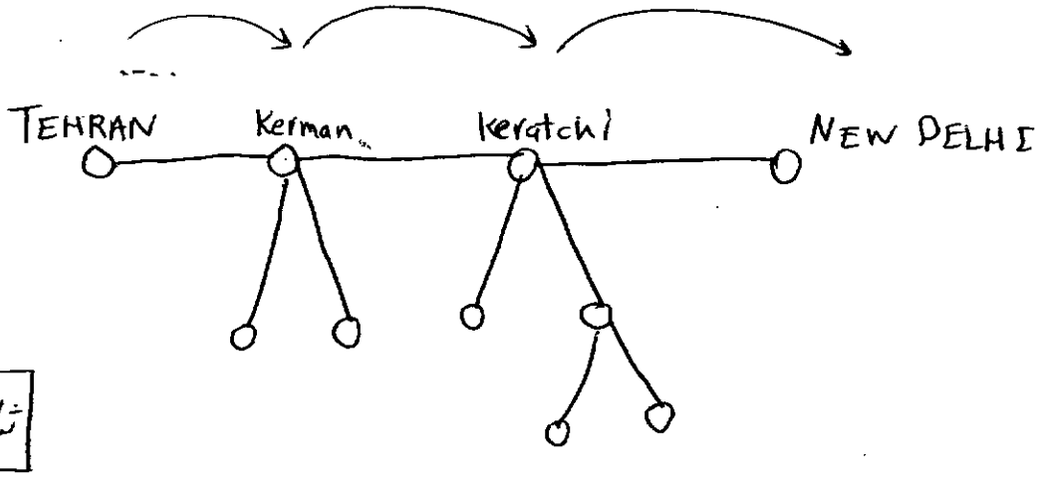
NETWORK MANAGEMENT:

⊗ وظایف مدیریت شبکه:

⊕
- خطاها، — Faults
- امنیت سیستم، — SYSTEM PROTECTION
- کیفیت سرویس سیستم،
- نظارت بر کاربران
- تجهیز شبکه و .....

SWITCHING: ⊕ روشهای سویچینگ:

Ⓐ شبکه‌های تلگراف: (مقدمه‌ای بر روش سویچینگ Store & Forward)



شکل (16)

★ یک اپراتور داریم که سیگنالهای الکتریکی نقطه و خط را می‌فرستد و در طرف دیگر نیز اپراتور دیگری هست که علایم را دریافت می‌کند.

★ در واقع در این شبکه‌ها، وظیفهٔ سویچینگ بر عهدهٔ اپراتورها می‌باشد.

MORSE CODE (DotLine)

- نقطه: سیگنال کوتاه الکتریکی
- خط: سیگنال بلندتر از نقطه

A	∘ —
B	— ∘ ∘ ∘
⋮	
E	∘

⊛ الفبای مورس برای ارسال اطلاعات پیام روی خطوط شبکه‌ی تلگراف

★ ایراتور تهران پیام را به کرمان می‌فرستد.

ایراتور کرمان پیام را می‌خواند و می‌فهمد که پیام مربوط متعلق به شهر دهلی است، پس صلاح خروجی به کراچی را انتخاب می‌کند و پیام را روی خطی می‌فرستد. (عمل سوییچینگ)

به همین ترتیب بقیه‌ی ایراتورهای شهرهای دیگر این کار را انجام می‌کنند تا پیام به دهلی برسد.

★ به این روش سوییچینگ روش MESSAGE Switching می‌گوییم.

① ⊛ Message Switching:

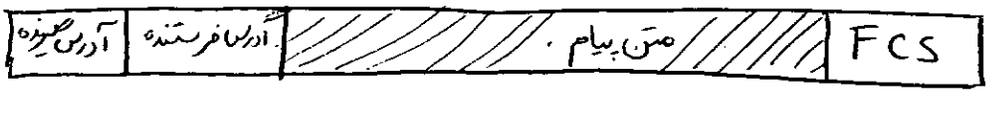
از همان ایده شبکه‌های تلگراف می‌توان در شبکه‌های کامپیوتری نیز استفاده کرد.

کنترل خطا: در هر نودی باید صحت اطلاعات دریافتی تأیید شود و اگر اطلاعات خراب بود، باید تقاضایی به نود قبلی جهت ارسال اطلاعات مجدد فرستاده شود.

FCS: Frame checksum

قسمتی از پیام که برای چک کردن صحت اطلاعات به آن اضافه می‌شود.

پیام ارسالی:



شکل ۷

① ⊕ به این روش STORE & FORWARD می‌گوییم.

⊛ نودهای بین راه در این روش وظایفی بعهده دارد، علاوه بر گفته‌های بالا در صورت درستی اطلاعات این نود باید یک Acknowledge به نود قبلی بدهد تا نود قبلی نسخه پیام را که برای مجدداً فرستادن در حافظه‌اش نگهداری می‌کند، پاک کند.

⊛ روش Store & Forward ایراد عمده‌ای دارد که Real Time نبودن ارتباط و تأخیر زیاد در عملیات فرستادن پیام است.

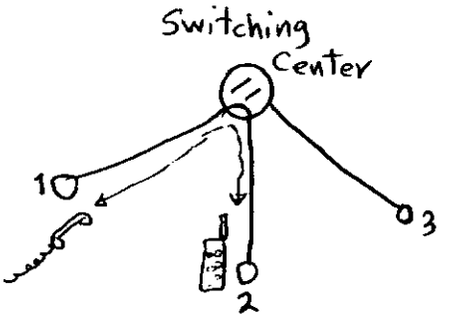
Ⓑ روش سویچینگ مرکز تلفن:

سویچینگ مداری

Circuit Switching

★ یک مرکز سویچ ارتباط مستقیم سیمی دو نود فرستنده و گیرنده را برقرار می‌کند.

★ ارتباط در این جا REAL TIME است. (تقریباً)



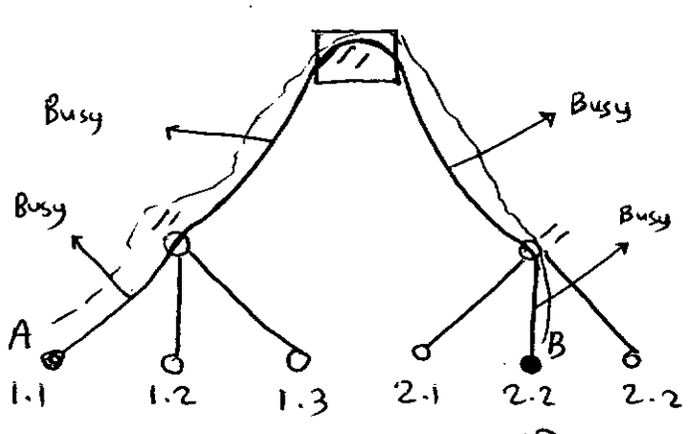
شکل (18)

★ مرکز تلفن و نودهای متصل به آن که تشکیل یک شبکه‌ی تلفنی را می‌دهند

★ ارتباط 1 و 2 برقرار است و دیگر 3 نمی‌تواند با هیچ کدام از آن‌ها ارتباط برقرار کند.

★ ولی در روش Store & Forward مشکل مشغول بودن خط وجود ندارد. (با فرنگیِ درهرنود.)

شکل (19)



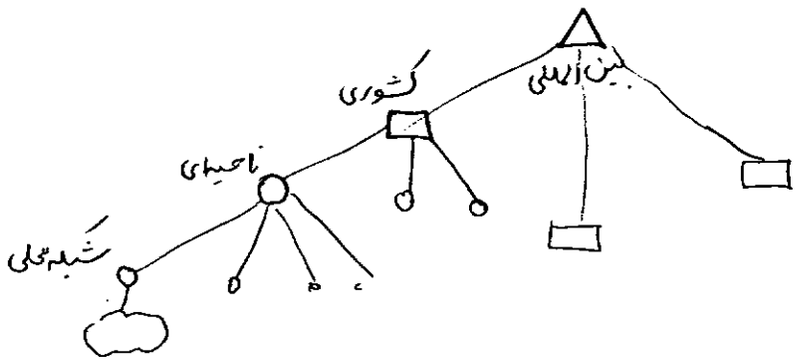
« شبکه تلفنی با مرکز سویچ سلسله مراتبی »

⊕ گسترش شبکه‌ی تلفن

★ اگر A بخواهد با B ارتباط برقرار کند، شماره ۲۲ را می‌گیرد که بدین طریق مرکز سویچ محلی خودش می‌فهمد که این اتصال محلی نیست و تقاضایی به مرکز بالاتری دهد و همین ترتیب تا به مقصد برسیم.

Mo

شکل ۵.



مرکز تلفن‌های سلسله مراتبی

* در شبکه‌های سوییچینگ مداری یک فاز برقراری ارتباط قبل از خود ارتباط مطرح است که این بحث در شبکه‌های Store & Forward وجود ندارد ولی بعد از ارتباط، ارتباط Real Time است.

« مقایسه روش‌های سوییچینگ مداری و سوییچینگ پیغامی »

Message Swithing
Vs
Circuit Switching

« معیارهای مقایسه »

۱- تخصیص خط LINE Pedication (در روش اول تخصیص خط (فاز اولیه) نداریم)

۲- تشخیص خطا ERROR CHECKING (در شبکه‌های سوییچینگ سیمی تشخیص خطا نداریم)

۳- REAL TIME بودن (روش اول REALTIME نیست)

۴- تأخیر DELAY (اندازه یا محاوتأثیر آن روی تأخیر)

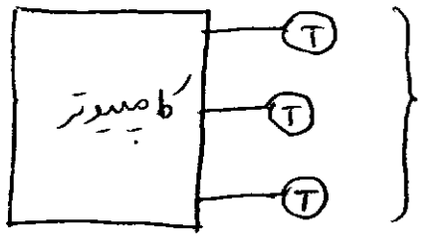
* همانطور که دیدیم در روش سوییچینگ قبلی معایب بسیاری داشتند که بخاطر همین دلیل از آنها کمتر در شبکه‌های کامپیوتری استفاده می‌شود.

* در شبکه‌های کامپیوتری از روشی موسوم به Packet Switching استفاده می‌شود.

* X * مروری بر سیر تاریخچه و تحول شبکه‌های کامپیوتری از آغاز: (شکل‌ها در صفحه بعد)

Terminal ها	PSTN و Modem شبکه‌های تلفنی فرینه‌ی بالا	Multiplexer DeMultiplexing تقسیم پهنای زمانی بین ترمینال‌های متصل به شبکه‌ی تلفنی	Front end Processor

« اتصال ترمینال های مختلف به کامپیوتر مرکزی »

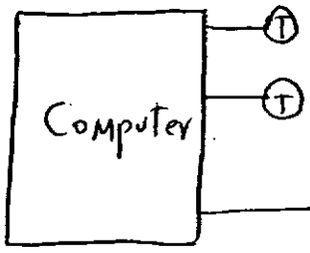


Terminals

شکل 21-1

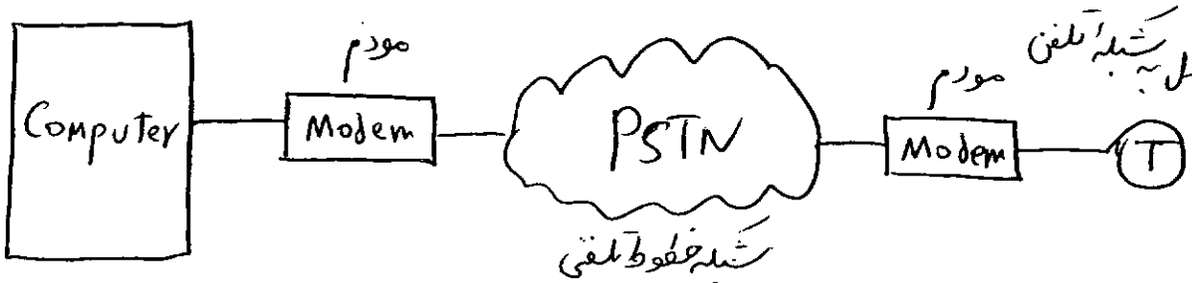
★ ترمینال، یک کامپیوتر نیست (قدرت پردازشی ندارد)

« تاریخچه و سیر تحول کامپیوترها و ارتباط آنها با یک سری ترمینال های مرتبط با کامپیوتر مرکزی »



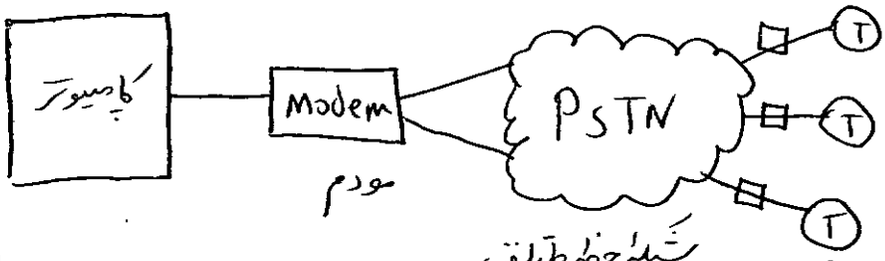
ترمینال نودتر

21-2



ترمینال متصل به شبکه تلفن

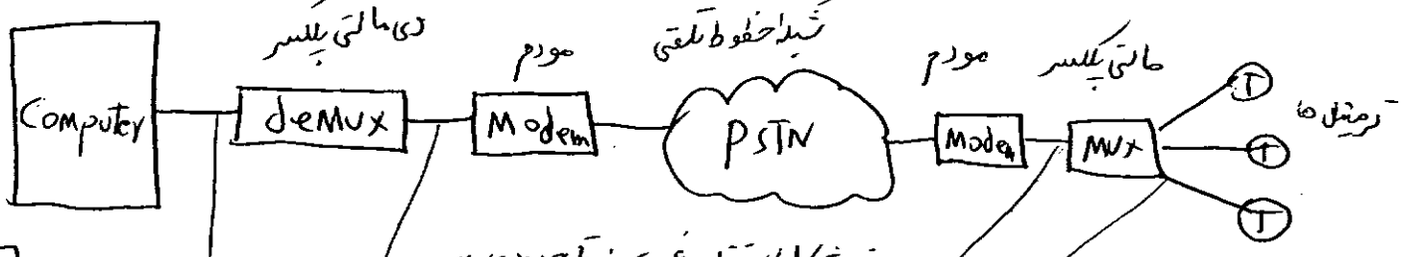
21-3



ترمینال های مرتبط به خطوط تلفن از طریق مودم

(در اینجا مشکل استفاده غیربهینه از لحاظ زمانی داریم)

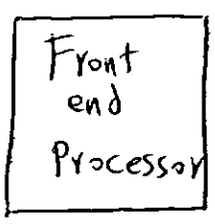
21-4



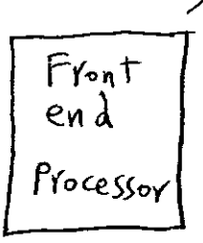
★ رفع مشکل استفاده غیربهینه تا حدودی »

21-5

21-6



تشخیص ترمینال فعال در هر لحظه و ارسال اطلاعات آن روی خط به طریق که پردازنده دیگر بفهمد اطلاعات از آن کدام یک است.



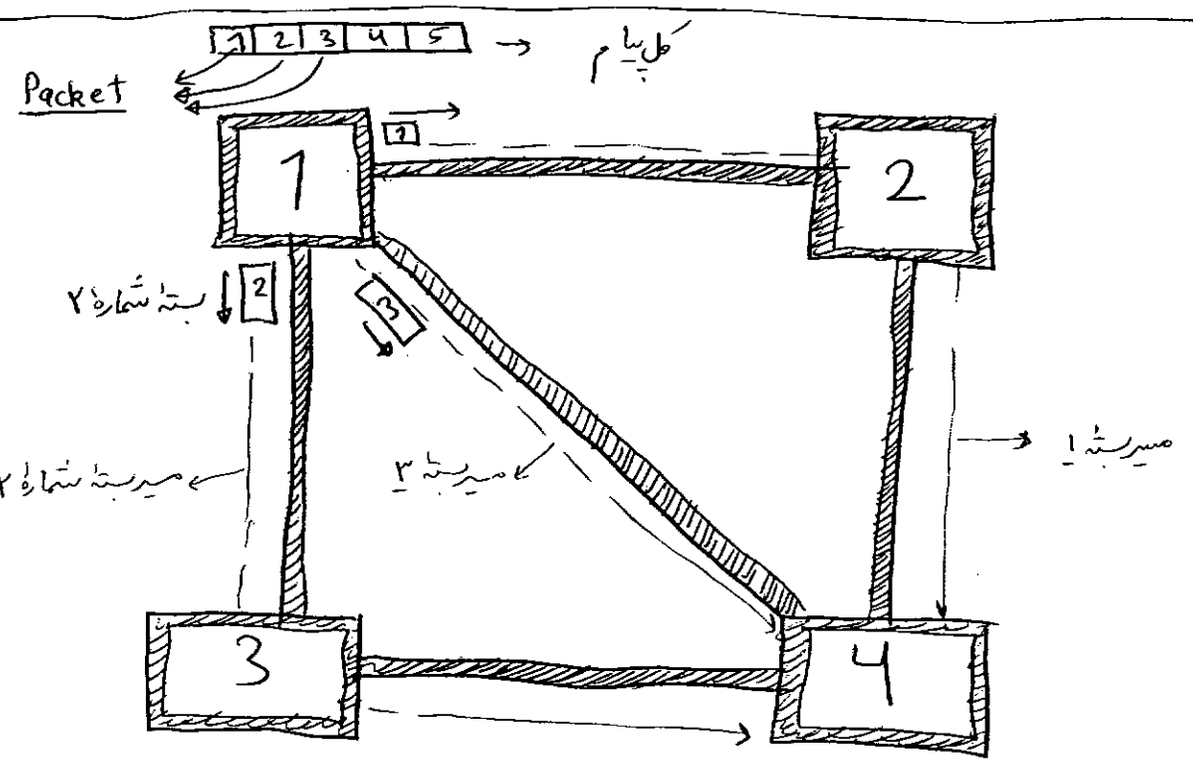
رفع کامل مشکل غیربهینه بودن استفاده از خطوط تلفن که هزینه بالایی دارند.

با استفاده از دو پردازنده متصل به مالتی پلکسر و دی مالتی پلکسر

★ (رکت های فوق تعدادی ترمینال و یک کامپیوتر داشتیم و نه یک سری کامپیوتر هم متصل !!!))

ⓒ روش بسته هسته ؛ Packet Switching

* تقسیم هر پیام به تعدادی بسته به اندازه‌ی معین و محدود و نسبتاً کمینه.
* از روش Message Switching استفاده می‌کنیم.
* مزایای روش Message Switching وجود دارد ولی معایب آن نه.
* گیرنده باید منتظر تمامی بسته‌های پیام بماند.



* ارسال پیام بسته بسته شده از ۱ به ۴ (هر بسته از هر مسیر دلخواه)

شکل (22)

ⓐ تعریف :

سوییچینگ :

متود ارتباط و ارسال اطلاعات بین نودهای یک شبکه. (انتخاب مسیر در هر نود و شیوه‌ی کلی اجرای این مسأله)

* سوییچینگ از Routing متمایز است.

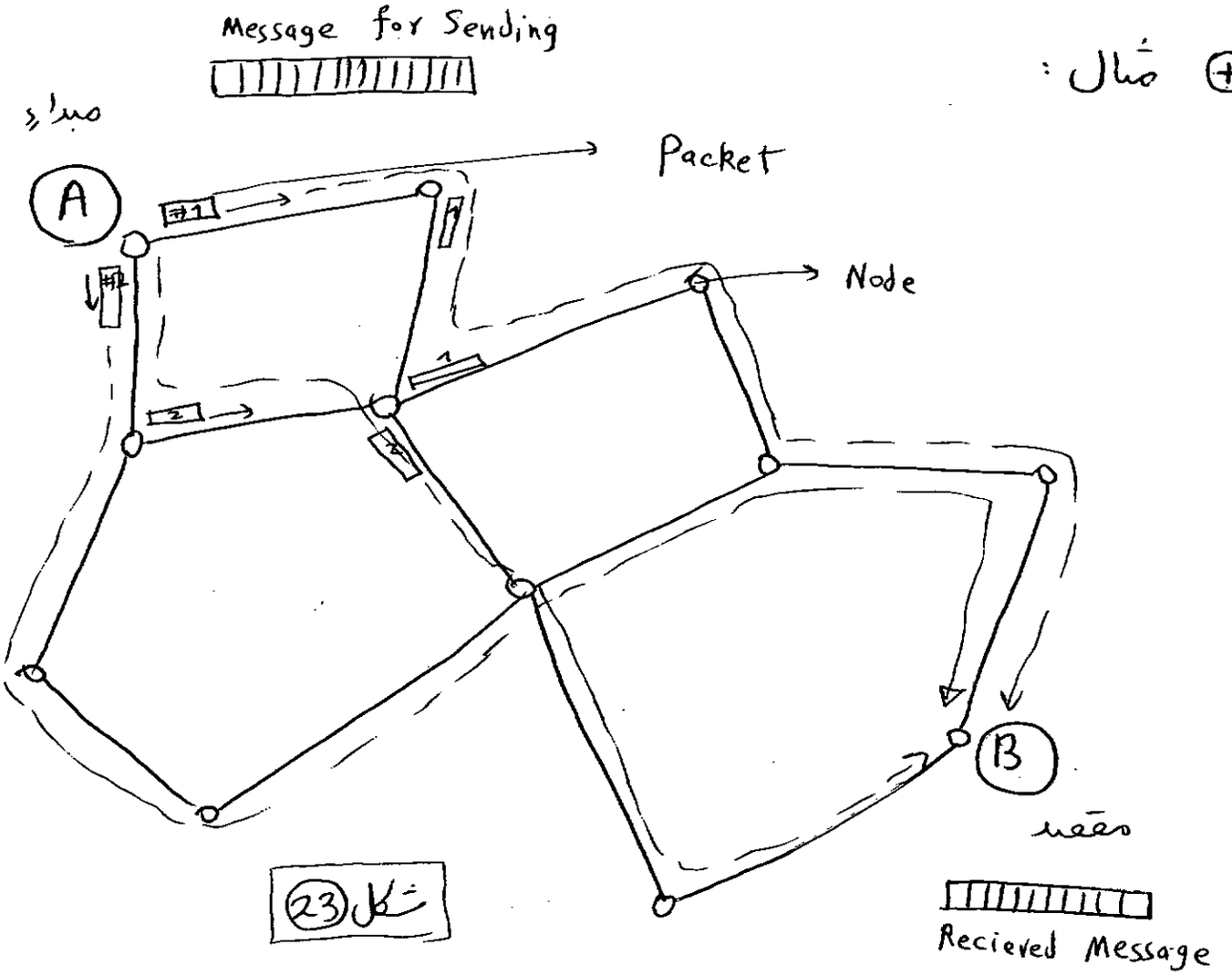
⊕ ادامه مبحث سوییچینگ :

سه روش سوییچینگ داریم:

1- MESSAGE Switching
2- CIRCUIT Switching
3- PACKET Switching

ادامه روش PACKET SWITCHING :

⊕ مثال :



شکل (23)

" شبکه کامپیوتری مورد بحث در مثال ارسال اطلاعات به روش سوییچینگ Packet "

★ فرستنده اطلاعات را به بسته های هم اندازه تقسیم می کند و با هر بسته به مانند پیامی در روش سوییچینگ پیام برخورد می شود.

ادامه مثال :

* گیرنده باید تمامی بسته‌ها را بگیرد،
* بدین منظور باید شماره‌گذاری انجام گیرد.
* در کامپیوتر مقصد عمل سرهم بندی روی بسته‌ها انجام می‌شود و پیام کامل بدست می‌آید.
* کامپیوتر مقصد باید فضایی را برای پیام در نظر گیرد، یعنی فضایی خالی برای کل پیام در نظر گیرد.
* کامپیوتر مقصد باید از طول پیام آگاهی داشته باشد.
* اگر فضای خالی در نظر گرفته نشود، در صورت پر شدن بافر کامپیوتر مقصد، داچار خلل در عملیات می‌شویم
* در انتخاب مسیر برای هر بسته، هر نود بهترین مسیر را انتخاب می‌کند.
* بهترین مسیر، مسیری است که تأخیر کمتری داشته باشد.
* عمل بسته بستن و انتخاب بهترین مسیر توسط هر نود، ترافیک روی کل شبکه توزیع می‌شود.
* روش Packet عمل کنترل ترافیک را انجام می‌دهد.
* در شبکه‌های اینترنت از روش سوییچینگ PACKET استفاده می‌شود.

⊕ تأخیر : DELAY

* در ارتباطات شبکه‌ای دو نوع تأخیر داریم: (تأخیر کلی مجموع این دو است $DELAY = t_x + t_p$

۱ـ تأخیر ناشی از ارسال اطلاعات : TRANSMISSION DELAY

رابطه ④

$$t_x = \frac{L}{R}$$

L → طول اطلاعات برحسب بیت

R → سرعت انتقال خطوط

۲ـ تأخیر ناشی از انتشار : PROPAGATION DELAY

$$t_p = \frac{d}{v}$$

d → فاصله

v → سرعت انتشار علایم

تأخیر ناشی از ارسال علایم درون سیم‌ها :

در فاصله‌های زیاد مثل ارتباطات ماهواره‌ای مطرح می‌شود.

روشها / معیارها	سویچینگ بسته PACKET Switching	سویچینگ پیام MESSAGE Switching	سویچینگ مداری CIRCUIT Switching
تجهیزات سخت افزاری NETWORK	کامپیوترهای شبکه اینترنت	دستگاههای تلگراف شبکه تلگراف	شبکه تلفن
سخت افزار HARDWARE	کامپیوتر	تلگراف	تلفن
اطلاعات INFORMATION	اطلاعات باینری	سیگنال کدمورس	سیگنال صوتی
آدرس دهی ADDRESSING	سلسله مراتبی چهار بایتی	جغرافیایی	شماره گیری
مسیرو انتخاب مسیر ROUTING	بسته مستقلاً و وابسته به ترافیک شبکه در هر لحظه.	توسط اپراتور	شماره گیری خود مسیر را مشخص می کند.
تأخیر DELAY تأخیر ناشی از ارسال اطلاعات.	تأخیر کمی دارد.	تأخیر فاز برقراری نداریم تأخیر فاز اصلی بسته به اندازه اطلاعات است.	تأخیر فاز برقراری ارتباط داریم. پس از برقراری ارتباط ارتباط بلادرنگ است.

« جدول شماره ① »

⊕ « مقایسه روشهای سویچینگ »

« COMPARING THREE SWITCHING METHODS »

⊛ انواع سرویسهای شبکه‌ای: (از منظر ارتباط Connection)

① CONNECTION ORIENTED

* وابسته به اتصال،
* اتصال باید برقرار شود و سپس سرویس انجام شود
* مانند سرویسهای شبکه های تلفنی.
* فاز برقراری ارتباط دارد.

② CONNECTION_LESS

* ارتباط مستقیم نداریم،
* مانند شبکه‌های تلگراف.
* فاز برقراری ارتباط ندارد.

✚ ادامه روش "PACKET SWITCHING"

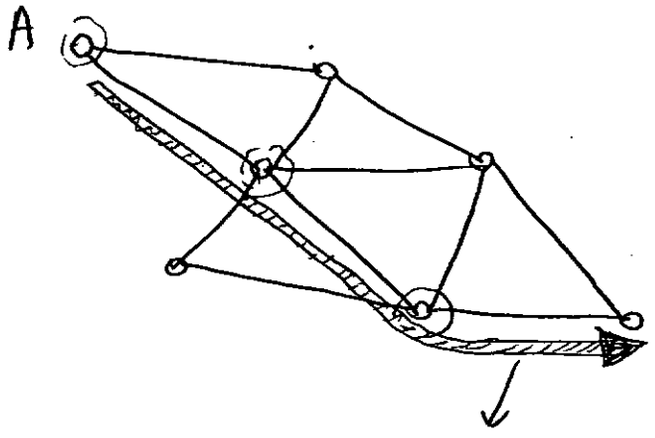
← روش PACKET می‌تواند هم وابسته به اتصال و هم غیروابسته به اتصال باشد.

1- در روش Connection Oriented یا VIRTUAL CIRCUIT

* فرستنده ابتدا تقاضایی به گیرنده می‌فرستد،
* گیرنده درصورت قبول از همان مسیر بسته قبول را می‌فرستد،
* و ارتباط مستقیم (مجازی) در همان مسیر برقرار می‌شود و بقیه اطلاعات از آن مسیر منتقل می‌شود.
* فایده این روش آن است که سرعت انتقال بالا می‌رود و پیام زودتر می‌رسد.
* نودهای میان باید حافظه‌ای داشته باشند که مسیر مجازی را حفظ کنند.
* در این روش فاز برقراری ارتباط داریم ولی پس از آن مانند روش سوییچینگ پیام در یک مسیر عمل می‌کند

روش VIRTUAL CIRCUIT

« مسیر مجازی » یا « مدار مجازی »



شکل (24)

* تمام اطلاعات بین مبدا و مقصد از طریق مسیر مجازی مبادله می‌شوند.

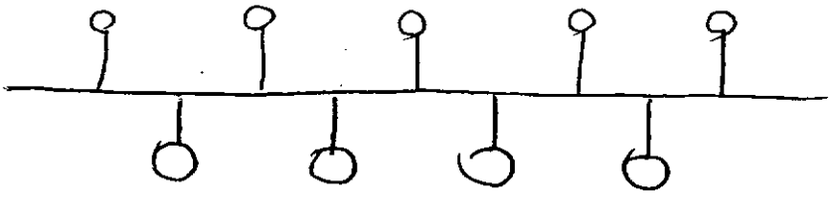
* نودهای مسیر مجازی باید حافظهٔ تشخیص مسیر را تا هنگام معتبر بودن مسیر مجازی دارا باشند.

۲- روش CONNECTION-LESS : DATAGRAM

* روشی که قبلاً بعنوان روش Packet گفتیم روش Datagram است.
* در این جا فاز برقراری ارتباط نداریم.
* ارتباط مستقیم نداریم و هر بسته از مسیری مستقل رد می‌شود.

⊕ ⊛ شبکهٔ LAN محلی کوچک با توپولوژی Bus :

مثال :



شکل (25)

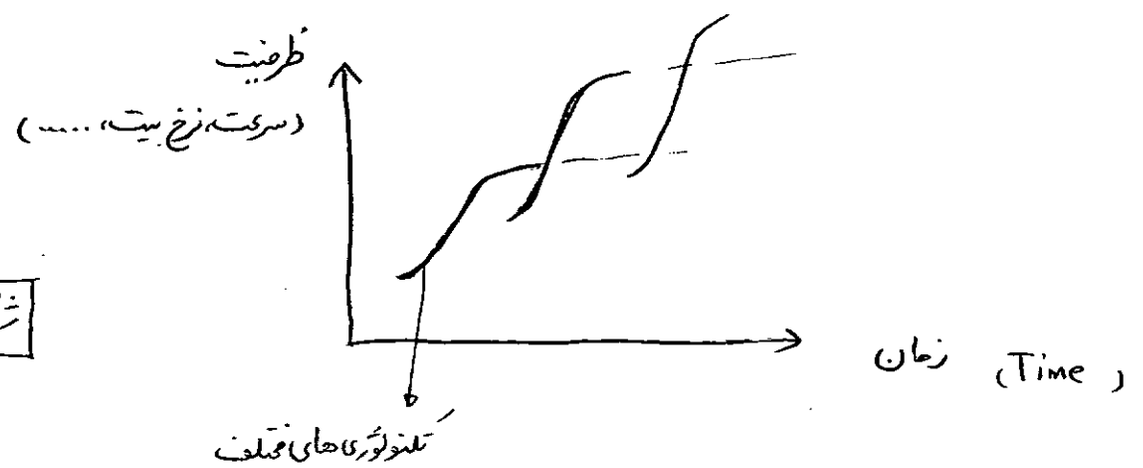
* تکنولوژی انتقال : در این جا از تکنولوژی Broadcast استفاده می‌کنیم.
* در این جا سوییچینگ نداریم.
* در این جا مسألهٔ در اختیار گرفتن خط را داریم.

«عوامل مهم در تحول شبکه‌های کامپیوتری»

① - تکنولوژی : TECHNOLOGY:

★ زمان پردازش‌ها کند بوده در شبکه‌ها تأخیر زیادی بوجود می‌آورد.

★ امروزه قدرت پردازشی تاحدی بالارفته که تقریباً تأخیری ندارند.



شکل (26)

« نمودار : ⊕تأثیر تکنولوژی و تأثیر تحول تکنولوژی روی تحول شبکه‌های کامپیوتری »

② - مقررات و قوانین : REGULATION:

★ می‌تواند باعث رشد یا جلوگیری از رشد هر پدیده تکنولوژیک از جمله شبکه باشد

③ - بازار : MARKET :

★ مدتهاست که تکنولوژی تلفن تصویری موجود است ولی به خاطر بالا بودن هزینه آن، وارد بازارهای جهانی نشده است.

④ - استانداردها : STANDARDS:

★« تأثیر استانداردهای جدید روی پدیده‌های تکنولوژیک و از جمله شبکه‌های کامپیوتری »

∞ تمرینهای فصل اول کتاب :

9 - 11 - 15 - 16 - 17 - 18

⊕ تا این جا مقدمات کلی و سرویسها و فعالیتهای شبکه های کامپیوتری را مروری کلی کردیم.
از این به بعد وارد مباحث اصلی می شویم.

سرفصل دوم: معماری لایه ای ، OSI ، TCP/IP ، دولایه پایین هرکدام

⊕

معماری لایه ای

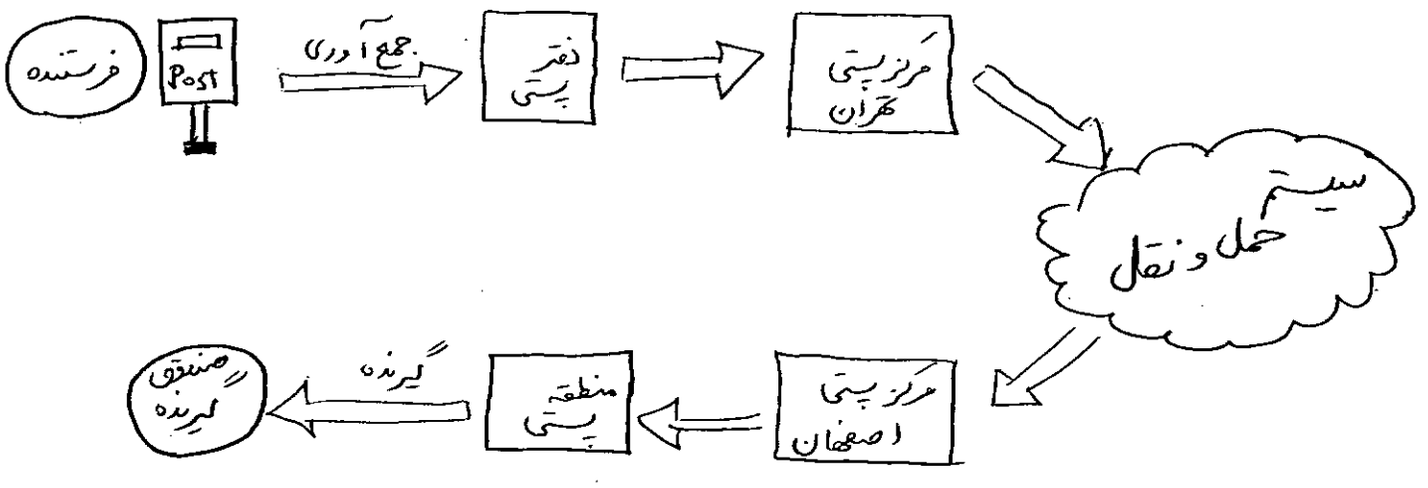
LAYERED ARCHITECTURE

★ برای دستیابی به آنچه که از مورد شبکه ها گفتیم، نیاز به معماری خاص، موسوم به معماری لایه ای داریم.

★ برای روشن تر شدن موضوع معماری لایه ای، مثالی از یک سیستم پستی می آوریم.

☒ مثال: سیستم پستی :

POSTAL SYSTEM:



شکل (27) « ⊕ معماری بخشی از سیستم پستی برای عملیات نامرسان تهران به اصفهان »

Mo

★ در سیستم پستی ، کارها و وظایف تقسیم شده اند.

★ مزیت اصلی این کار تسهیل جریان کار و امور پستی است .

★ مزیت دیگر این کار آن است که در مقابل ترافیک جدید و تکنولوژی های جدید ، قابلیت انعطاف بالاتری داریم.

★ مثلاً اگر سیستم حمل ونقل با ماشین انجام می شد، می توان از یک هواپیما استفاده کرد بدون اینکه خللی به بقیه بخشهای سیستم وارد شود در صورتی که اگر معماری ما لایه ای نبود این کار به سهولت انجام نمی شد.

★ در شبکه های کامپیوتری تمام موارد بالا معتبر است .

(★★★) دید هرکدام از عوامل شامل سیستم پستی نسبت به کل سیستم:

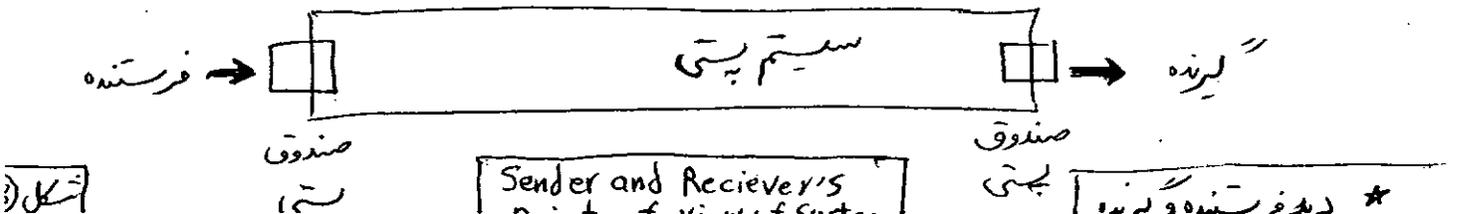
(1) فرستنده :
فقط صندوق پست را می بیند، به بقیه امور دیدی ندارد.

(2) دفتر پستی : نامه های صندوق ها را جمع می کند و به سیستم مرکز پست می دهد، نامه های رسیده از مرکز را به پستچیان می دهد تا به مقصد ببرند.

(3) مرکز پستی : نامه ها را از دفتر پستی می گیرد، (یا می دهد)
دسته بندی می کند،
آماده حمل ونقل می کند،
به سیستم حمل ونقل می دهد. (یا می گیرد.)

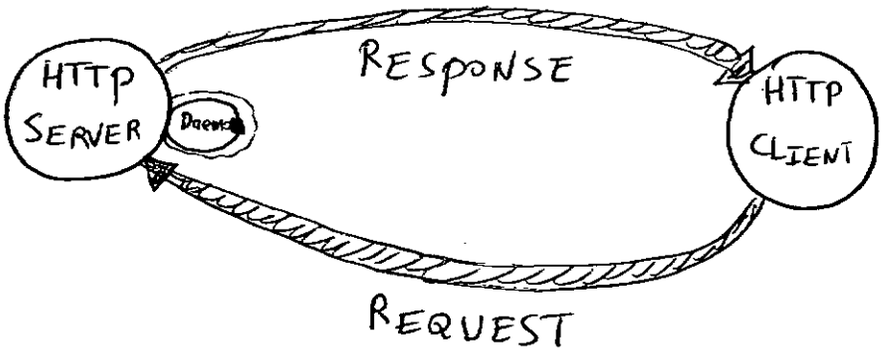
(4) سیستم حمل ونقل : نامه ها را از مرکز می گیرد و به مرکزی دیگری برد.

(5) گیرنده : نامه را می گیرد !! همین وبس ؟!



شکل (2)

مثال: یک ارتباط WEB:



شکل (29)

★ Client و Server هیچ کدام کاری به نحوهٔ رسیدن تقاضا یا پاسخ ندارند و فقط کل قضیه برایشان اهمیت دارد.

مراحل انجام کار: (مثالی از کتاب)

1- User Select Document: الف- کاربر لینک خاصی را انتخاب می‌کند.

2- Client Network Software Locates Server Address and Stablishes a Connection

ب- نرم‌افزار شبکهٔ Client محل و آدرس Server را مشخص می‌کند و ارتباطی با Server برقرار می‌کند

★ در این جا ارتباط Client و Server از نوع Connection Oriented است.

3- Client Send Message Requesting Document.

ج- Client درخواست فایل را می‌فرستد.

GET / inform / index.html HTTP/∅

4- Server Daemon listening to a Tcp Port inputs Message.

د- مشتری از سرور که دایماً روی خط گوش می‌دهد پیغام را می‌گیرد

5- Daemon Sends a Result Code and information

هـ- سرور نتیجه و اطلاعات مورد نیاز Client را به او می‌دهد

6- Daemon Reads the file and sends it through Tcp Port 80.

و- سرور فایل را خوانده و از طریق پورت ۸۰ خود برای Client می فرستد.

7- Daemon Disconnects the Connection.

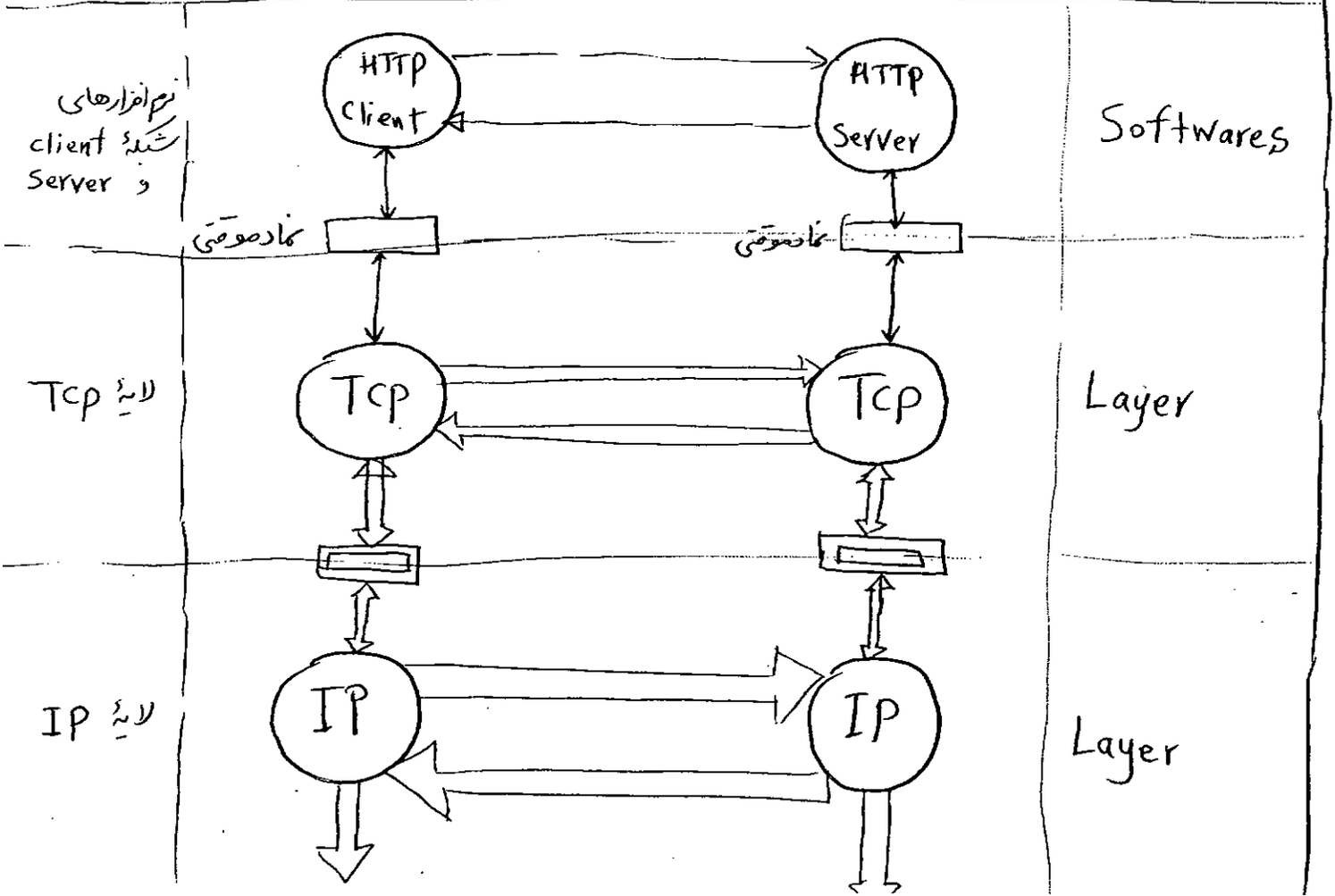
ز- سرور ارتباط را می بندد و خاتمه می دهد.

8- TEXT is Displayed on the client's Software which Interprets HTML Code.

ح- فایل روی نرم‌افزاری از Client که کد HTML را تفسیر می‌کند نشان داده می شود.

* فی الواقع ارتباط این دو از طریق لایه هایی انجام می شود که آن دو از آن ها بی خبرند.

شکل (30) « نمودار ⊕ ارتباط Server و Client در واقع از طریق لایه های زیرین »



* همانطور که در جلسه‌ی سوم اشاره شد تأخیر انواعی دارد.

۱- تأخیر ناشی از ارسال اطلاعات

۲- تأخیر ناشی از انتشار سیگنال در سیم‌ها.

$t_x = \frac{L}{R}$

$t_p = (\frac{v}{d})^{-1}$

* ۳- تأخیر ناشی از انتظار و معطلی اطلاعات درون نودهای درون مسیر انتقال. t_{Nodes}

* در این میان تأخیر ناشی از انتشار از همه کمتر و در حدود (سرعت نور) تأخیرهای الکتریکی بسیار کم است.
* تأخیر نودها به قدرت پردازشی و توان محیاتی دیگر نودهای درون مسیر بستگی دارد.
* تأخیر ناشی از ارسال اطلاعات به اندازه فایل و روش سوییچینگ بستگی دارد.

رابطه ⑥

$$T = t_x + t_p + t_{Delay\text{-}of\text{-}Nodes}$$

* در معیارهای ارزیابی در درس شبکه تنها t_x و t_p اهمیت دارند.

معیار $T = t_x + t_p$

* حتی می‌توان گفت t_x پراهمیت‌ترین فاکتور است چون t_p حتی در صورت طولانی بودن مسیر هم باز از t_x معمولاً بسیار کمتر است.

---

* حسن‌های استفاده از معماری لایه‌ای زیاد است.

یکی آنکه قابلیت انعطاف‌پذیری بالایی دارد

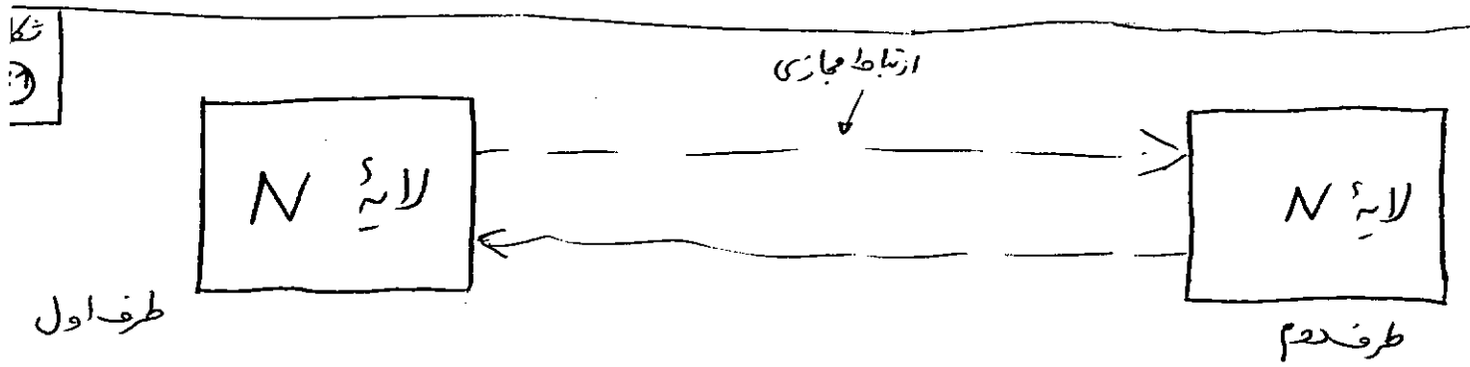
و قابلیت تغییر تکنولوژی بیشتری شود.

و وظایف تقسیم می‌شوند.

* در این جلسه به بررسی بیشتر لایه‌های شبکه می‌پردازیم.

Layered Architecture

⊕ «ارتباط مجازی بین لایهٔ N ـام دوطرف ارتباط»



★ ارتباطی مجازی بین لایهٔ Nام دوطرف ارتباط برقرار است.

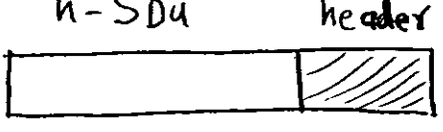
⊕ PDU : PROTOCOL DATA UNIT

تعریف: واحدهایا بسته‌هایی که از طریق آنها دو لایهٔ N از دوطرف ارتباط با هم ارتباط برقرار می‌کنند. (واحدی که فرستنده می‌خواهد به گیرنده ارسال کند)

★ همهٔ بسته‌ها یک header دارند که هماهنگی بین آنها را برقرار می‌کند.

★ بخش دیگر پیام Service Data Unit یا SDU است.

★ با توجه به اینکه این دو لایه PDU های مخصوص خودشان را دارند به PDU های لایهٔ nام، n-PDU می‌گوییم.



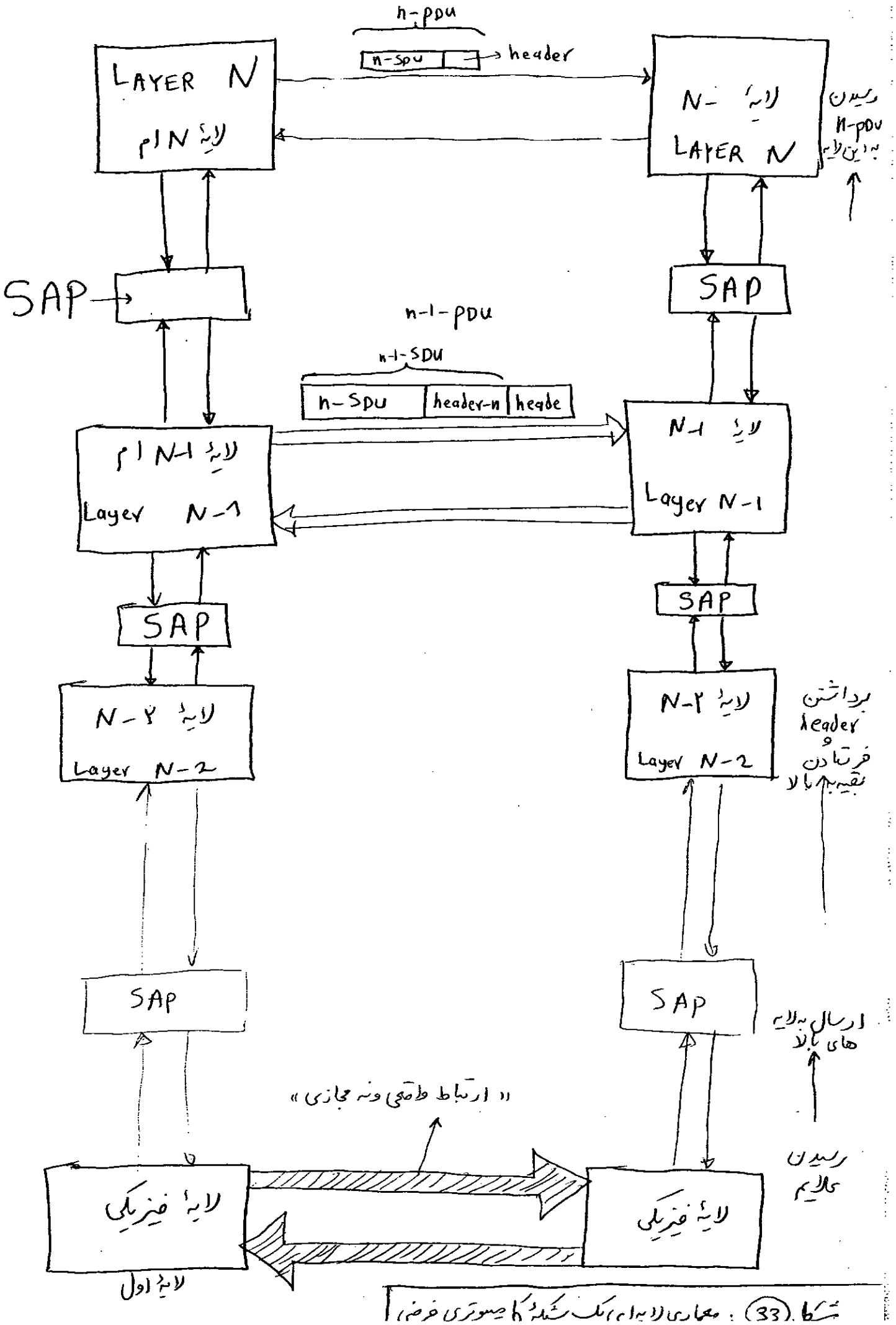
شکل (32) ★ یک n-PDU

★ مبادلهٔ n-PDU بین دوطرف فی الواقع مجازی است و از طریق لایه‌های زیرین میسر می‌گردد.

★ لایهٔ زیرین (N-1) از کل n-PDU بعنوان n-1-SDU استفاده می‌کند.

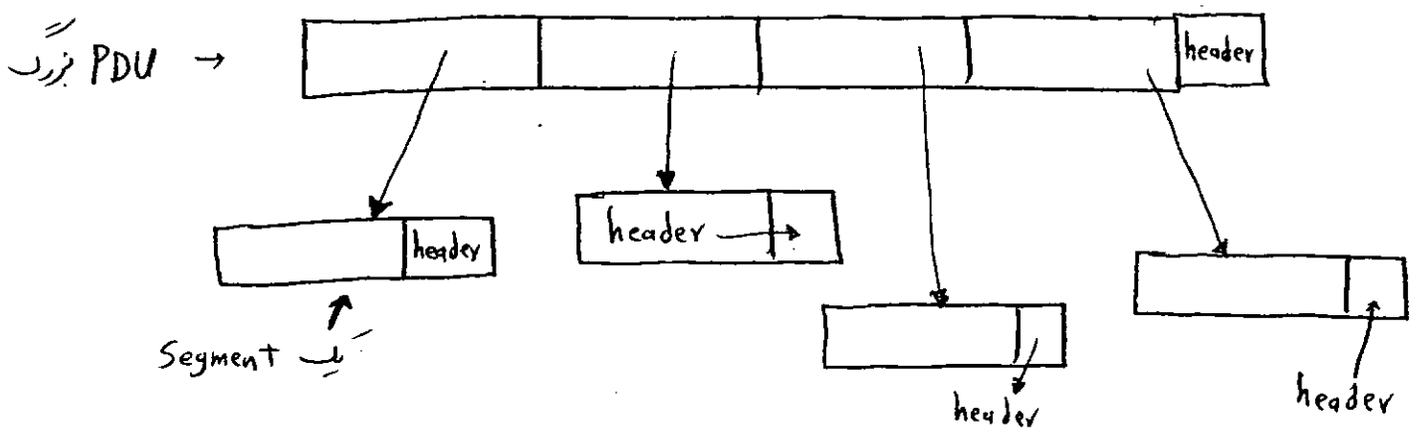
⊕ تعریف SAP : «Service Access Point»

★ رابط بین دو لایه را SAP می‌گوییم.



شکل (33): معماری لایه‌ای یک شبکه‌ی کامپیوتری فرضی

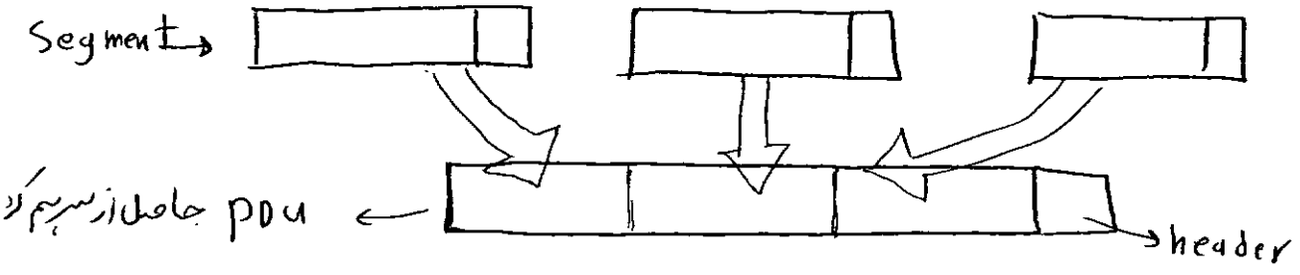
* در سیستم پستی مبادله‌ها از طریق سیستم پستی انجام می‌گیرد.
* در سیستم‌های کامپیوتری، یک لایه‌ فیزیکی در زیرهمه لایه‌ها قرار دارد که مبادله‌ علایم الکتریکی را به عهده دارد.
* ممکن است اطلاعاتی که از لایه‌های بالاتر می‌آید بسیار بزرگ باشد، بنابراین در بحث معماری لایه‌ای، بحث بخش کردن اطلاعات بزرگ پیش می‌آید که به آن Segmentation می‌گوییم.



شکل (34) "SEGMENTATION"

* بسته‌های Segment شده در header خود باید دارای اطلاعاتی از قبیل شماره‌ توالی باشند تا درمقصد بتوان آنها را سرهم کرد.

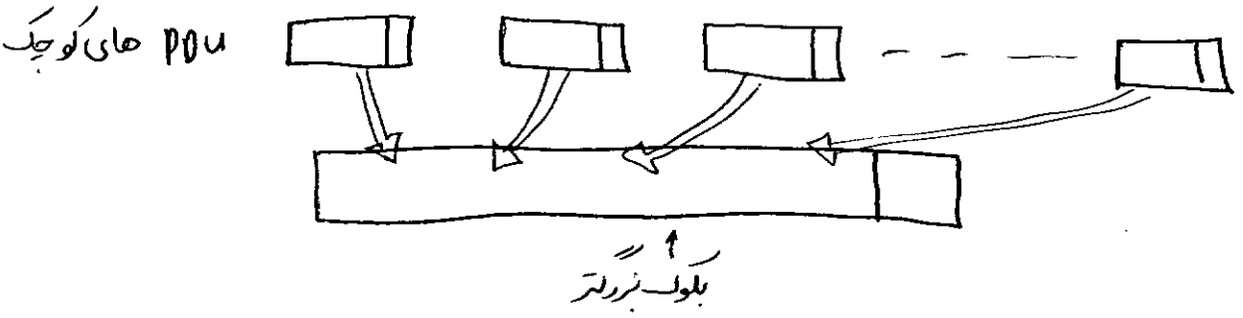
⊕ REASSEMBLY : عمل سرهم کردن بسته‌های Segment شده.



شکل (35) "REASSEMBLY".

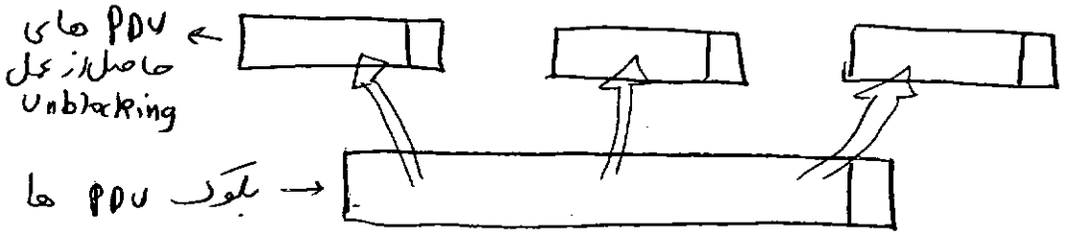
* حال اگر بسته‌هایی که دو لایه‌ دو طرف ارتباط می‌خواهند با هم مبادله کنند بسیار کوچک باشند، به جهت اینکه هر بسته یک header دارد ارتباط بهینه نیست.

⊕ BLOCKING : سرهم کردن PDU های کوچک درون یک FRAME.



شکل (36) BLOCKING.

(*) عکس عمل BLOCKING را UNBLOCKING گویند.



شکل (37) UNBLOCKING

* وقتی پیام در لایه‌ی فیزیکی به مقصد رسید، هر لایه header مربوطه را بررسی کرده و بقیه پیام (SDU) را به لایه‌ی بالاتر انتقال می‌دهد.

* بدین ترتیب آنچه که به لایه‌ی N می‌رسد همان PDU ای است که لایه‌ی N طرف دیگر ارسال کرده. بدین خاطر است که ارتباط مجازی بین دو لایه‌ی N برقرار می‌شود.

* حال به بررسی مدل OSI که یک معماری لایه‌ای است می‌پردازیم.

* OSI یک مدل ۷ لایه‌ای است.

* OSI برای آموزش معماری لایه‌ای است و در عمل کاربرد چندانی ندارد.

* مدل OSI توسط موسسه Iso (International Standard Org.) برای شبکه‌های ارتباطات باز ارایه شده است.

مدل OSI:

★ ۷ لایه دارد که از بالا به پایین به صورت کلی آنها را بررسی می کنیم.

⊕ لایه هفتم: «لایه کاربرد» "Application Layer"

★ بالاترین لایه که برنامه های کاربردی در آن مطرح اند.

مثل:

SMTp
HTTP
FTP
Telnet

⊕ لایه ششم: لایه نمایش PRESENTATION LAYER

★ لایه زیرین لایه کاربرد.

★ اطلاعاتی که در کامپیوتر مبداء به فرمت خاص آن کامپیوتر است در این لایه به فرمت قابل فهم برای شبکه مقصد تبدیل می شود.

⊕ لایه پنجم: «لایه جلسه» "SESSION LAYER"

★ وظیفه هماهنگی ارتباط را برعهده دارد.

★ هماهنگی شروع ارتباط، نحوه ارتباط و پایان ارتباط.

⊕ لایه چهارم: «لایه حمل» "TRANSPORT LAYER"

★ وظیفه اش حمل پیامها است.

★ روشهای سوییچینگ، روشهای وابسته به Connection و یا غیروابسته به آن در این لایه پیاده می شود.

★ عملیات Segmentation و Reassembly، Blocking و UnBlocking همه در این لایه انجام می شوند.

⊕ لایه‌ی سوم: «لایه شبکه» "NETWORK LAYER"

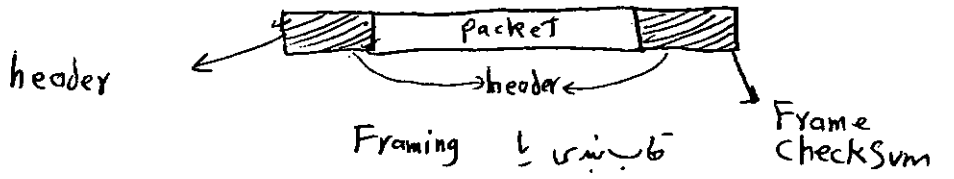
★ بحث های مسیردهی (Routing) و کنترل ترافیک شبکه در این لایه انجام می شود.

⊕ لایه‌ی دوم: «لایه‌ی پیوند داده ها» "DATA Link Layer"

★ وظیفه آن Framing است.

⊕ Framing: اضافه کردن دو header به سر و ته پیام، برای آنکه گیرنده محدوده‌ی داده را بداند.

مثال (38)



★ وظیفه‌ی دیگر آن کنترل خطا است.

★ کنترل جریان از دیگر وظایف این لایه است.

★ مثلاً در ارتباط دو طرف، سرعت دو طرف را با هم هماهنگ می کند.

⊕ لایه‌ی اول: «لایه‌ی فیزیکی» PHYSICAL LAYER

★ تبدیل اطلاعات بایتی به علایم قابل عبور در کانال ارتباطی.

★ در واقع فرستنده و گیرنده در دو طرف، لایه‌ی فیزیکی دو طرف هستند.

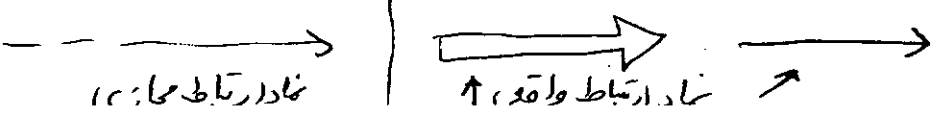
★ در تمامی لایه ها، ارتباط از انتها به انتها به صورت مجازی داریم به جز، سه لایه پایین.

★ در سه لایه‌ی پایین ارتباط انتها به انتهای مجازی نداریم، بلکه ارتباط مجازی بین دو نود متوالی است.

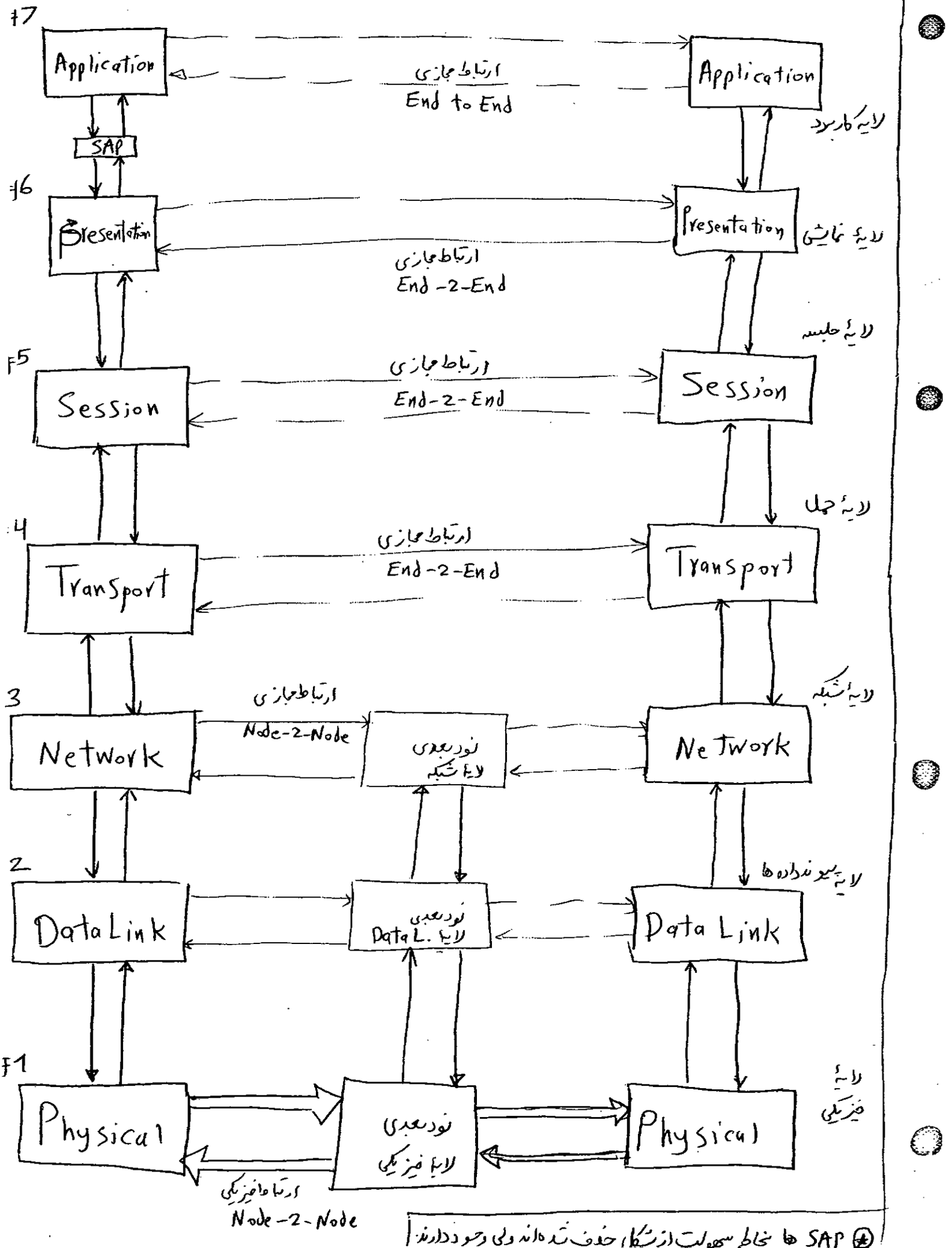
★ در لایه‌ی فیزیکی ارتباط مجازی نیست.

★ ارتباط در لایه‌ی فیزیکی با نود بعدی است.

⊕ شکل (39)



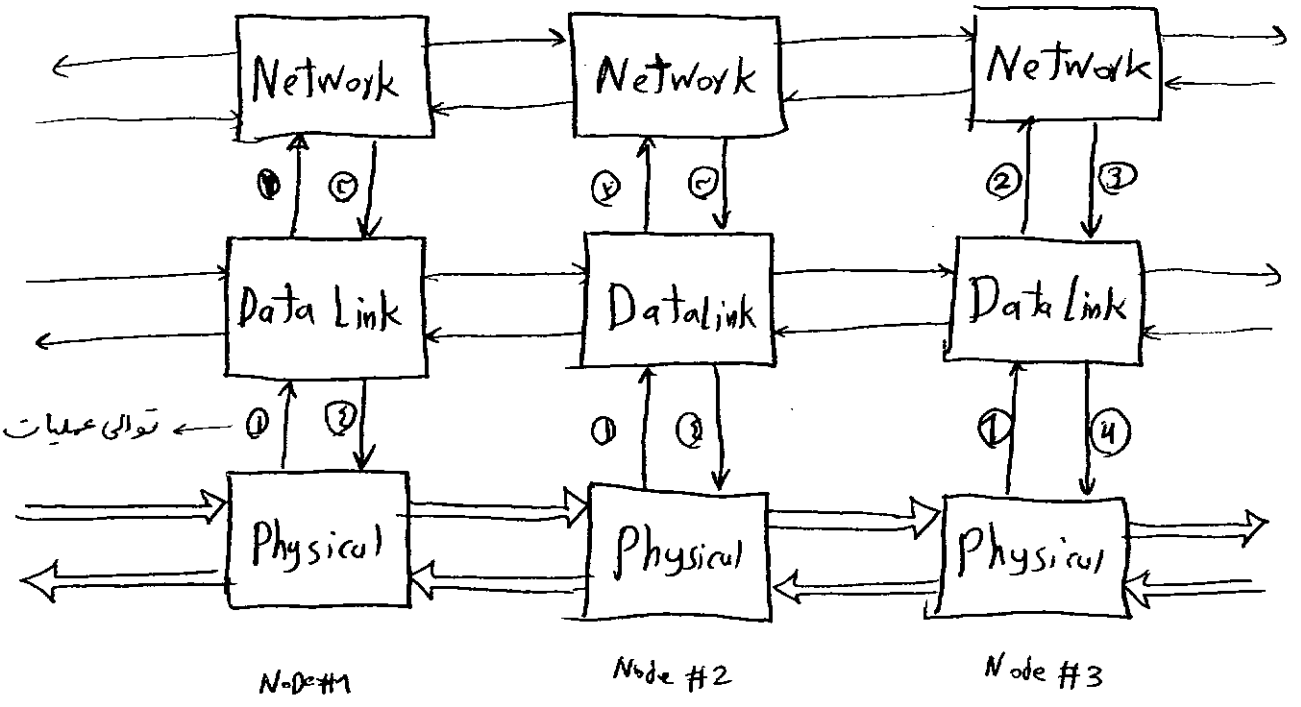
⊕ شکل (40) مدل لایه‌ای OSI



⊕ SAP ها نخاط سهولت از شکل حذف شده‌اند ولی وجود دارند!

* نودهای میان راه در مسیر بین مبدأ و مقصد، سه لایه پایینی خود را در اختیار این ارتباط قرار می‌دهند.

* بدین ترتیب که وقتی اطلاعات از لایه‌ی فیزیکی به آنها رسید، آن را به لایه‌ی Data Link خود می‌دهند و سپس از آن به لایه‌ی Network و سپس دوباره به لایه‌ی Data Link و لایه‌ی فیزیکی برای انتقال به نود بعدی، (شکل ⑪ ↓)



⊕ شکل ⑪ ارتباط سه لایه‌ی پایین مدل OSI برای نودهای میر و توالی عملیات انجام شده.

⊕ ارتباط‌ها در معماری OSI دو نوعند. (ارتباط لایه‌ی N ام دو طرف ارتباط)

۱- END to END : برای چهار لایه‌ی بالا. (ارتباط مجازی)

۲- NODE to NODE : برای سه لایه‌ی پایین. (برای لایه فیزیکی ارتباط غیرمجازی)

⊕ ارتباط نودها با هم [سه لایه‌ی پایین] دو نوع است.

۱- Connection-Oriented

۲- Connection-Less

« غزال اگر به کمند او فتد عجب نبود » « عجب فتادن مرد است در کمند غزال »

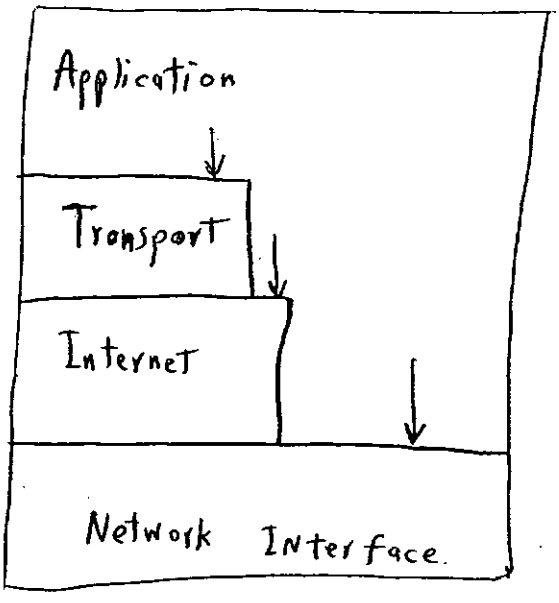
مدل TCP/IP:

* در حال از این مدل بیشتر از OSI استفاده می‌شود.

⊕ لایه‌های مدل TCP/IP (و معادل آن‌ها در مدل OSI)

OSI	معادل با	TCP/IP
Application Presentation Session	≡	Application
Transport	≡	Transport
Network	≡	Internet
Data Link physical Layer	≡	NI Network Interface
Net* *[در OSI نداریم] شبکه	≡	physical Layer

* تفاوت عمدهٔ TCP/IP و OSI در این است که لایه‌های بالاتری دارند مستقیماً با لایه‌های پایینی ارتباط برقرار کنند. (بدون دخالت لایه‌های میانی)

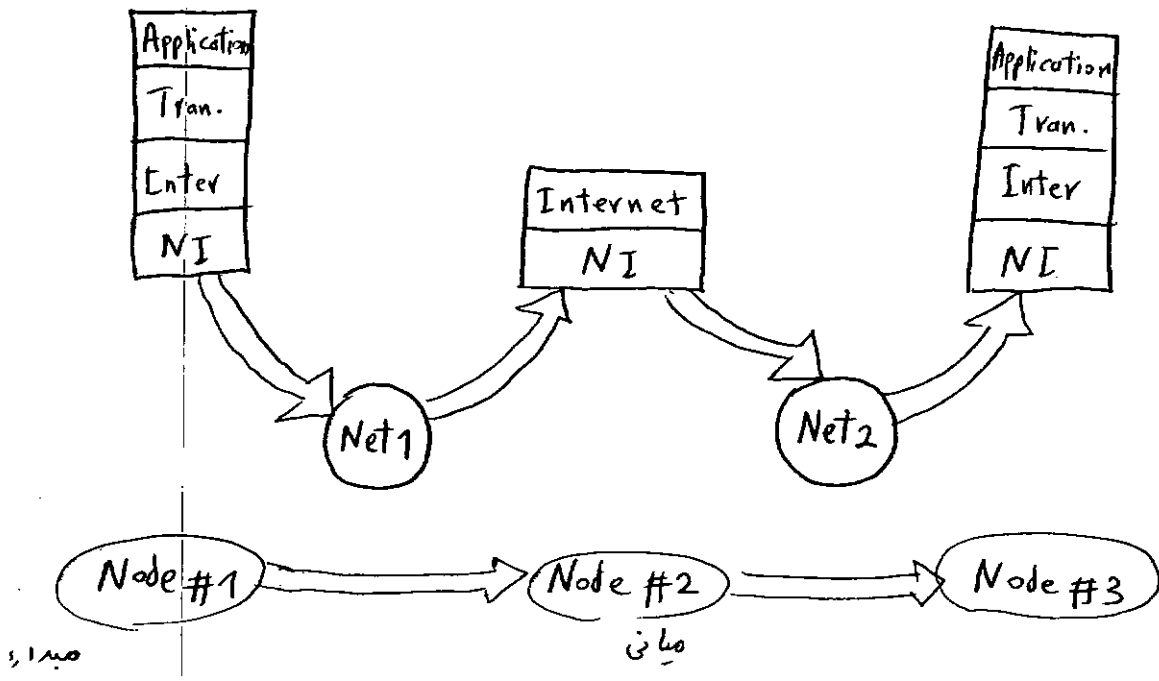


TCP/IP Architecture

شکل (42) تفاوت TCP/IP و OSI

★ مدل TCP/IP برای اتصال شبکه های گوناگون به یکدیگر مناسب است.

★ به علت دارا بودن لایه ی physical که درواقع همان Net یا شبکه محلی است.

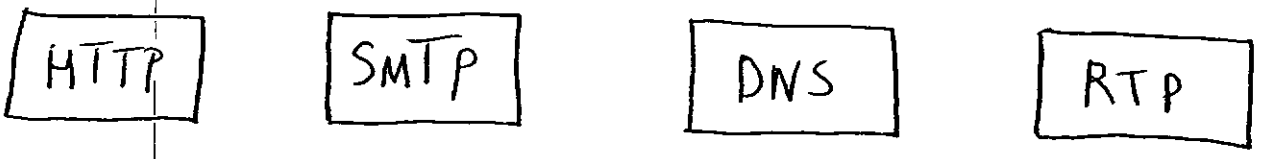


شکل (۴۲)
ارتباط از طریق شبکه های میانی راه

★ هرکدام از (Net1) یا (Net2) خودمی توانند یک شبکه محلی باشند و نه لزوماً یک خط.

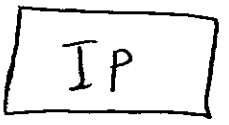
★ پروتکل های HTTP ، FTP ، SMTP و ... درواقع متعلق به معماری TCP/IP هستند.

★ برخی پروتکل های لایه Application در مدل TCP/IP



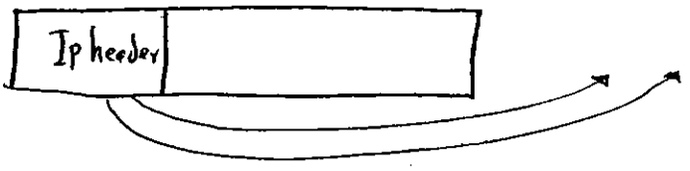
★ برخی پروتکل های لایه Transport در مدل TCP/IP

★ پروتکل لایه INTERNET



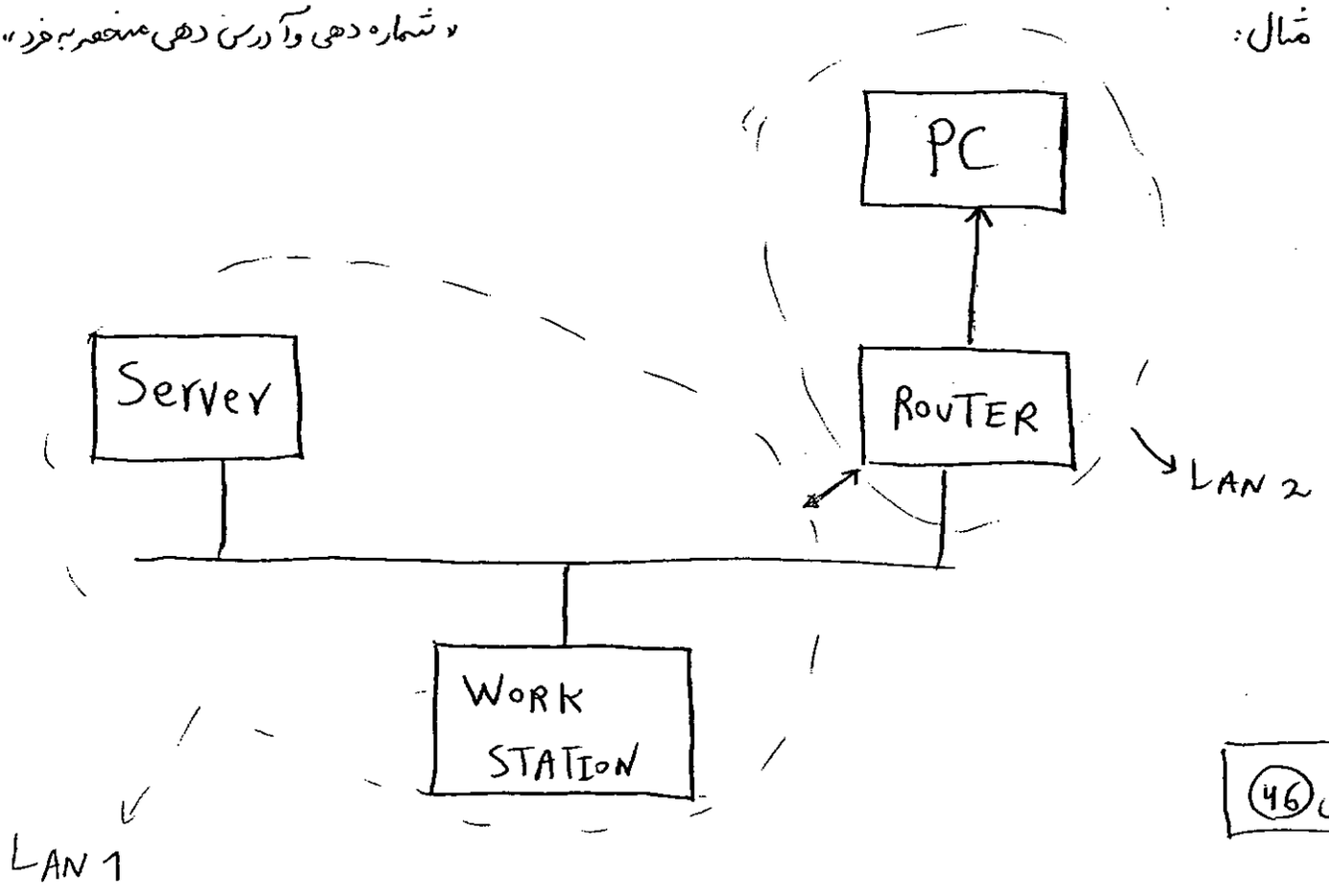
(★) بعلت استفاده فراوان از دو پروتکل TCP و IP به این مدل لایه ای، TCP/IP می گویند

⊕ شکل (45)



آدرس Ip مبدا
آدرس Ip مقصد

⊕ مثال: « شماره دهی و آدرس دهی منحصر به فرد »



شکل (46)

★ به هر کامپیوتر و هر شبکه شماره‌ای اختصاص داده می‌شود. (شبکه شکل (46) را در نظر بگیرید.)

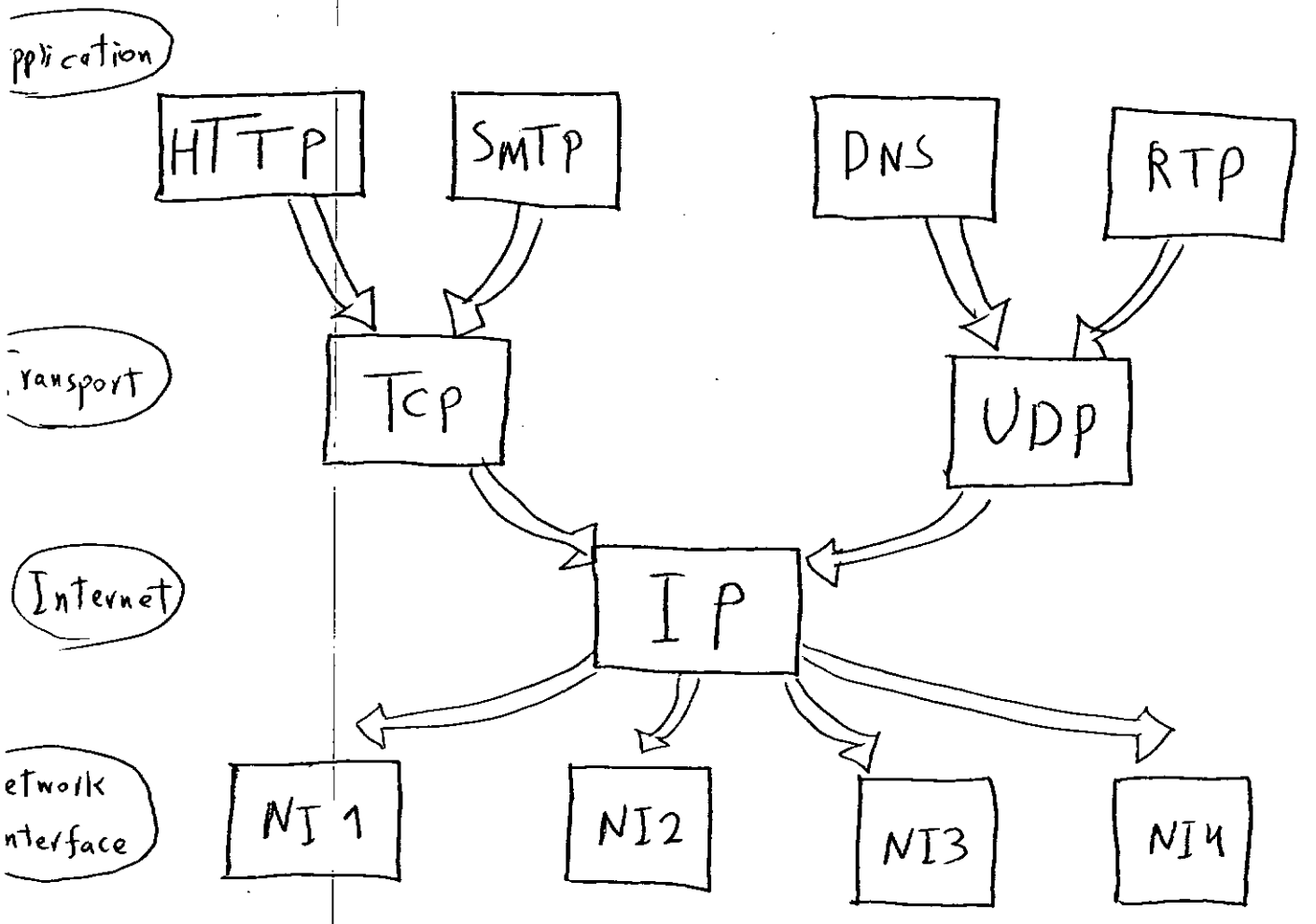
LAN 1: Server, WS, Router ⇒ درون شبکه 1
LAN 2: Router, Pc ⇒ درون شبکه 2

Server : (1,1)
WS : (1,2)
Router : (1,3), (2,1)
Pc : (2,2)

⇒ « آدرس های منحصر به فرد

★ آدرس Ip آدرسی چهار بایتی و سلسله مراتبی به مانند همین مثال است.

* پروتکل های مختلف در لایه های پروتکل Tcp/Ip

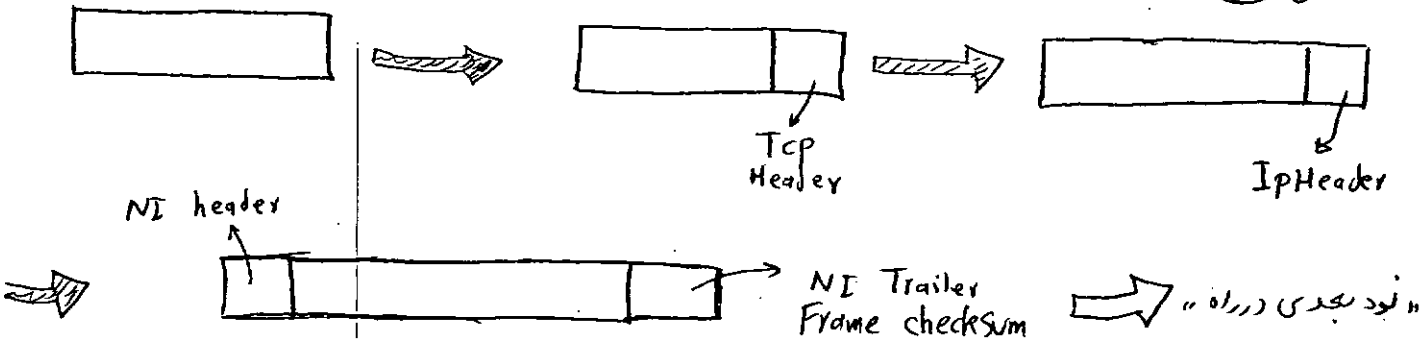


شکل (43) ارتباط پروتکل های لایه های Tcp/Ip ، (نقش کلیدی IP در این مدل)

* پروتکل Tcp ، Connection-Oriented است.
* پروتکل UDP ، Connection-Less است.
* پروتکل Ip نقش بسیار مهمی در هیئلی کار های لایه های بالاتر خود را دارد.

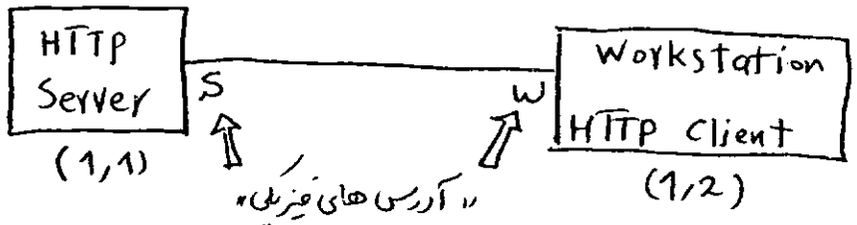
* مثال: ارتباط دو طرفه با پروتکل HTTP : (با توجه به شکل صفحه ۳۸)

شکل (44) «اضافه شدن header به پیام توسط هر پروتکل در هر لایه»



⊕ ادامهٔ مثال (شماره گذاری منحصر به فرد):

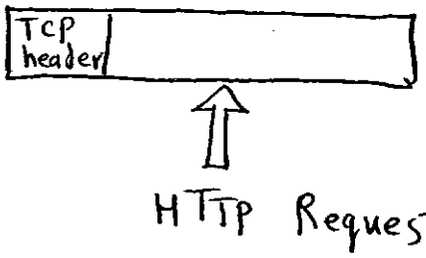
⊕ فرض کنیم Server(1,1) ، یک سرور HTTP و Workstation(1,2) ، Client باشد.



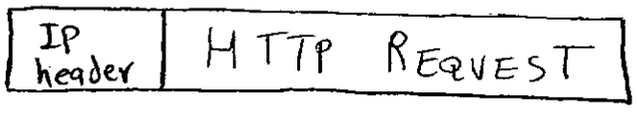
شکل (47) آدرس فیزیکی (کارت شبکه)

PDU بصورت یک Request

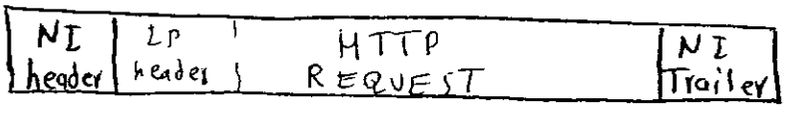
« مرحلهٔ اول » : پروتکل Tcp



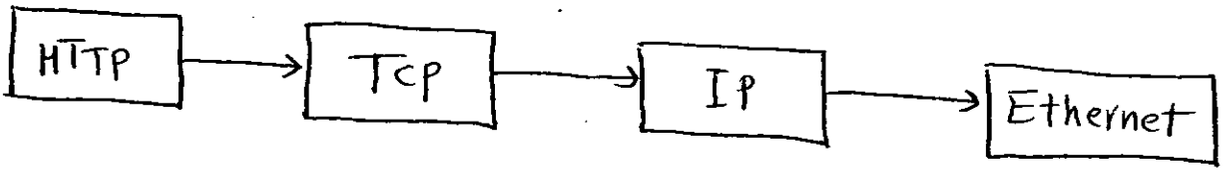
مرحلهٔ دوم : پروتکل IP



مرحلهٔ سوم : Network Interface



⊕ مرحلهٔ بعدی قرارگرفتن اطلاعات حاصل، روی خطوط فیزیکی است.

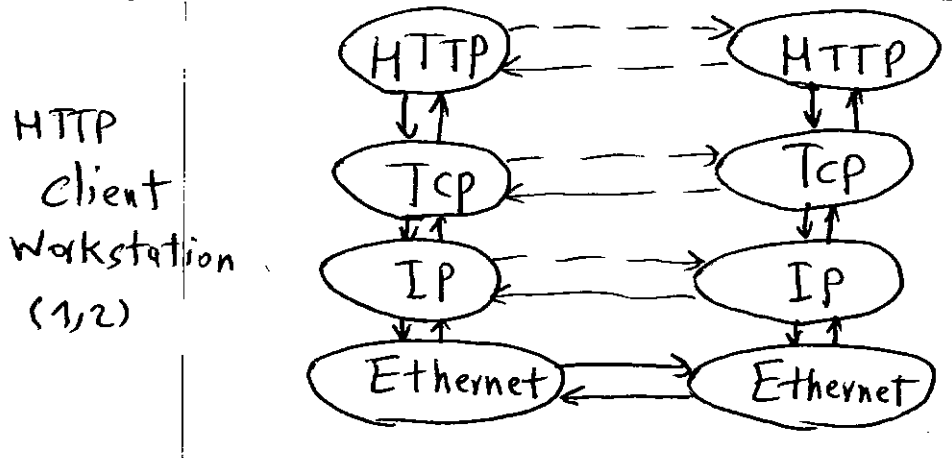


⊕ پروتکل‌های درگیر در این ارتباط « HTTP » (در طرف Client)

شکل (48)

* کارِ رابطِ Network Interface را در محیط های واقعی، کارت شبکه انجام می دهد.
* هر نودی یک آدرس سخت افزاری دارد.
* فرض کنیم a آدرس سخت افزاری Server و b آدرس سخت افزاری Client است.
* در توپولوژی Bus ، همه روی خط گوش می کنند و اطلاعات مربوط به خودشان را برمی دارند.

* کلاینم توسط لایه‌ فیزیکی Client روی BUS قرار می‌گیرد.
* سرور کلاینم را پردازش می‌کند و پروتکل های لایه های مختلف آن، عکس عملیات قبلی را انجام می‌دهد.
* هر کارت شبکه آدرس منحصر به فردی دارد که توسط کارخانه‌ سازنده مشخص می‌شود.



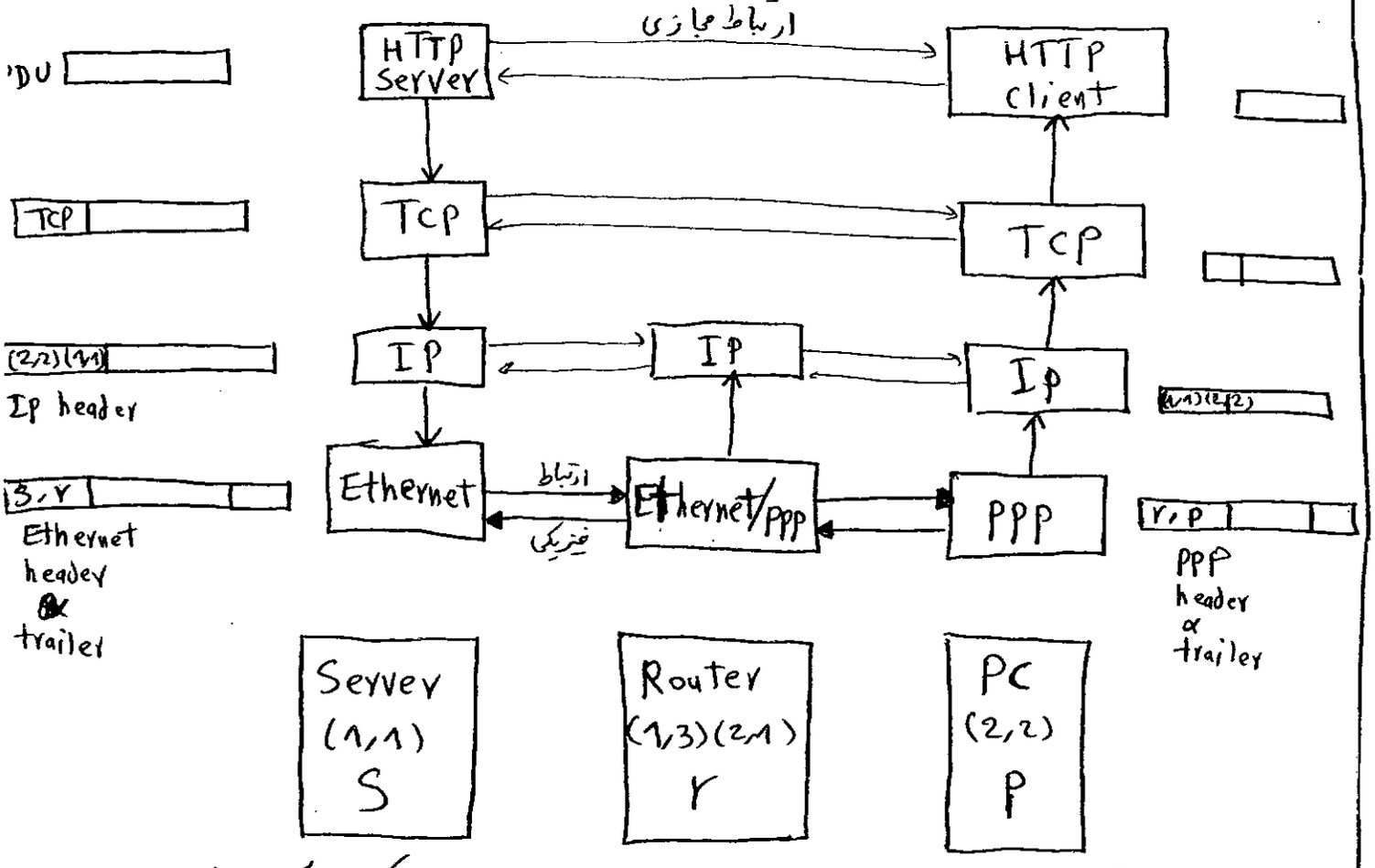
شکل (49) ⊕ ارتباط لایه های مختلف Server و Workstation

مثال: (*) در همان شبکه‌ فرضی مثال قبل فرض کنیم Server به آدرس (1,1) می‌خواهد با pc به آدرس (2,2) ارتباط برقرار کند.

* ابتدا Server می‌خواهد پیامی به pc بفرستد.
* در لایه‌ IP، IP header مربوطه روی پیام نوشته می‌شود که باید شامل آدرس مبداء و مقصد باشد.

شکل (50) Ipheader

* لایه NI اگر آدرس کارت شبکه‌ مقصد را داشت آنرا در header مربوطه می‌نویسد و اگر آن را نداشت آنرا بصورت پیش فرض قرار می‌دهد. (آدرس Router در اینجا)
* فرض کنیم آدرس فیزیکی Router ۲ است.
* پس نود بعدی Router است.
* پیام به Router می‌رسد. لایه NI پیام را گرفته به لایه IP از Router می‌دهد.
* لایه IP تشخیص می‌دهد که پیام برای Router نیست و در نتیجه آن را به لایه NI (خود که در اینجا PPP است) می‌دهد که آن لایه پیام را روی خط می‌گذارد. (آدرس کارت شبکه‌ pc را هم در header می‌گذارد.)



شکل (51) ⊕ ارتباط Server و PC از طریق Router (بدون اینکه سرور آدرس ساخت افزاری PC را بداند)

⊕ نمایش ارتباط به صورت مختصر به فرد توسط سه تایی:

(#port , protocol , Ip address)

⊕ آدرس سوکت مبدا: Source Socket Address

سه تایی که در آن اطلاعات مبدا قرار دارد.

⊕ آدرس سوکت مقصد: Destination Socket Address

سه تایی که در آن اطلاعات مقصد قرار دارد.

(#port , Protocol , Ip Address)

⊕ با یک پنج تایی می توان کل ارتباط را به صورت مختصر به فرد شناسایی کرد.

* برای مثال قبل:

آدرس سوکت مبدا: (#80, Tcp, (1,1))

آدرس سوکت مقصد: (#موقتی, Tcp, (2,2))

پنج تایی کل ارتباط: (#80, Tcp, (1,1), #موقتی, (2,2))

---

* فرض کنیم اتصال دیگری بخواهد بین Pc و Server برقرار شود.
* بدین منظور شماره موقتی دیگری به ارتباط داده می شود.

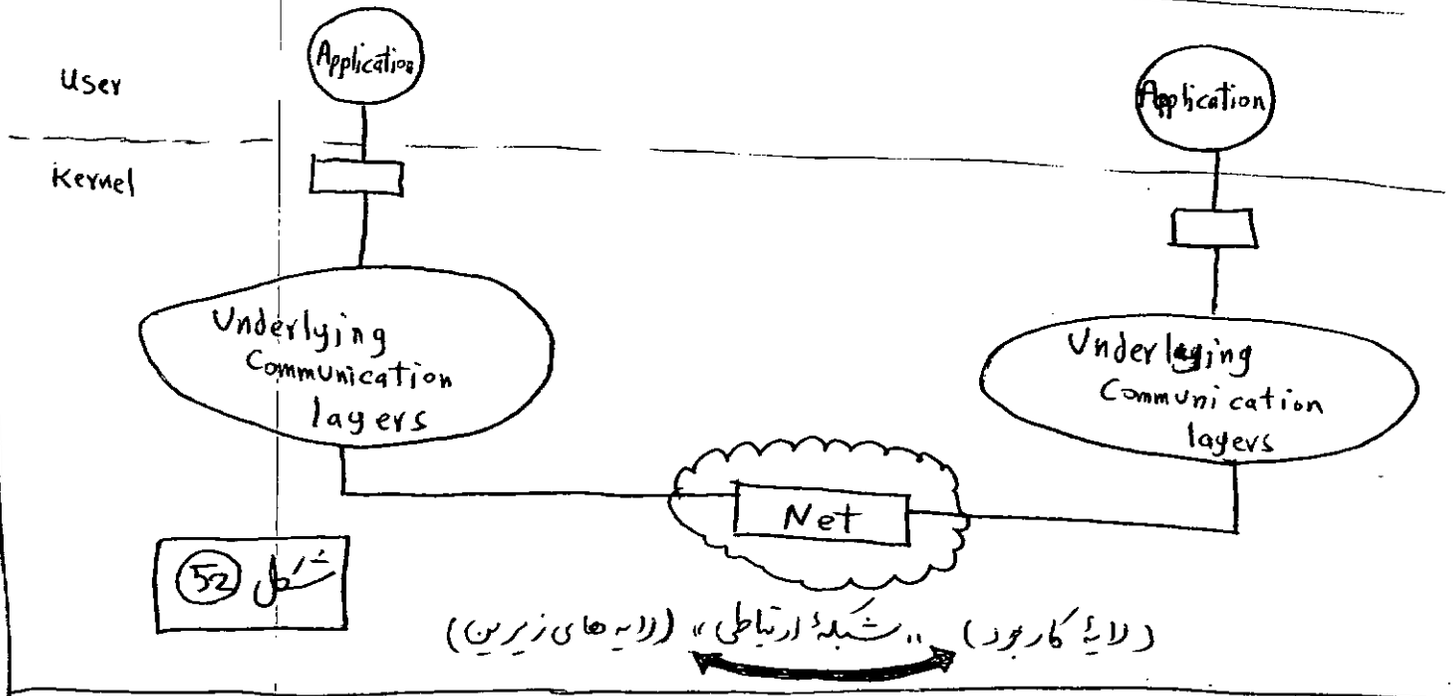
(#80, Tcp, (1,1), # موقتی ۲, (2,2))

⊕ ارتباط شماره ۲

* چون port های موقتی دو اتصال با هم متفاوتند، این دو ارتباط کاملاً از هم مجزا هستند و هرکدام به صورت منحصر به فردی شناسایی می شوند.

⊕ لایه های ارتباطی زیرین UNDERLYING CONNECTION LAYERS

* لایه های زیر لایه Application را بررسی می کنیم.



شکل (۵۲)

* برای هر ارتباط یک Socket ایجاد می شود. برنامه نویس تنها به همین جنبه از ارتباط کار دارد و بقیه کارها را لایه های زیرین انجام می دهند.

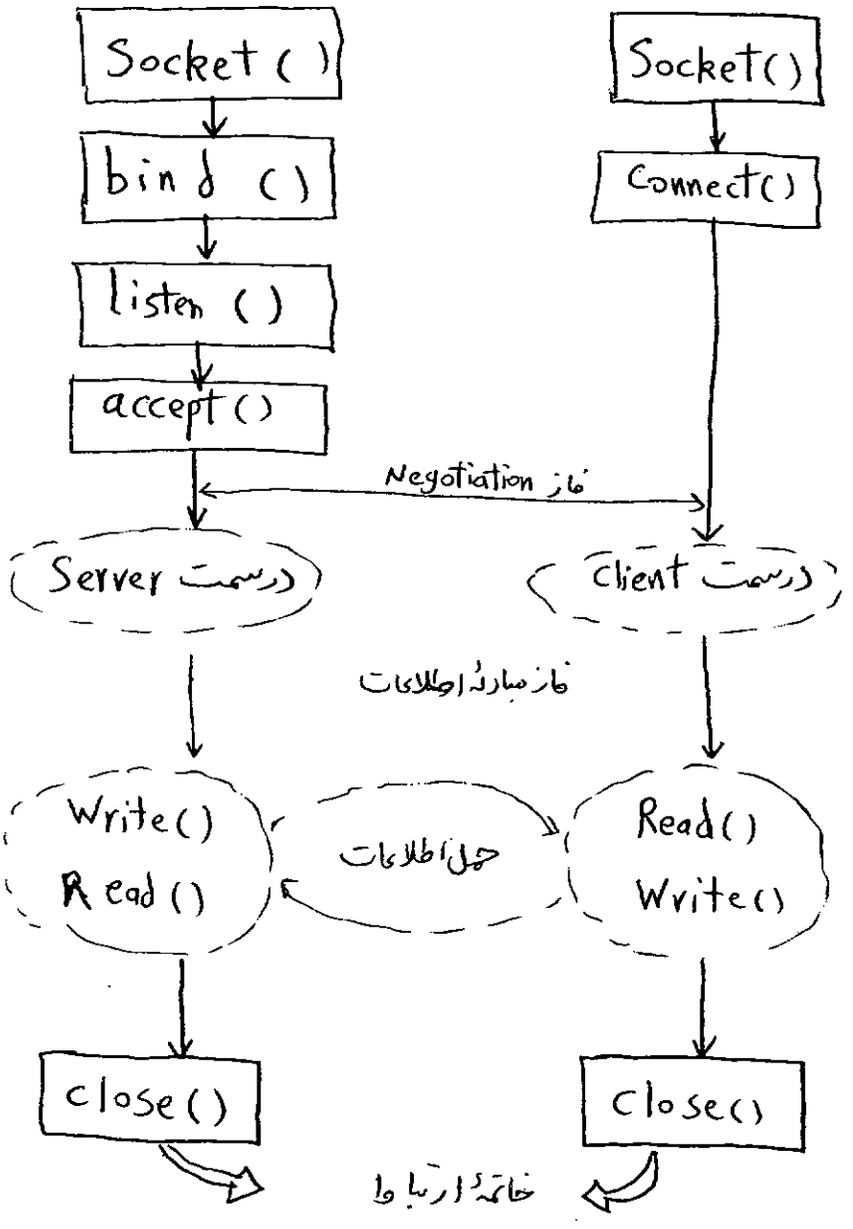
⊕ برای برنامه نویسی شبکه Library هایی وجود دارند که برای ایجاد سوکت مفید هستند.

⊕ API : Application Programming Interface

* در کتاب، کتابخانه های Berkely API مطرح شده است که به بررسی برخی از توابع آن می پردازیم.

* برای بررسی دو مورد Connection Oriented و Connection-Less را مطرح می کنیم.

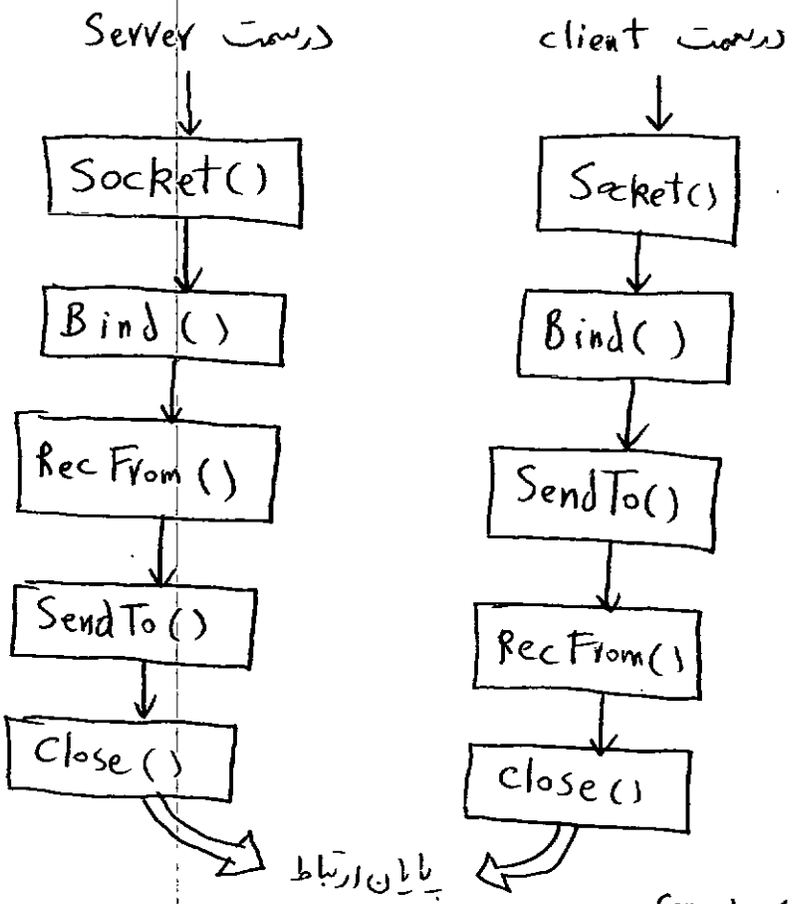
Ⓐ Connection Oriented (TCP)



شکل (53)

توابع Berkely API برای ارتباط Connection ORIENTED

(B) Connection-Less (UDP)



★ در اینجا دیگر خازبرقراری ارتباط نداریم.

شکل (54) توابع API برای ارتباط Conn-Less

★ برای اطلاع بیشتر از برنامه‌نویسی سوکت [Socket Prog.] به کتاب، بخش های ۱-۴-۲ و ۲-۴-۲ مراجعه کنید.

★ پیش از بررسی لایه های معماری OSI و TCP/IP به ارایه چند مثال از برنامه‌های کاربردی می پردازیم.

مثال ⟸ برنامهٔ TELNET

★ دو کامپیوتر را از طریق پروتکل TCP به هم وصل می‌کند.
★ یک کامپیوتر را به عنوان ترمینال کامپیوتر دیگر شبیه سازی می کند.
★ انتقال از طریق پروتکل NVT (Network Virtual Terminal) است.
★ انتقال دوطرفه Full Duplex است.
★ در هر طرف دو پورت برای ارسال و دریافت موجود است.
★ دستورات از کدهای ASCII ۷ بیتی پیروی می کنند

* دستورات مانند دستورات Terminal ها هستند.
* Password بصورت واضح مطرح است و ارسال می‌شود.

شکل (55)

Telnet

مثال: پروتکل FTP:

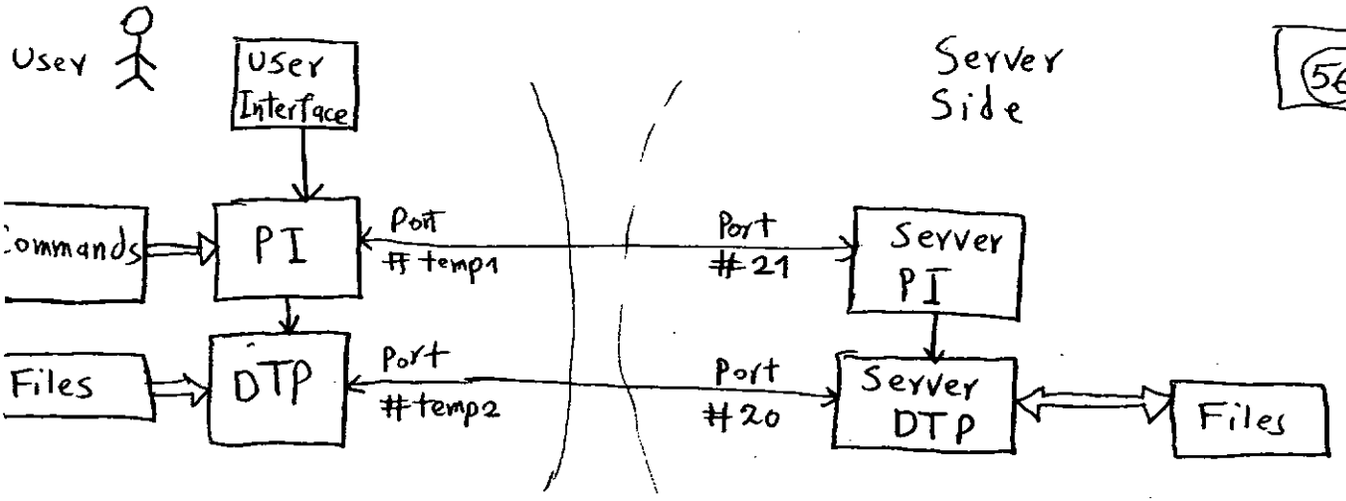
FILE TRANSFER PROTOCOL:

* دو اتصال Tcp است.
* یکی برای انتقال داده‌ها و دیگری برای مبادله‌ی فرمان‌ها.
* قادر است ساختارفایل های متفاوت در سیستم عامل های متفاوت را باهم جابجا کند و انتقال دهد.

Tcp Connection 1 : Protocol Interpreter (PI) — مفسر و انتقال دهنده فرمان

Tcp Connection 2 : Data transfer Protocol (DTA) — منتقل کننده‌ی داده ها

* سه فاز دارد:
  ۱- فاز برقراری ارتباط
  ۲- انتقال اطلاعات
  ۳- فاز خاتمه‌ی ارتباط

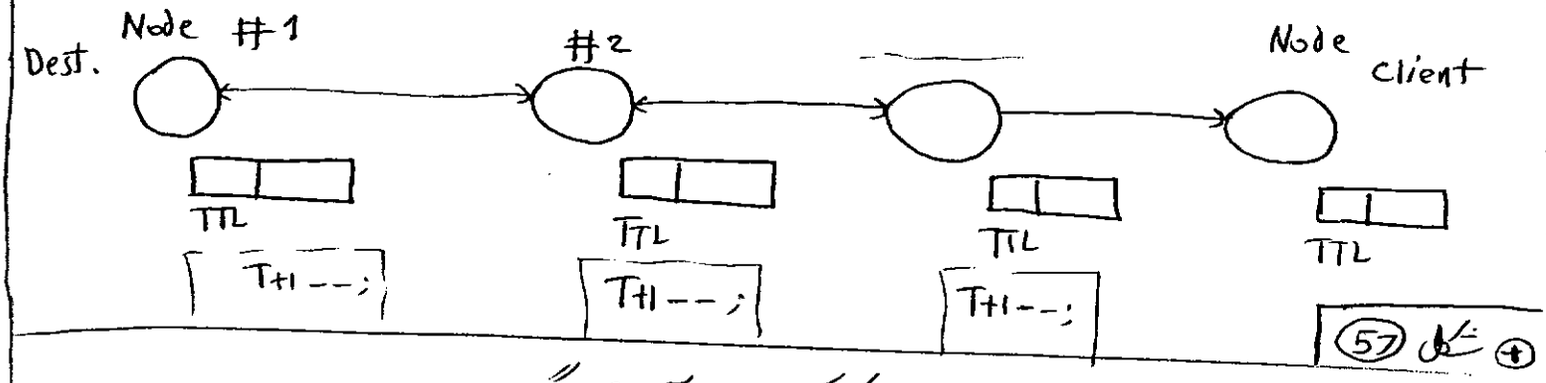


شکل (56)
FTP

* بررسی برخی Utility های لایهٔ اینترنت می پردازیم که به آنها IP Utilities می‌گوییم

مثال: PING

* Utility: برای اینکه بفهمیم طرف مقابل در چه وضعیتی قرار دارد (پروتکل IP آن فعال یا در دسترس است یا خیر)
* از پروتکل موسوم به ICMP (Internet Control Message Protocol)
* از TTL (Time-to-Live) که پارامتر خاصی است استفاده می‌کند.



شکل (57)

* TTL در header قرار داده می‌شود. که یک عدد در آن قرار می‌گیرد.
* پیام به هر نودی که می رسد یک شماره از عدد TTL مربوطه کم می‌شود.
* بدین ترتیب می توان هم در دسترس بودن طرف مقابل و هم تعداد نودهای میان راه را فهمید.

P.S.:

* سرعت انتشار نور:

۷-۱	در خلاء	$۳ \times ۱۰^۸ \ m/s$
۷-۲	در کابل مسی	$۲/۸ \times ۱۰^۸ \ m/s$
۷-۳	در فیبرهای نوری	$۲ \times ۱۰^۸ \ m/s$

رابطهٔ (۷)

★ Ping ، برای کارهای مدیریت شبکه بسیار کاربرد دارد.

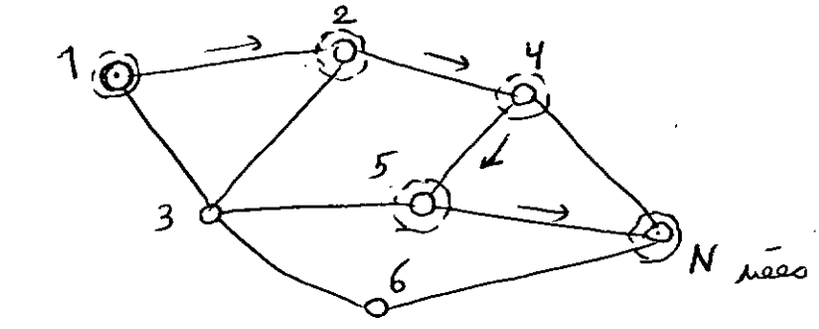
★ از Ping می توان برای حملهٔ Denial of Service استفاده کرد.

★ تعداد بسیار زیادی از دستورات Ping متوالی باعث می شود که طرف مقابل سرویسهای عادی خود را متوقف کند و فقط به Ping ها جواب بدهد.

مثال: TRACE ROUTE utility

پیگیری مسیر

★ به ما اطلاعاتی در مورد مسیر ارتباط می دهد.



شکل (58)

⊕ مسیر 1-2-4-5-N

Rout

★ از پروتکل ICMP استفاده می کند.

★ از پارامتر TTL استفاده می کند.

★ فرض اساسی در Trace Rout آن است که در مدت بسیار کوتاه مسیرها عوض نمی شوند و همان مسیر قبلی باقی می مانند.

⊕ روش کار TRACE Rout

★ ابتدا TTL را مقدار 1 می گذارد و آنرا با آدرس مقصد به نود بعدی می فرستد.

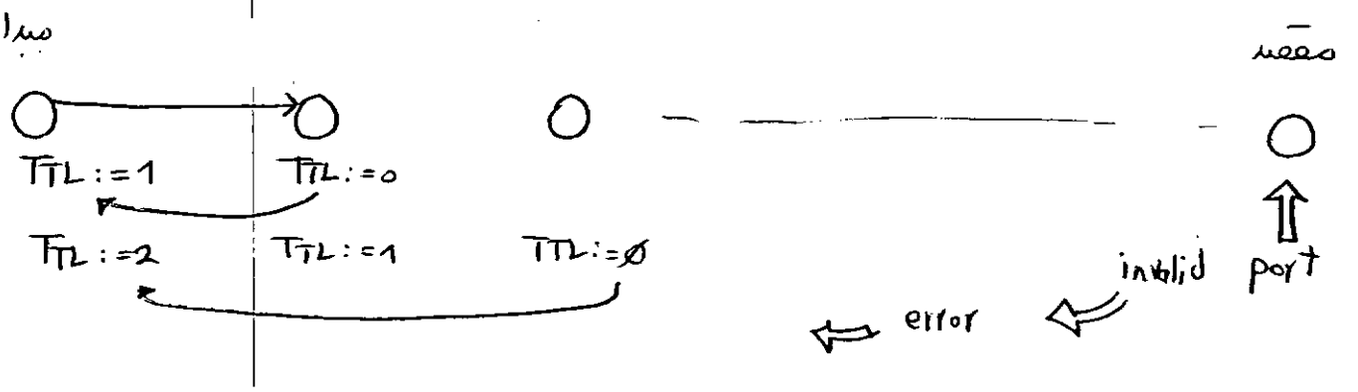
★ نود بعدی TTL را صفر می کند و پیغامی به نود فرستنده برمی گرداند.

★ فرستنده می فهمد که نود بعدی در راه مقصد، نودی است که TTL را صفر کرده.

★ حال TTL را مقدار 2 قرار می دهد و آنرا می فرستد تا نود دوم در راه را پیدا کند.

★ فرستنده TTL را آنقدر مقداردهی - بصورت یکی یکی افزایشی - می کند تا به مقصد برسد.

★ برای آنکه مقصد بسته را برندارد، فرستنده Port Number آن را اشتباه می گذارد تا پیغامی برگردد، تا بفهمد که پیغام به مقصد رسیده است.



شکل (59) ⊕ شکل: روش کار Utility ، TRACE Rout

مثال: NETSTAT utility

★ گزارشی در مورد وضعیت کنونی شبکه ارایه می دهد.

# Packet in	تعداد بسته های وارد شده
# Packet out	تعداد بسته های خارج شده
# error packet	تعداد بسته های خطا
Routing table	جدول مسیردهی
Tcp Connection	ارتباطات Tcp

★ NETSTAT برای کارهای مدیریتی و همچنین جدول های مسیردهی و کنترل ترافیک کاربرد دارد.

مثال: Tcp dump

★ Packet های وارد شده را dump - ذخیره - می کند.

★ کارت های شبکه مود کاری تحت عنوان Promiscuous دارند که در آن مُد نه تنها بسته های مختص خودش را، که تمامی بسته هایی که از خط عبوری کنند را - یک کپی از آن ها - برمی دارد و Tcp dump نسخه های بسته ها را ذخیره می کند.

تمرینهای فصل دوم: 16-17-21-26-27-29-37-53

فصل سوم : «بررسی دقیق تر لایه های معماری»

★ برای بحث آموزش ترکیبی - ۵ لایه ای - از Tcp/Ip و OSI مطرح می شود.

Physical – Data Link – Network – Transport – Application

۱- لایهٔ فیزیکی Physical System

★ کار این لایه انتقال اطلاعات دیجیتال به علایم قابل عبور از کانال است. (وانتقال آنان)

انواع اطلاعات : (از لحاظ طبیعت اطلاعات)

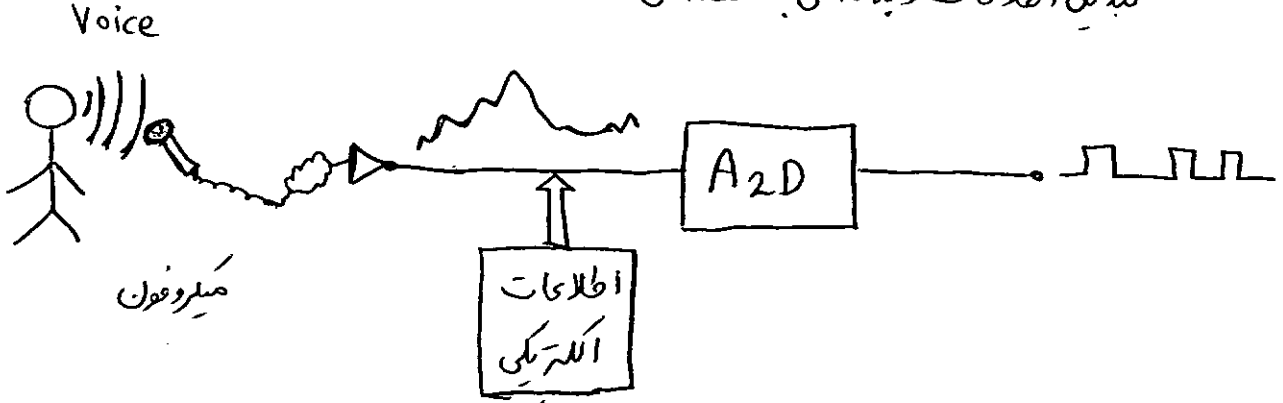
۱- دنباله ای : Stream
صوت، تصویر

۲- قطعه ای : Block
اطلاعات دیجیتال

★ هر سرویس لزوماً اطلاعات قطعه ای جابجا می کند.

★ هر سرویس نیازها - Requirements - ی مورد نظر خودش را دارد.
مثلاً سرویسی ممکن است تأخیر حداقل را مورد نظر داشته باشد.

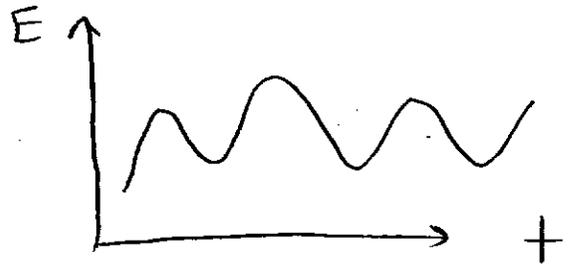
تبدیل اطلاعات دنباله ای به قطعه ای :



شکل (60) ⊕ تبدیل صوت به علایم الکتریکی

⊕ سیگنال های ANALOG:

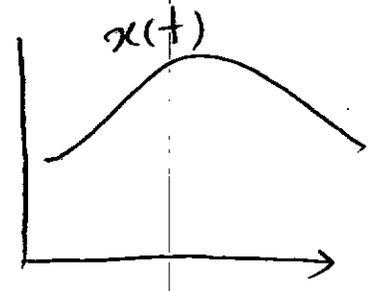
* سیگنالهایی که تغییرات آنها نسبت به زمان پیوسته باشد (مثل علایم الکتریکی ناشی از صوت)



شکل (61)
Analogous Signal

⇦ تبدیل سیگنال Analog به Digital:

⊕ نمونه برداری: SAMPLING

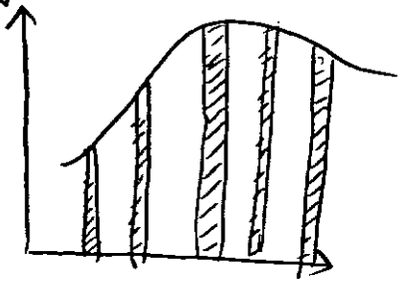


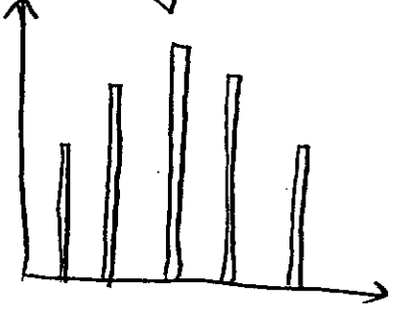
شکل (62) نمونه برداری (تئوریک)

Analog سیگنال

SAMPLING

نمونه برداری از سیگنال آنالوگ



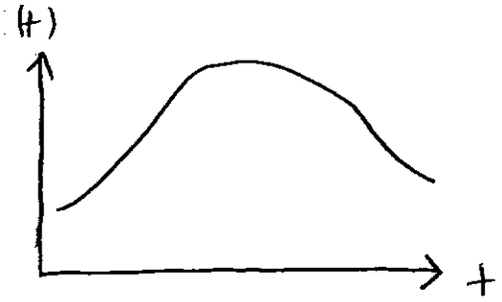


نمونه «SAMPLE»

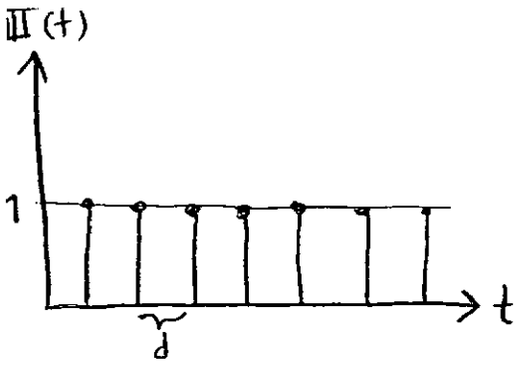
قضیه نمونه برداری نایکوئیست: (Nyquist Sampling)

★ نمونه برداری از سیگنال آنالوگ، باعث از بین رفتن اطلاعات آن نمی شود.

⊕ بیان سادهٔ ریاضی قضیه: (شکل 63)



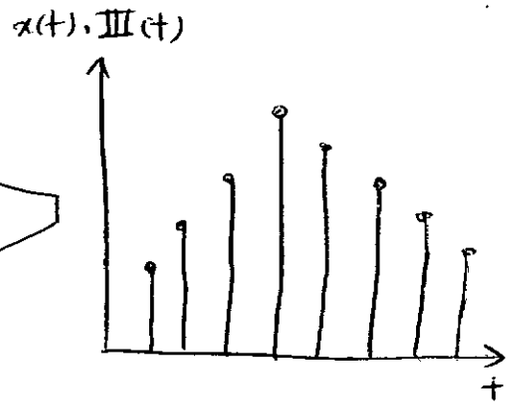
سیگنال $x(t)$



قطار پالس

« نمونه برداری واقعی با استفاده از تابع ضربه و ضرب آن در سیگنال »

« ضرب دو سیگنال در هم »



ضرب $x(t)$ در قطار پالس

$d \to 0$

★ اگر d به سمت صفر میل کند $x(t)\cdot Ш(t)$ برابر $x(t)$ است.

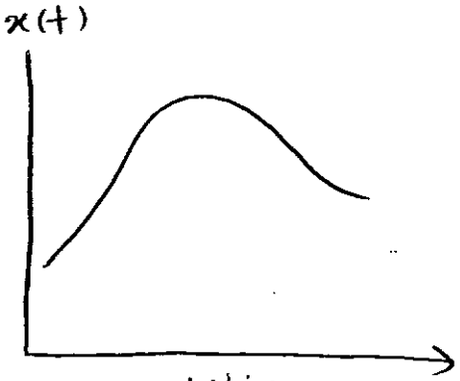
⊛ تبدیل فوریه:

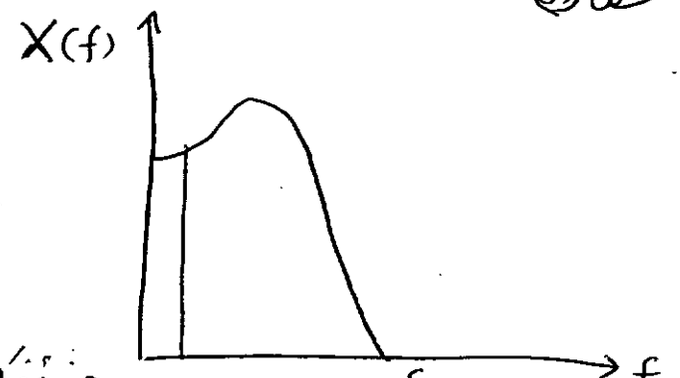
رابطهٔ (8)

$$F[x(t)] = \int_{-\infty}^{+\infty} x(t) e^{-j2\pi ft}\, dt$$

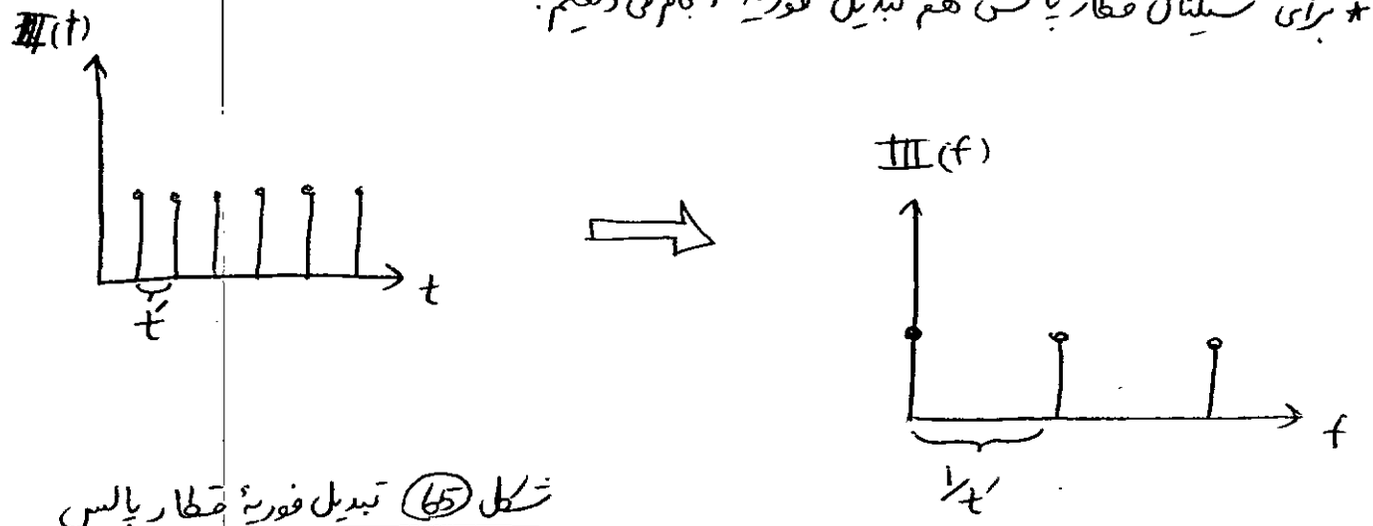
★ هر سیگنال را می توان بصورت مجموعی از توابع کسینوسی نشان داد.

شکل (64)



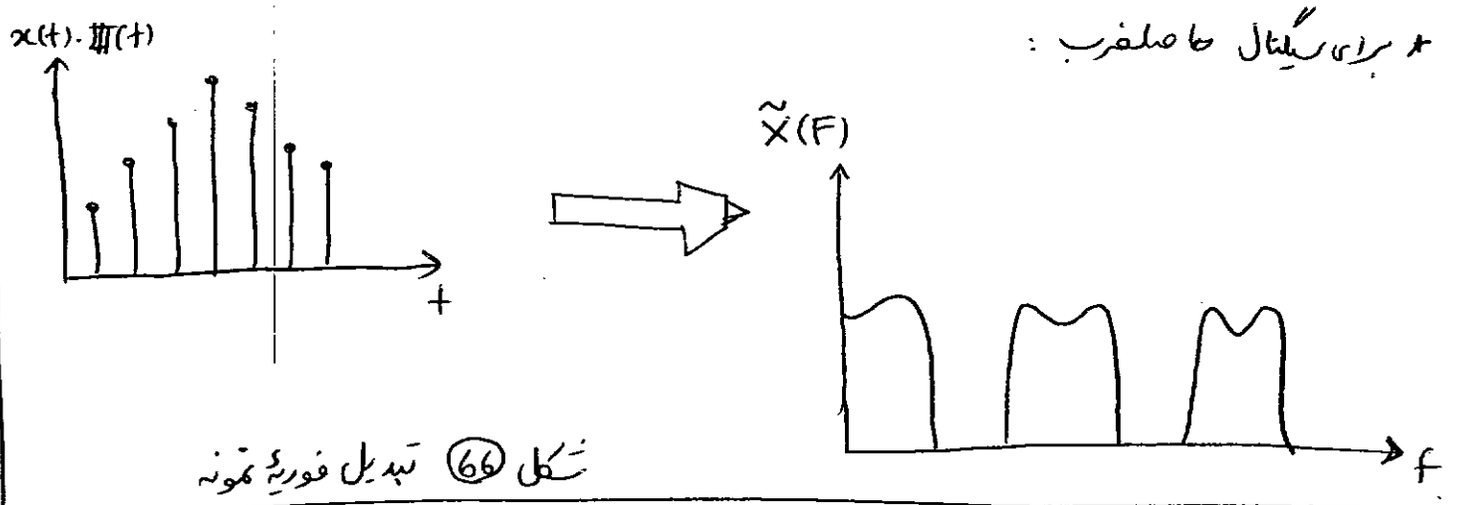


* برای سیگنال قطار پالس هم تبدیل فوریه انجام می دهیم.



شکل (65) تبدیل فوریهٔ قطار پالس

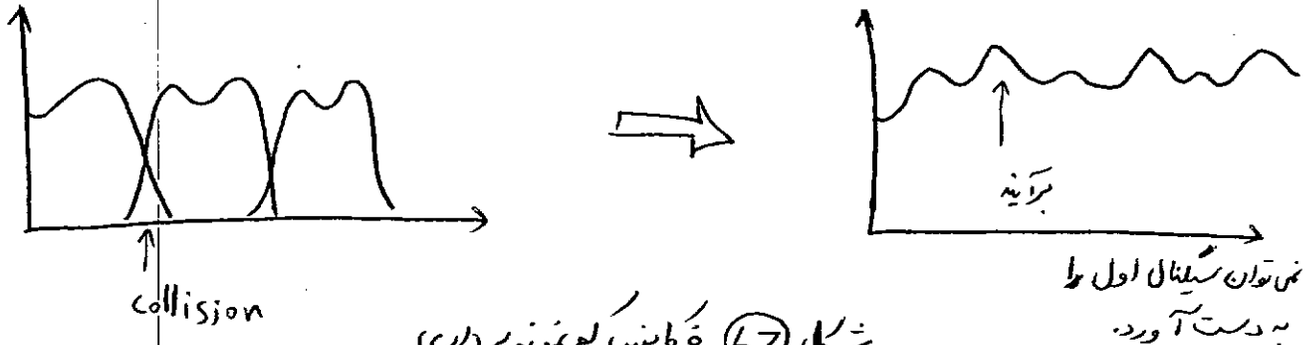
* برای سیگنال حاصلضرب :



شکل (66) تبدیل فوریهٔ نمونه

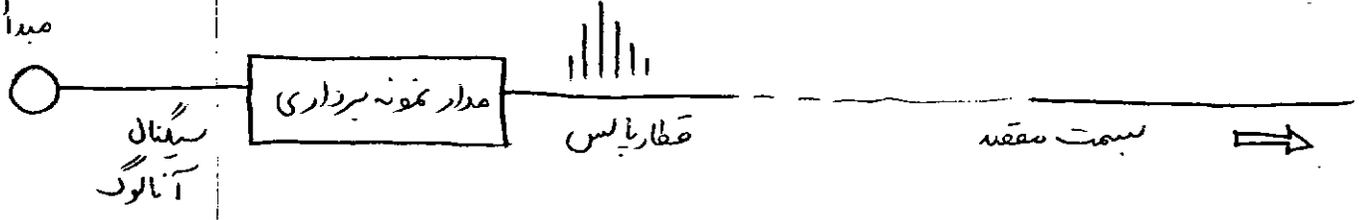
* هرچه T کوچکتر باشد فاصلهٔ تکه‌ها در حوزهٔ فرکانس بیشتر می‌شود.

* تکه‌ها نباید با هم تداخل کنند، چون نمونه‌ها درون هم می روند



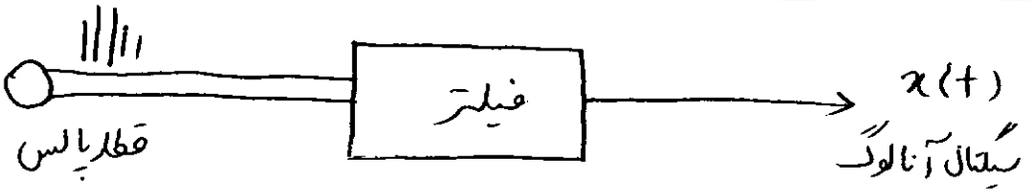
نمی‌توان سیگنال اول را به دست آورد.

شکل (67) فرکانس کم نمونه برداری



(شکل (68))

* مدار نمونه بردار تابع ضربه را در $x(t)$ ضرب می‌کند و قطار پالس تشکیل می دهد.



* فیلتر محدودهٔ خاصی از فرکانس ها را ردمی‌کند، و نمونه ها را سرهم می‌کند. (شکل 69)

⊕ بیان قضیه نایکوییست:

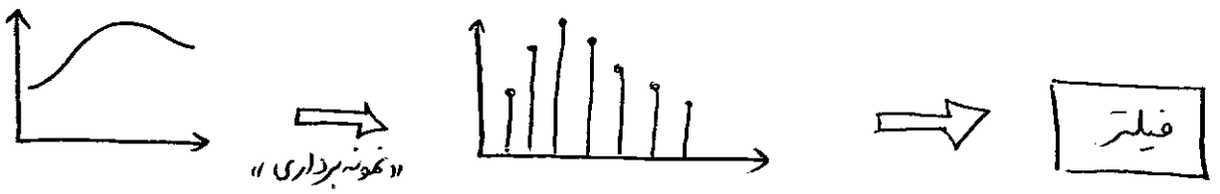
* هر سیگنال $x(t)$ را می‌توان با نمونه‌هایش بازسازی کرد، به شرط آنکه:

فرکانس نمونه برداری $\left(f_s=\frac{1}{T}\right)$

شرط اساسی نمونه‌برداری: $f_s \geq 2f_M$ رابطهٔ (9)

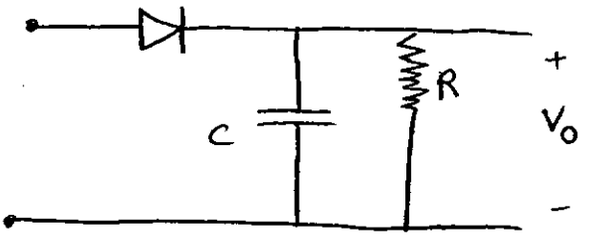
f_M = بزرگترین مولفه فرکانسی $x(t)$

⊕ بیان دیگری از قضیهٔ نمونه برداری: (مدار فیلتر)

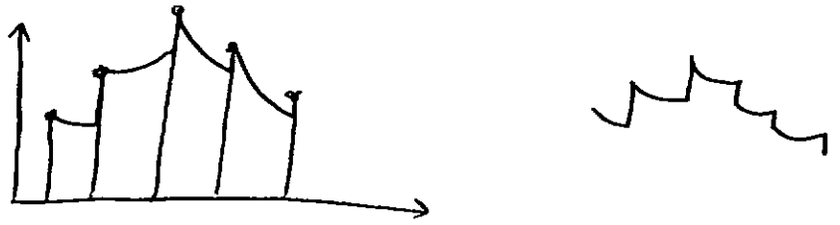


شکل (70)

یک فیلتر ساده:



شکل (71)



$V_o \simeq x(t)$

* سیگنال بدست آمده از فیلتر تقریباً برابر $x(t)$ ورودی به نمونه بردار است.

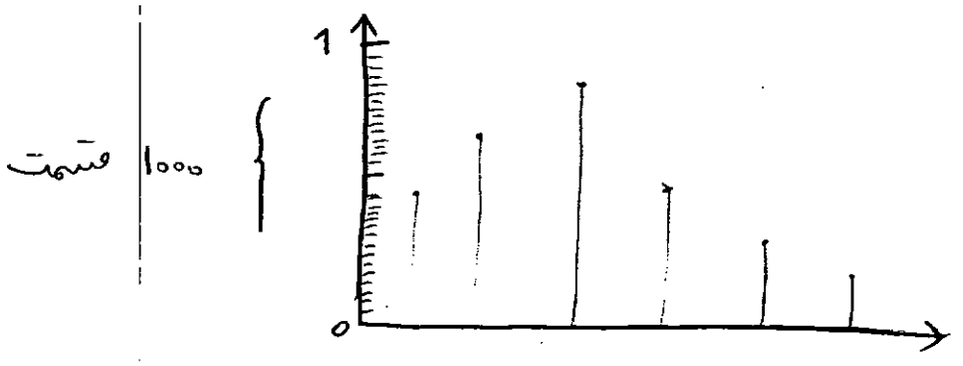
* هرچه مقادیر R و C خوب انتخاب شوند و فرکانس نمونه برداری بالا باشد V_o بیشتر شبیه $x(t)$ است

نتیجه ⇐ از این پس به جای سیگنالها می‌توانیم از نمونه آنها استفاده کنیم (به جز بیانات ریاضی‌اش این به بعد کاری نداریم :))

Digitization: دیجیتال کردن عالم:

* تعداد سطوح ممکن برای سیگنال نمونه $\tilde{x}(t)$ را کم می‌کنیم.
* مثلاً فرض می‌کنیم ۱۰۰۰ سطح ممکن بین سطوح ۰ و ۱ قرار دارد.
* اگر نمونه‌ای بین دو سطح بود، آنرا به نزدیکترین سطح تقلیل یا افزایش می‌دهیم.

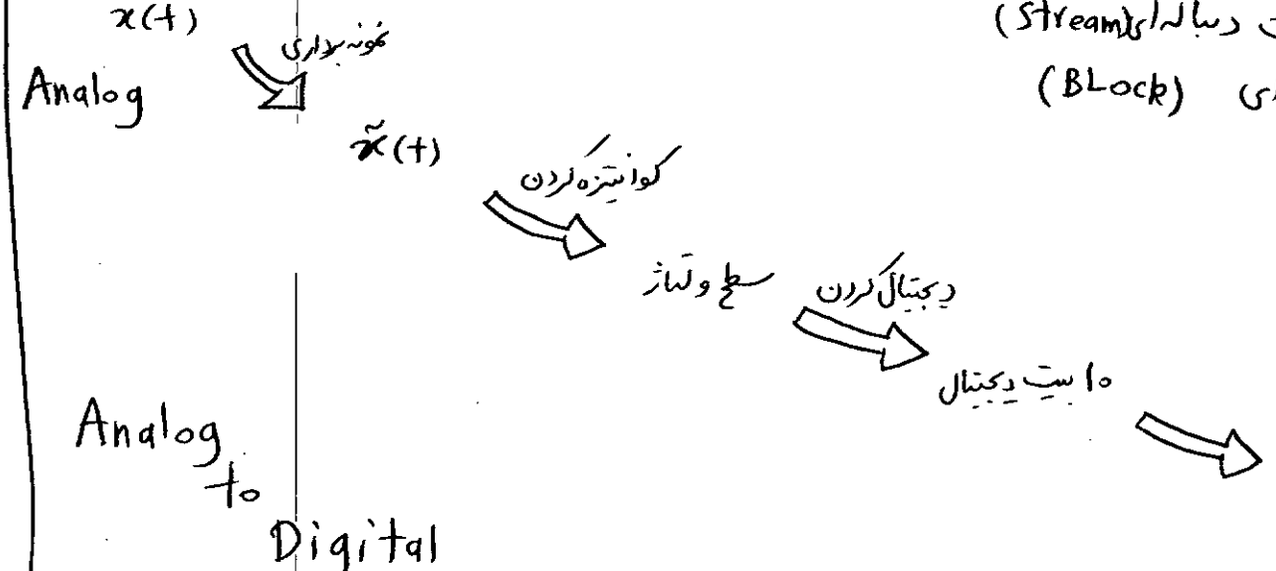
Quantization «کوانتیزه کردن»



شکل (72)

* با n بیت می‌توان سطح ولتاژ هر نمونه را ذخیره و انتقال داد.

مراحل تبدیل اطلاعات دنباله‌ای (Stream) به اطلاعات قطعه‌ای (BLock)



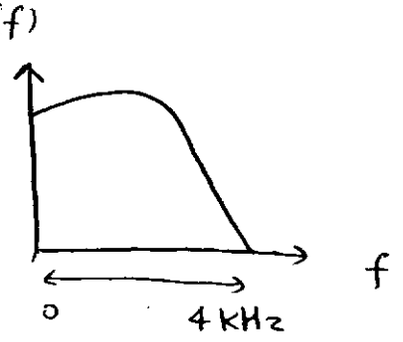
* عکس مراحل بالا قابل انجام است. (تبدیل اطلاعات قطعه‌ای به Stream)

مثال «ارتباط تلفنی»

$x(t)$

سیگنال تلفنی

شکل (73)



* فرض کنیم می‌خواهیم سیگنال های تلفنی را از طریق شبکه کامپیوتری منتقل کنیم.

$f_M = 4000$ Hz

* قضیه نمونه برداری

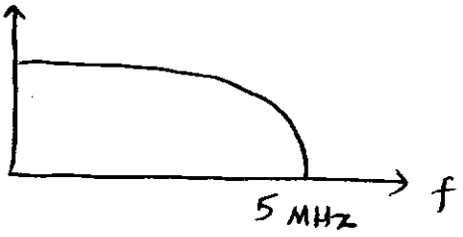
$f_S \geq 8000$ Sample/Second

* برای نمایش هر نمونه از ۸ بیت استفاده می‌کنیم. (۲۵۶ سطح ولتاژ)

تعداد بیت خروجی در هر ثانیه $= 64\,000$ b/sec

* برای انتقال سیگنال، خطوط باید bitrate، ۶۴۰۰۰ را حمایت کنند

مثال: «تصاویر تلویزیونی»



$f_M = 5$ MHz

* بزرگترین فرکانس موجود در موج تصویر

$f_S \geq 2 f_M \Rightarrow f_S \geq 10 \times 10^6$ Sample/Sec *

* هر Sample را - برای دقت بهتر - با ۲۴ بیت نمایش می‌دهیم.

* نرخ بیت برثانیه:

$24 \times 10 \times 10^6 \Rightarrow$

* حداقل نرخ انتقال اطلاعات در کانال ها:

240 Mb/s

* هر سرویسی نیازهای Requirements متفاوتی دارد.

+ یکی از این نیازها، دقت Accuracy است.

⊗1 دقت : ACCURACY:

* در فاز کوانتیزه کردن، تعداد سطوح از اهمیت زیادی برخوردار است.

هر Sample ⟵ ۴ بیت ⟵ تعداد سطوح = $2^4 = 16$

هر Sample ⟵ ۸ بیت ⟵ تعداد سطوح = $2^8 = 256$

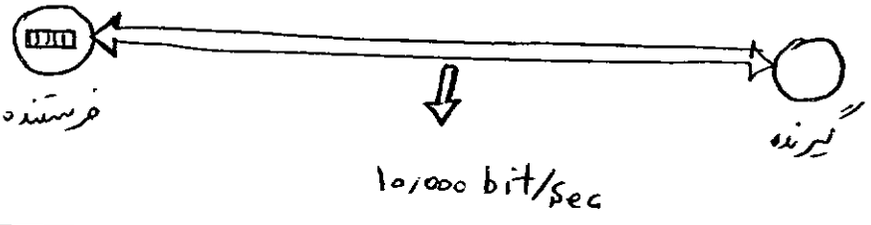
هر Sample ⟵ ۱۰ بیت ⟵ تعداد سطوح = $2^{10} = 1024$

* هرچه تعداد سطوح بیشتر باشد، خطای ناشی از Map کردن سطوح و تبدیل نمونه به یکی از سطوح کوانتوم کمتر می‌شود و نتیجتاً دقت بالا می‌رود.

* سرعت خط باید متناسب با bitrate نمونه بردار طرف فرستنده باشد، وگرنه این عمل با تأخیر انجام می‌شود. (بدین نیاز، زمان بری‌مان گوییم.)

⊕2 زمان بری TIMELINESS

«زمان کل» $[t_x + t_p]$



شکل (۷۴)

bitrate

$f_s = 8000 \times 8bit = 64000\ bit/sec$ ⟵ ۶٫۴ ثانیه تأخیر

$8000 \times 10bit = 80000\ bit/sec$ ⟵ تأخیر ۸ ثانیه

$8000 \times 1bit = 8000\ bit/sec$ ⟵ بدون تأخیر

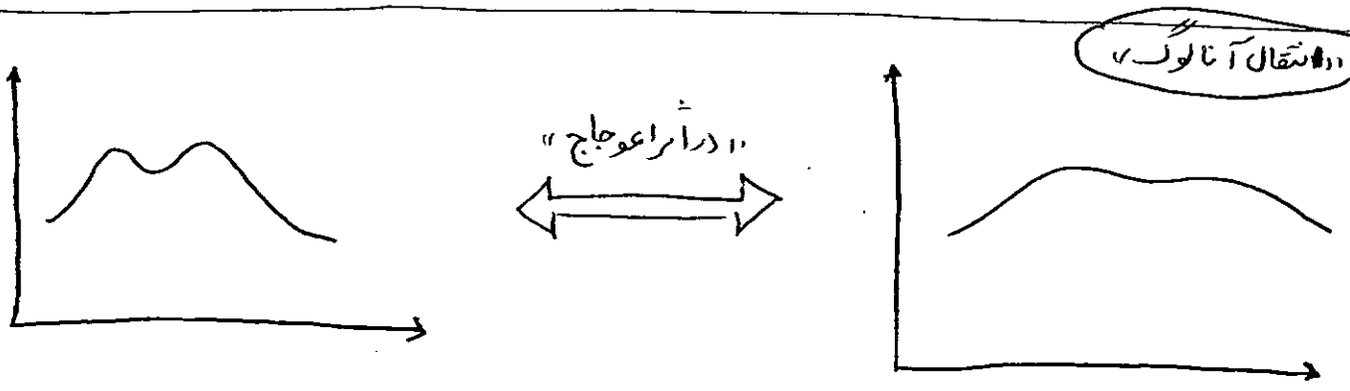
* در مبحث زمان بری - Timeliness - دو پارامتر t_p و t_x مطرح هستند.

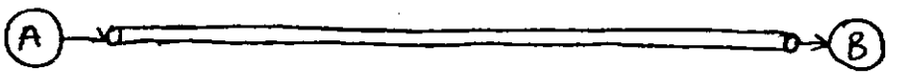
t_x = زمان ارسال = L/R

t_p = زمان انتشار = d/v

⇦ در ارسال اطلاعات روی یک خط، پدیده‌ی Jitter، بیتها لزوماً به ترتیب فرستاده شده دریافت نمی شوند.

سؤال: چرا انتقال بصورت دیجیتال - ونه آنالوگ - صورت می گیرد؟





«فرستنده» «گیرنده»

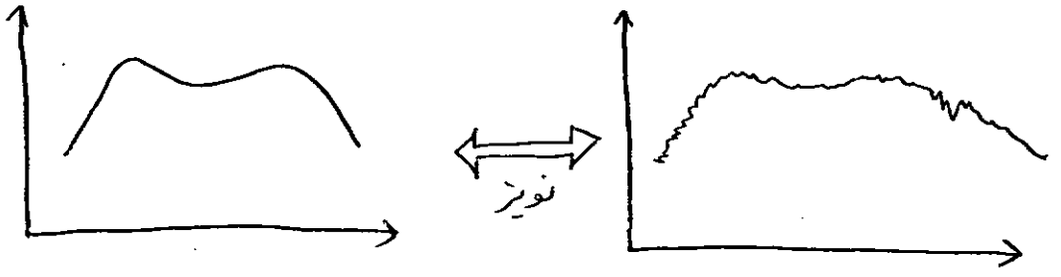
«شکل (75) اعوجاج در انتقال آنالوگ»

* گیرنده باید اعوجاج را تشخیص دهد و مشکلات ناشی از آن را حل کند.

⊕ دستگاه EQUALIZER، در انتهای خط قرار می گیرد و مشکلات تضعیف سیگنال را حل می کند.

* ساخت دستگاه اکولایزر هزینه بر است.

⊗ تشخیص و رفع نویز - NOISE - در انتقال آنالوگ کار آسانی نیست.

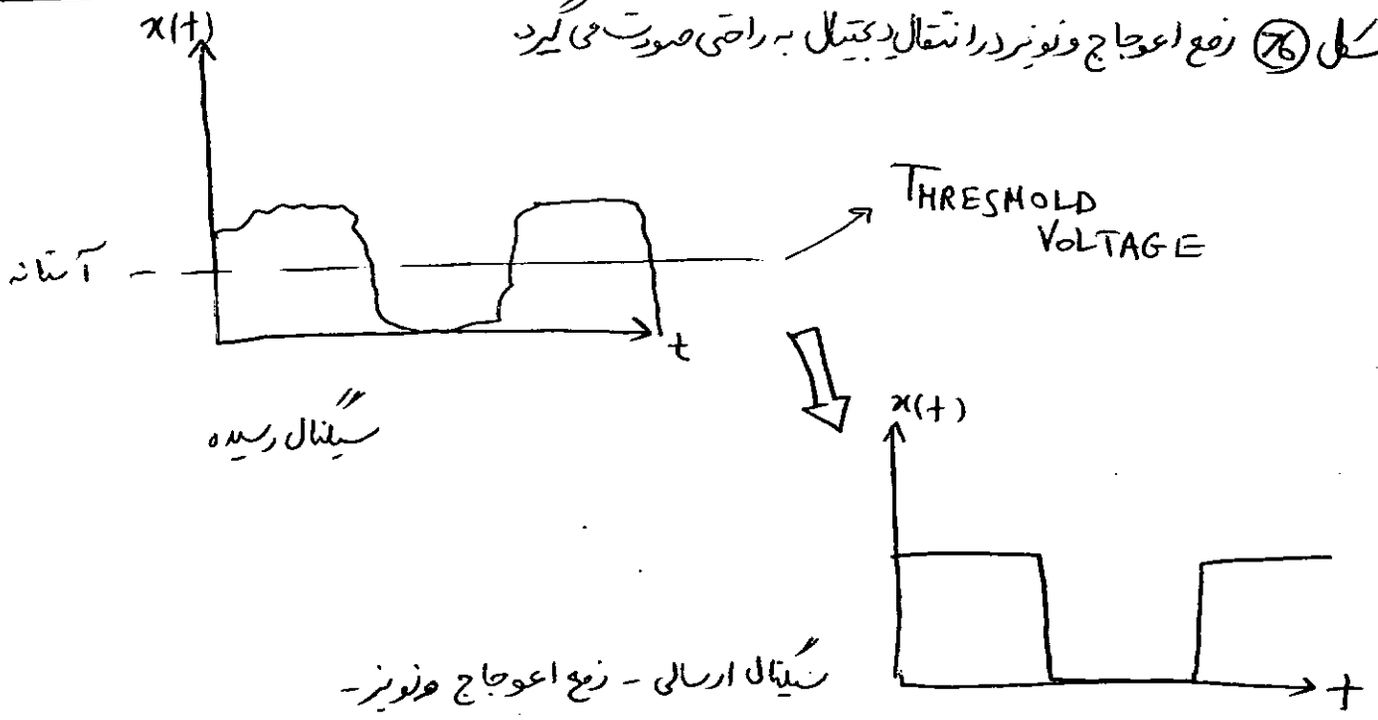


شکل (75)
نویز، انتقال آنالوگ

* در انتقال دیجیتال مساله اعوجاج به راحتی حل می شود.

* سطح ولتاژی را به نام آستانه - THRESHOLD - در نظر می گیریم، اگر سیگنال رسیده از آن بیشتر بود، آن را یک و در غیر اینصورت آنرا صفر در نظر می گیریم.

شکل (۷۶) رفع اعوجاج و نویز در انتقال دیجیتال به راحتی صورت می‌گیرد.

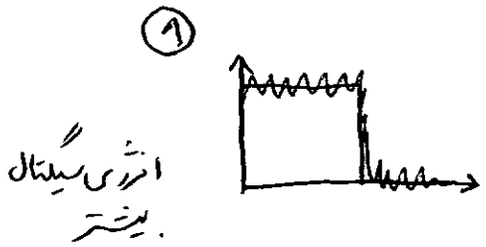


★ عمل رفع اعوجاج خود بخود نویز را برطرف می‌کند.

★ به دلیل هایی که ذکر شد، انتقال دیجیتال به آنالوگ ترجیح داده می‌شود.

۱۱ پارامترهای اثرگذار بر نرخ بیت خطوط انتقالی

① انرژی سیگنال: ENERGY



★ رفع نویز در حالت اول راحت‌تر است.

② فاصله DISTANCE

③ نویز: NOISE

★ برخی کانال‌ها پرنویز و برخی کم نویز‌اند. (مشخصاً کانال‌های پرنویز نرخ بیت ارسالی کمتر است.)

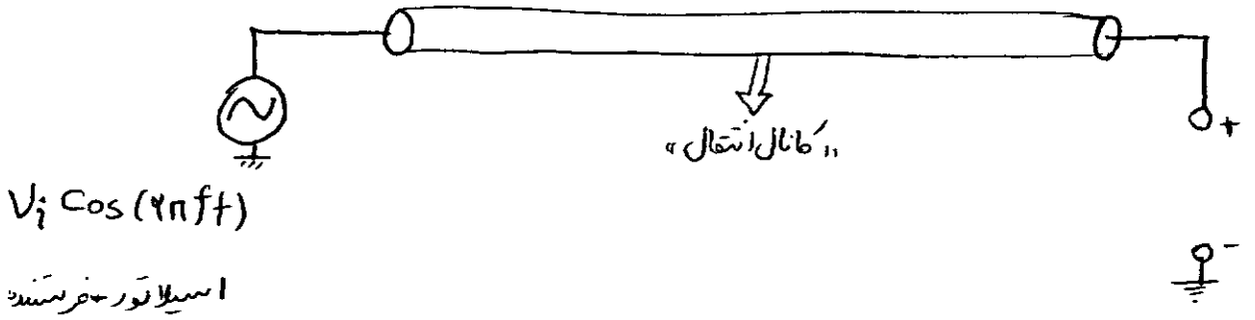
④ پهنای باند: BANDWIDTH :

⊛ به بررسی این مفهوم می پردازیم

⊕ مشخصه کانال انتقال: CHANNEL CHARACTERISTICS:

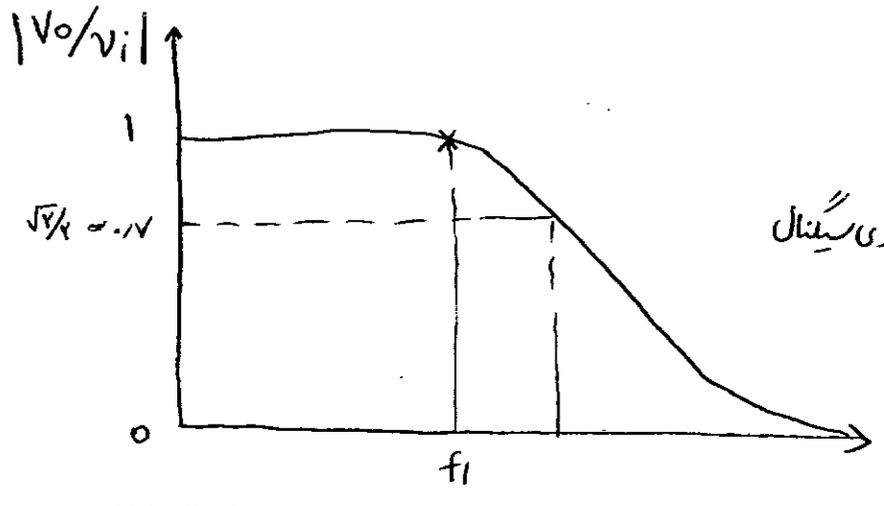
∞ الف- مشخصه میدان فرکانس « FREQUENCY DOMAIN CHARCTERISTICS »

★ در طرف فرستنده یک اسیلاتور - OSCILLATOR - می گذاریم تا سیگنال های سینوسی روی کانال بفرستد.



« شکل (77) ارسال و دریافت امواج سینوسی »

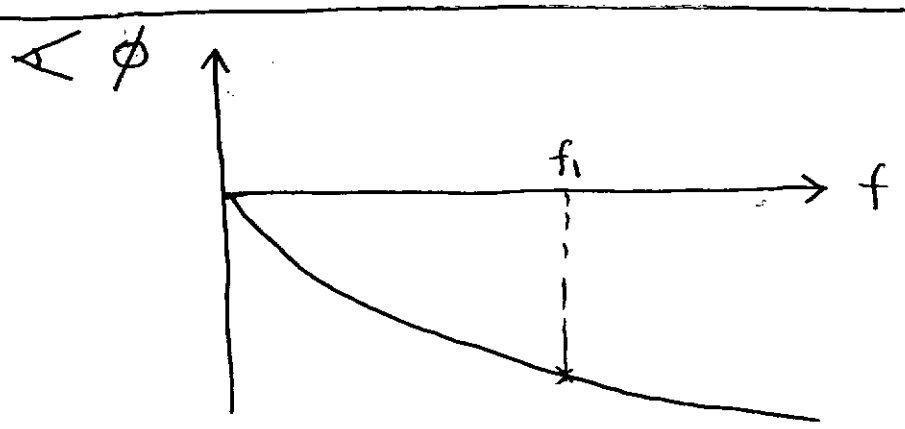
★ نسبت $\left|\frac{V_o}{V_i}\right|$ را برای فرکانس های مختلف اندازه گیری می کنیم.



شکل (78)
هرچه فرکانس بیشتر شود، از دقت سیگنال دریافتی کمتر می شود.

★ ممکن است در برخی کانال ها، منحنی V_o/V_i به شکل بالا نباشد، ولی شکل تیپیک آن همین است.

★ بحث دیگر ϕ، اختلاف فاز سیگنال ورودی به خروجی است.



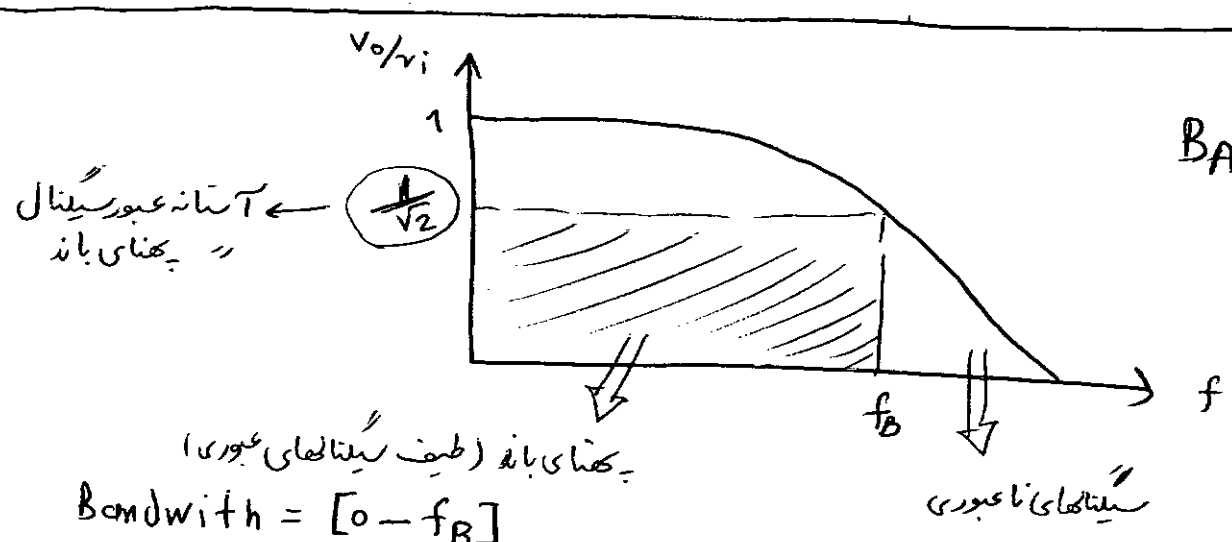
شکل (79)
نمودار ارتباط اختلاف فاز سیگنال دریافتی نسبت به فرکانس سیگنال ارسالی

⊛ به مقادیر V_o/V_i بین ۱ تا $\frac{1}{\sqrt{2}}$ اصطلاحاً سیگنالهای عبوری از کانال نامیده می‌شوند.

⊛ به بقیه سیگنالها، سیگنالهای نا عبوری می‌گوییم

⊕ پهنای باند، BANDWITH

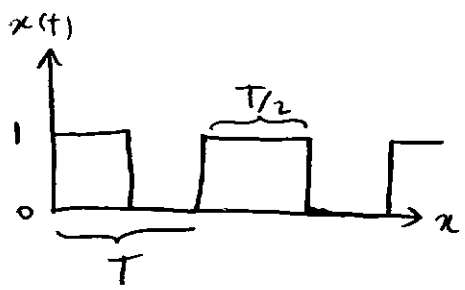
طیف سیگنالهای عبوری یک کانال را پهنای باند می‌گوییم.



شکل (80)
BANDWIDTH

⊛ مثال (پهنای باند):

$x(t)$ یک سیگنال پریودیک با دوره تناوب (T) است.



* برای سیگنالهای تناوبی بسط فوریه مطرح می‌شود:

$$x(t) = a_0 + 2\sum_{n=1}^{\infty}\left[a_n \cos\frac{2n\pi t}{T} + b_n \sin\frac{2n\pi t}{T}\right]$$ بسط فوریه

$$a_0 = \frac{1}{T}\int_{-T/2}^{T/2} x(t).dt \qquad a_n = \frac{1}{T}\int_{-T/2}^{T/2} x(t)\cos\frac{2n\pi t}{T}dt \qquad b_n = \frac{1}{T}\int_{-T/2}^{T/2} x(t)\sin\frac{2n\pi t}{T}dt$$

بسط فوریه سیگنال مثال

★ برای سیگنال $x(t)$ این مثال داریم:

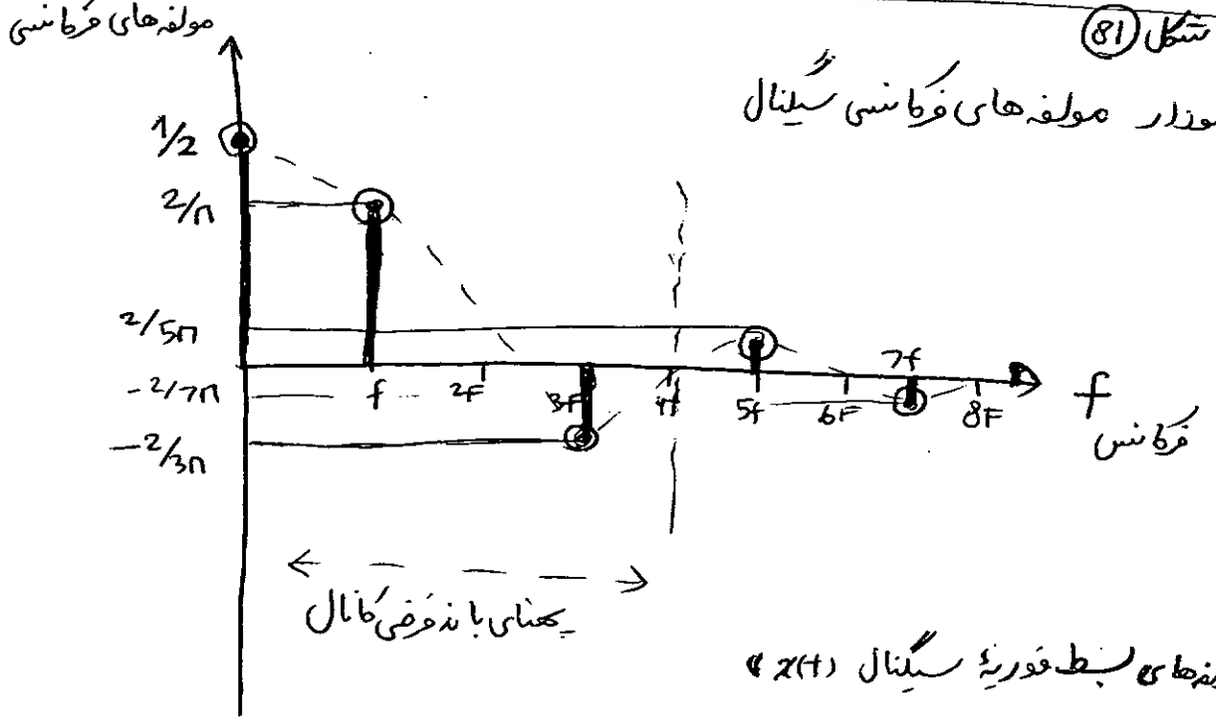
رابطه ⑤ $$x(t) = \frac{1}{2} + \frac{2}{\pi}\cos 2\pi ft + \frac{-2}{3\pi}\cos 2\pi(3)ft + \frac{2}{5\pi}\cos 2\pi(5)ft + \dots$$

مولفه DC ⇐ — مولفه فرکانس f ⇐ — مولفه $3f$ ⇐

$$f = 1/T$$

شکل (81)

نمودار مولفه‌های فرکانسی سیگنال



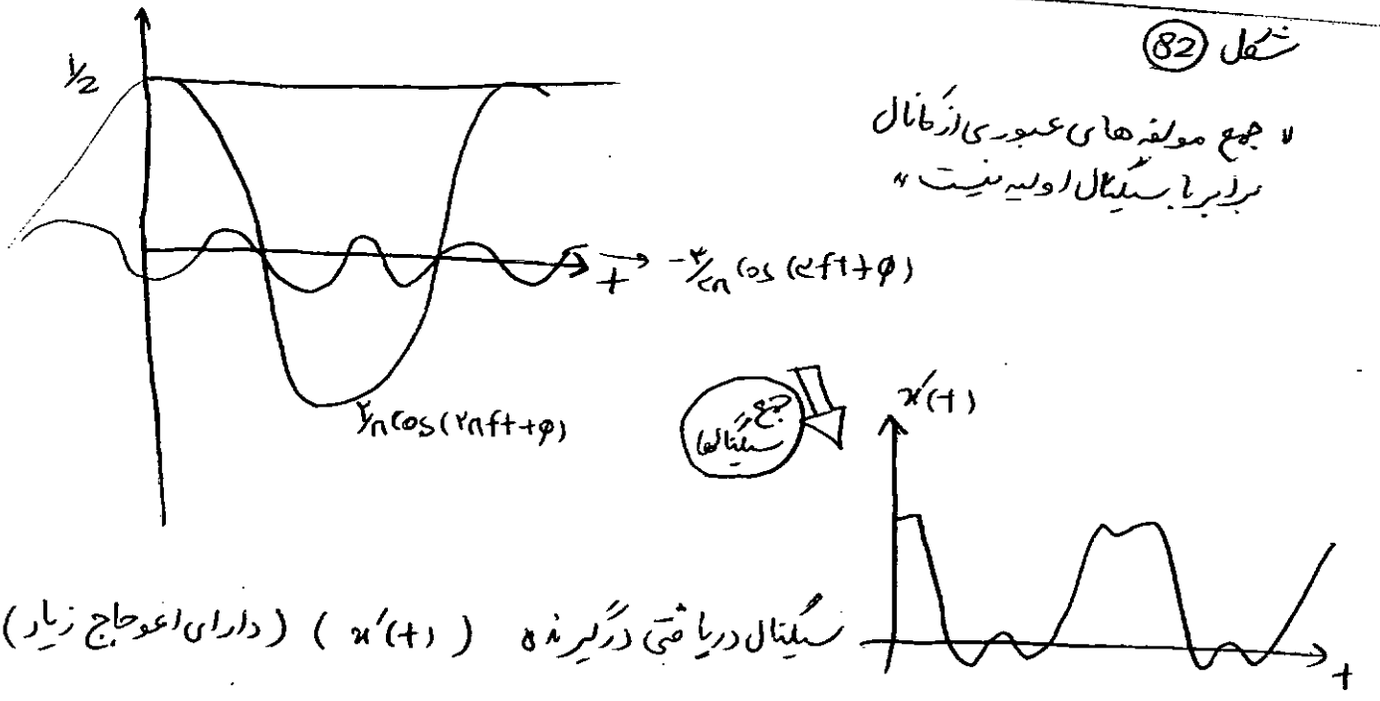
«مولفه‌های بسط فوریه سیگنال $x(t)$»

★ کانال انتقال یک سری از فرکانسها را از خود عبور می‌دهد و یک سری را عبور نمی‌دهد.

★ فرض بنیم کانال تا فرکانس $3f(\frac{5}{T})$ را عبور می‌دهد.

شکل (82)

«جمع مولفه‌های عبوری از کانال برابر با سیگنال اولیه نیست»



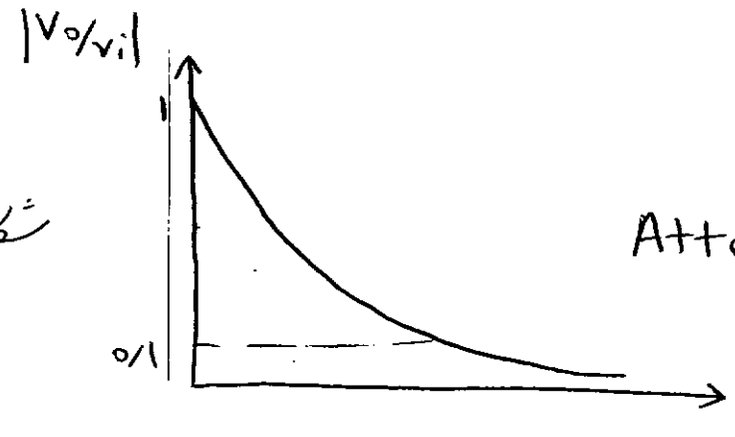
سیگنال دریافتی دریافت کننده ($x'(t)$) (دارای اعوجاج زیاد)

* درنتیجه محدودیت پهنای باند باعث ایجاد اعوجاج در سیگنال دریافتی می‌شود.
* درنتیجه پهنای باند عامل موثری در نرخ بیت ارسالی است.

① - طول خط:

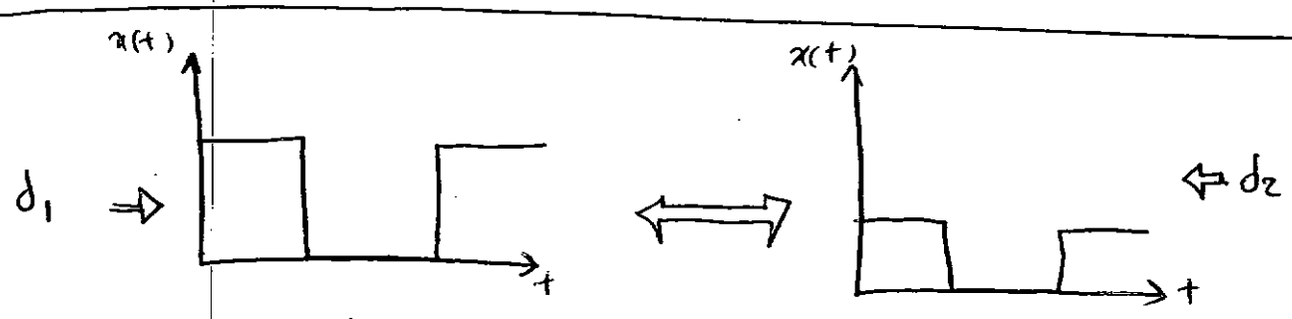
* فاصله در تضعیف سیگنال اثر دارد.

Attenuation (تضعیف)



شکل (83)

* تضعیف نسبت $|V_o/V_i|$ باعث کمتر شدن پهنای باند است.



طول کمتر — طول بیشتر

« شکل (84) نقش فاصله در پهنای باند و تضعیف انرژی سیگنال »

$d_1 > d_2$

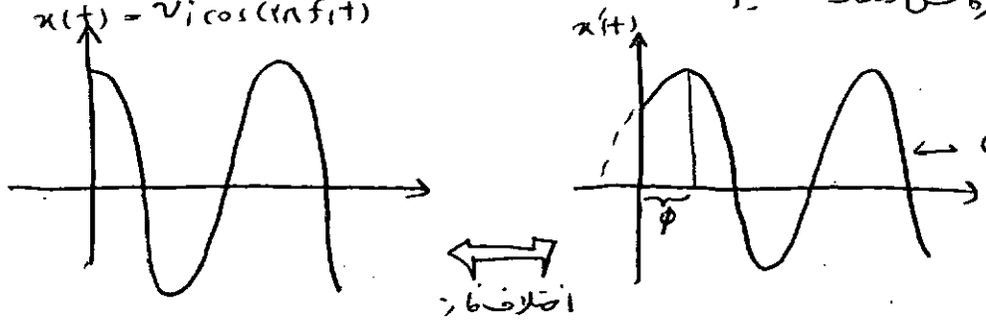
⇦ با ازدیاد طول خط تضعیف بیشتر مولفه‌ها داریم. (حد اکثر نویز سخت ترمی شود.)

② اختلاف فاز ϕ

a) $V_{o1} = \cos(2\pi f_1 t + \phi_1)$

b) $V_{o2} = \cos(2\pi f_2 t + \phi_2)$

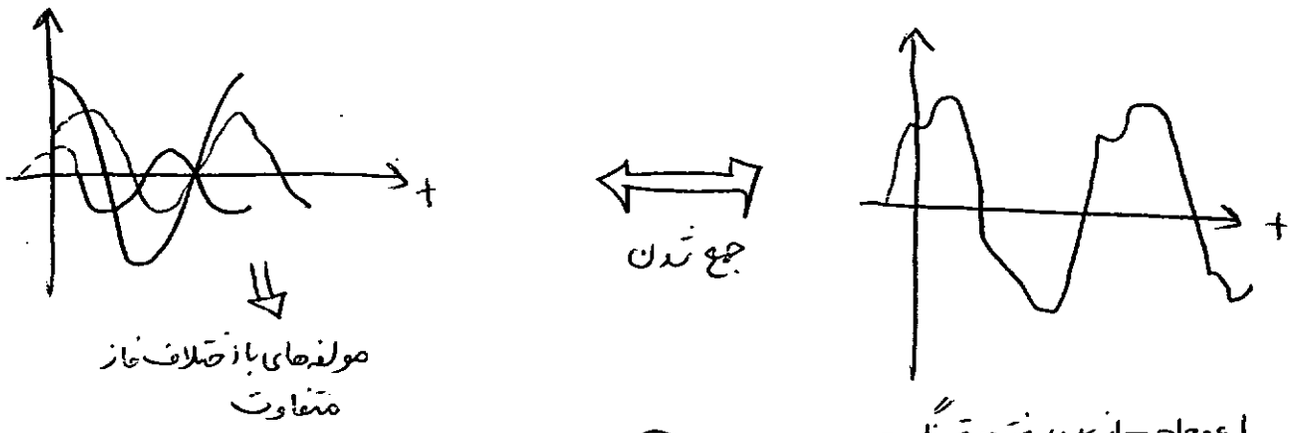
* V_i را 1 در نظر می‌گیریم.
* V_o/V_i را هم 1 در نظر می‌گیریم.
* برای دو فرکانس متفاوت داریم:



شکل (85) تأثیر فرکانس در اختلاف فاز سیگنال در ریاضی

* تأخیر فاز در گیرنده به مانند تأخیر زمانی است. (یعنی همان اثر را در گیرنده می گذارد.)

* جمع شدن مولفه های فرکانسی طیف فوریه سیگنال $x(t)$ در گیرنده، به سبب اختلاف فازهای مختلف در هر مولفه (فرکانس های مختلف) سبب اعوجاج در سیگنال دریافتی می شود.



شکل (86)

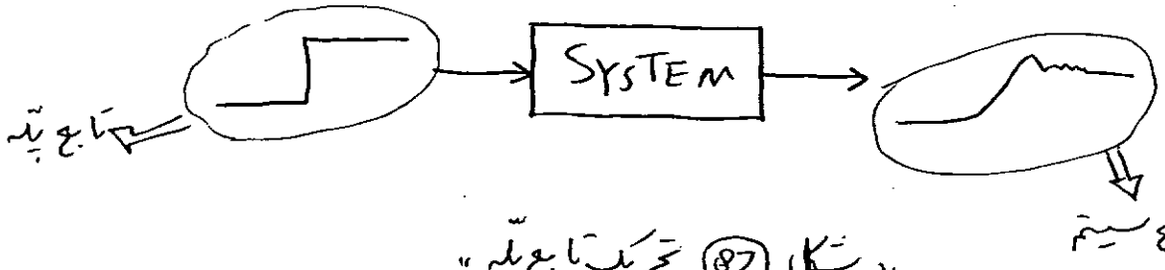
⇐ اختلاف فاز برای فرکانس های متفاوت مولفه‌های سیگنال، سبب اعوجاج سیگنال دریافتی است.

مشخصه کانال:

ب - مشخصه میدان زمان

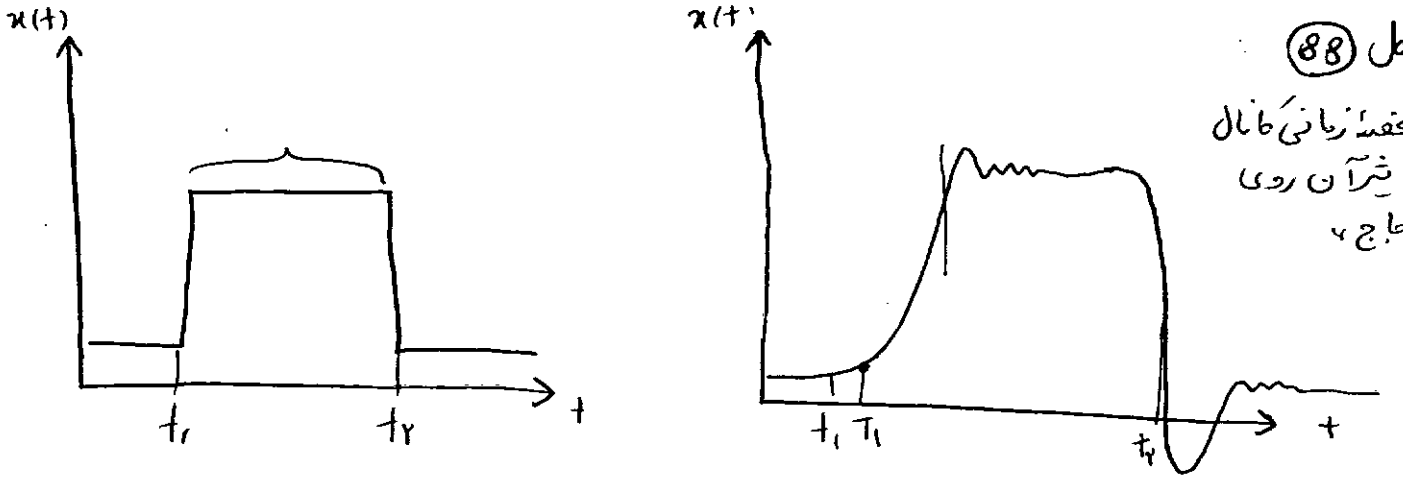
TIME DOMAIN CHARAC.

* بررسی خروجی سیستم با تحریک ورودی توسط تابع پله.



« شکل (87) تحریک تابع پله »

« نگفتم از غم دل راه بوستان گیرم گدام سرو به بالای دوست مانند است؟ »



شکل (88) مشخصهٔ زمانی کانال و تأثیر آن روی اعوجاج

* مشخصهٔ زمان برای کانال پارامترهای سیگنال دریافتی – زمان تأخیر، زمان Rise – ... است.

* با کاهش فاصلهٔ t_1 و t_2 در تابع تحریک پله‌ای سیگنال خروجی بسیار دارای اعوجاج می‌شود.
* اگر سیگنال پریودیک با پالس‌های باریک داشته باشیم، مشخصهٔ زمان کانال در انتقال صحیح آن نقش اساسی دارد.

به پدیده درهم‌رفتن سیگنالها Symbol Interference می‌گوییم.

Symbol Interfer ⇦4 در اعوجاج سیگنال دریافتی نقش اساسی دارد.

⊕ خلاصهٔ مشخصات کانال: (پارامترهای اثرگذار روی نرخ بیت کانال)

⇦4 ۱ – محدودیت پهنای باند
⇦ ۲ – طول خط (هرچه بیشتر شود، سیگنال تضعیف می‌شود)
⇦ ۳ – اختلاف فازها (در مولفه‌های فرکانسی سیگنال)
⇦ ٤ – مشخصهٔ زمانی و پدیدهٔ Inter Symbol Interference

* در جلسهٔ آینده، مشخصات کانال با مثالهای عددی بررسی می‌شود.

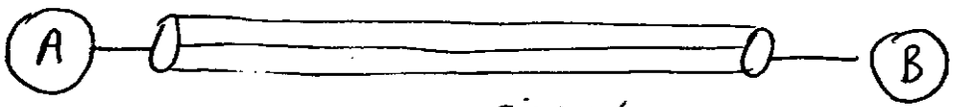
سلام، خسته نباشید لطفاً کمی به من استراحت بدهید ؛): متشکرم، خودکار یک متشکلی رنگ.

★ مشخصات کانال انتقال در چگونگی سیگنال ورودی تأثیرگذار است.

⊕ مثال: «بررسی مشخصات کانال انتقال با مثالهای عددی معین»

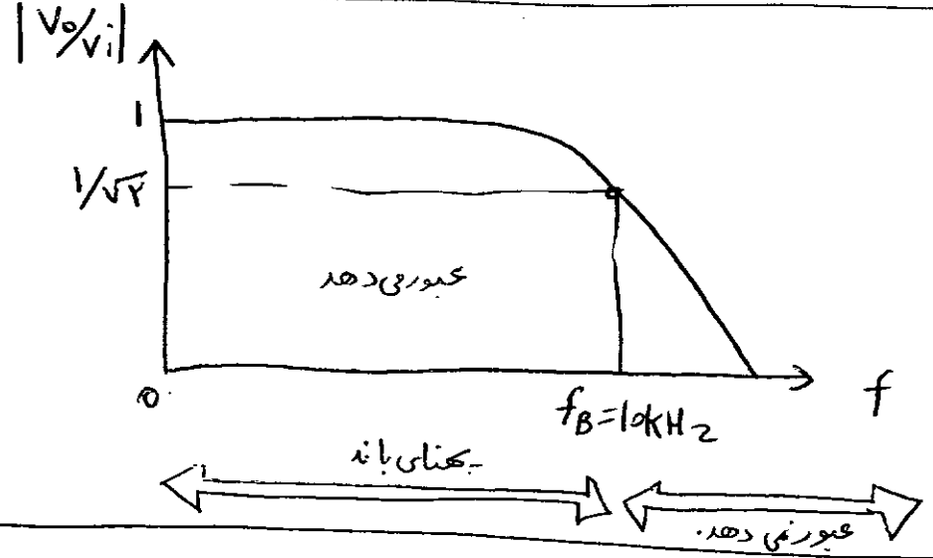
کانال انتقالی داریم با مشخصه‌های زیر:

از فرکانس ۰ تا $10KH_z$، V_0/V_i تقریباً تضعیف نمی‌شود و از آن به بعد به شدت تضعیف می‌شود



کانال انتقال

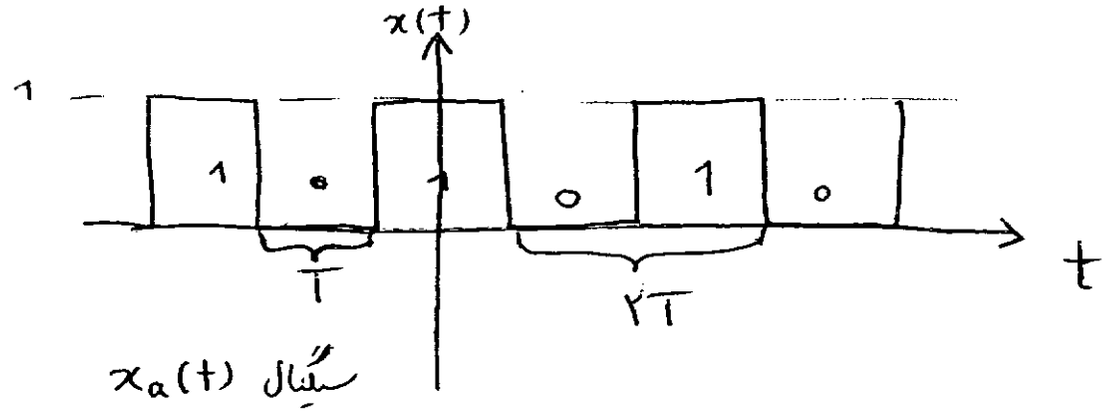
شکل (89) کانال مورد بحث (پهنای باند ۱۰ کیلوهرتز)



شکل (90)

طیف سیگنالهای عبوری — پهنای باند با در نظر گرفتن همه‌ی فاکتورها (فاصله، اختلاف فاز...)

★ سیگنال مفروض $x_a(t)$ از کانال در حال گذر است (فرستنده آن را فرستاد)



شکل (1) سیگنال $x_a(t)$

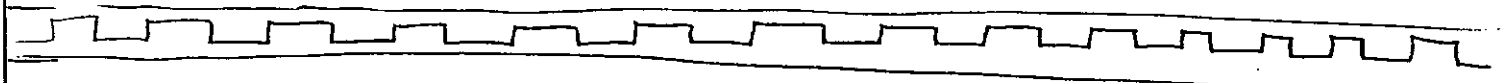
★ سیگنال را رشته‌ای از بیتها در نظر می‌گیریم.

★ طول هر بیت را T در نظر می‌گیریم در نتیجه پریود این سیگنال 2T است.

★ بسط فوریه سیگنال $x(t)$:

$$x(t) = \frac{1}{2} + \frac{2}{\pi}\cos 2\pi ft - \frac{2}{6\pi}\cos 2\pi(3f)t + \frac{2}{10\pi}\cos 2\pi(5f)t - \cdots$$

$$f = \frac{1}{\text{پریود}} \Rightarrow f = \frac{1}{4T}$$



$T = 1\,ms$

⑨ فرض اول:

فرض کنیم طول هر بیت 1×10^{-3} ثانیه باشد.

$$x(t) = \tilde{f}_0 + \tilde{f}_1 + \tilde{f}_2 + \cdots$$

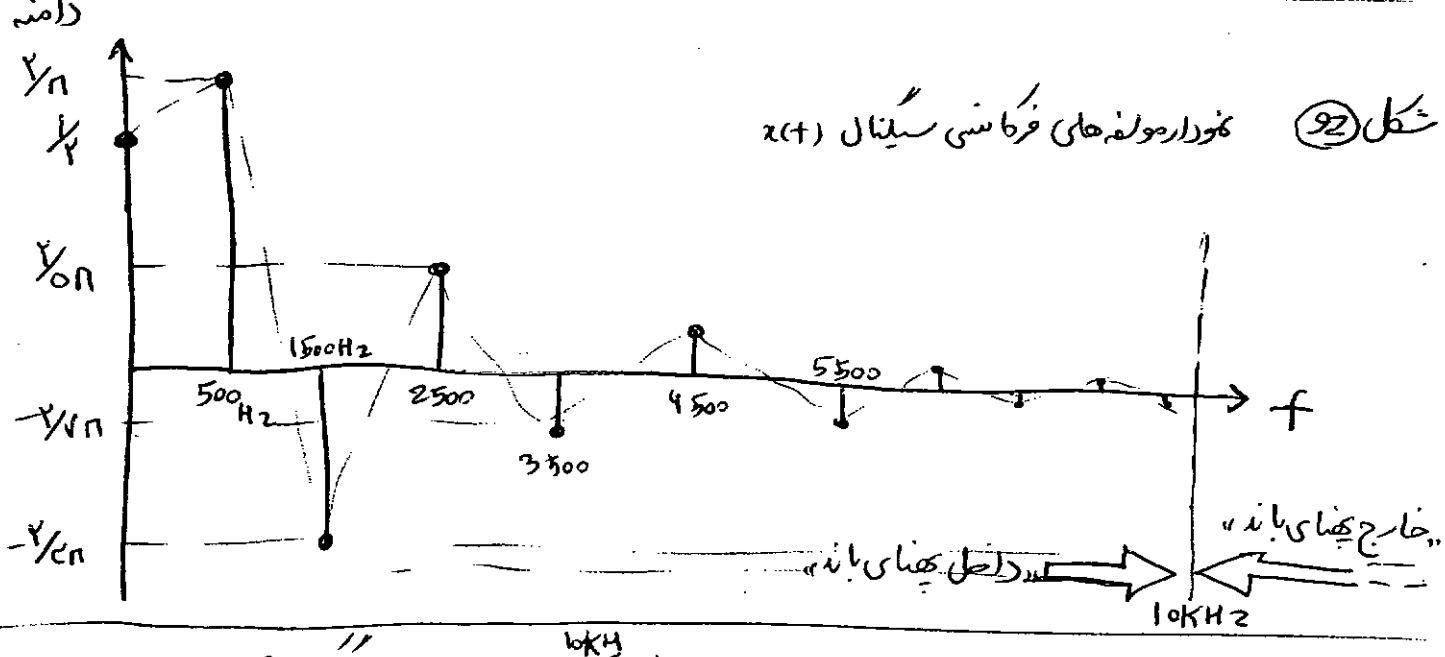
$$x(t) = \frac{1}{2} + \frac{2}{\pi}\cos 2\pi f_1 t - \frac{2}{6\pi}\cos 2\pi f_2 t$$

$$f_1 = \frac{1}{2\times1\times10^{-3}} \Rightarrow f_1 = 500\,Hz$$

$$f_2 = \cdots \rightarrow f_2 = 1500\,Hz$$

$$f_3 = \cdots \rightarrow f_3 = 2500\,Hz$$

$$f_4 = \cdots \rightarrow f_4 = 3500\,Hz$$



شکل (92) نمودار مولفه‌های فرکانسی سیگنال $x(t)$

$y(t) = x(t)$ جمع سیگنال‌های عبوری $= \sum_{f=0}^{10KH}$ مولفه‌های بسط

$T = 1\,ms \Rightarrow y_a(t) \simeq x(t)$

★ درحالتی که $T=1ms$ است تقریباً همه‌ی سیگنال‌های $x(t)$ عبور می‌کنند.

(b) حالت دوم: $x_b(t)$: $T = 0/1\ ms$

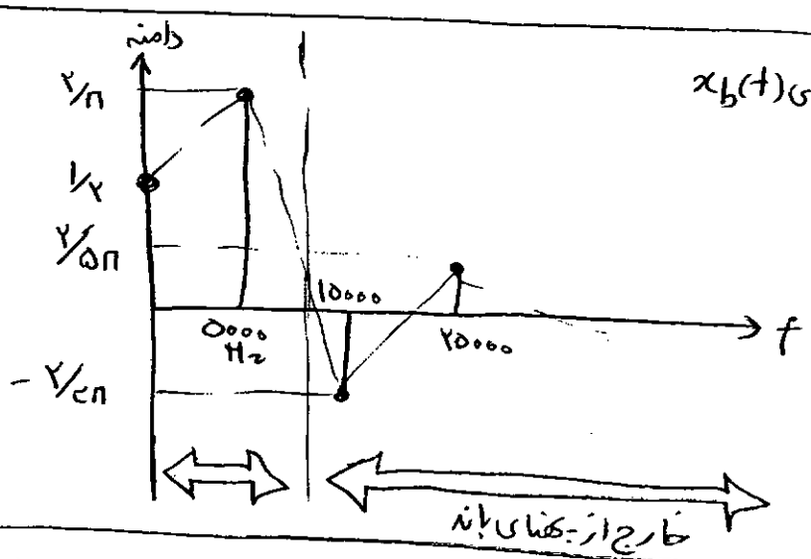
فرض کنیم طول هر بیت $1\times10^{-4}\ s$ است.

$$f_1 = \frac{1}{2\times1\times10^{-4}} \Rightarrow f_1 = 5000\ Hz$$

$$f_2 = \text{---} \Rightarrow f_2 = 10000\ Hz$$

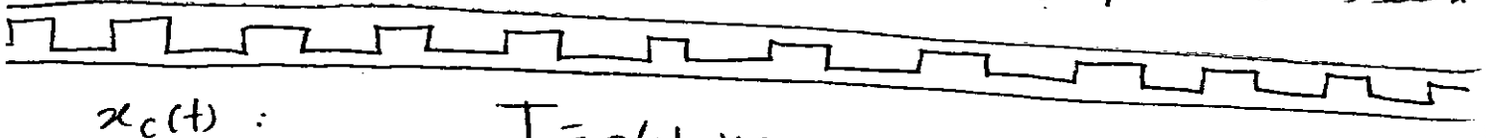
$$f_3 = \text{---} \Rightarrow f_3 = 20000\ Hz$$

شکل (92) مولفه های فرکانسی $x_b(t)$



$$y_b(t) = \frac{1}{2} + \frac{2}{\pi}\cos 2\pi\times5000\times t$$

★ فقط مولفه های اول و دوم عبور پیدا کردند.



$x_c(t)$: $T = 0/01\ ms$

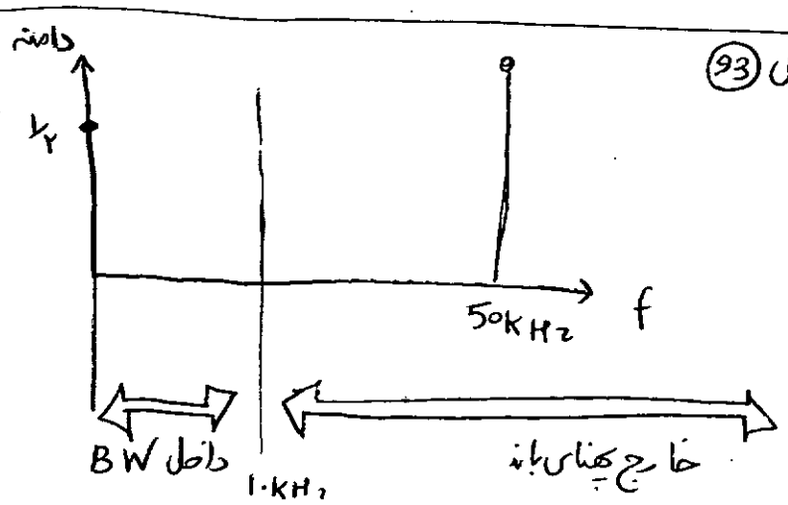
(c) حالت سوم:

فرض کنیم طول هر بیت $1\times10^{-5}\ s$ باشد.

$$f_1 = \frac{1}{2\times1\times10^{-5}} \Rightarrow f_1 = 50\ kHz$$

$$f_2 = \text{---} \Rightarrow f_2 = 100\ kHz$$

شکل (93)



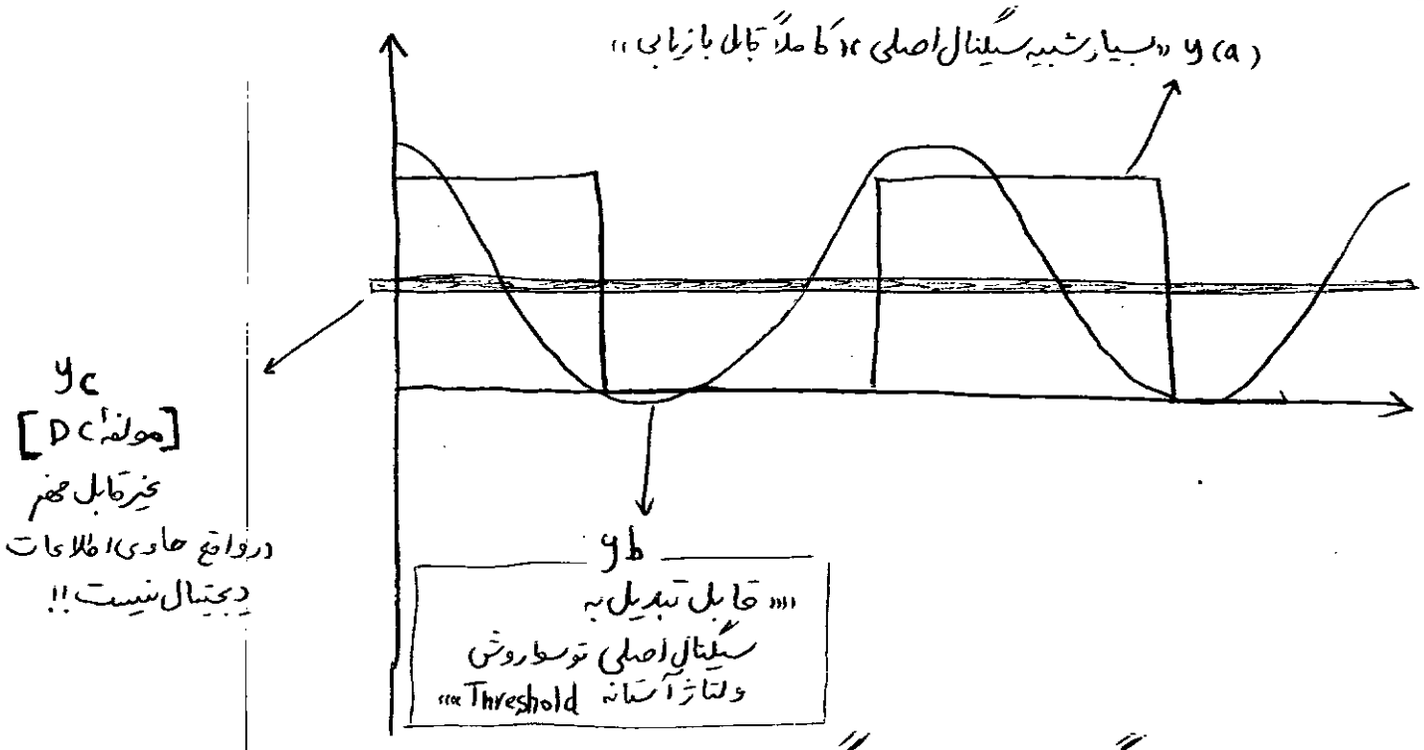
$$y_c(t) = \frac{1}{2}$$

در این حالت فقط مولفه DC از کانال عبور می کند!

حالت ⓐ $y_a(t) \simeq x_a(t)$

حالت ⓑ $y_b(t) = \frac{1}{2} + \frac{2}{\pi} \cos 2\pi \times 5000 \times t$

حالت ⓒ $y_c(t) = \frac{1}{2}$



شکل (۹۶) مقایسه سیگنال دریافتی توسط گیرنده در حالتهای a و b و c (T = ۱/۵۰۰ ms)

★ دیدیم که مشخصهٔ کانال در اعوجاج سیگنال ارسالی نقش دارد.

★ اگر در حالت b فرستنده‌ای سیگنال را روی کانال بفرستد، گیرنده می‌تواند سیگنال را به روش ولتاژ آستانه THRESHOLD بازسازی کند.

★ در حالت c هیچ کاری برای بازسازی x(t) نمی‌توان انجام داد.

«نتیجه» ⇐ اگر مولفهٔ اول فرکانسی سیگنالی ـ دیجیتال ـ از کانالی عبور کند، در طرف گیرنده می‌توان به روش ولتاژ آستانه Threshold سیگنال اولیه را بازسازی کرد.

در مثال قبل اگر مولفه اول فرکانسی سیگنالی کوچکتر از 10kHz باشد، سیگنال قابل عبور از کانال است.

$$\left.\begin{array}{l} T = 0/5\ ms \\ f_1 = 10\ kHz \end{array}\right\}$$ حداقل طول هر بیت

$T = 0/5 ms$

$\Rightarrow$ نرخ بیت $= 20\ kbit/Sec = R$

$\Leftarrow$ نرخ بیت ۲۰ هزار بیت در ثانیه حداکثر نرخ قابل عبور از کانال مثال قبل است.

برای یک کانال انتقال که حداکثر فرکانس عبوری از آن f_w باشد، حداکثر نرخ عبوری از آن برابر است با:

Nyquist Signaling Rate

$$R_{Max} = 2 \times f_w = 2 \times Band\ Width$$

رابطه (11)

که آن را نرخ سیگنالینگ نایکوییست می گویند.

---

* با افزودن تعداد سطوح (از یک و صفر به چهار سطح) می توان با هر سطح دو بیت را نمایش داد.

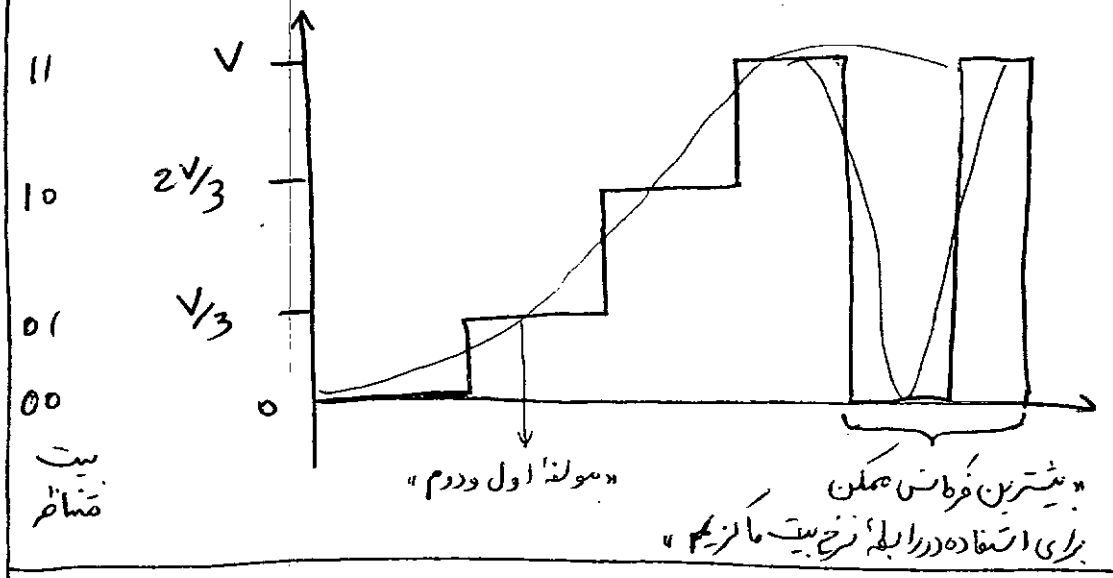
* با گذراندن مولفه اول می توان این سیگنال را هم بازسازی کرد.

* پریود این سیگنال برابر با چهار بیت است

$$f = \frac{1}{4T}$$ رابطه (12)

شکل (96)

«چهارسطح به جای دوسطح ولتاژ»



* نتیجه: برای حالتی که چهارسطح داریم، اگر کانال دارای پهنای باند f_{BW} باشد داریم:

نرخ گذر بیت علایم (بیشینه) $R_{Maxbit} = 4 \times f_{BW}$ bit/sec

رابطه (13)

⊛ تعمیم: برای حالتی که 2^n سطح ولتاژ داریم:

نرخ بیت $R_{max} = 2 \times n \times f_{BW}$ bit/sec

رابطهٔ (14)

رابطهٔ (15) اگر M سطح داریم: «$R = 2 \cdot f_{BW} \cdot \log M$» بیت بر ثانیه

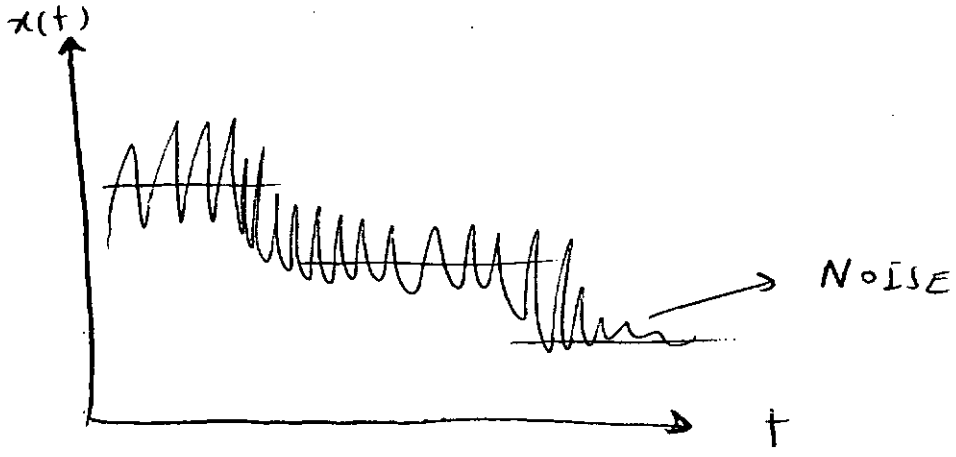
رابطهٔ (16) $\text{baud rate} = 2 \times f_{BW}$

baud rate ⟸ تعداد علایم عبوری در هر ثانیه. (با تعداد سطوح رابطه‌ای ندارد. (واستگی ندارد.))

⊛ محدودیتی برای M از لحاظ تئوریک وجود ندارد.

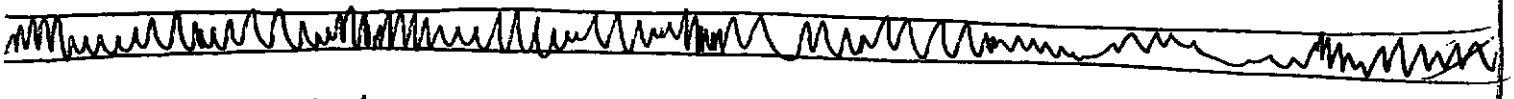
عاملی که باعث محدودیت M در عمل می‌شود، نویز NOISE است که کار بازسازی سیگنال اولیه را - وقتی تعداد سطوح زیاد و سطوح به هم نزدیکند - مشکل می‌کند.

⊛ برای جداسازی M سطح ولتاژ به $M-1$ ولتاژ آستانه نیاز داریم.



سطوح نزدیک به هم

شکل (97) کار بازسازی در این حالت دچار اختلال می‌شود. (به دلیل نزدیک بودن انرژی نویز و سیگنال)



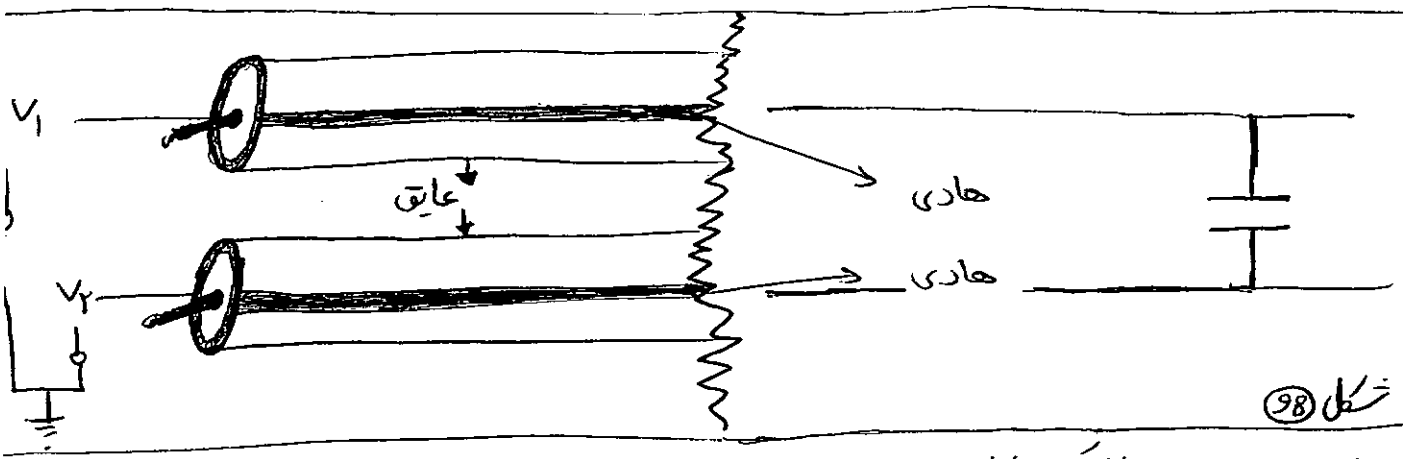
NOISE نویز:

⊕ تعریف: انتشاشات، سیگنالهای ناخواسته‌ای که سوار سیگنال اصلی می‌شوند.

⊕ منشاء های نویز: NOISE RESOURCES

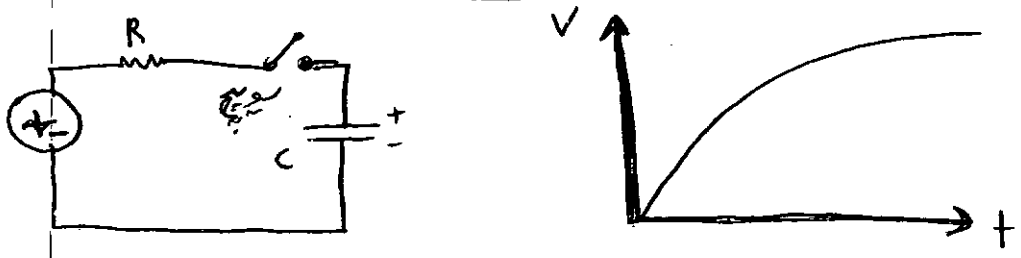
(A) نویز CROSS-TALK:

دو سیم موازی را در نظر بگیرید:



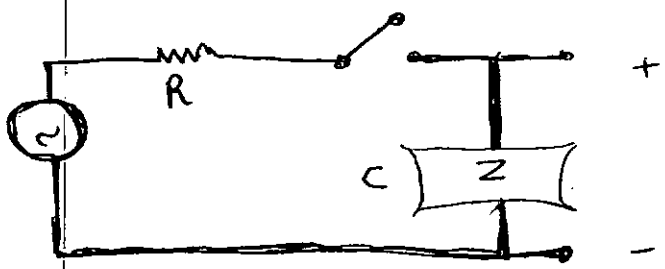
شکل (98)

* این وضعیت به مانند یک خازن است.
* خازن در سیگنالهای AC، امپدانس دارد.
* هادی | عایق | هادی ⇐ خازن



شکل (۹۹) ⊛ انتقال باز در حالت DC

شکل (۱۰۰)



خاصیت امپدانسی در حالت AC

⊛ وجود این وضعیت باعث سوارشدن سیگنالهای سیم اول روی سیم دوم می‌شود.

★ بسته به اندازه و ظرفیت خازن مجازی این نویز تغییر می‌کند.

(B) نویز IMPULSE

★ فرض کنیم کابلی داریم که در آن سیگنالهای دیجیتال جریان دارد.

★ اگر در اطراف این کابل جریانهای الکتریکی تغییر ناگهانی کنند، در محیط ایجاد میدان الکتریکی و القاء جریان در سیمها می‌شود.

★ فرض کنیم رعد و برقی به مدت ۰/۱ میلی ثانیه روی خطوط شبکه جریان زیادی القا کند.

★ فرض کنیم سرعت انتقال ۱۰ Mb/sec باشد.

★ این رعد و برق ۱۰۰۰ بیت در حال انتقال - را از بین می‌برد.

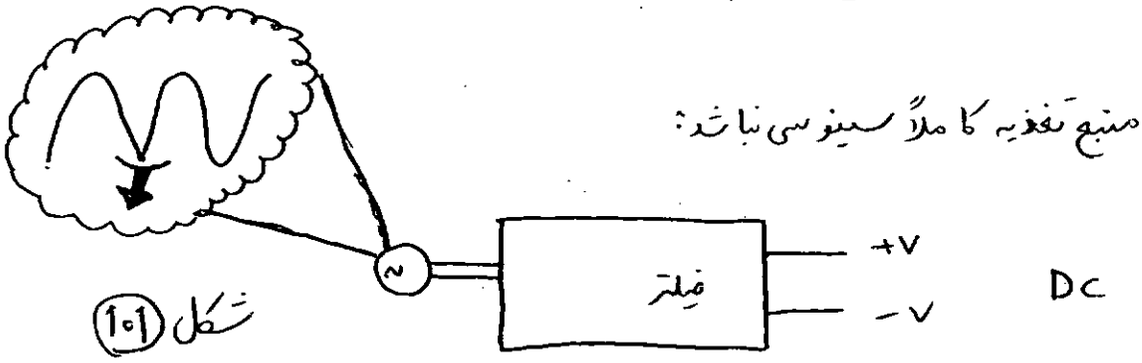
★ نویز Impulse گاهی رخ می‌دهد،

قسمتی از اطلاعات را از بین می‌برد،

ولی به باقی اطلاعات کاری ندارد.

★ بسته به شدت جریان القایی میزان اطلاعات از بین رفته، تغییر می‌کند.

© نویز PHASE JITTER :



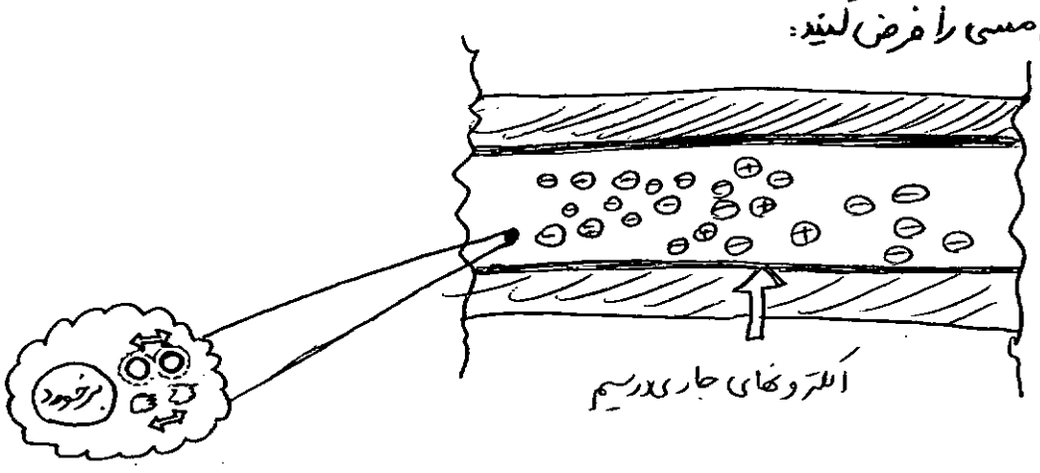
شکل (101)

* اگر شکل موج منبع تغذیه کاملاً سینوسی نباشد:

* طبق بحث بسط فوریه اگر سیگنال ورودی کاملاً سینوسی نباشد، علاوه بر مقدار DC ، نویز دیگری- ناشی از مولفه‌های عبوری از فیلتر - روی جریان خروجی سوار می‌شود.

(D) نویز حرارتی : THERMAL NOISE:

⊛ ساختار داخلی یک کابل مسی را فرض کنید:



شکل (102)

⊛ برخورد نامنظم الکترونها در اثر حرارت باعث ایجاد جریان ونویز می‌شود.

(A) نویز IMPULSE نویزی خارجی است که با حایل بندی خوب حل می‌شود.

(B) نویز CROSS-TALK هم خارجی است و با حایل خوب حل می‌شود.

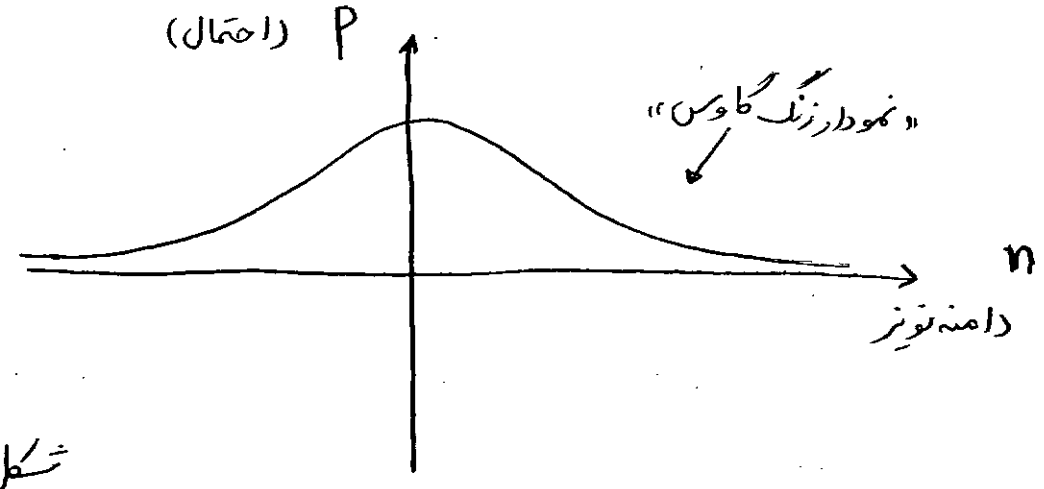
© نویز PHASE JITTER با فیلتر خوب قابل حل است. (منشاء خارجی)

(D) نویز حرارتی منشاء ذاتی دارد و نمی‌توان آن را برطرف کرد.

(€) نویز ناشی از تشعشعات کیهانی هم همراه با نویز حرارتی بحث می‌شوند.

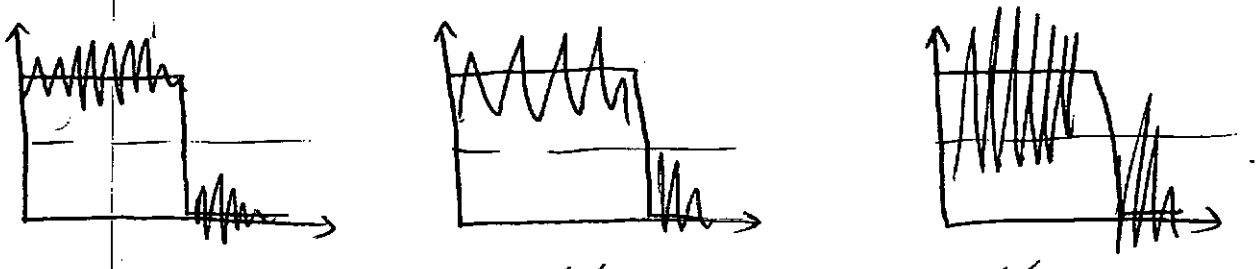
* از این پس فقط روی نویزهای حرارتی بحث می‌کنیم:

* نویز حرارتی کاملاً تصادفی است. به همین جهت با توابع احتمالاتی روی آن بحث می‌شود.



شکل (103)

* با توجه به اینکه برای نویز فرمول خاصی نداریم، روی انرژی آن بحث می‌کنیم. (N)



شکل (104)

* پارامتر انرژی سیگنال به انرژی نویز: (توان متوسط سیگنال به توان متوسط نویز)

$$S/N = \text{Signal To Noise Ratio}$$

* رابطهٔ شانون: (رابطهٔ (17))

اصلاح رابطهٔ $R = 2f_{BW} \cdot \log_2^M$ به صورت:

$$C = f_{BW} \cdot \log_2 (1 + S/N)$$

ظرفیت خط — در محیط‌های نویزی

در این رابطه تمامی پارامترهای ذکر شده واگذار روی نرخ بیت ظاهرند.

* واحد S/N را معمولاً دسی بل در نظر می گیرند.

$$(S/N)_{dB} = 10 \log_{10} S/N$$ رابطهٔ (۱۸)

⊙ مثال: فرض کنیم حداکثر مولفهٔ فرکانسی در خطوط تلفن 3400 هرتز است.
(پهنای باند آن از ۰ تا 3400 هرتز است!)
اگر در این کانال نرخ سیگنال به نویز ۴۰ دسی بل باشد. حداکثر نرخ قابل عبور از این کانال چه قدر است؟

$f_{B.W} = 3400\ Hz$
$(S/N)_{dB} = 40\ dB$
$C = ?$

حل:

$$(S/N)_{dB} = 10 \log_{10} S/N = 40$$
$$\Rightarrow S/N = 10000$$

$$C = f_{BW} \times (\log_2(1 + S/N))$$
$$\Rightarrow C = 3400\ Hz \times \log_2(1 + 10^4)$$
$$\Rightarrow \boxed{C = 44.8 \times 10^3\ b/sec}$$

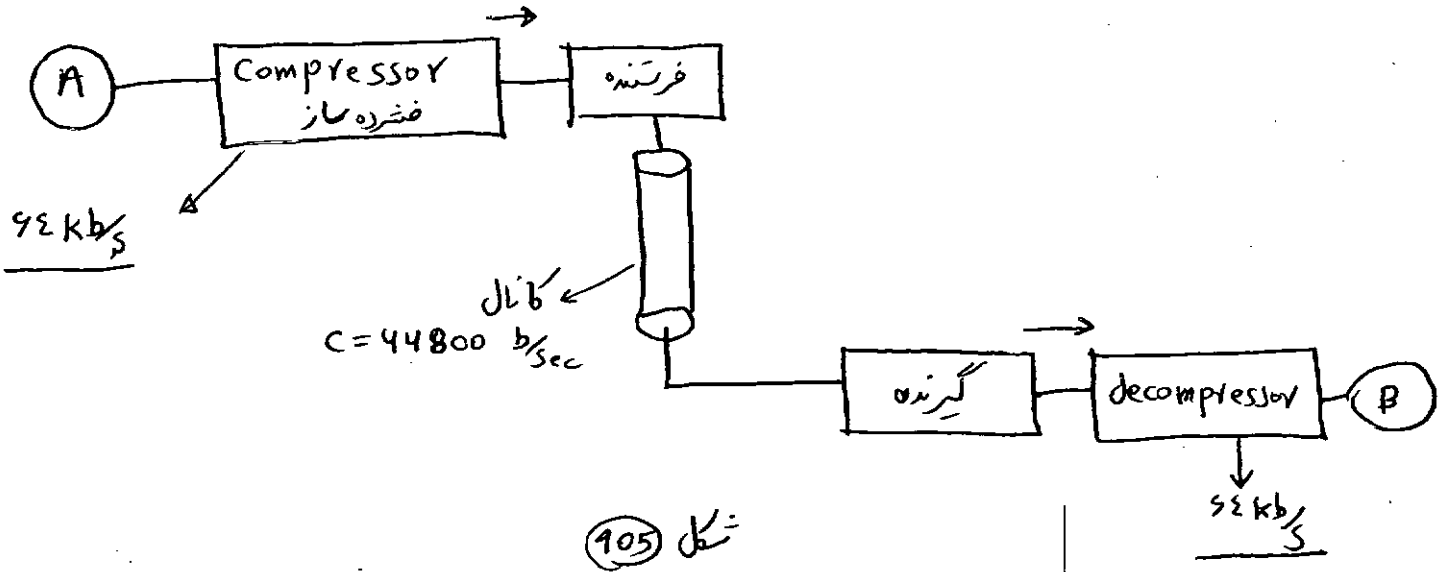
با استفاده از چندین سطح ولتاژ

⊕ فشرده سازی اطلاعات : COMPRESSION:

* فرض کنیم فایلی داریم شامل ۱۰۰۰ کاراکتر A پشت سر هم!!
* به جای ارسال کل (واقعی) فایل می توانیم بگوییم A و ۱۰۰۰ را بفرستیم!
* یعنی به جای ۷۰۰۰ بیت در این مثال از ۳۵ بیت استفاده می شود

عمل decompression عکس عمل Compression است.

⊕ مثال: فرض کنیم کانالی داریم با پهنای باند 3400 هرتز و با S/N ، 40 dB مانند مثال قبل.



شکل (905)

★ اردیب A و B ارتباط با سرعت 64 Kb/s انجام می‌شود. درحالیکه در کانال این نرخ کمتر است.

⇐ فشرده سازی هم عامل دیگری در نرخ بیت (اردیب فرستنده و گیرنده) است.

خلاصه: پارامترهای اثرگذار در نرخ بیت:

S ↑	→	C ↑
N ↓	→	C ↑
f_w ↑	→	C ↑
Compression ↑	→	C ↑
سطوح ولتاژ S/N ↑↑	→	C ↑

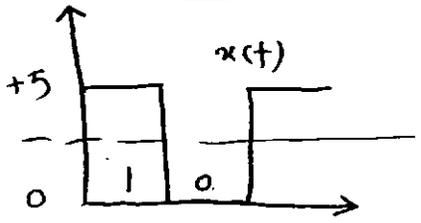
روابط (19)

LINE CODING : کدینگ خط :

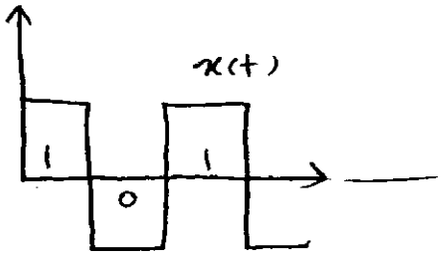
* روش تبدیل اطلاعات باینری به علایم دیجیتال

Binary Data ⇒ Line Coding ⇒ Digital Signals

⊕ مقایسه دو روش :

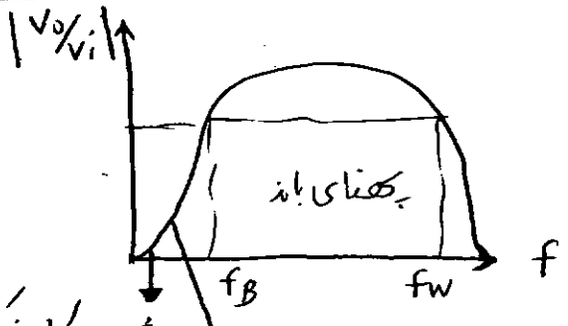


a) ۰ باینری را با ولتاژ ۰
۱ باینری را با ولتاژ +5V نشان می دهیم.



b) ۰ باینری را با -5V و
۱ باینری را با +5V نشان می دهیم

* در روش اول مقدار متوسط موج سیگنال - ۲٫۵ ولت است. که می تواند باعث اتلاف حرارتی و ... شود. ضمناً مولفهٔ DC این سیگنال ۲٫۵ ولت است و در کانالهایی که فرکانسهای کم را از خود عبور نمی دهند این روش دچار اخلال می شود.

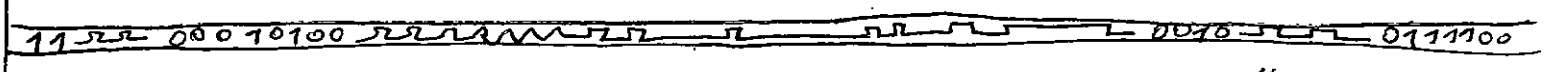


شکل (106)

کانالی که فرکانس های کم - مولفهٔ DC - را از خود عبور نمی دهد.

روش a کار نمی کند و دچار اعوجاج می شود. ولی روش b چون مولفهٔ DC ندارد به راحتی و بدون اعوجاج گذری کند.

روش کدینگ نقش زیادی در کارآیی انتقال دارد.



روشهای کدینگ : (از لحاظ تعداد سطوح برای نمایش ۰ و یا ۱)

Full Binary ⓐ

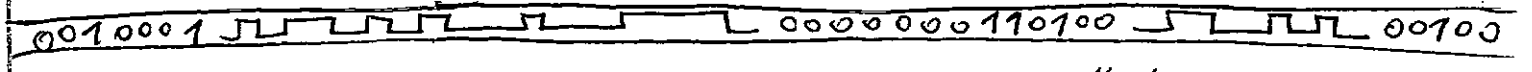
+ برای ۰ و ۱ یک ولتاژ معین داریم. (یا یک روش (لبه) معین)

Half Binary ⓑ

+ فقط برای یکی ـ ۰ یا ۱ ـ ولتاژ مشخص می‌کنیم. (یا لبه)

Multiple Binary ⓑ

+ استفاده از چند سطح ولتاژ برای ۰ یا ۱ یا هردو.



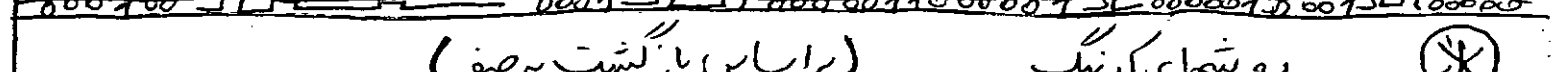
روشهای کدینگ (براساس پولاریته)

UNi POLAR Ⓐ

+ فقط در یک سمت سطح ولتاژ داریم ـ مثبت یا منفی ـ

Bi POLAR Ⓑ

+ سیگنال در هر دو جهت مثبت و منفی جریان دارد.



روشهای کدینگ (براساس بازگشت به صفر)

Return to Zero RZ ⓐ

+ هر بیت در طول زمان خودش به سطح صفر باز می‌گردد.

Non Return 2 Zero NRZ ⓑ

+ بیت ها به صفر برنمی‌گردند ـ در طول زمان بیت ـ

⊕ در روشهای اول و دوم (مثال ⊛ و مثال ⊛⊛) اگر تعداد زیادی ۱ پشت سر هم بیایند گیرنده ممکن است به این شک کند که خط قطع شده است، ولی در روش سوم (مثال ⊛⊛) این مشکل را نداریم !!

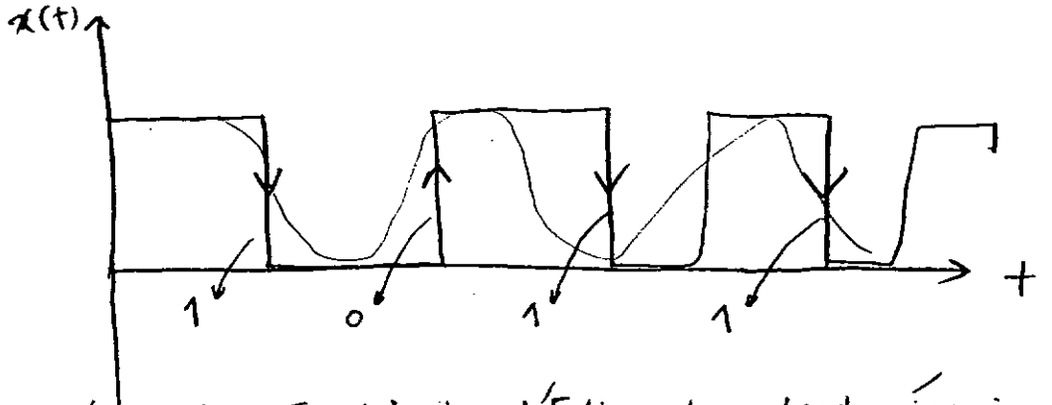
⊛⊛ مثال چهارم : MANCHESTER CODING

★ ۰ها و یک ها نه با سطح که با لبهٔ ولتاژ نشان داده می‌شوند. (حساس به لبه)

Edge Triggered

۱ : لبهٔ پایین رونده

۰ : لبهٔ بالارونده



شکل (۱۱۰)

⊛ این روش در شبکه های محلی Ethernet کاربرد دارد. (در سیستمهای هوایی نیز کاربرد دارد.)

★ این روش : full Binary است.

Bipolar است. (محور + را هرطور در نظر بگیریم)

نصف موج در بالا و نصف آن پایین محور + است.

RZ است :

چون هر بیتی در طول زمانی خودش حتماً به سطح صفر باز می‌گردد.

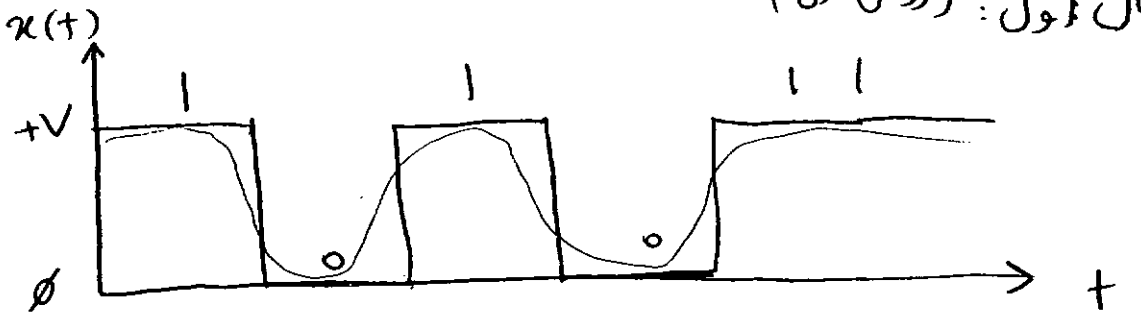
⊕ در روش سوم اگر تعدادی صفر پشت سر هم بیایند باز هم شکل ایجاد می‌شود. (مقطعی خط)

ولی در روش Manchester در هیچ حالتی بجز قطعی خط سیگنال ثابت نمی‌ماند.

⊕ در روش Manchester طول پهنای پالس‌ها نصف می‌شوند.

« پس فرکانس این روش حداکثر ۲ برابر روشهای دیگر است. »

مثال اول: (روش اول)

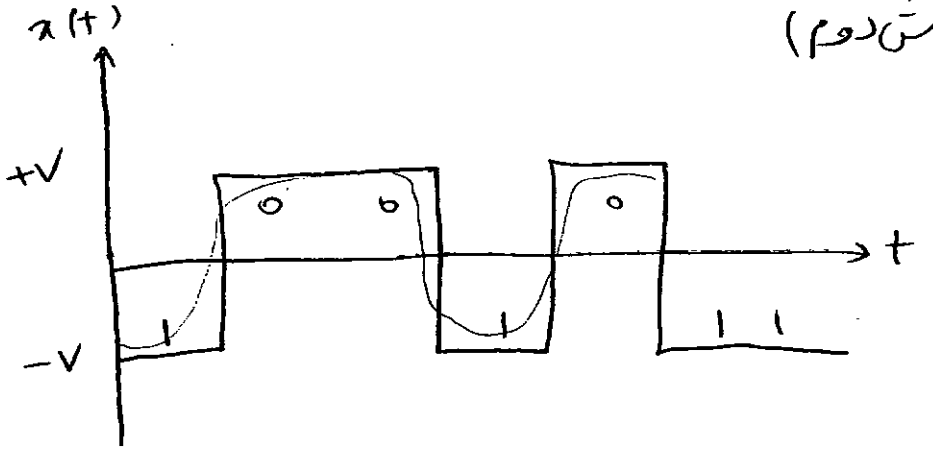


شکل (107)

★ نحوهٔ کدینگ:

full Binary
Uni polar
NRZ

مثال دوم: (روش دوم)

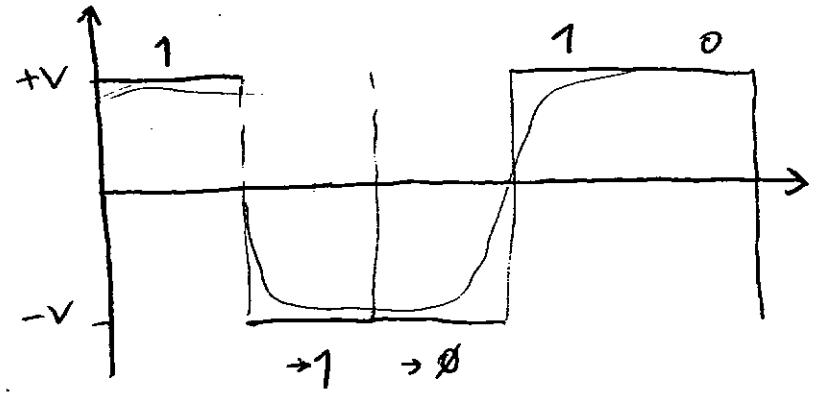


شکل (108)

★ نحوهٔ کدینگ:

full Binary
Bi polar
NRZ

مثال سوم: (روش سوم)



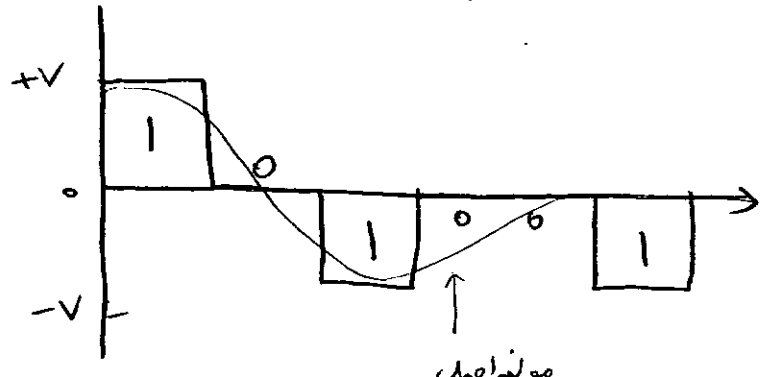
شکل (109)

★ سطوح ولتاژ +V و -V هردو کسر انشان می دهند - به شرطی که 1 بعدی که آمد جهت (±) عوض شود و اگر عوض نشد بیت صفر است. پس صفر خودش سطح ولتاژ ندارد.

Half Binary , Bipolar , NRZ

روش پنجم: Dou-Binary

* سطوح +V و -V برای یک منطقی
* سطح ۰ برای ۰ منطقی
* اگر تعداد صفرهای بین دو یک فرد باشد پولاریته عوض می‌شود، ۱ بعدی در پولاریته بعدی است.

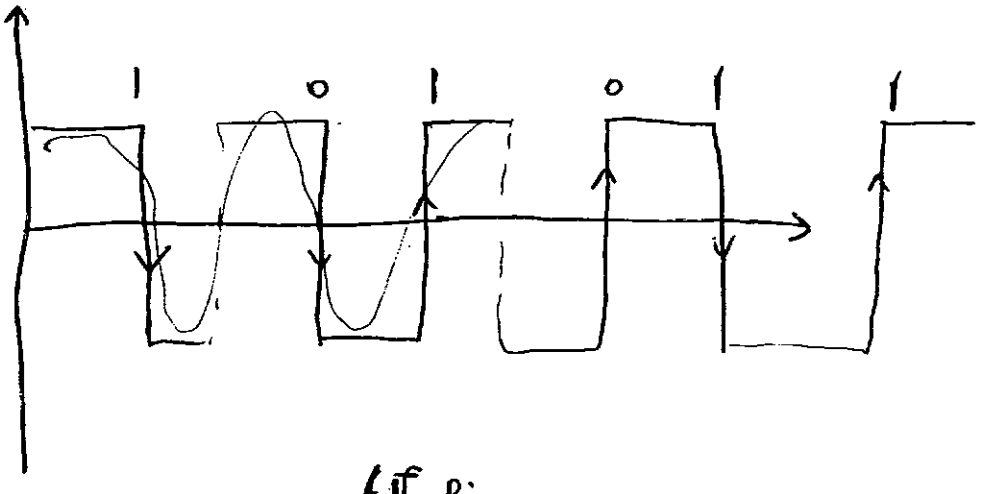


شکل (111)

Muttiple Binary
Bipolar
NRZ

روش ششم: Differential Manchester:

* باز هم با لبه‌ها سروکار داریم.
* اگر بیت بعدی فرق می‌کند - با بیت فعلی - ۰ بعد از ۱ لبه عوض نمی‌شود ولی اگر ۱ باشد لبه عوض می‌شود.
* پالس‌های با رنگ کمتر شدند - فقط وقتی تعدادی صفر پشت سر هم داریم یا صفر بعد از یک.



شکل (112)

half Binary
Bipolar
RZ

⊛ مقایسه مولفه اصلی روشهای مختلف:

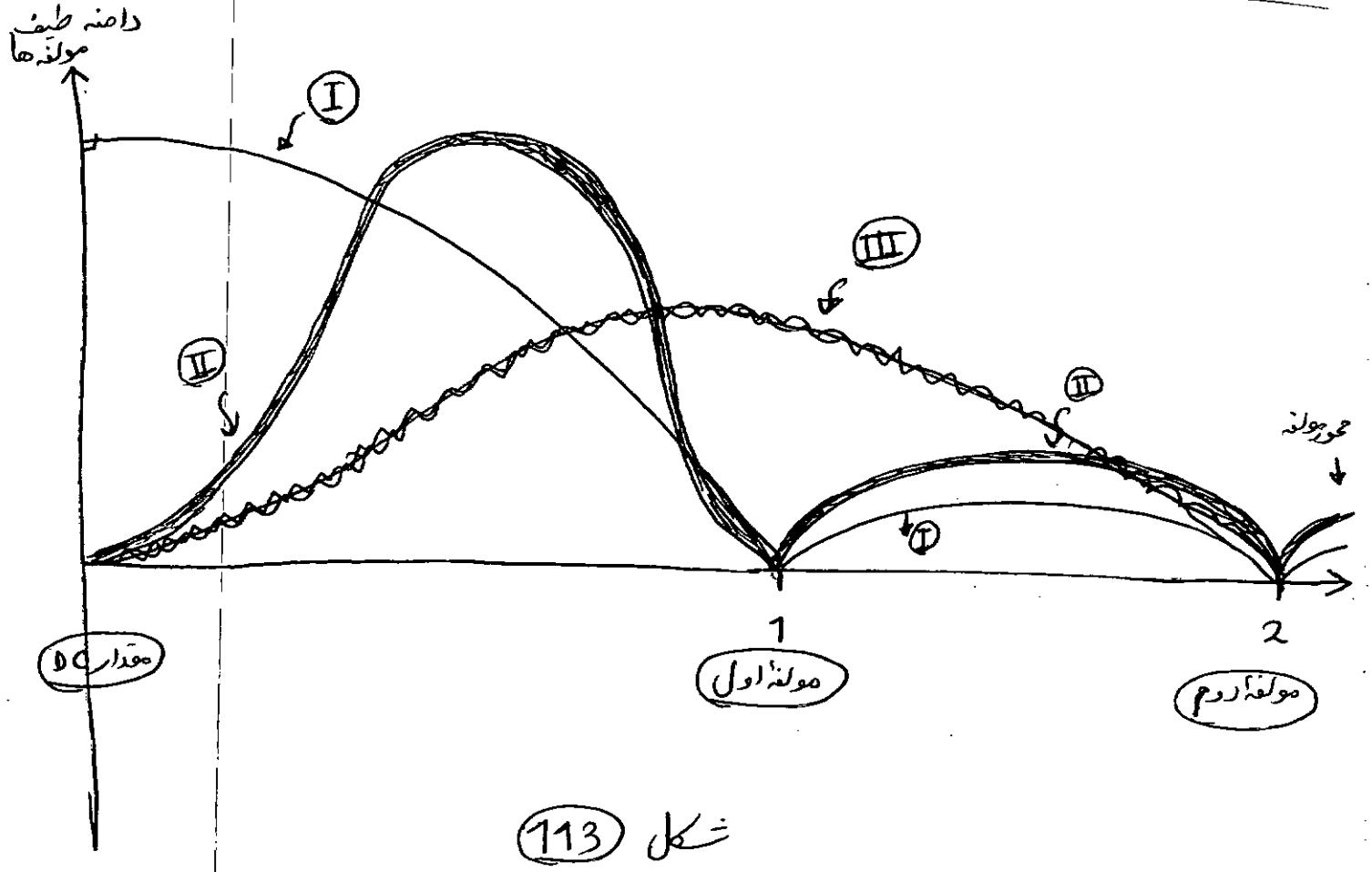
* روش Dou-Binary از همه روشها فرکانس کمتری دارد.
* روش Dou-Binary برای کانالهایی که پهنای باند محدود دارند کاربرد دارد.

مقایسه:

I: روش اول - معمولی - ۰ و ۷+ ولت.
II: روش Dou-Binary
III: روش Manchester

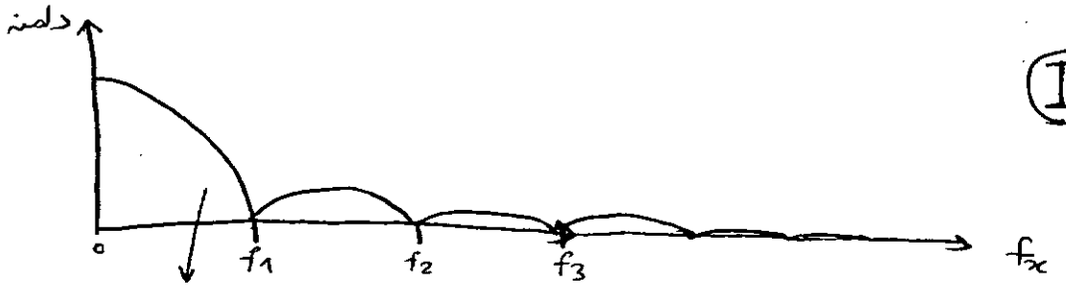
* تولید رشته بیت تصادفی.

و مقایسه مولفه وطیف فرکانسی:



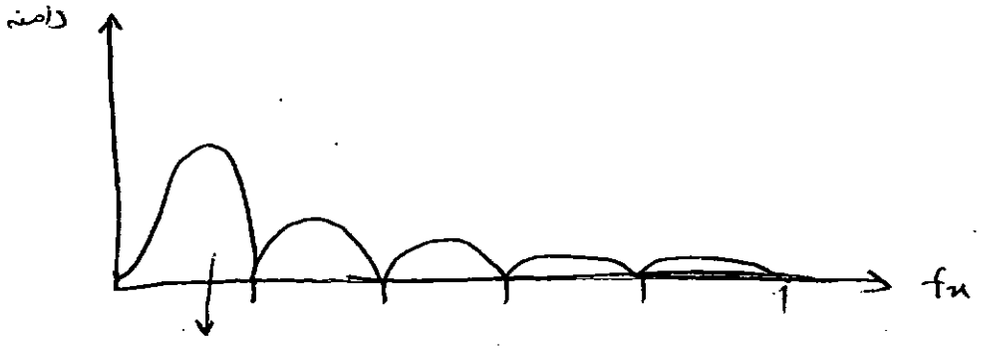
شکل (113)

مقایسه طیف فرکانسی روشهای I و II و III برای سیگنالی تصادفی «Random».



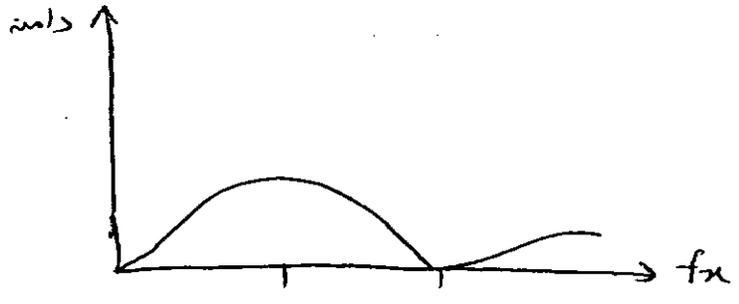
شکل (14)

★ عمده انرژی در مولفه اول است. پس با عبور دادن مولفه اول عمده سیگنال عبور می‌کند.



★ باز هم می‌توان با عبور دادن مولفه اول عمده سیگنال را بازسازی کرد ولی با اعوجاج بیشتر نسبت به روش (I)

شکل (15)



شکل (16)

★ با پهنای باند کیان با روشهای (I) و (II) اعوجاج بسیار زیاد است.

★ برای عبور مناسب به پهنای باند حدوداً دو برابر آنها نیاز دارد.

⇦ با پهنای باند کیان:

روش اول از دوم II بهتر است. اعوجاج کمتری دارد.

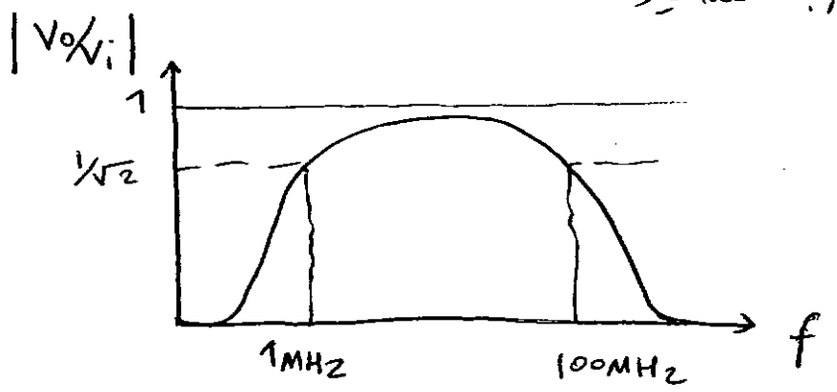
و هردوی I و II از III بهترند.

⊕ تمرینهای فصل سوم:

1, 2, 3, 14, 15, 16, 18, 19, 20, 21

مدولاسیون : MODULATION

- کانال انتقالی داریم با مشخصهٔ زیر:

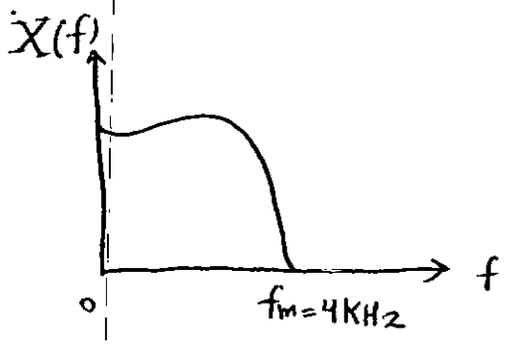


شکل (۱۱۷) مشخصهٔ فرکانسی کانال

پهنای باند

- سیگنالی داریم با مشخصهٔ زیر:

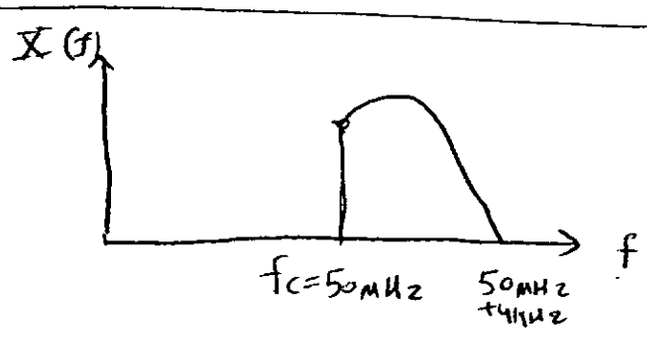
سیگنال صوتی میکروفون



شکل (۱۱۸) مشخصهٔ فرکانسی

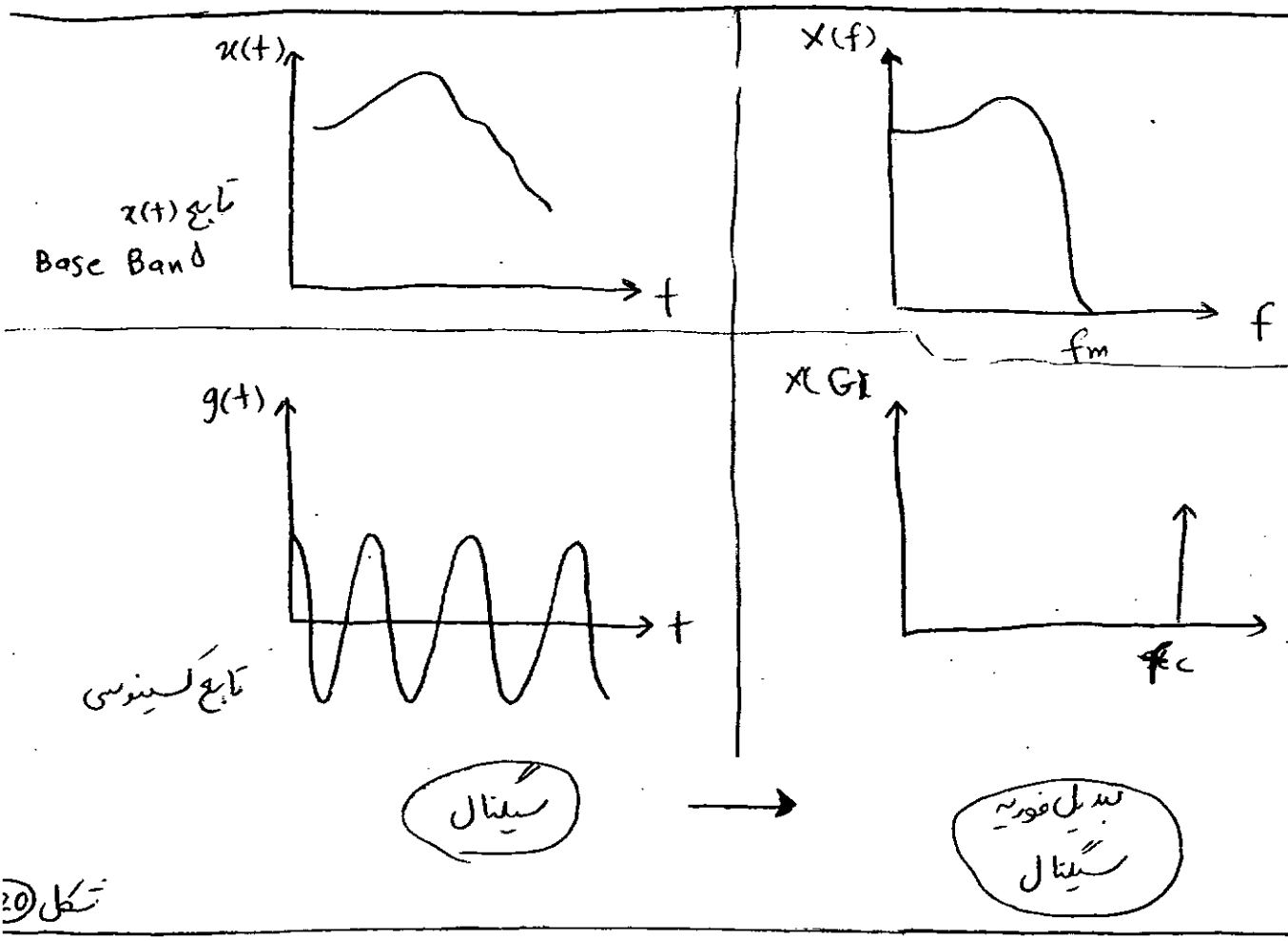
* همانطور که می دانیم سیگنال $x(t)$ از داخل کانال فرببر عبور نمی کند.
* مدولاسیون به ما کمک می کند که سیگنالهایی مانند این را از کانالی مانند بالا عبور دهیم.

سیگنال مدوله شده :



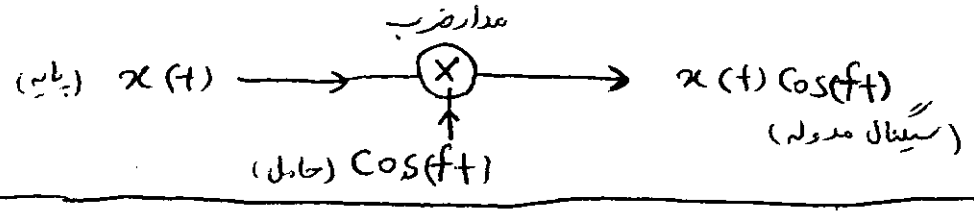
شکل (۱۱۹)

* با انتقال سیگنال می توان آن را از کانال عبور داد.
* مدولاسیون عمل انتقال سیگنال جهت عبور دادن آن از کانالی است که سیگنال در حالت عادی از آن گذر نمی کند.



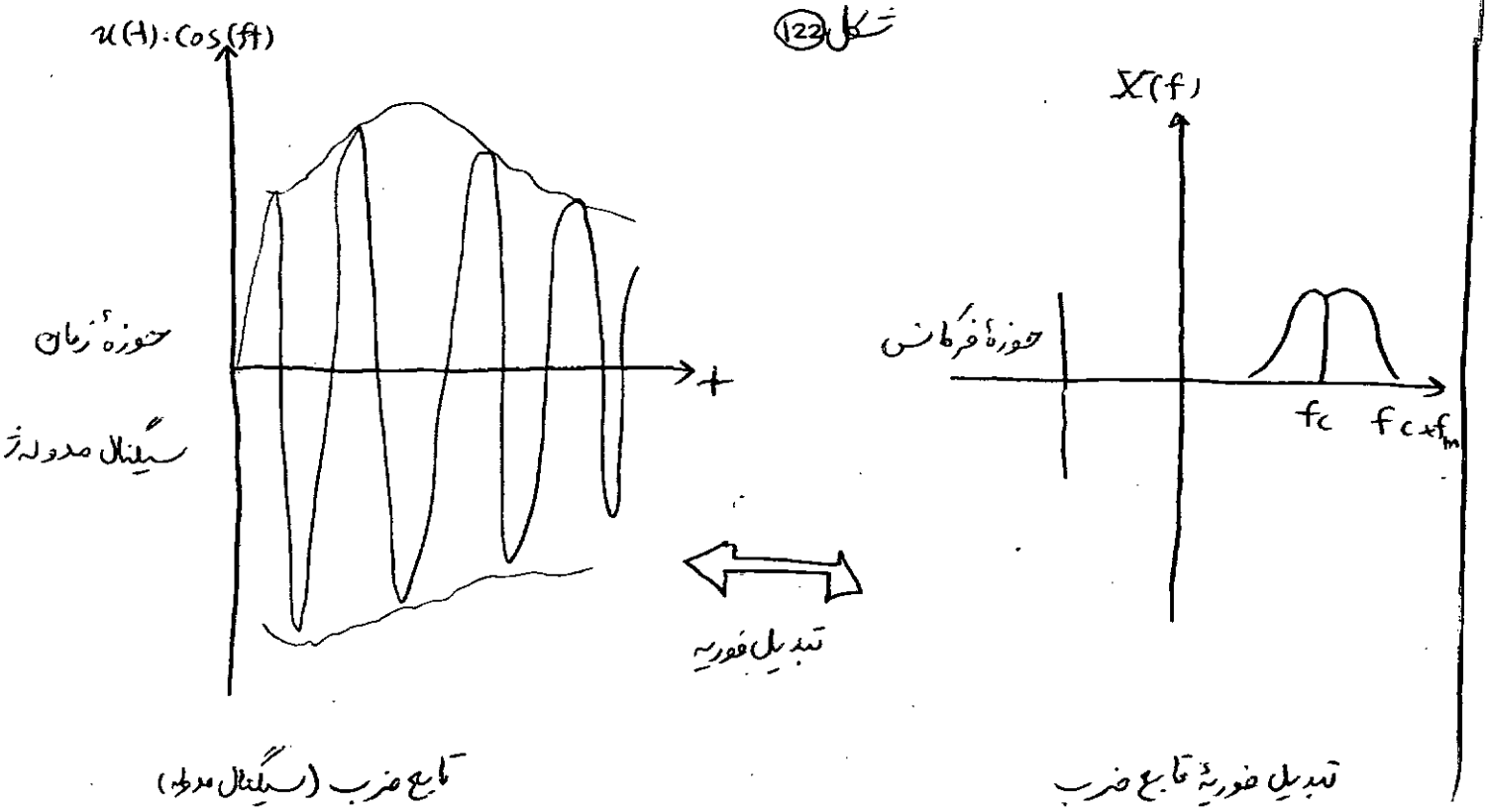
شکل (120)

* تابع x(t) ـ سیگنال ـ را با مدار ضرب کننده در سیگنال کسینوسی ضرب می‌کنیم.



شکل (121)

شکل (122)



* محل ضرب سیگنال در سیگنال سینوسی ، مدولاسیون نام دارد.

⊕ تعریف :

مدولاسیون : انتقال طیف فرکانسی به حول یک فرکانس معین مانند f_c

فواید مدولاسیون :

⊛ 1 - سیگنال $X(f)$ از کانال قابل عبور می‌شود. - مهمترین فایدهٔ مدولاسیون -

⊛ 2 - استفادهٔ بهتر از امکانات کانال انتقال .

درمثال قبل :

از پهنای باند ۹۹ مگاهرتزی تنها ۸ کیلوهرتز آن اشغال شد.

پس با انتخاب f_c های متفاوت می‌توان سیگنال های زیادی را از یک کانال عبور داد. (Simultaneously)

اگر پهنای باند ۹۹ مگاهرتز باشد و سیگنال ها دارای fm ، 4 کیلوهرتز باشند ، با انتخاب f_c های متفاوت می‌توان

$$\frac{BW}{2 \times fm}$$ تعداد

یعنی تعداد 12000 سیگنال را همزمان روی کانال فرستاد.

⊛ 3 - مخابرهٔ راحت‌تر سیگنالها.

⊛ امکان استفاده از آنتن Anthenna

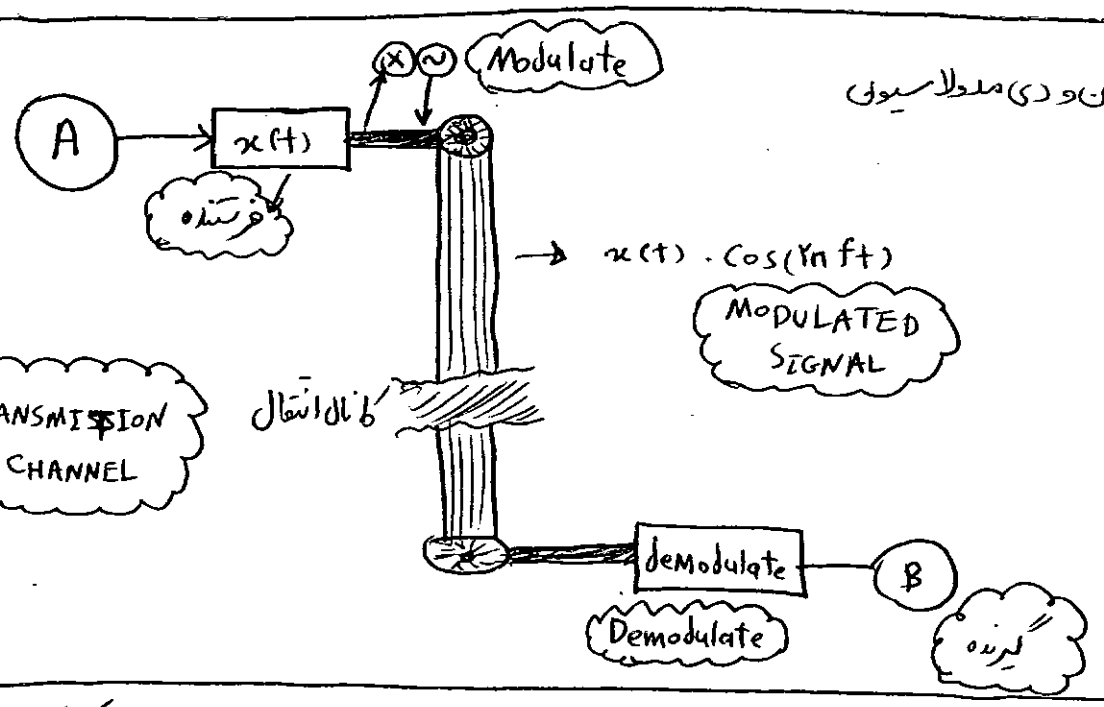
⊛ 4 - امنیت بیشتر در برابر نویزها

* درجلسات بعد مختصراً توضیح داده می‌شود.

⊕ سیگنال باند پایه : BASE BAND

تعریف ← سیگنال اولیه که هنوز مدوله نشده است و احتمالاً غیرقابل عبور از کانال است.

BASEBAND ≡ $x(t)$



شکل (123)

+ مدولاسیون و دی مدولاسیون

* گیرنده برای آنکه سیگنال مدوله شده را به سیگنال اولیه "Baseband" تبدیل کند آن را دوباره در $\cos 2\pi ft$ ضرب می‌کند.

$x(t).\cos 2\pi ft$ ضرب (در فرستنده)

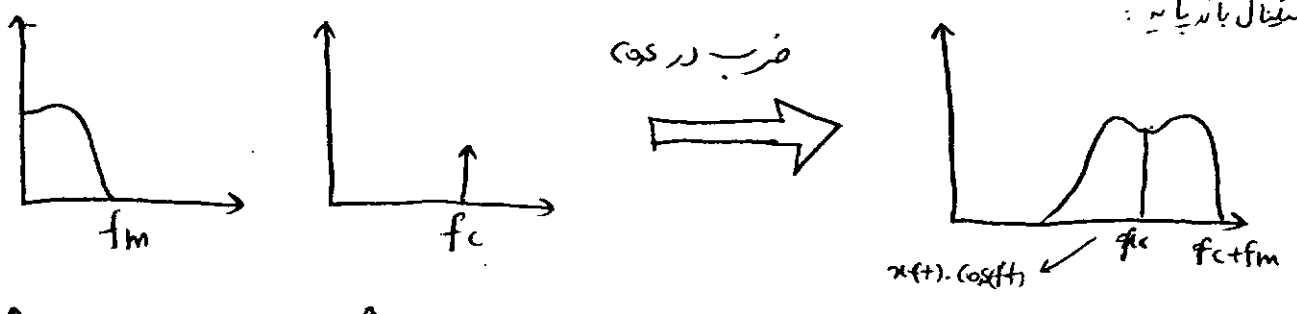
$x(t).\cos(2\pi ft)$ سیگنال دریافتی

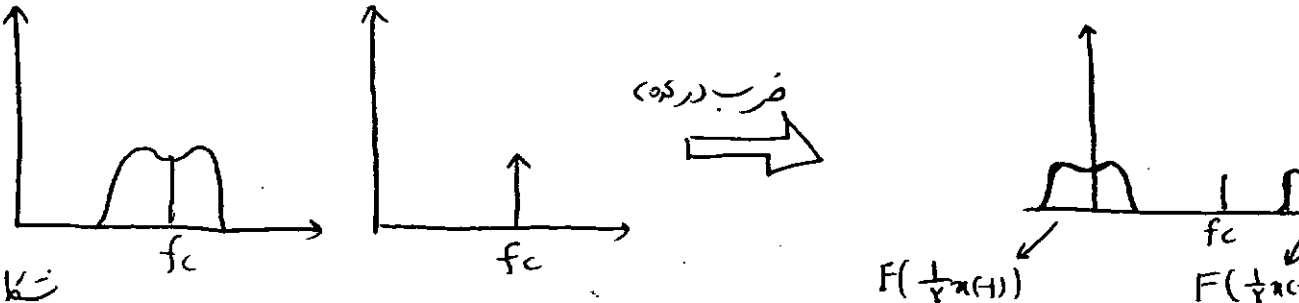
$Y = x(t).\cos^2(2\pi ft)$ ضرب (در گیرنده)

$Y = \frac{1}{2}x(t) + \frac{1}{2}x(t).\cos(4\pi ft)$ → فیلتر کردن

* گیرنده تنها یک سیگنال را می‌تواند از درون کانال بگیرد.
* با ضرب کردن سیگنال دریافتی در فرکانسهای مختلف، گیرنده می‌تواند سیگنال مورد نظرش را انتخاب کند.

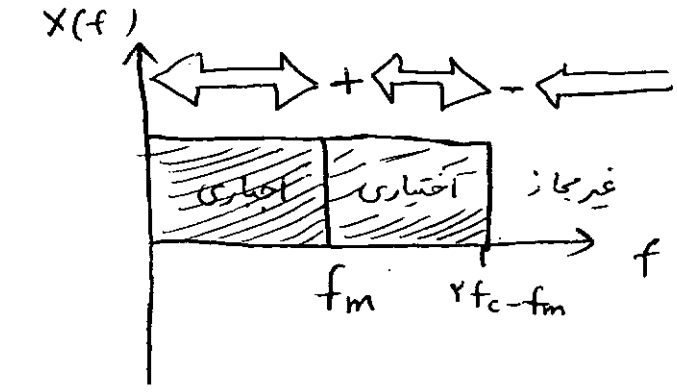
بازیابی سیگنال باند پایه:





شکل (124)

⊕ با عبور دادن سیگنال نهایی ۲ از یک فیلتر مناسب می‌توان $\frac{1}{2}x(t)$ را از بقیه سیگنال جدا کرد.



[پهنای باند فیلتر]

شکل (125) مشخصهٔ فرکانسی فیلتر مناسب برای جداسازی $\frac{1}{2}x(t)$ (در گیرنده). (محدودهٔ مجاز، اجباری و غیرمجاز)

نامساوی محدودهٔ پهنای باند فیلتر برای بازسازی سیگنال با مدولاسیون A.M

$$f_m < f_w < 2f_c - f_m$$ رابطهٔ (20)

⊕ پهنای باند فیلتر مناسب

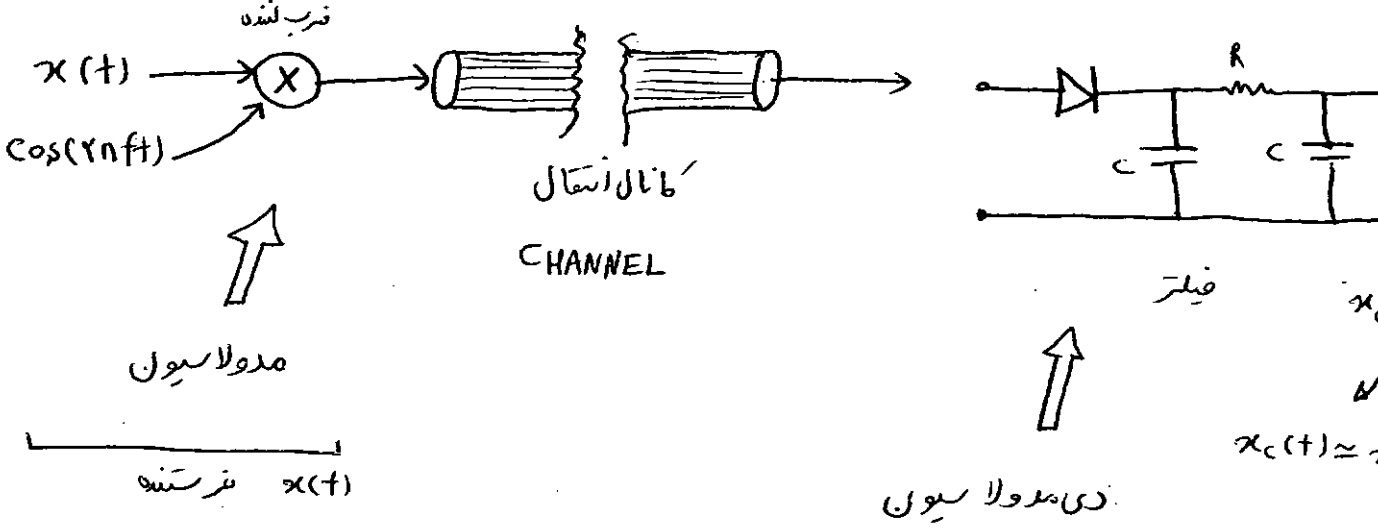
نامساوی محدودهٔ f_c:

$$2f_c \geq 2f_m$$ رابطهٔ (21)

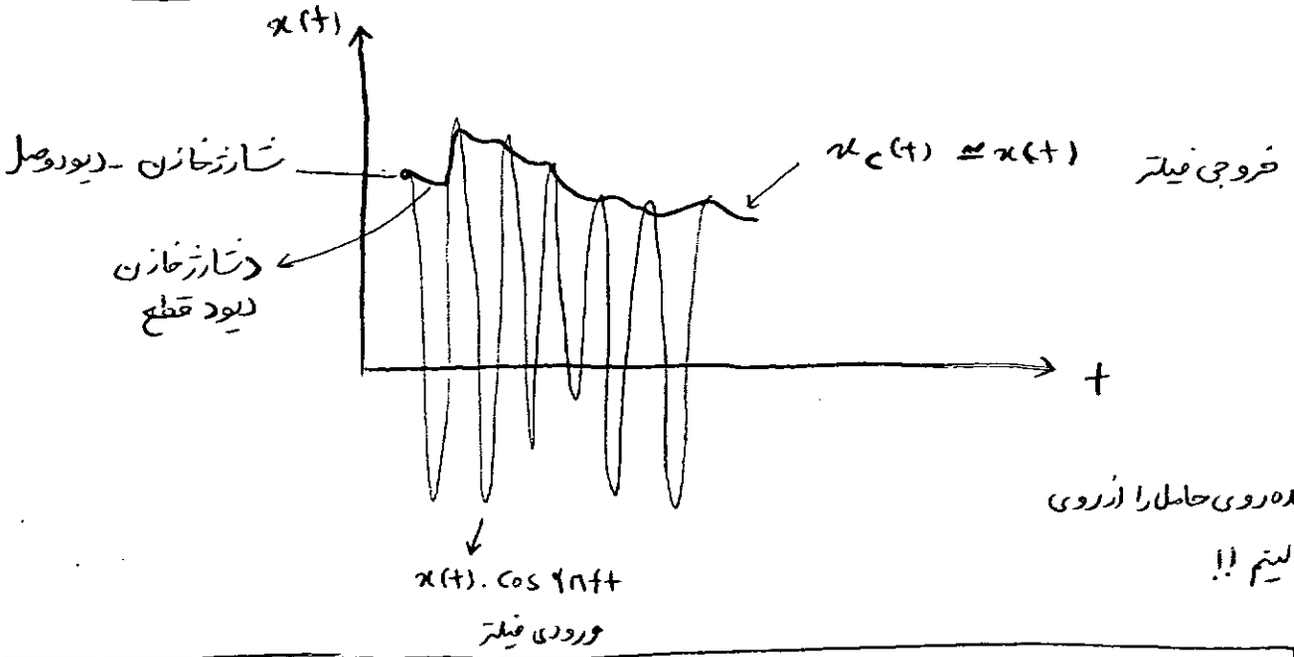
* $2f_c$ باید حتماً از $2f_m$ بزرگتر یا مساوی باشد.

اگر اینگونه نباشد مولفه‌های $\frac{1}{2}x(t)$ و $\frac{1}{2}x(t)\cos(4\pi f_c t)$ با هم مخلوط می‌شوند.

* بحثهایی که کردیم تئوری بودند به بحث مدولاسیون در عملی پردازیم. (بدون توجه به جنبه‌های ریاضی صرف)



شکل (126) فیلترخازنی (مدولاسیون و دی‌مدولاسیون در عمل)



سیگنال سوار شده روی حامل را از روی آن پیاده می کنیم !!

شکل (127) سیگنال $x(t)$ - پایه - تقریباً بازسازی شده است. (بدون نیاز به ضرب مجدد در تابع کسینوسی - حامل -)

* اعوجاجات خروجی فیلتر، با انتخاب f بیشتر (فرکانس تابع کسینوسی) کمتر می شود.

⊕ مدولاسیون در لغت به معنای هماهنگ کردن است.

⇦ دامنه سیگنال کسینوسی با سیگنال $x(t)$ هماهنگ شد.

⊗ سیگنال کسینوسی را سیگنال حامل - CARRIER - گوییم.

چون به تغییری سیگنال $x(t)$ روی آن سواری شود.

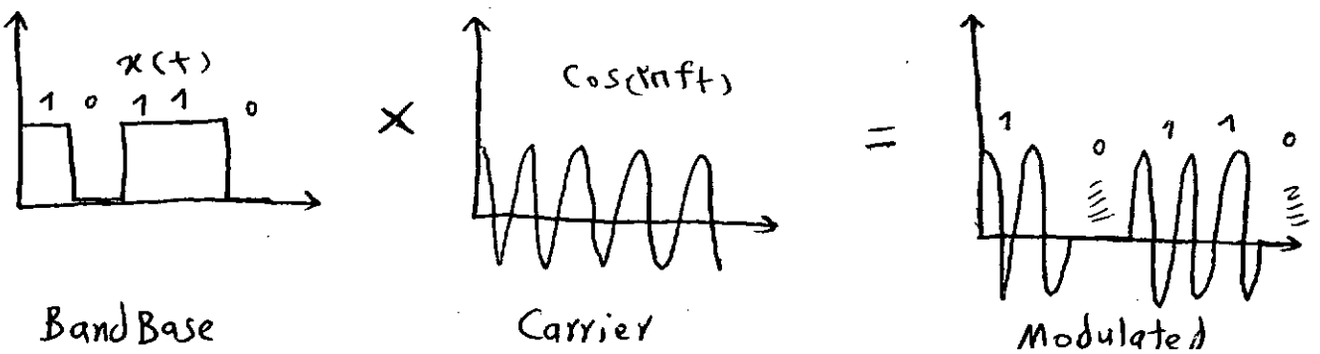
در عمل در مدولاسیون سیگنال را از روی Carrier پیاده می کنیم.

---

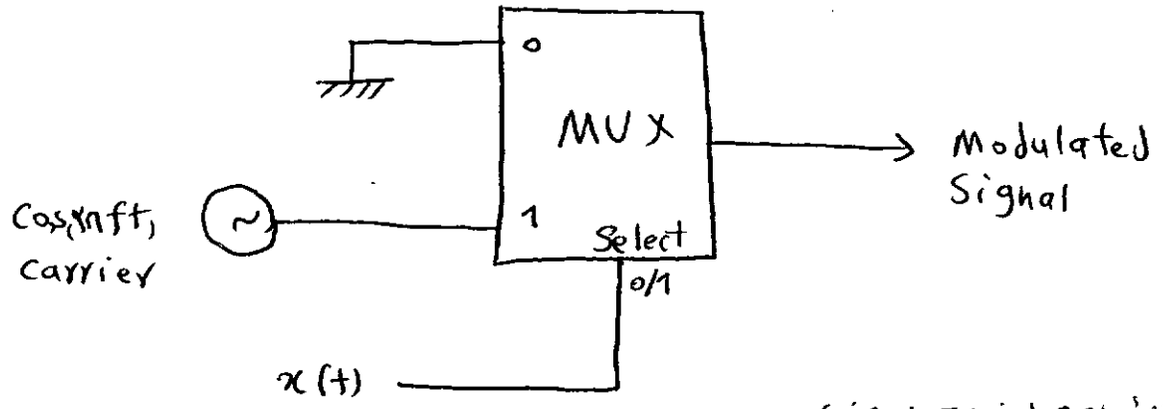
* بحث بیشتر مدولاسیون در درس مخابرات صناعی است.
* تا اینجا روی سیگنالهای آنالوگ بحث کردیم.
* سیگنالهای مادیجیتال هستند.

شکل (128)

مدولاسیون سیگنالهای دیجیتال :



⊕ ساختن سیگنال مدوله‌شده بسیار راحت است:



شکل (129) مدولاسیون کنندهٔ ASK (صفر و یک برای صفر)

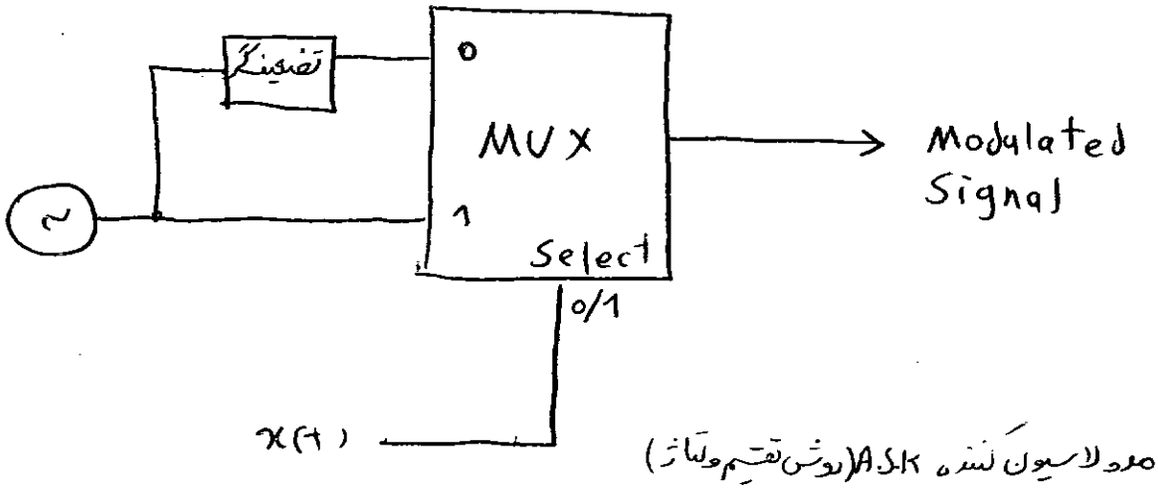
به عمل بالا AMPLITUDE-SHIFT-KEYING می‌گوییم.

ASK ⊕

* ایراد این روش این است که قطعی خط از بیت های صفر پشت سر هم تمیز داده نمی‌شود.
* برای رفع این مشکل برای بیتهای صفر به جای فرستادن صفر، سیگنال تضعیف‌شدهٔ کسینوسی را می‌فرستند.

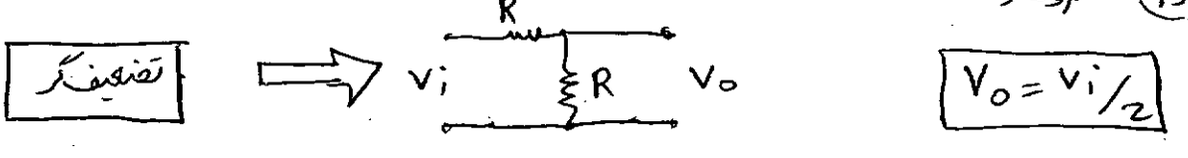
AMPLITUDE MODULATION

⊕ مدولاسیون دامنه:



شکل (130) مدولاسیون کننده ASK (روش تقسیم ولتاژ)

شکل (131) مقسم ولتاژ



⊕ یک سیگنال کسینوسی دارای خواص زیر است:

- دامنه
- فرکانس
- فاز

رابطه (22)

$$A_c \cos(2\pi f_c t + \varphi_c)$$

A_c ← دامنه ، f_c ← فرکانس ، φ_c ← فاز

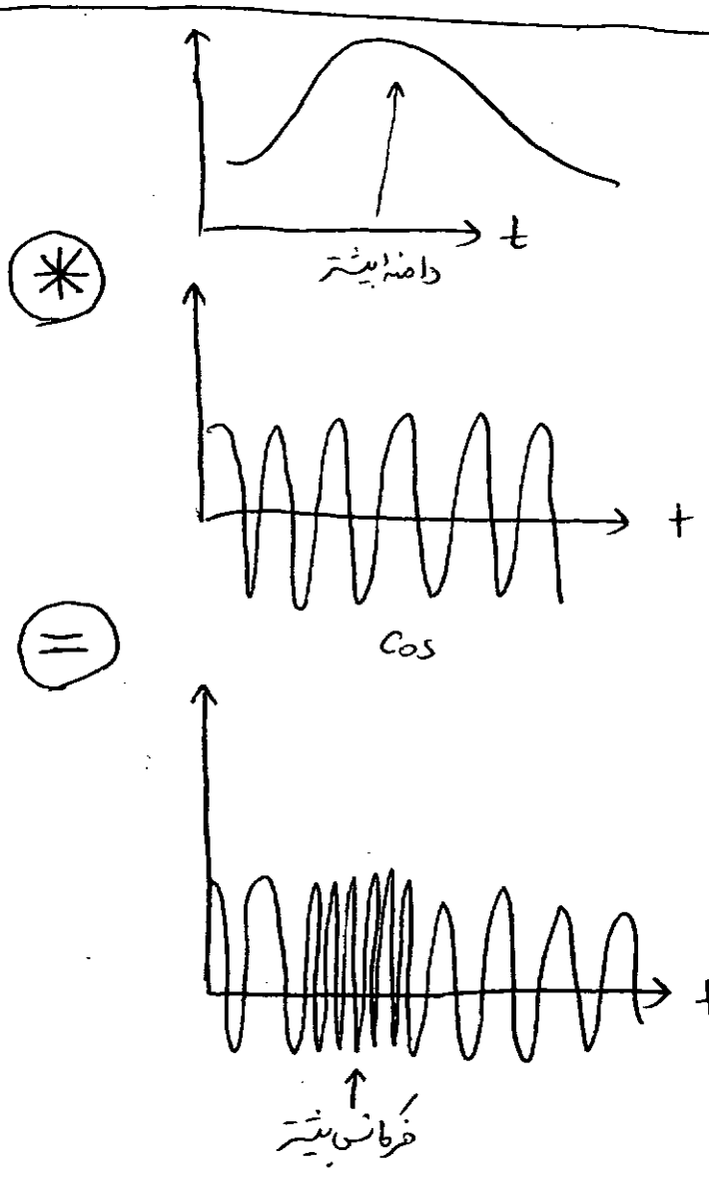
هرکدام از خواص سیگنال کسینوسی را می‌توان مدولاسیون کرد.

* تا به حال مدولاسیون دامنه AM یا Amplitude Modulation را بررسی کردیم.

⊕⊕ FM ← 2 : [مدولاسیون بسامد] Frequency Modulation

هماهنگ کردن فرکانس با دامنهٔ $x(t)$

شکل (132) مدولاسیون فرکانس



Base Band
سیگنال آنالوگ باندپایه

(*) → «تبدیل دامنه به فرکانس»

Cos 2πft ≡ Carrier
سیگنال حامل

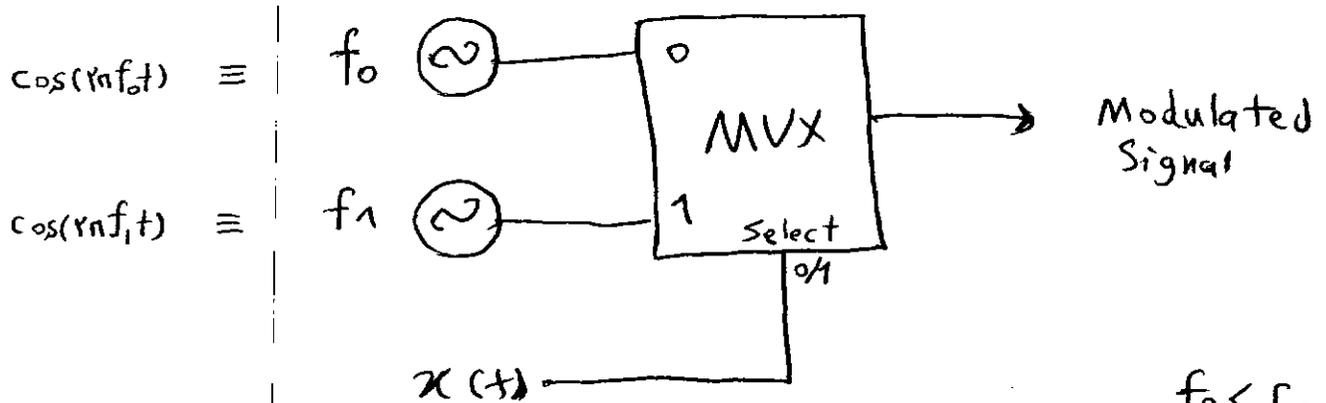
(=)

«سیگنال مدوله شده تحت فرکانس»
Modulated
(Freqency)

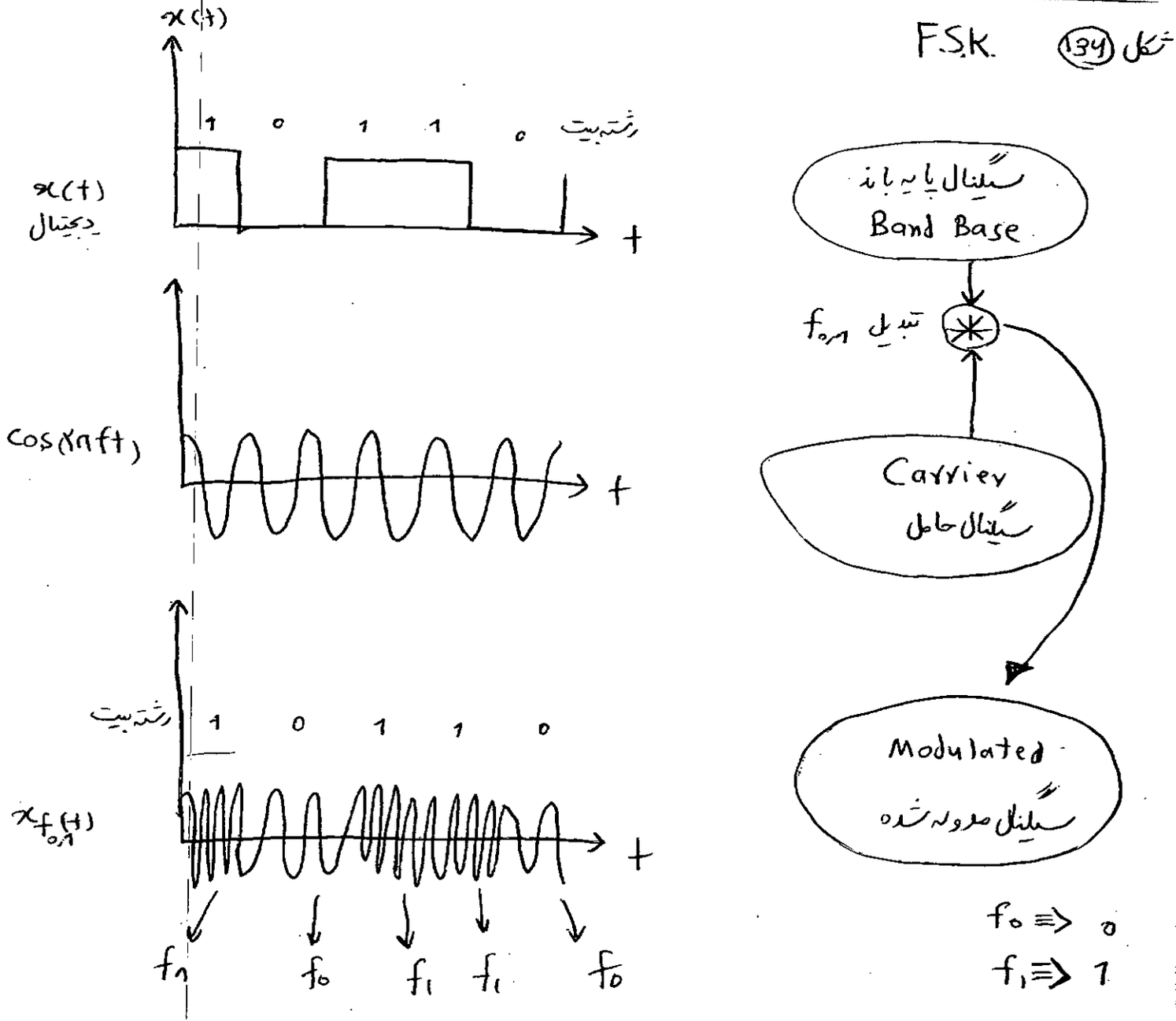
✳ مدوله کردن (فرکانسی) سیگنال دیجیتال :

FREQUENCY – SHIFT – KEYING

FSK



شکل (133) مدولاسیون FSK به روش کلید زنی



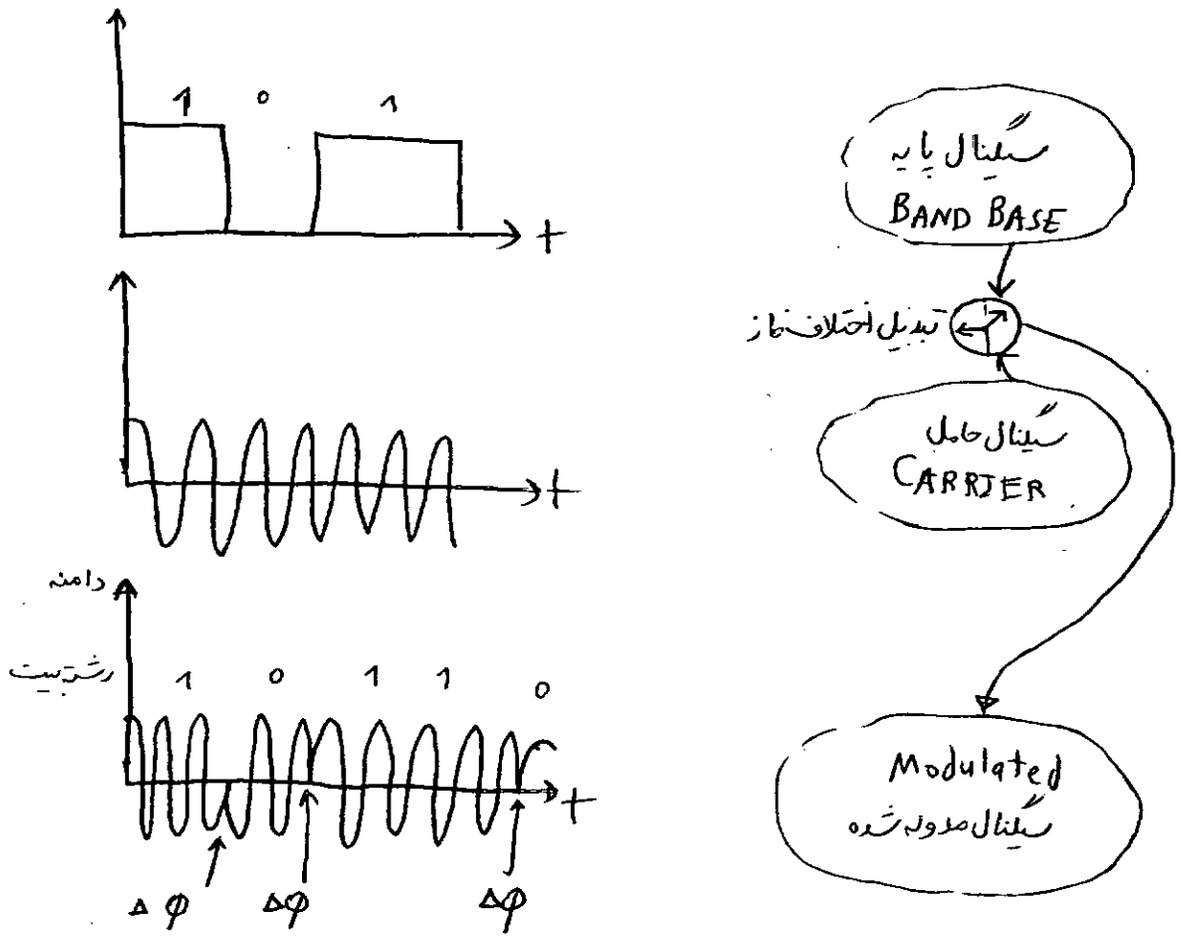
PHASE MODULATION مدولاسیون فاز: ⊕3

P.M.

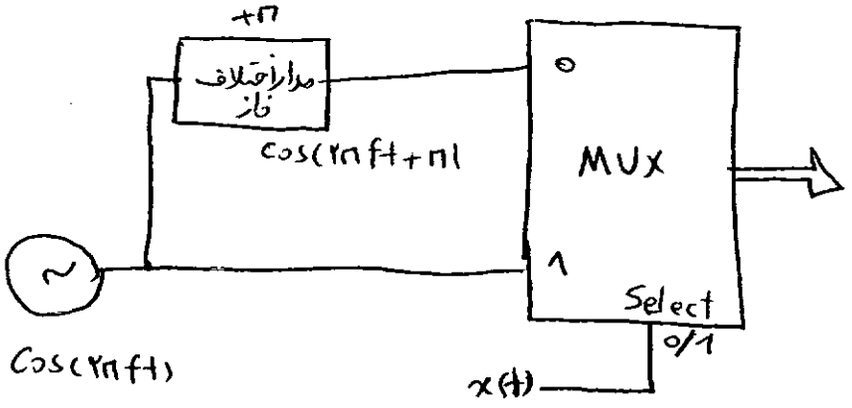
★ هماهنگ کردن فاز با دامنهٔ سیگنال $x(t)$ (فقط بحث دیجیتال را مطرح می کنیم)

⊛ مدوله کردن سیگنال دیجیتال : (توسط فاز P.M.)

شکل (135)



$\Delta\varphi \Rightarrow$ Trigger $\begin{matrix}1 \text{ to } 0\\ 0 \text{ to } 1\end{matrix}$



شکل (136) مدار مدولاسیون P.S.K

Phase-Shifting-Key P.S.K ⇦

مدوله کردن دیجیتال توسط فاز

مقایسهٔ روشهای مختلف مدولاسیون (دیجیتال)

EXPENCES :

● معیار اول (هزینه)

مدولاسیون	P.S.K	F.S.K	A.S.K
	یک عدد مالتی پلکسر MUX	یک مالتی پلکسر MUX	یک عدد مالتی پلکسر MUX
	یک اسیلاتور OSC	دو اسیلاتور OSC	یک عدد اسیلاتور OSC
	یک مدار اختلاف فاز		یک مقسم ولتاژ
	هزینهٔ متوسط	هزینهٔ زیاد	هزینهٔ کم
	هزینه های دی مدولاسیون در ادامهٔ بحث خواهد آمد. صفحه های 97 و 98		
دی مدولاسیون	یک ضرب کننده	دو عدد ضرب کننده	یک عدد ضرب کننده و فیلتر
	یک فیلتر	یک مقایسه گر	یا یک فیلتر خازنی (دو عدد خازن، یک مقاومت)
	یک بازساز		روش بدون ضرب مجدد در تابع کسینوسی
	یک هم آهنگ کننده فاز فرستنده و گیرنده		
	هزینهٔ متوسط، کمتر از F.S.K	هزینهٔ متوسط	هزینهٔ کم

جدول شمارهٔ (2)

نتیجه ⇐ از لحاظ معیار هزینه روشهای مدولاسیون دارای اولویت زیر هستند:

Modulation

1- A.S.K مدولاسیون دامنه

2- P.S.K مدولاسیون فاز

3- F.S.K مدولاسیون بسامد (فرکانس)

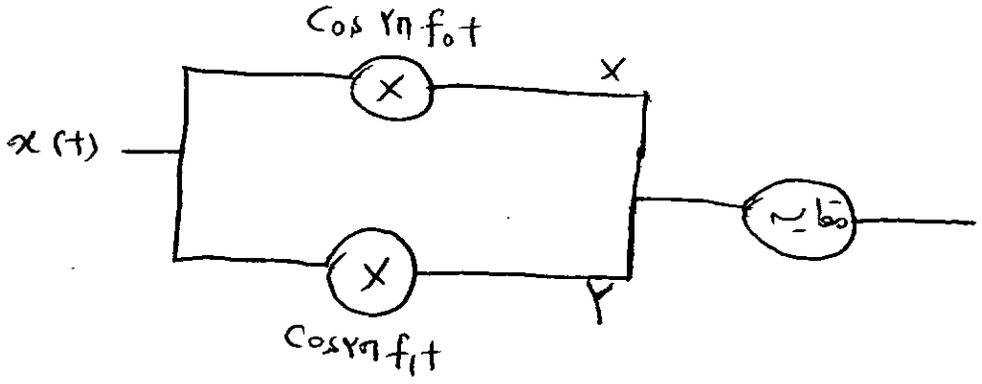
★ دو معیار دیگر جهت مقایسهٔ این سه روش مدولاسیون در جلسهٔ آینده معرفی خواهند شد.

⊕ دی مدولاسیون روش A.S.K

★ روش اول: فیلتر خازنی که توضیح داده شد (صفحه 90)

★ روش دوم: ضرب در سیگنال (cos) و فیلتر کردن با فیلتر پهنای باند $[f_m < f_w < 2f_c - f_m]$ صفحه (

⊕ دی مدولاسیون به روش F.S.K:

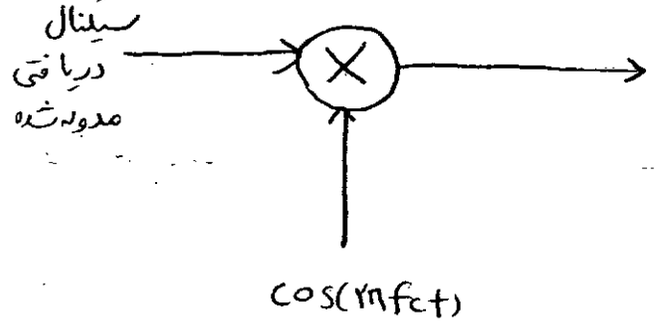


⊕ تجهیزات نیاز: مدار a- دوضرب کننده
+
b- یک مقایسه گر

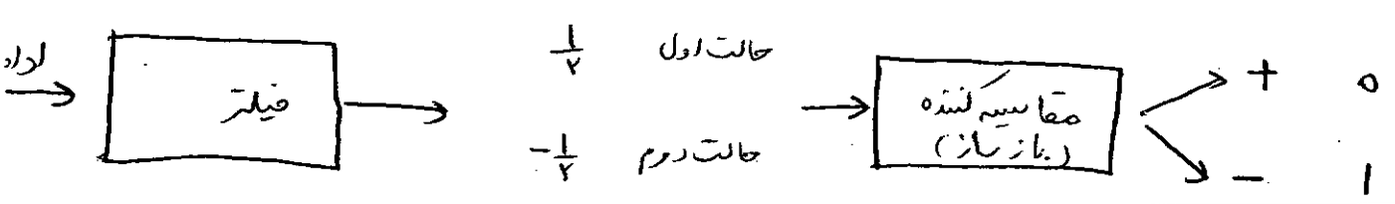
شکل (137) مدار دی مدولاسیون F.S.K

⊕ دی مدولاسیون به روش P.S.K:

⊕ فرض اساسی: $\cos(2\pi f_c t)$ کاملاً هم فاز با سیگنال فرستنده است.



شکل (138) مولد دی مدولاسیون روش P.S.K



⊛ تجهیزات مورد نیاز: P.S.K برای دی مدولاسیون

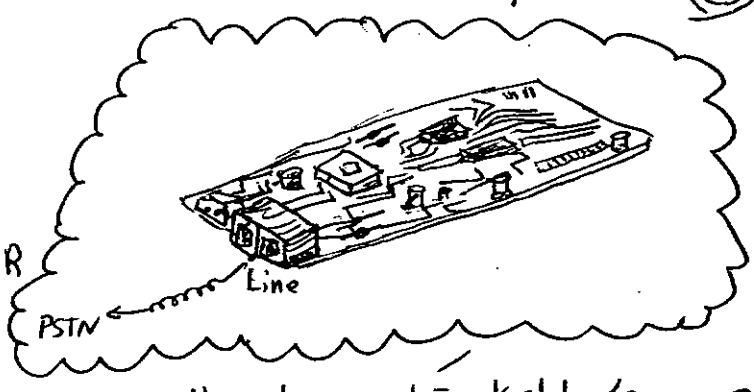
a — ضرب کننده
b — یک فیلتر
c — یک بازساز
d — یک هم‌زمان‌کننده اسیلاتور فرستنده و گیرنده

⊕ مودم:

MODEM :

MODULATOR

DEMODULATOR



⊛ وسیله‌ای است که هر دو عمل مدولاسیون و دی مدولاسیون را برای کامپیوترهای شخصی انجام می‌دهد و عموماً جهت ارتباط با کانال‌های خطوط شبکهٔ تلفن - PSTN - به‌کار می‌رود.

⊛ !!! رادیو: RADIO

* مطلب خوب درس نبوده و خودم آن را نوشته‌ام! شاید غلط باشد، . عجب !! ؟؟؟؟!!

⊕ رادیو، گیرنده (دی مدولاسیون‌کننده) امواج رادیویی منتشر شده در فضا است.

⊕ می‌دانیم که در فضای اطراف ما امواج گاما و فضا طیس بسیاری وجود دارند. (تکنولوژی انتقال ≡ Broadcast)

⊗ در واقع فضا کانال انتقال بسیار بسیار وسیعی است که سیگنال‌های بسیاری در آن جریان دارند،

⊛ + هر سیگنال در محدودهٔ خاصی مدولاسیون شده تا با بقیه تداخل نکند + (کار فرستنده‌ها و قرارداد آنها با هم)

⊗ رادیو با انتخاب فرکانس مناسب جهت دی مدولاسیون یکی از این سیگنال‌ها را دی مدوله کرده و پس از تقویت و تبدیل مجدد به علایم صوتی (سیگنال آنالوگ است یعنی خودش علایم الکتریکی است و نیازی به تبدیل دیجیتال به آنالوگ نیست -) آنرا از طریق بلندگو به گوش ما می‌رساند.

⊕ فیلترخانی با چرخاندن پیچ، تغییر مشخصه فرکانسی می‌دهد: علت ناملایمی FM و AM در رادیو

نوع مدولاسیون امواج است پس رادیو هم به روش F.M و هم A.M می‌تواند سیگنال‌ها را دی مدوله کند!!!

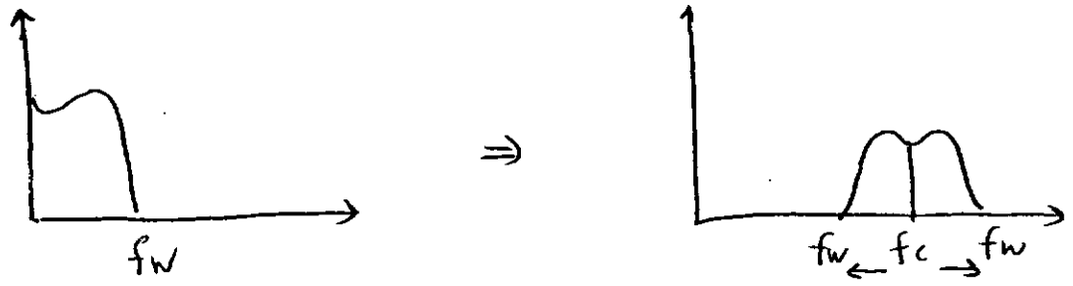
(۲) ⊕ معیار دوم مقایسه:

BAND WIDTH

پهنای باند:

---

(1) A.S.K



حالت اول: Single Side Band

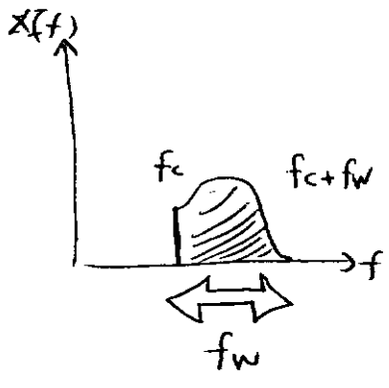
پهنای باند fw طیف است. [fc ⇔ fc+fw] رابطه (23)

$BW = fw$

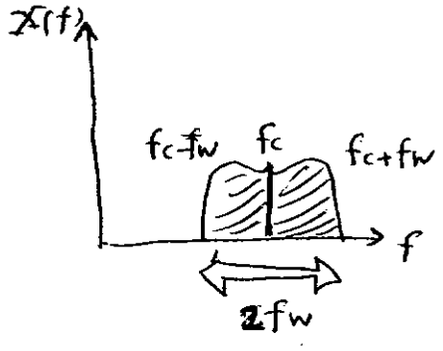
حالت دوم: Double Side Band

$BW = fw * 2$

پهنای باند 2fw لازم است. [fc-fw , fc+fw] رابطه (24)



[Single Side]

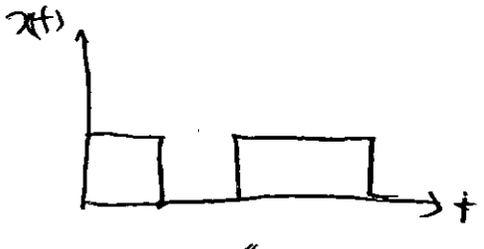


Double Side

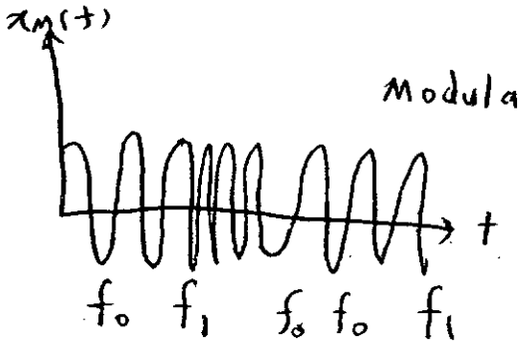
شکل (39) پهنای باند روش A.S.K

---

(2) F.S.K : دو فرکانس f_0 و f_1 داریم:

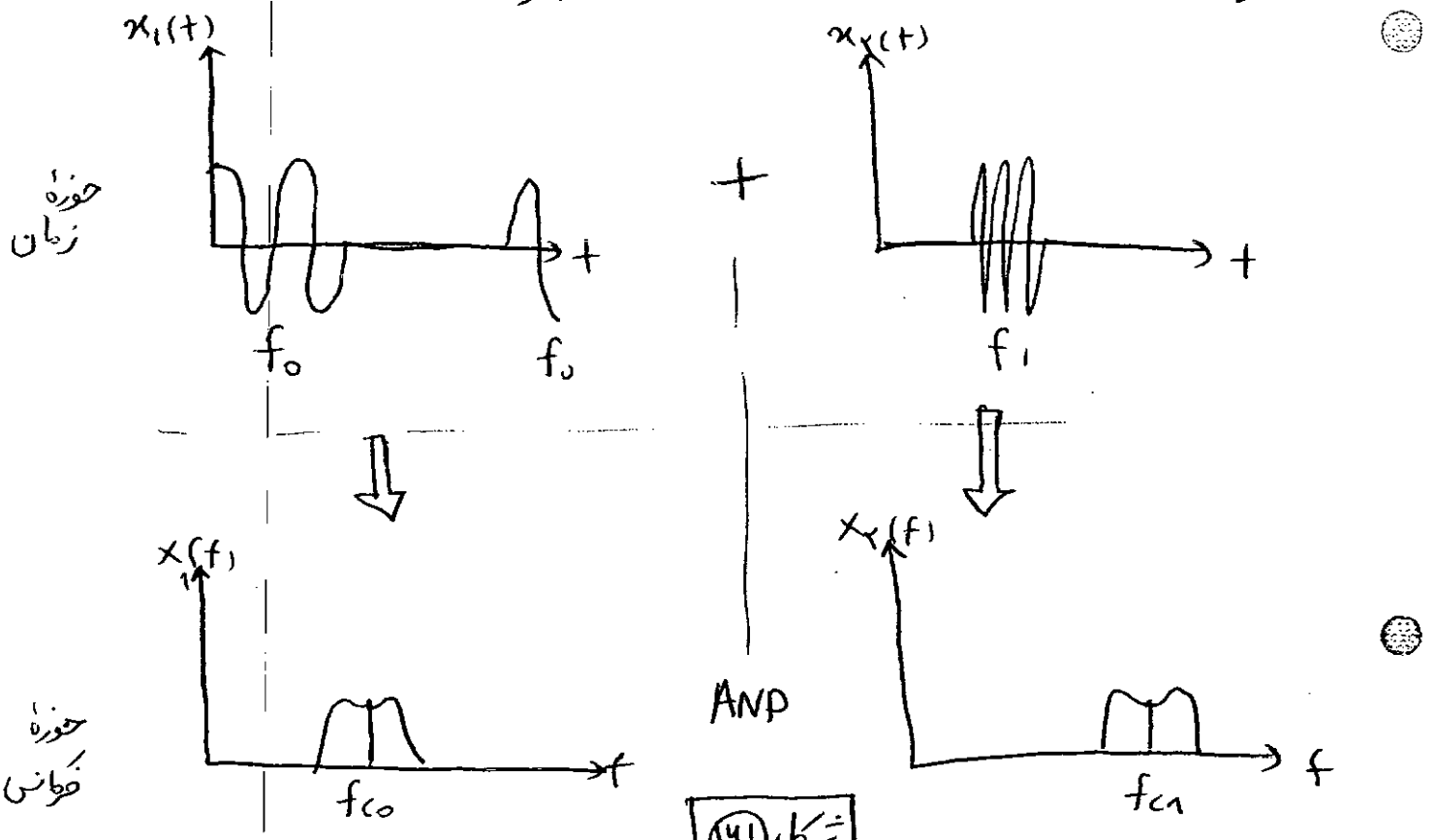


Band Base سیگنال



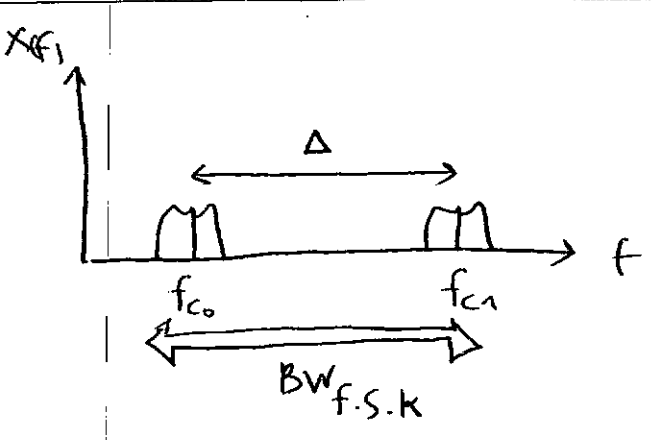
شکل (40)

این سیگنال مدوله شده را می‌توانیم مجموع دو سیگنال زیر در نظر بگیریم:



شکل (۱۴۱)

پهنای باند لازم: (روش F.M)



شکل (۱۴۹)

حالت اول:

روابط (25)

$$BW_{f.s.k} = 2\times 2f_w + \Delta[-2\times f_w]$$ پهنای باند

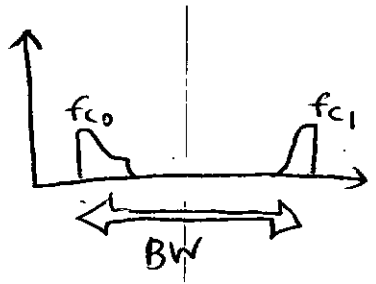
و

$$f_{c1} - f_{c0} = \Delta$$ تعریف Δ

و

$$\Delta > 2f_w$$ جهت عدم تداخل دو مولفه

حالت دوم:

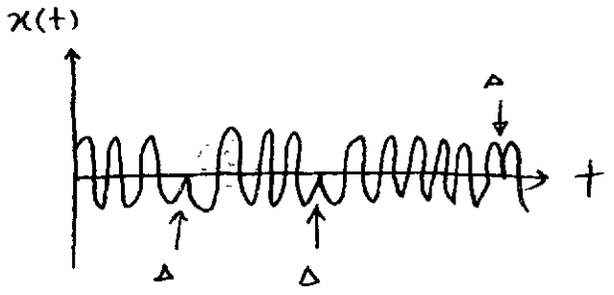


شکل (۱۴۲)

$$BW_{f.s.k} = 2\times f_w + \Delta[-2*f_w]$$

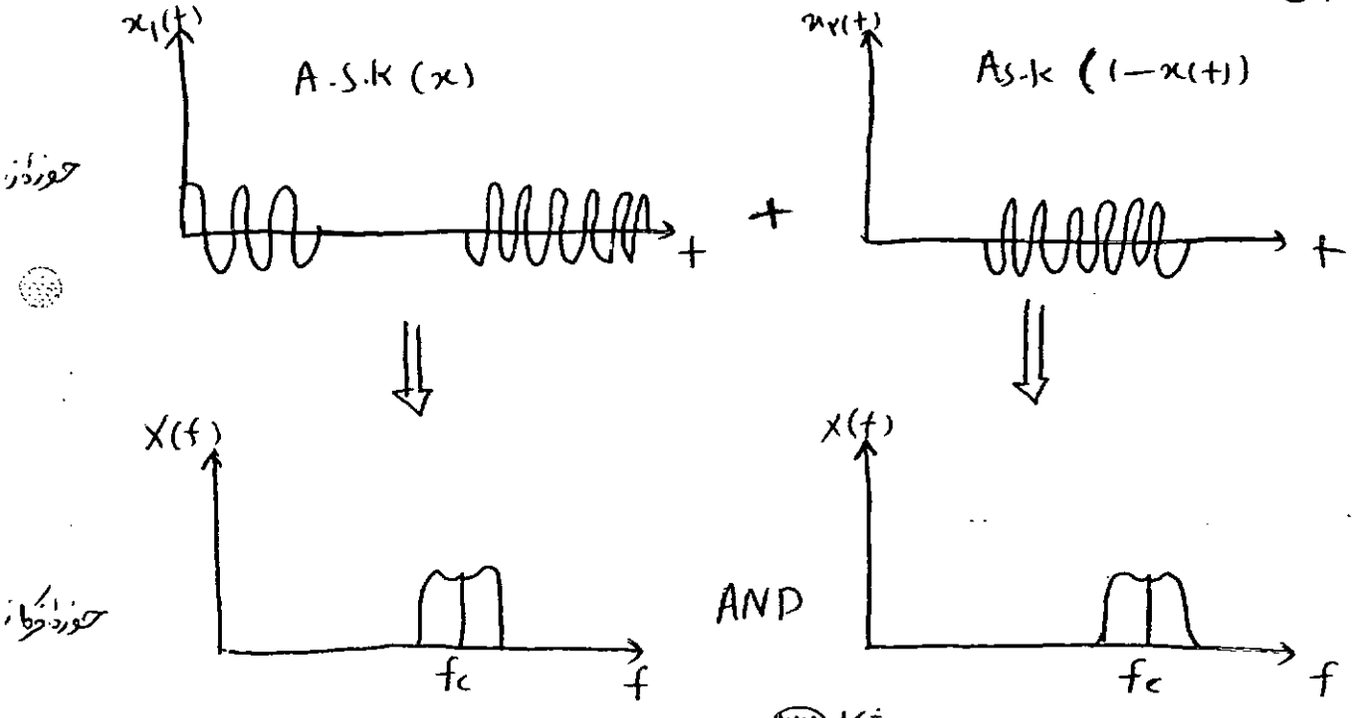
رابطه (26)

P.S.K (3)



شکل (143)
سیگنال مدوله شدهٔ P.S.K

← باز هم می‌توان سیگنال مدوله شده را
مجموع دو سیگنال کسینوسی در نظر گرفت.



شکل (144)

* بررسی دقیق نیست: چون در سیگنال x(t) تیزی‌هایی داریم ( ) که مولفه‌های فرکانسی بیشتری را دارند.

$$BW_{P.S.K} \simeq BW_{A.S.K} = \begin{cases} fw & \text{Single} \\ 2fw & \text{Double} \end{cases}$$

روابط (27)

* پهنای باند P.S.K تقریباً برابر A.S.K است (کمی بیشتر از آن ـ به خاطر تیزی‌ها ـ)

نتیجه:

A.S.K → P.S.K → F.S.K → B.W

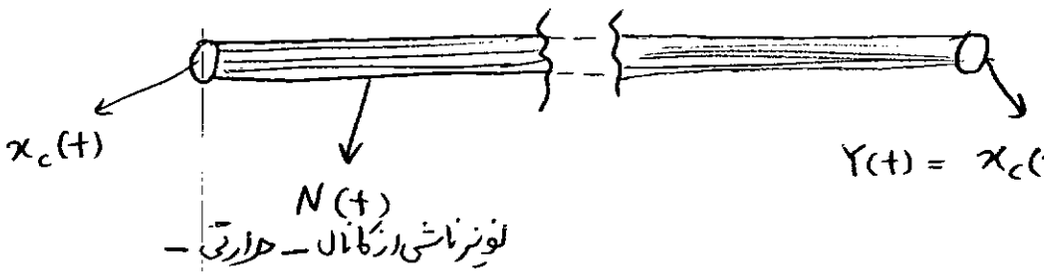
کمترین پهنای باند را احتیاج دارد

بیشترین پهنای باند را احتیاج دارد.

۳ - ⊕ معیار سوم مقایسه:

امنیت در برابر نویز: NOISE IMMUNITY :

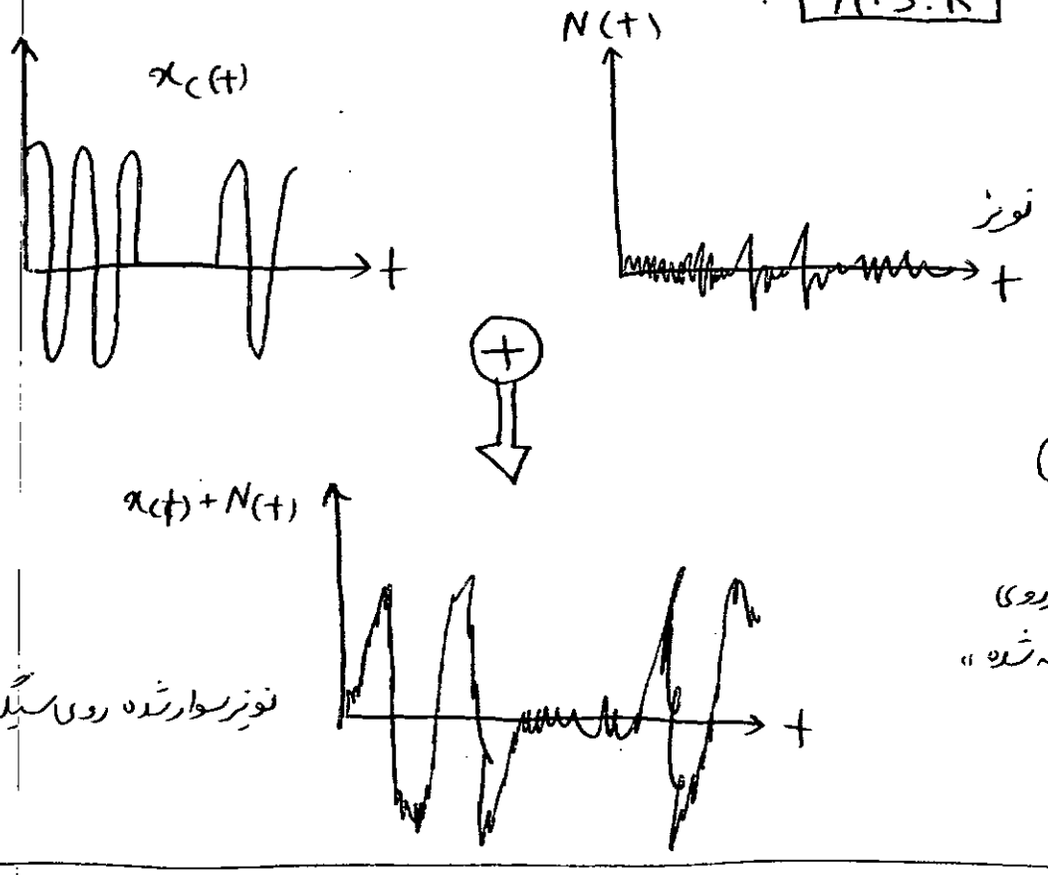
* نویز حرارتی یک نویز جمع شونده است: بدین معنی که با سیگنال گذرا از درون کانال جمع می‌شود.



رابطهٔ (28)

شکل (145)

: A.S.K ⓐ



شکل (146)

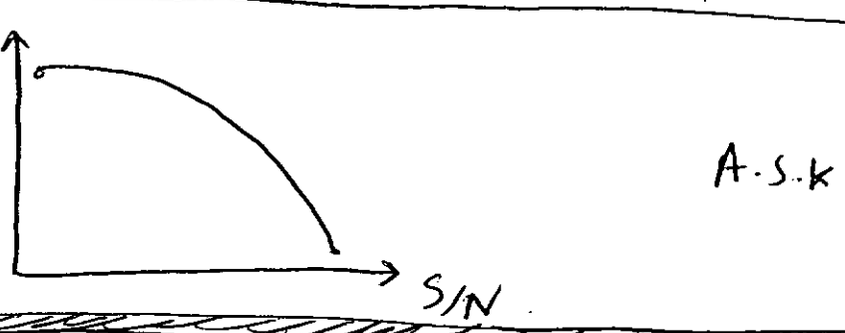
سوار شدن نویز روی سیگنال مدوله شده

---

* در اینجا شکل خاصی نداریم. مگر آنکه نویز ناخواسته‌ای با دامنه بسیار زیاد داشته باشیم.
* تفکیک به روش Threshold وقتی S/N زیاد است براحتی انجام پذیرفت.

* بحث احتمالاتی می‌کنیم.

S/N = انرژی سیگنال به نویز P_e = احتمال خطا رابطهٔ (29)



شکل (۱۴۷)
احتمال خطا در روش A.S.K

---

b) F.S.K:

* چون تشخیص سیگنال از روی فرکانسهای f_0 و f_1 است، نویز به سختی می‌تواند کاری کند که فرکانسها غیرقابل تفکیک شوند.
* در نتیجه احتمال خطا در F.S.K کمتر از A.S.K است.

شکل (۱۴۸)
احتمال خطا در روش F.S.K

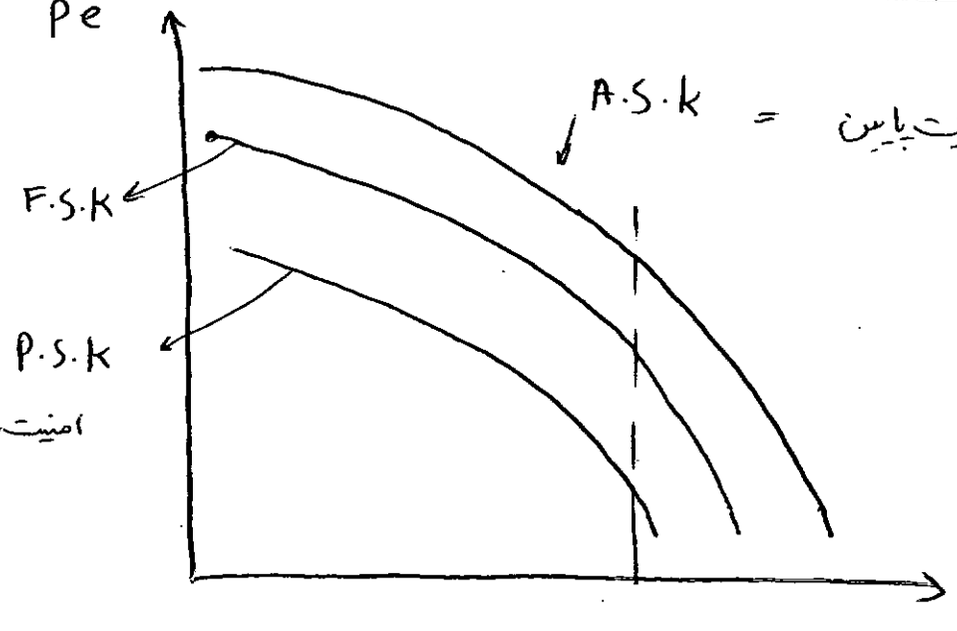
---

c) P.S.K:

* نویز باید کاری کند که گیرنده فاز را اشتباه کند؛ که بسیار بعید است پس احتمال آن از هر دو روش قبلی کمتر است.

شکل (۱۴۹)
احتمال خطا در روش P.S.K

مقایسهٔ امنیت در برابر نویز:



شکل (۱۵۰)
مقایسه امنیت در برابر نویز سه روش

« Recommendations »

* درجایی که اطلاعات محدود، نویز کم است، پهنای باند محدودی در اختیار داریم:
مثلاً دو کامپیوتر نزدیک هم : روش A.S.K بهتر است.

* جایی که پهنای باند زیاد است و کیفیت خیلی خوب مورد نظر است:
مثلاً سیگنال های رادیویی FM که کیفیت بالایی می خواهد F.S.K بهتر است.

* ارتباطات ماهواره ای، نرخ بیت بالا، امنیت خوب در برابر نویز، پهنای باند محدود تر،
روش P.S.K بهتر است. (هزینه اش هم خوب است).

⊕ در MODEM ها معمولاً از P.S.K برای نرخ بالای بیت و A.S.K برای نرخ کمتر استفاده می شود.

روابط (30)

Expence (A.S.K) < Expence (P.S.K) < Expence (F.S.K)

Bandwith (A.S.K) ≤ Bandwidth (P.S.K) ≤ Bandwidth (F.S.K)

Noise Immunity (P.S.K) > Noise Immunity (F.S.K) > Noise Immunity (A.S.K)

بدترین ⟷ متوسط ⟷ بهترین

COMPARING CRITERIA [A.S.K, P.S.K, F.S.K]

⊕ مقایسهٔ سه روش APF با سه معیار EBNI

هرگه که دل به عشق دهی خوش دمی بود
درکار خیر حاجت هیچ استخاره نیست.

«حافظ»

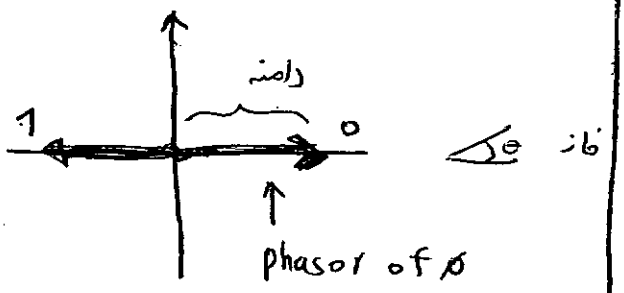
⊕ مطالعه‌ی بیشتر PSK:

* امنیت بالا در برابر نویز و پهنای باند کم باعث شده تا از این روش در ارتباطات دیجیتال بسیار استفاده شود.

---

* نمایش فازوری: Phasor Representation:

$$\emptyset = \cos(2\pi f_c t + 0)$$
$$1 = \cos(2\pi f_c t + \pi)$$

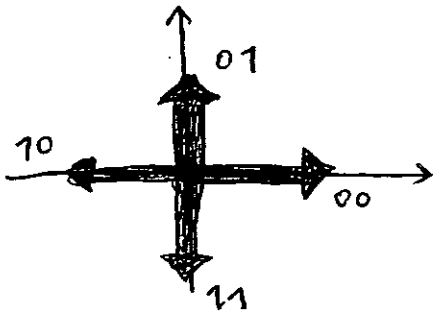


شکل (151) P.S.K دوفاز

---

* می‌توان این روش را تا چهار فاز گسترش داد.

$\cos(2\pi f t + 0^\circ)$
$\cos(2\pi f t + \pi/2)$
$\cos($

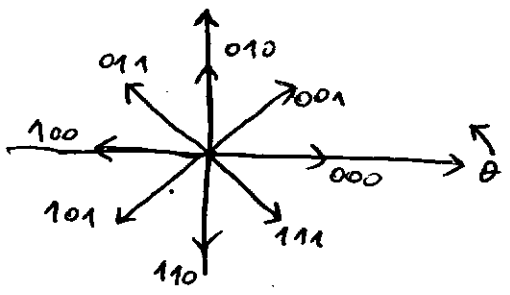


شکل (152) P.S.K چهارفازه

---

* اگر baudrate ما ثابت باشد P.S.K چهارفازه نرخ بیتی، دوبرابر نرخ بیت P.S.K دوفازه دارد:
چون هر علامت در اینجا حاوی دو بیت است.

+ تعمیم: می‌توان P.S.K 8 فازه طراحی کرد.

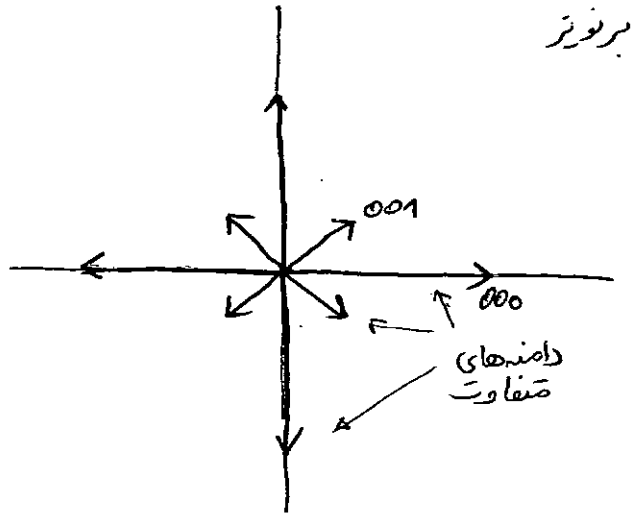


شکل (153) 8 فاز P.S.K

⊛ با اضافه کردن تعداد فاز حساسیت به نویز بالا می رود و به همان نسبت امنیت در برابر نویز پایین می آید.

(AP) تلفیق دوروش A.S.K و P.S.K :

* برای امنیت بالاتر در برابر نویز



شکل (154)

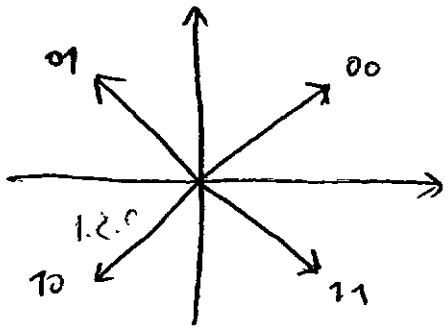
⊛ دوفاز مجاور دامنه‌های متفاوتی بهم دارند که باعث می شود امنیت در برابر نویز بالا برود.

⊞ P.S.K چهارفازه QAM

Quadrature
Amplitude
Modulation

چهارفاز:

$\sin(2\pi ft + \pi/4)$
$\sin(2\pi ft + 3\pi/4)$
$\sin(2\pi ft + 5\pi/4)$
$\sin(2\pi ft + 7\pi/4)$



$\pi/4 \equiv 00$
$3\pi/4 \equiv 01$
$5\pi/4 \equiv 10$
$7\pi/4 \equiv 11$

شکل (155) QAM

* بردار 00 را می توان بصورت جمع دوفاز با زوایای ۰ و $\pi/2$ ساخت (و بهمین ترتیب برای بقیه فازها)

$$\sqrt{2}\sin(2\pi ft + \pi/4) = \sin(2\pi ft + \pi/2) + \sin(2\pi ft)$$
$$\sqrt{2}\sin(2\pi ft + 3\pi/4) = \sin(2\pi ft + \pi) + \sin(2\pi ft + \pi/2)$$
$$\sqrt{2}\sin(2\pi ft + 5\pi/4) = \sin(2\pi ft + 3\pi/2) + \sin(2\pi ft + \pi)$$
$$\sqrt{2}\sin(2\pi ft + 7\pi/4) = \sin(2\pi ft) + \sin(2\pi ft + 3\pi/2)$$

می‌توان چهار فاز را با دو سیگنال با فازهای ۰ و $\pi/2$ و جمع و تفریق آنها ساخت.

$$[\pi/4]' = [0]' + [\pi/2]'$$
$$[3\pi/4]' = -[0]' + [\pi/2]'$$
$$[5\pi/4]' = -[0]' + [\pi/2]'$$
$$[7\pi/4]' = [0]' - [\pi/2]'$$

$[\ ]'$ نمایشگر فازور [phasor] است.

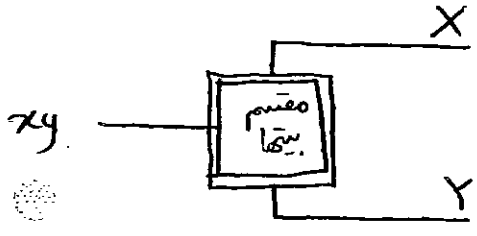
XY	
00	$\pi/4$
01	$3\pi/4$
10	$5\pi/4$
11	$7\pi/4$

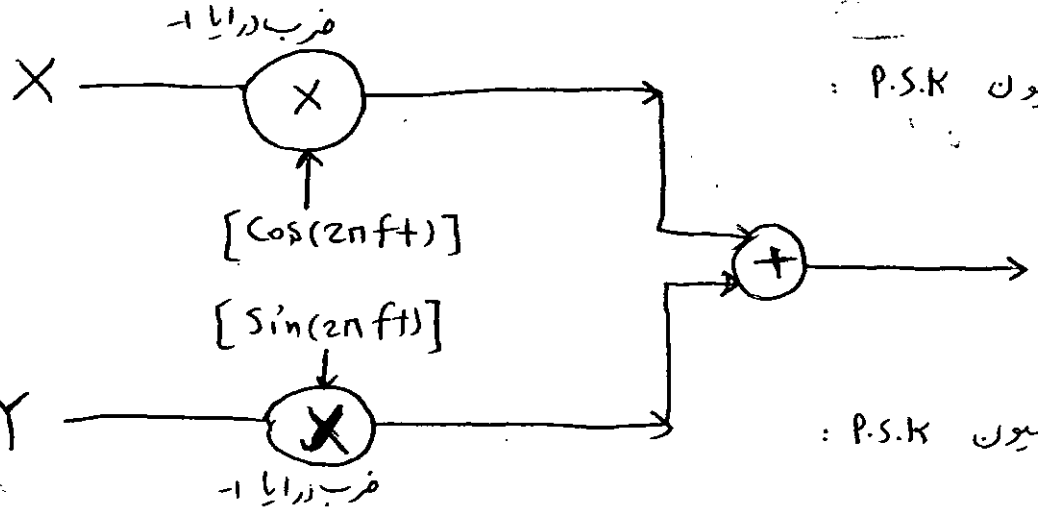
⇨

$Y=0$: $\sin(2\pi ft)$
$Y=1$: $-\sin(2\pi ft)$
$X=0$: $\cos(2\pi ft)$
$X=1$: $-\cos(2\pi ft)$

★ مداری می‌سازیم که معادلات بالا را انجام دهد:

ابتدا دوبیت را از هم جدا می‌کنیم





مدولاسیون P.S.K :

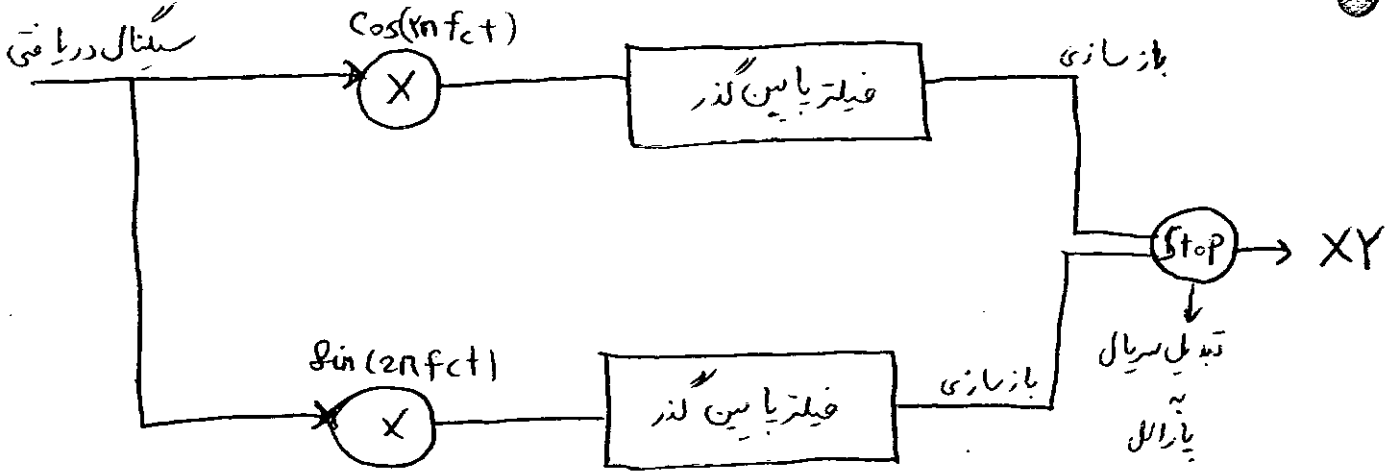
مدولاسیون P.S.K :

شکل (156) مدار مدولاسیون QAM

پس در کل با دو مدولاسیون دوقازهٔ P.S.K می‌توان QAM را پیاده سازی کرد.

البته سیگنالهای carrier این دو با هم فرق دارند (یکی Sin و دیگری Cos)

⊕ دی مدولاسیون QAM :



شکل (157) مداردی مدولاسیون QAM

E.e.1 ✱ بعنوان تمرین عملکرد مدار بالا را بررسی کنید. [هر چهار حالت را بررسی کنید]

CHANNEL CHARACTERISTICS IN DIGITAL TRANSMISSION

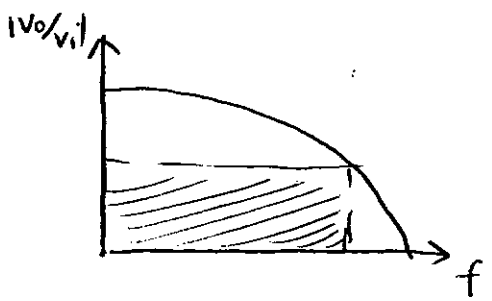
انتقال دیجیتال :

⊕ بررسی چند ویژگی کانالهای انتقال :

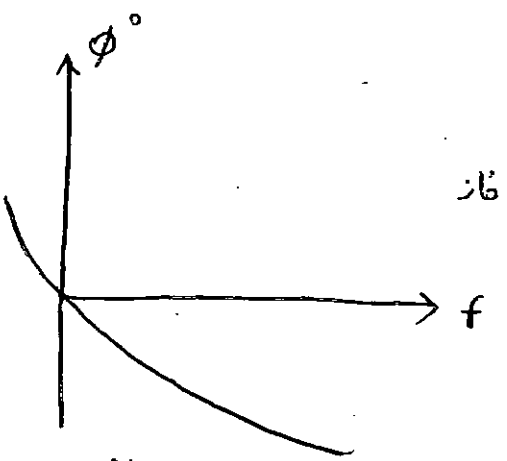
① مشخصهٔ فرکانسی :

AMPLITUDE RESPONSE V_o/V_i - a

PHASE RESPONSE فاز - b



Amplitude Response



Phase Response

شکل (158) پاسخ فرکانسی : دامنه و فاز

② نسبت سیگنال به نویز (SNR)

SIGNAL 2 NOISE RATIO

نسبت انرژی یا توان سیگنال به نویز $\equiv [S/N]$: واحد دسی بل dB

* عامل مهمی در انتقال دیجیتال است.
* بعضی کانال ها نویز زیاد و برخی نویز کمتری دارند

③ سرعت انتشار: PROPAGATION Rate

اگر ε سرعت نور باشد:

سرعت انتشار $v = c/\sqrt{\varepsilon}$

* ε به جنس کانال بستگی دارد.
* ε برای هوا 1 و برای بقیه اجسام بیشتر از 1 است. $\varepsilon \geq 1$.

← λ : طول موج، طول یک پریود.

طول موج $\lambda = \frac{v}{f}$ سرعت انتشار علایم / فرکانس

④ هدایت علایم در کانال انتقال: CONDUCTION of SYMBOLS

* مثلاً علایم منتشره در هوا احتیاج به تبدیل به موج دارند.
* در فیبر نوری باید اطلاعات تبدیل به فرمت قابل عبور شوند. (رشته‌های نور)
+ بر همین اساس دو دسته بندی برای کانالها داریم:

1- WIRELESS 2- WIRED

① WIRE-LESS : [بی سیم]

اطلاعات در کل فضا پخش می‌شود. [Broadcast]

از این رو به آنها عنوان CONTINUOUS NETWORK TOPOLOGY «توپولوژی پیوسته» اطلاق می‌شود.

② WIRED : [با سیم]

اطلاعات می‌توانند نقطه به نقطه منتقل شوند. از این رو به آنها عنوان DISCRETE NETWORK TOPOLOGY «توپولوژی گسسته» اطلاق می‌شود.

⊕ تضعیف : ATTENUATION ≡

در شبکه‌ها یا کانال‌های Wired داریم:

تضعیف = $k \cdot d$

(k: ضریب، d: فاصله)

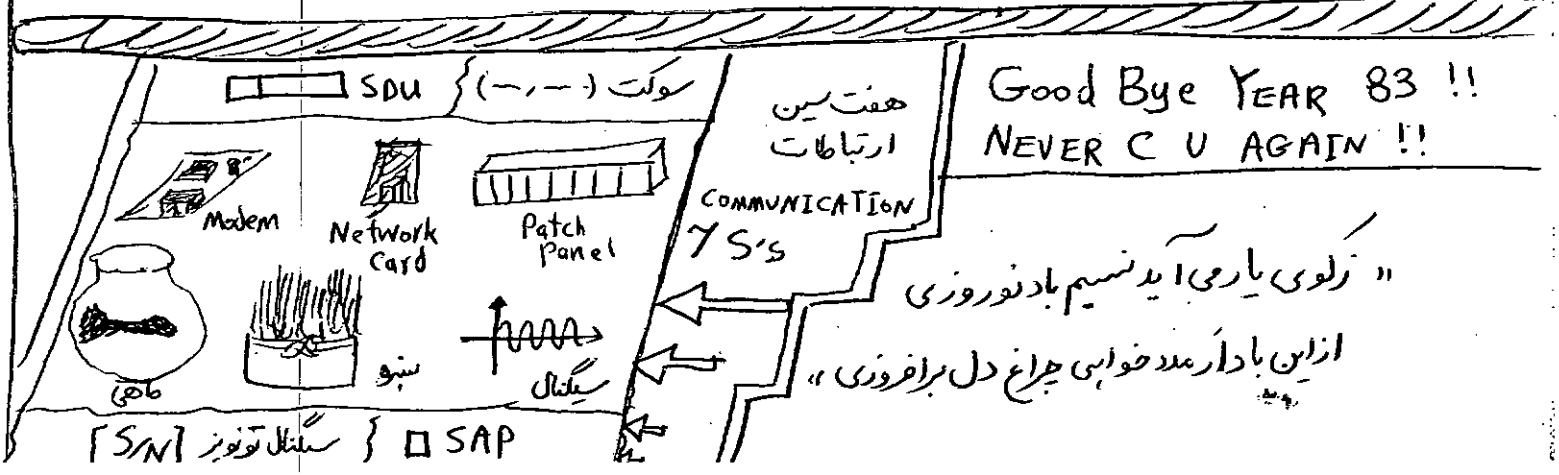
در کانال‌های Wireless داریم:

تضعیف = $\eta \log_2 d$ (dB)

پس در Wireless ها تضعیف کمتر است.

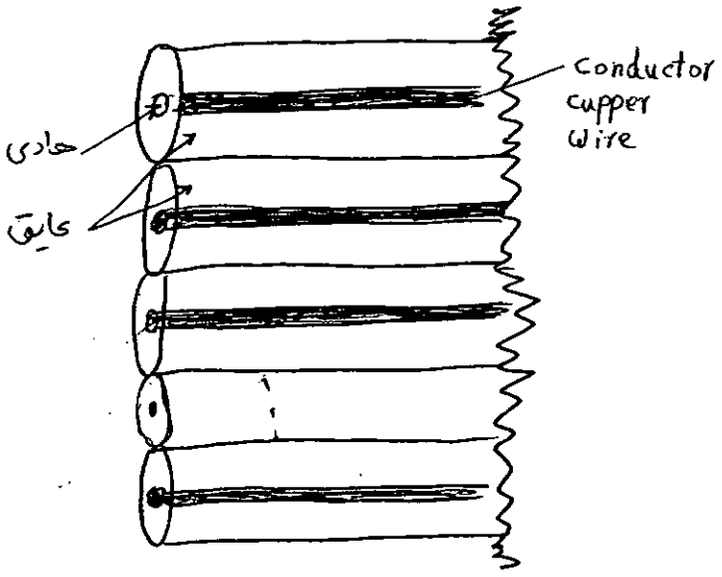
⊕ حق راه :

در سیم‌کشی‌های Wired سیم از هر جایی امکان عبور ندارد ولی در شبکه‌های بی سیم این مشکل وجود ندارد.



⊕ کانالهای انتقال: TRANSMISSION CHANNELS

⊕ ۱- سیم‌های روبانی:



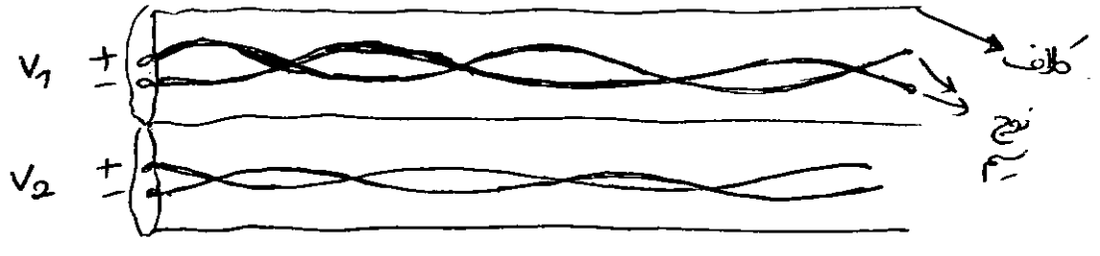
چند سیم موازی با هم که با عایقهایی از یکدیگر جدا شده‌اند.

شکل (159) رشته‌ای از سیم‌های روبانی با زمین مشترک.

* برای ارتباطات نزدیک. - مثل کامپیوتر به پرینتر - به کار می‌رود. (یا داخل خود کامپیوتر، BUSها)
* در این نوع کانال انتقال، به دلیل خاصیت خازنی عایقها، نویز Cross Talk داریم.
* این کانالها برای سرعتهای زیاد مناسب نیستند.
* حداکثر نرخ بیت می‌تواند 20 Kb/s باشد.
* حداکثر طول کانال، حداکثر چندین متر است.

← برای از بین بردن اثر Cross-talk از زوج سیمهای به هم تابیده استفاده می‌شود.

⊗ 2- زوج سیم‌های به هم تابیده: Twisted Pairs:



Twisted Pairs

شکل (160)

* در اینجا دیگر سر مشترک نداریم، پس اثر Cross-talk از بین می‌رود.

زوج سیم‌های به هم تابیده دو نوع دارند:

Ⓐ Shielded

دور زوج سیم‌ها یک حفاظ فلزی پیچیده می‌شود.
که از تشعشعات اضافی جلوگیری می‌کند.

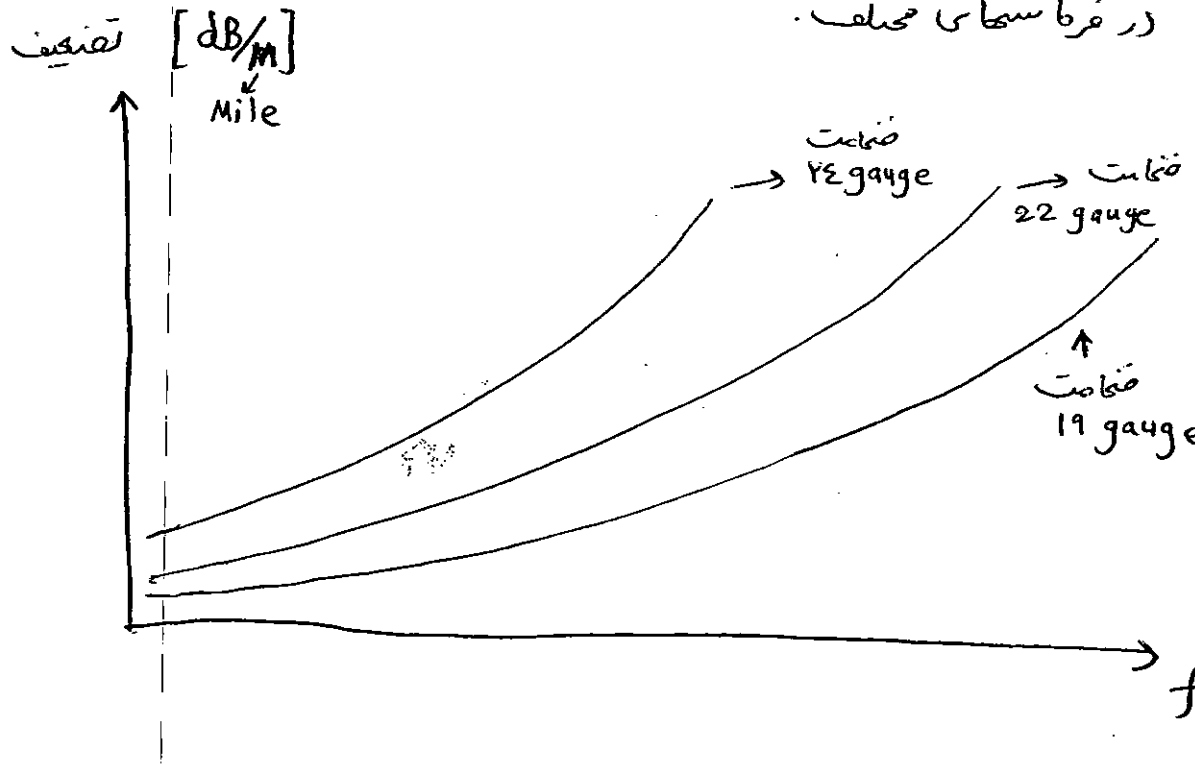
Ⓑ UnShielded

حفاظ فلزی وجود ندارد. (هزینه کمتری دارد)

---

* فواصل بیشتری را می‌توان با این کانال پوشش داد: حدود صدمتر
* تا نرخ بیت 1 Mb/s را می‌توان از این کانالها عبور داد.

Ⓐ بررسی تضعیف در زوج سیم‌های به هم تابیده در فرکانسهای مختلف.



24 gage = 0.016 :: واحد gauge

شکل (161)
تضعیف در Twisted Pairs

تضعیف:
1 KHz — 1 dB/Mile
500 KHz — 10 dB/Mile

کاربردهای Twisted Pairs : APPLICATIONS:

⊕ E1 : نرخ بیت 2Mb/s

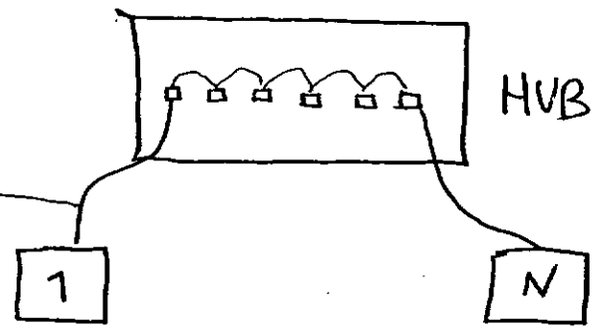
⊕ T1 : نرخ بیت 1.5 Mb/s طول 5.5Km

⊕ ISPN
Integrated Services Digital Networks
B : 64 kb/s
D : 16 kb/s

⊕ ADSL
Asymmetric Digital Subscribe Line
ارتباطات نامتقارن : یک طرف اطلاعات بیشتری را می فرستد
⇐ تقسیم پهنای باند { صوت / دیجیتال } QAM
رفت : کاربر به شبکه : 64 تا 640 kb/s
برگشت : شبکه به کاربر : 1.5 تا 6.4 Mb/s

⊕ شبکه‌های محلی Ethernet :

* برای تسهیل سیم بندی از جعبه‌ای به نام Hub استفاده می‌کنند که دارای سوکتهایی است که امکان ارتباطات را برقرار می‌کند. برای برقراری ارتباط کامپیوتر با Hub از Twisted Pair استفاده می‌شود



شکل (162) HUB Ethernet

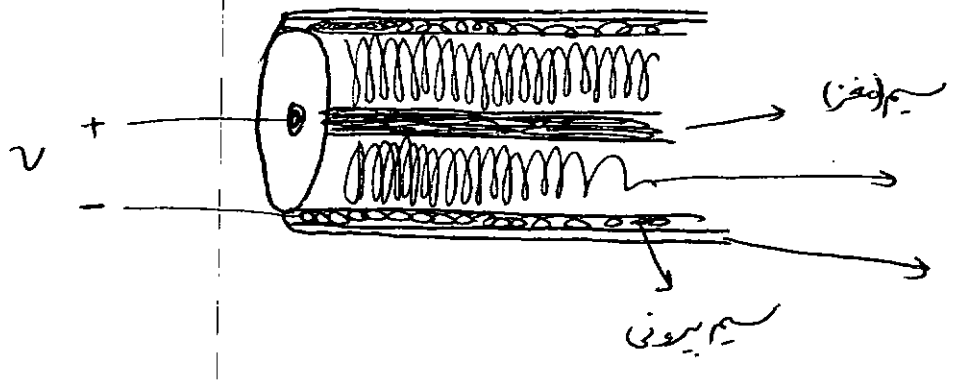
★ نمونه‌ای از کابلهای شبکهٔ Ethernet :

10 Base T
- 10 Mb/s
- سیگنال مدوله نمی‌شود.
- Twisted Pair

★ نمونه‌ای از کابل در شبکهٔ Fast Ethernet :

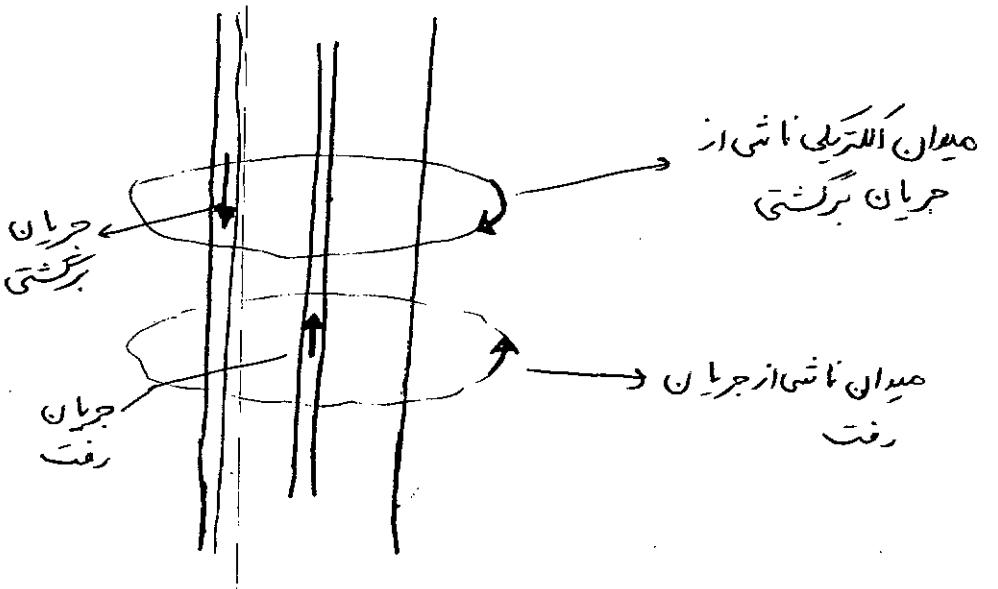
100 Base T
- 100 Mb/s
- Base Band
- Twisted

* 3 - کابل هم‌محور: CoAxIAL



شکل (163) کابل کواکس

(★★) میدانهای ناشی از جریانها یکدیگر را خنثی می‌کنند.



شکل (164) خنثی شدن میدان در کابلهای کواکس: یک مزیت بزرگ.

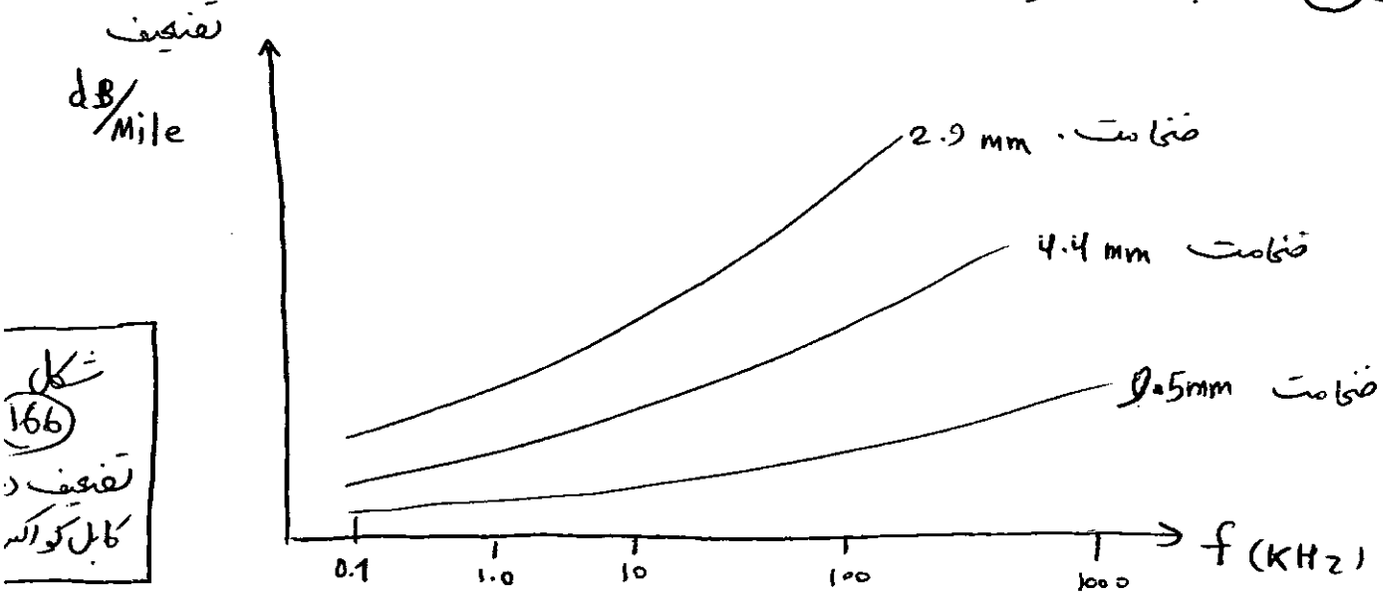
* فاصله‌های بیشتر و نرخ بیت بیشتر را می‌توان از کابلهای هم محور گرفت.

* فاصله‌های تا چند صد متر و نرخ بیت‌های تا چند ده مگابیت برثانیه.

⊛ نوع دیگری از کابلهای هم محور داریم که برای انعطاف بیشتر سیم‌های بیرونی بصورت کلافهایی بهم بافته می‌شوند. (در آنتن‌های تلویزیون همونه‌اش رادیده‌ایم)

شکل (165) کابل کوآکس تابیده شده.

⊖ بررسی تضعیف در فرکانسهای مختلف: (کابل کوآکس)



شکل (166) تضعیف در کابل کوآکس

⊕ کابل کوآکسیال: نقاط ضعف و قوت:

+ BANDWIDTH	+ پهنای باند بهتری دارد.
+ NOISE IMMUNITY	+ در برابر نویز کیفیت بالاتری دارد.
— EXPENCES	— پرهزینه است.
— CABLING	— کار کردن با آن مشکل است.

☆ انواع کابلهای کوآکس :

الف)

10-Base-5

↓

10 Mb/s

Base Band

ضخامت 10 mm

فاصله 500 m

Thick COAX

« کابل کوآکسیال ضخیم »

ب)

10 - Base - 2

10 Mb/s

ضخامت 5 mm

فاصله 200 m

Base Band

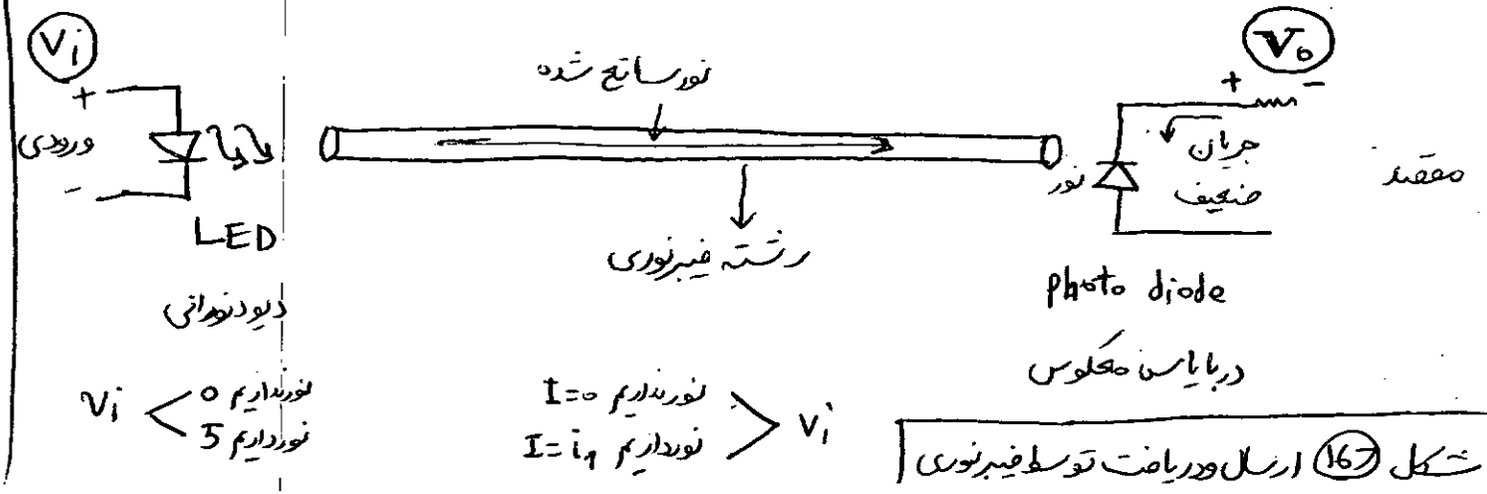
Thin COAx

« کابل کوآکسیال باریک »

⊗ 4- فیبر نوری : FIBER OPTIC

* برخلاف سه نوع قبلی ، علایم دراین جا سیگنال الکتریکی نیستند، بلکه نور هستند.

* برای تبدیل علایم نوری به الکتریکی از تجهیزات خاصی استفاده می شود.

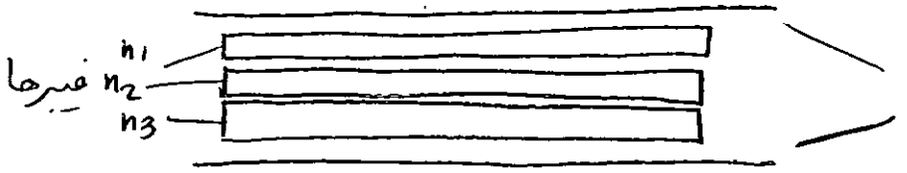


شکل (16) ارسال و دریافت توسط فیبر نوری

⬅ مهمترین مزیت فیبرنوری آن است که (راینجا نوری نداریم.

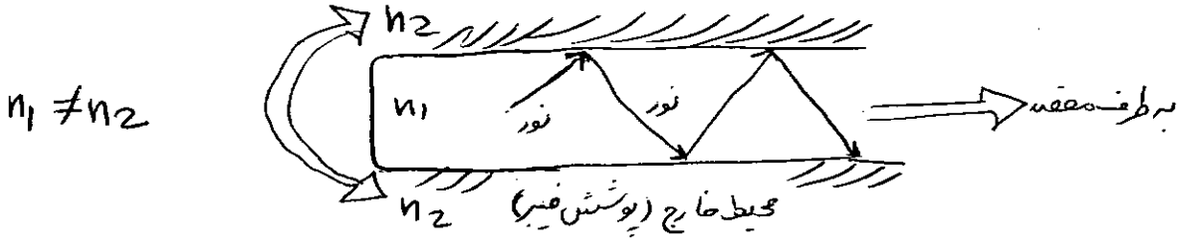
* برای جلوگیری از تداخل نور در فیبر از محافظهای پلاستیکی مناسب باید استفاده کرد.
* کارکردن با فیبرنوری سخت است.
* در مسیرهای غیرمستقیم نور نباید از فیبر خارج شود.

۱ ضرایب شکست متفاوت »



شکل (۱۶۸)

ضرایب شکست متفاوت باعث می‌شود که نور از محیط فیبر خارج نشود.

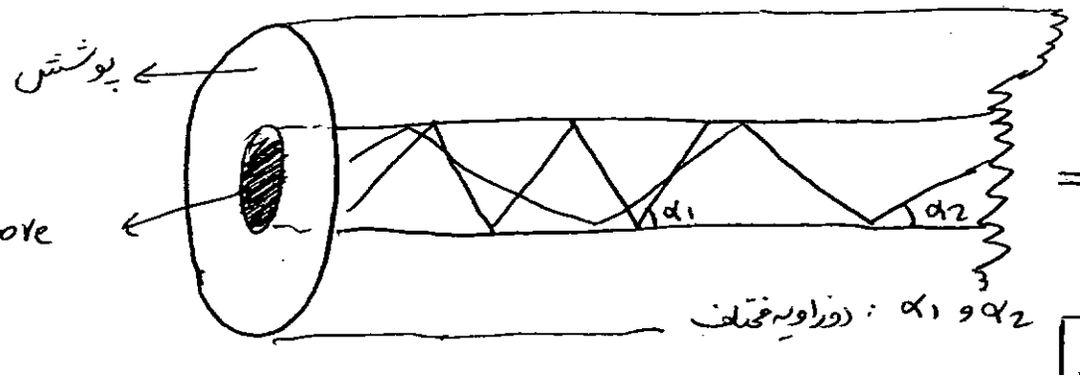


شکل (۱۶۹)

دو حالت استفاده از فیبرنوری :

◉ SINGLE MODE : یک نور هدایت می‌شود.

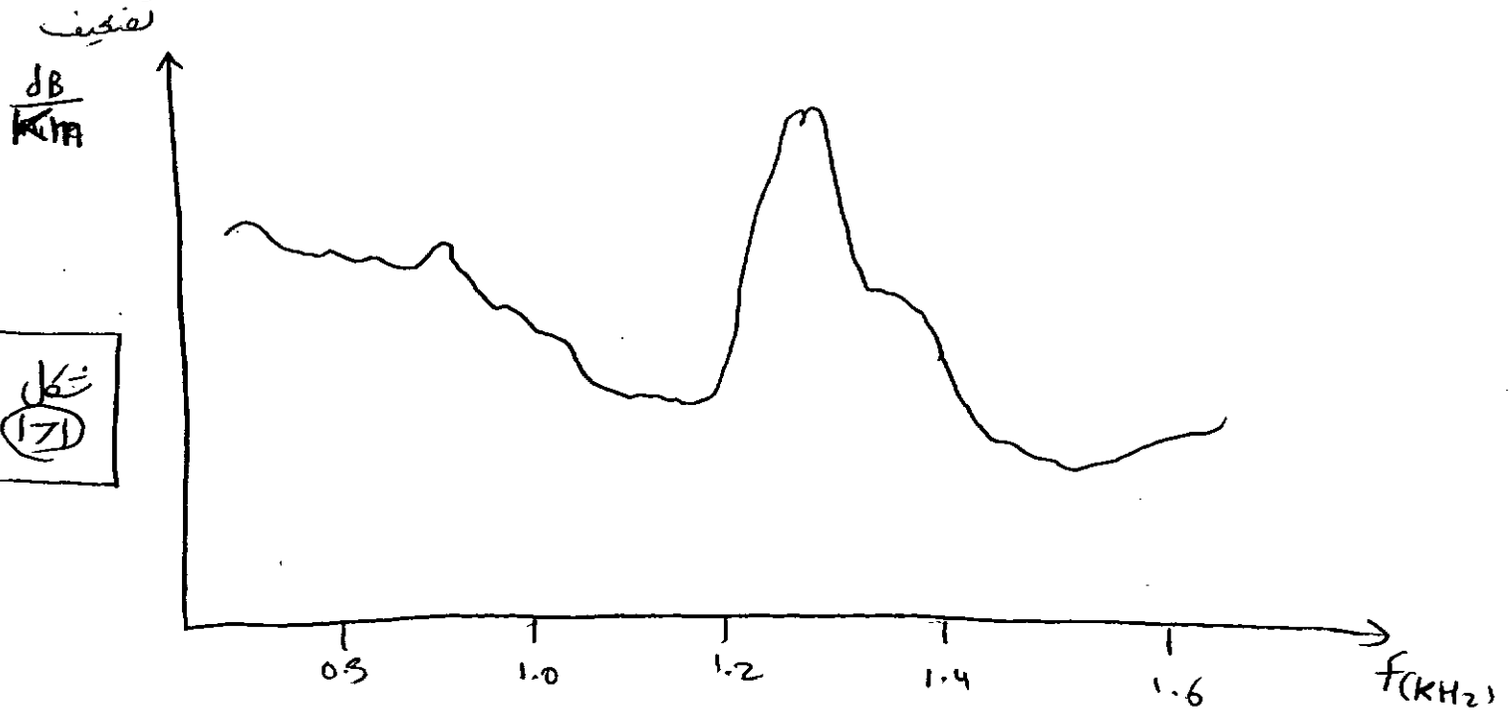
◎ MULTI MODE : چندین نور با زوایای تابش مختلف روی فیبر متشعر می‌شوند.



شکل (۱۷۰)

* معمولاً به جهت نرخ بیت بالا در BACK BONE ها از آن استفاده می شود.

بررسی تضعیف در فیبر نوری در فرکانسهای مختلف:



شکل (۱۷)

⊛ با توجه به اینکه تضعیف در فرکانسهای مختلف متفاوت است از فرکانسهای خاصی جهت انتشار علایم در فیبر نوری استفاده می شود. (طول موجهای خاصی)

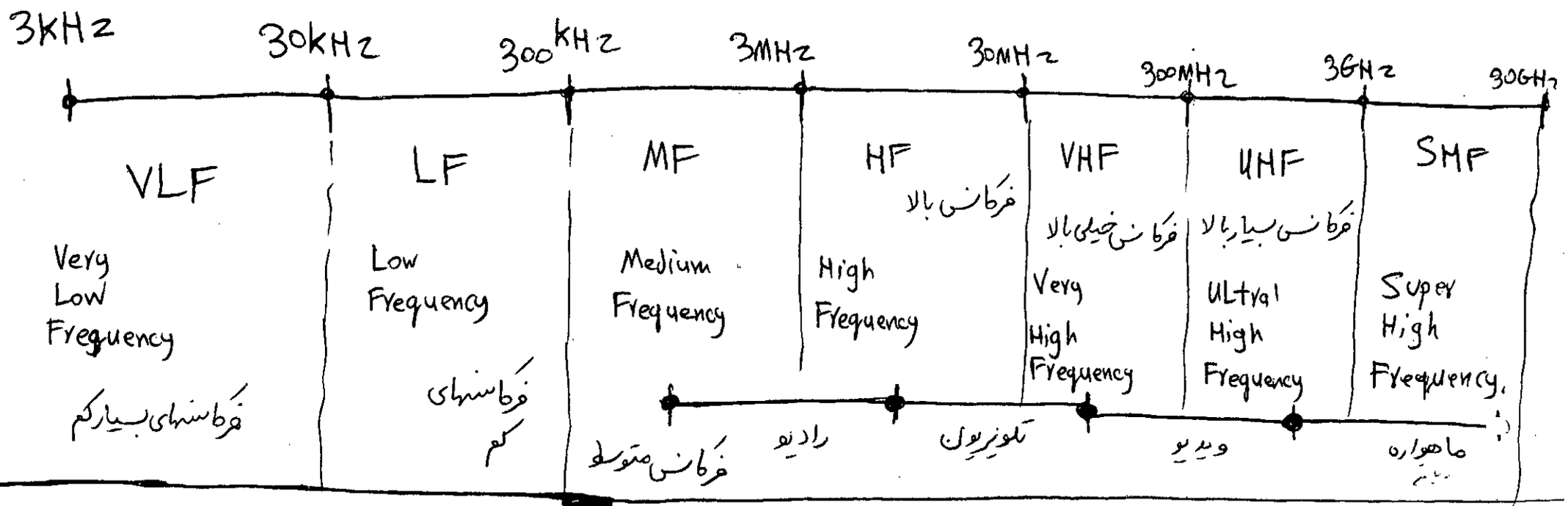
طول موج های مورد استفاده (فرکانسها) λ	850 nm	0.5 dB/Km	تضعیف Attenuation
	1300 nm	0.2 dB/Km	
	1550 nm	0.2 dB/Km	

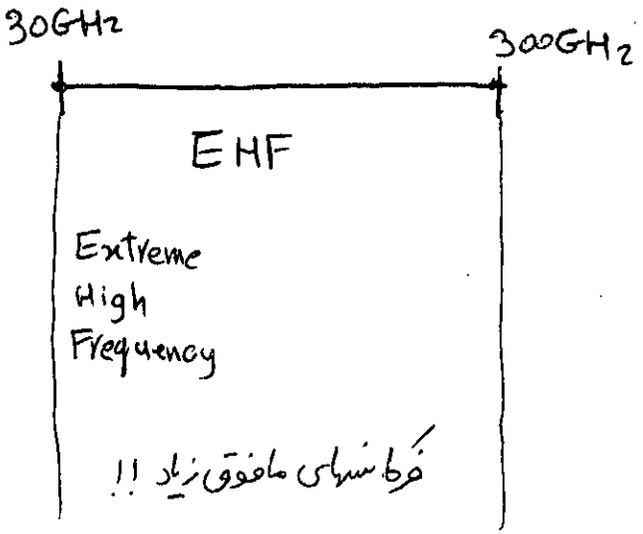
WAVE LENGTH DIVISION MULTIPLEXING [W.D.M]

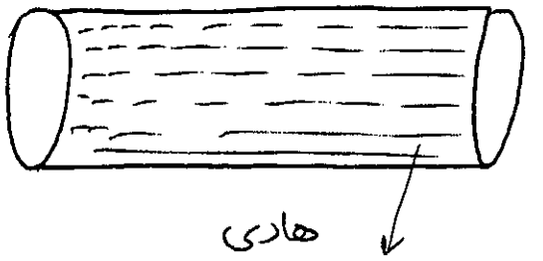
⊕ ارسال علایم با طول موجهای متفاوت روی یک فیبر.

* انشعاب گیری از فیبر نوری بسیار مشکل است لذا برای ارتباطات نودبه نود در Back Bone از آنها استفاده می شود.

جدول ③ ✱ FREQUENCY SPECTRUM طیف فرکانسی ✱







شکل (172) «اثر پوستی»

✱ هرچه فرکانس بالا برود الکترونها به سمت سطح رسانا متمایل می شوند. و انگار سطح مقطع کم شده و درنتیجه مقاومت بالا می رود.

انواع فیبرهای نوری :

الف)

10-Base-FP :

10 Mb/s

2 Km فاصله

850 nm طول موج

CODING: MANCHESTER

ب)

100-Base-FX :

100 Mb/s

2 km فاصله

800 nm طول موج

CODING: 4B5B inverted

ج)

1000-Base-X :

1000 Mb/s

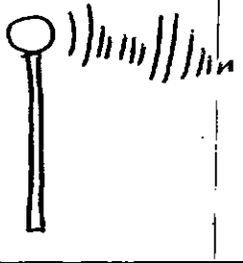
5Km فاصله

550 nm Multimode } فاصله
5 Km Single Mode }

CODING:

4- انتقال رادیویی : Radiation Transmission

ارسال سیگنال در فضا - توسط فرکانسهای بالا -
و یک آنتن.



* تلویزیون، رادیو، موبایل، LAN های بی سیم و ارتباطات ماهواره ای.

پدیده پوستی : SKIN EFFECT:

« بالارفتن مقاومت هادی در فرکانسهای بالا در اثر (کاهش سطح مقطع) حرکت الکترونها به سمت پوسته. »

پدیده تشعشع : RADIATION:

جداشدن سیگنال از سطح هادی در فرکانسهای بالا و تبدیل آن به امواج الکترومغناطیسی « Electro Magnetic »

برای ارسال سیگنال از طریق تشعشع طول هادی باید برابر طول موج باشد.

⊕ مثال:

$$\begin{cases} f_1 = 3\times10^6 \text{ Hz} \\ \lambda_1 = 100\text{m} \end{cases}$$

$\lambda = c / f$ (سرعت نور / فرکانس)

$$\begin{cases} f_2 = 30\times10^6 \text{ Hz} \\ \lambda_2 = 10\text{ m} \end{cases}$$

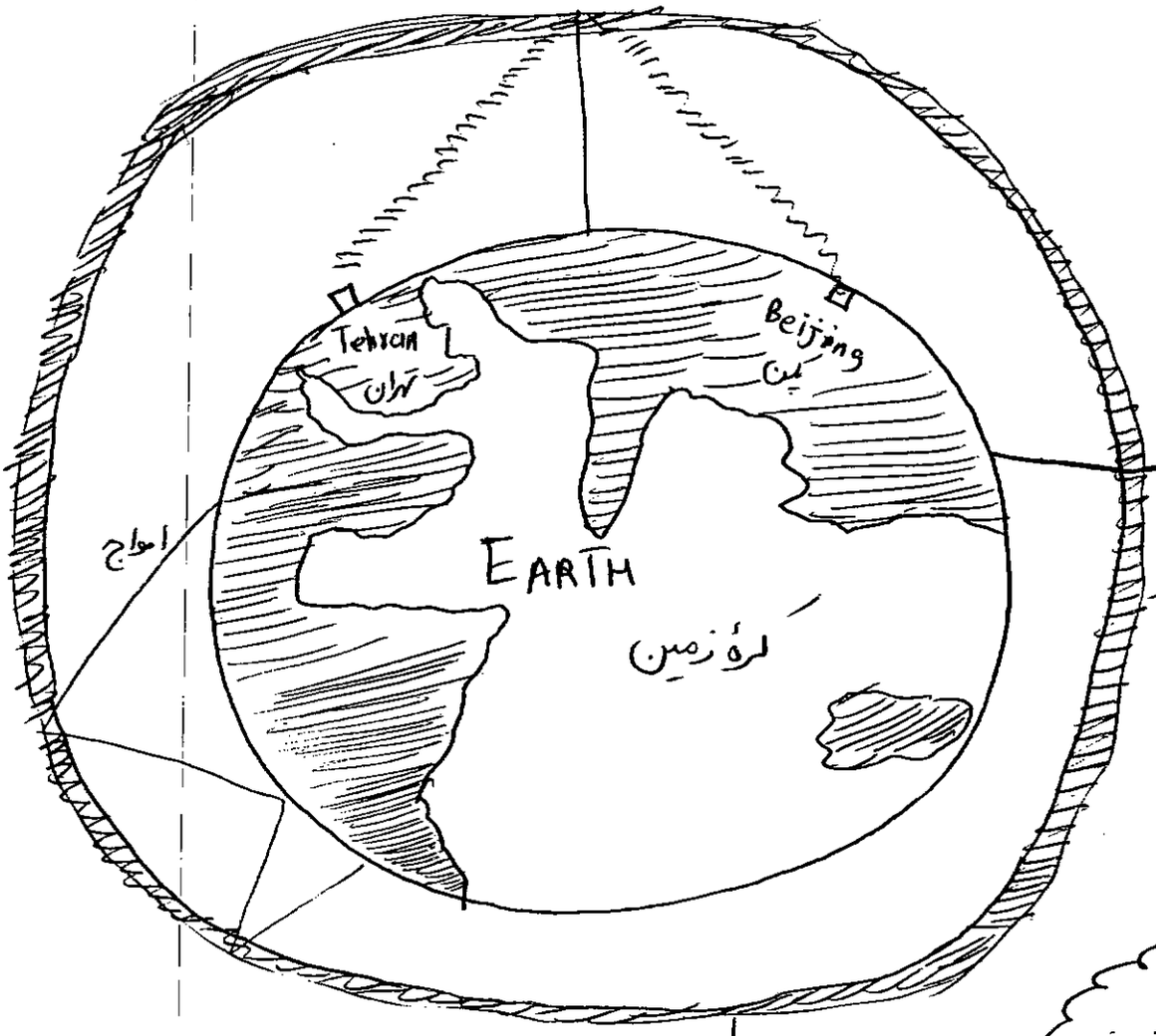
$$\begin{cases} f_3 = 300\times10^6 \text{ Hz} \\ \lambda_3 = 1\text{ m} \end{cases}$$

هرچه فرکانس بالاتر باشد، طول آنتن کمتری مورد نیاز است.

$$\begin{cases} f_4 = 3\times10^9 \text{ Hz} \\ \lambda_4 = 0.1\text{ m} \end{cases}$$

* موج FM رادیو [88 تا 107 MHz] به آنتنی حدود یک متر نیاز دارد.

ارتباطات رادیویی : از طریق لایهٔ یونوسفر

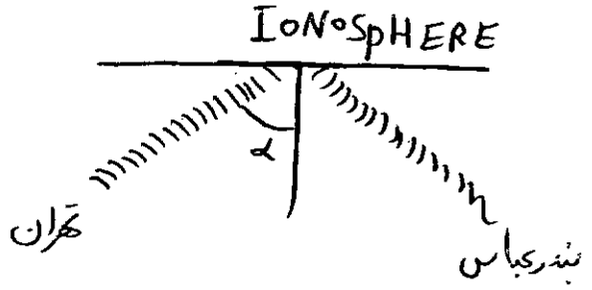


شکل (۱۳)
اثر و نقش لایهٔ یونوسفر در ارتباطات رادیویی در فرکانسهای نسبتاً متوسط.

* برای فرکانسهای کم، تشعشع در سطح زمین است و هرچه فرکانس بالاتر رود تشعشع به سمت بالا می رود.
* برای ارتباطات ماهواره‌ای فرکانسهای بالایی نیاز است تا سیگنال از لایهٔ Ionosphere عبور کند و به ماهواره برسد.

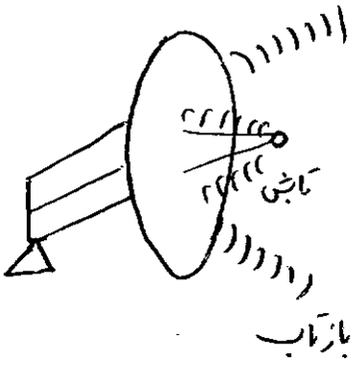
طول آنتن : ANTHENNA LENGTH:

طول آنتن باید برابر طول موج، امواج الکترومغناطیس باشد.



شکل (۲۶) بازتاب سیگنال در فرکانس‌های بالاتر از فرکانس بحرانی توسط لایه یونوسفر

* زاویه α با فرکانس سیگنال مرتبط است.
* برای فرکانس‌های کمتر تابش در سطح زمین است - و بالعکس -



* آنتن بشقابی: برای ارسال امواج مایکروویو.
* بحث بیشتر در مورد طراحی آنتن در درس به همین نام - درس آنتن - در دانشکده برق ارایه می‌گردد.

⊕ ارسال ماهواره‌ای: SATELLITE RADIATION

* استفاده از ماهواره به جای لایه یونوسفر جهت ارتباط ایستگاه‌های زمینی.

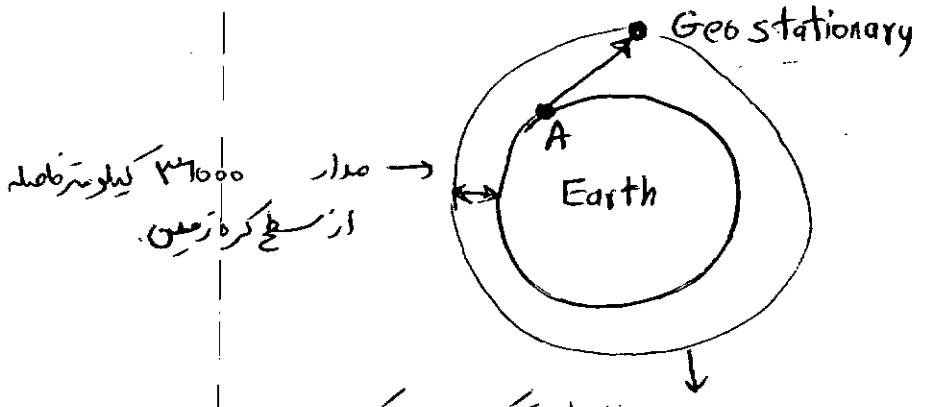
⊕ انواع ماهواره:

GeoStationary -1

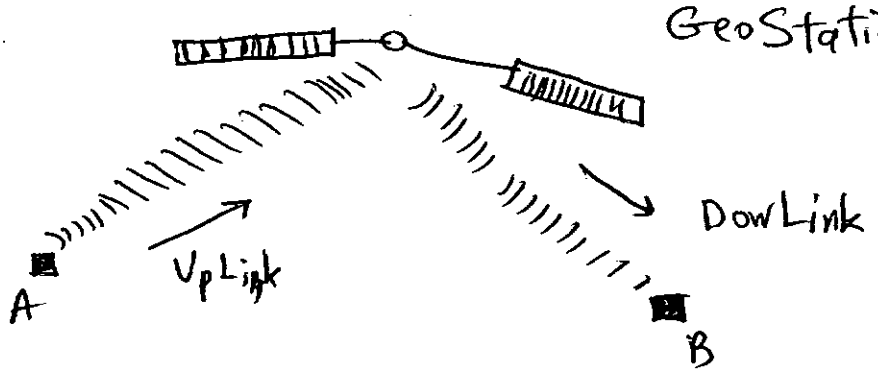
ایستا نسبت به زمین.
هر ۲۴ ساعت یک دور به دور کره زمین می‌گردد. فاصله حدوداً ۳۶۰۰۰ کیلومتر.
از دید ناظر روی زمین جای این ماهواره‌ها همواره ثابت است.

LEO Low Earth Orbit -2

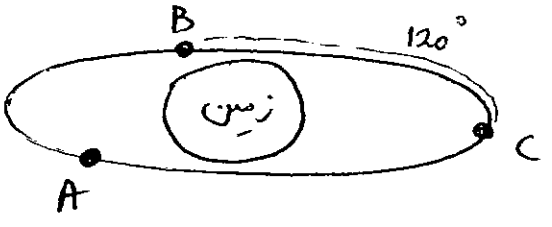
نزدیکتر به زمین است، و دارای سرعت بالاتر - تا جذب کره زمین نشود -
از دید ناظر روی زمین جای آن متغیر است.



شکل (175) ماهواره GeoStationary



شکل (176)



شکل (177)

★ قراردادن سه ماهواره LEO در یک مدار جهت پوشش دادن کل ساعات شبانه روز برای تمامی نقاط روی زمین.

⊕ ارتباطات :

1- UpLink : ارتباط از ایستگاه زمینی به سمت ماهواره

6 تا 14 GHz

2- Down Link : از ماهواره به زمین

4-11 GHz

⊕ خطای انتقال : TRANSMISSION ERROR

- کابل روبانی : طبیعت تصادفی RANDOM دارد. نویز حرارتی.
- Twisted Pair : $P = 10^{-6}$ احتمال خطای تیپیک.
- فیبر نوری : $P = 10^{-9}$ احتمال خطا
- ارسال رادیویی : $P = 10^{-3}$ به سبب تشعشعات لیهانی

! تشخیص و رفع خطا : ERROR DETECTION:

* در ارتباط تلفنی در صورت بروز خطا طرفین خود متوجه شده و در صدد رفع آن برمی آیند.
* ادامه ارتباطات شبکه های کامپیوتری چه باید کرد؟

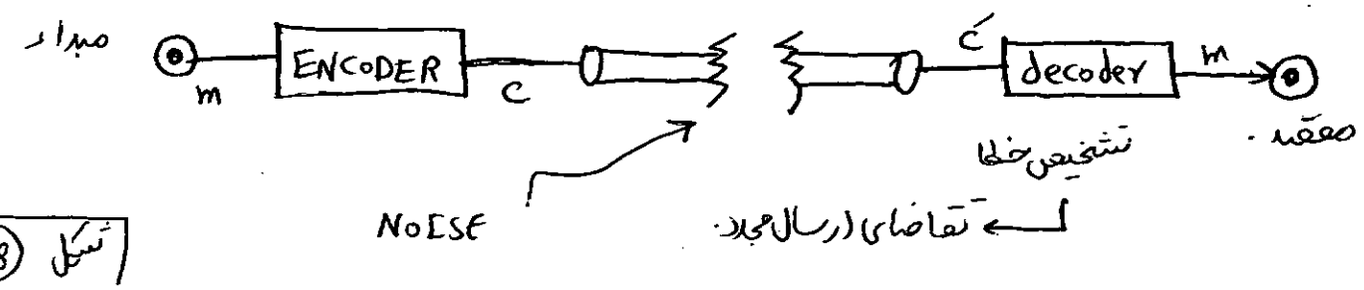
⊕ کدهای تشخیص خطا، کدهای تصحیح خطا:

⊕ روشها :

---

① روش FEED BACK :

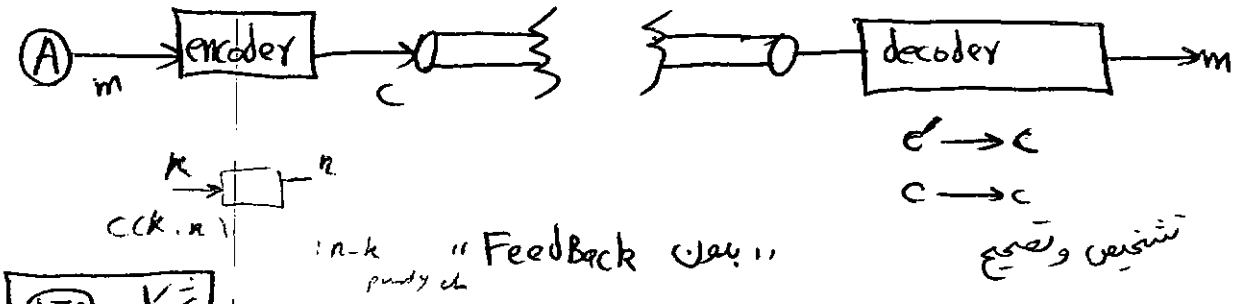
در حبداء مداری داریم به نام encoder ، دریافت کننده هم decoder داریم.
decoder در صورت تشخیص خطا تقاضای ارسال مجدد می دهد. سرنکردی



شکل (178)

۲ روش FORWARD ERROR CONTROL :

* باز هم مدارهای encoder و decoder داریم با این تفاوت که خود decoder در صورت بروز خطا خودش خطا را تصحیح می‌کند و پیام اصلی را بازیابی می‌کند.



شکل (۱۷۹)

بررسی : رشته‌ای n بیتی از کانالی در حال گذر است. اگر احتمال خطا رسیدن هر بیت p باشد:

+ احتمال درست رسیدن کل رشته :

$$P_1 = (1-p)^n$$

+ احتمال درست رسیدن $n-w$ بیت و خطا رسیدن w بیت :

$$P_2 = (1-p)^{n-w} p^w$$

$$\boxed{P_2 = (1-p)^n \left(\frac{p}{1-p}\right)^w}$$

نتیجه + اگر p بسیار کم باشد - که غالباً هست - احتمال آنکه تعداد بیتهای زیادی خراب شوند خیلی کم است.

/// تعریف :

فاصلهٔ همینگ : HAMMING DISTANCE

تعداد بیتهایی که دو رشته با هم در آن‌ها اختلاف دارند.

$a = 0010$
$b = 0011$ → distance = 1

نمونه

طرح اول

پیام m مانند زیر است:

$m = 00 \quad 01 \quad 10 \quad 11$

+ تعریف: CODE WORD

کلمه‌ای کد شده که از روی پیام اصلی و بنابر الگوریتم خاصی ساخته می‌شود.

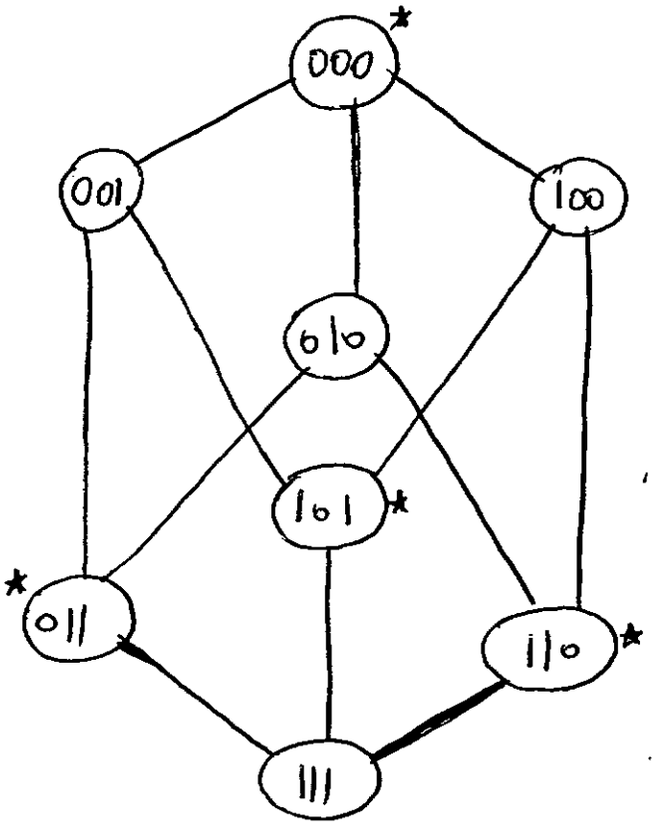
برای m مزبور فرض کنیم:

$C = 000 \quad 011 \quad 101 \quad 110$

ادامه

برای رشته‌های سه بیتی و براساس فاصلهٔ همینگ دیاگرام زیر را رسم می‌کنیم.

* هر خط بیانگر فاصلهٔ همینگ ۱ است.
* یعنی دو نود مجاور فاصلهٔ همینگ ۱ دارند.
* بیتهای ظاهر در C مزبور با ★ مشخص شده‌اند و فاصله همینگ همگی ۲ است.



شکل (۱۸۵)
« نمودار حالات »

* اگر در کانال فقط یک بیت خراب شود (فاصلهٔ همینگ ۱) گیرنده می‌تواند خطا را تشخیص دهد. زیرا کدهای موجود همگی با هم فاصلهٔ ۲ دارند.
* اگر دو بیت خطا داشته باشیم دیگر در این مثال خطا در طرف گیرنده قابل تشخیص نیست.

نمونه

طرح دوم

m	0	1
C	000	111

پیام
کلمهٔ کد

* در این جا یک بیت خطا قابل تشخیص است.

مجموعهٔ A = قابل تشخیص: 001، 010، 100 / درست: 000	000 ← خطا در یک بیت
مجموعهٔ B = قابل تشخیص: 110، 101، 011 / درست: 111	111 ← خطا در یک بیت

مجموعهٔ A و B کاملاً متمایزند.

⊕ نتیجه: گیرنده علاوه بر تشخیص خطا می‌تواند محل تقریبی خطا را نیز انجام دهد.
به این گونه که فاصلهٔ همینگ رشته بیت دریافتی اگر از 000 کمتر از 111 باشد m را صفر و اگر بیشتر باشد m را یک در نظر می‌گیریم.

(!) استثناء: اگر دو بیت خطا رخ دهد، خطا قابل تشخیص اما غیر قابل تصحیح است.

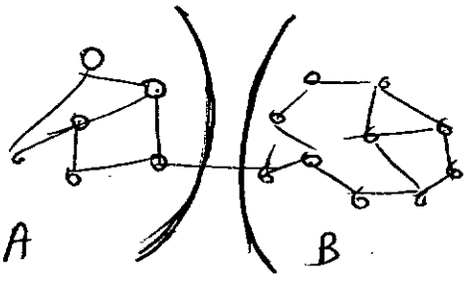
تعمیم

قانون کلی:

اگر فاصلهٔ همینگ مجموعه کلمات کد همگی برابر d باشد،

⊛ تا $d-1$ بیت خطا را می‌توانیم تشخیص دهیم.

⊛ تا $\left[\frac{(d-1)}{2}\right]$ بیت خطا را می‌توان تصحیح کرد.



تصحیح خطا:
در صورت دو مجموعهٔ مجزا برای هر کلامت.

بررسی چندین طرح کد: CODE WORDS:

① بیت توازن: PARITY BIT:

* از کدهای تشخیص خطا است.
* یک بیت بیشتر از پیام اصلی است.

اندازه $|c|$ = اندازه $|m|$

$$|c| = |m| + 1$$

⊕ روش: تمامی بیت‌های m با هم XOR می‌شوند و بیت حاصل به انتهای پیام اضافه می‌شود تا کلمهٔ کد را تشکیل دهد.

⊛ بیت آخر درصورت فرد بودن یک‌های m، یک و درصورت زوج بودن یک‌ها، صفر است

⊛ به این طرح EVEN PARITY می‌گوییم زیرا تعداد یک‌های کدهمواره زوج است.

$$m = b_1 b_2 b_3 \cdots b_{L-1}$$
$$c = b_1 b_2 b_3 \cdots b_{L-1} | b_L$$

$$b_L = b_1 \oplus b_2 \oplus \cdots \oplus b_{L-1}$$

$$b_L = \begin{cases} 1 & \text{تعداد یک‌های m فرد} \\ 0 & \text{" " m زوج} \end{cases}$$

⇦ طرح ODD PARITY:

در این طرح b_L مکمل b_L در طرح EVEN است.

$$m = b_1 \cdots b_{L-1}$$
$$c = b_1 \cdots b_L$$

$$b_L = (b_1 \oplus \cdots \oplus b_{L-1})'$$

$$b_L = \begin{cases} 1 & \text{تعداد یک‌های m زوج} \\ 0 & \text{" " " فرد} \end{cases}$$

⊛ در این طرح تعداد یک‌های کلمهٔ کدهمواره فرد است.

⊛ دریافت کننده - در طرح EVEN - کافی است همهٔ بیت‌های کد را با هم XOR کنیم که اگر حاصل صفر بود درست و اگر یک بود غلط دریافت شده است.

⊕ با این طرح تعداد فرد بیت خطا را می‌توان تشخیص داد و تعداد زوج را نمی‌توان.

① احتمال عدم تشخیص خطا در طرح Parity:

$$P_{\text{عدم تشخیص}} = P(\text{۲ بیت خطا}) + P(\text{۴ بیت خطا}) + \text{———}$$

$$P(\text{۲ بیت خطا}) = \binom{n}{2} p^2 \times (1-p)^{n-2}$$

(جایگشت دو بیت خطا ↗ ، دو بیت خطا ↑ ، بقیه درست ↑)

$$P(\text{۲K بیت خطا}) = \binom{n}{2K} p^{2K} (1-p)^{n-2K}$$

* Parity کاربرد بسیار زیادی در عمل دارد.
* خطای روی خود بیت توازن هم درگیرنده قابل تشخیص است.
* در این طرح هیچ گونه امکانات تصحیح خطایی نداریم.

② کد توازن دوبعدی:

2D PARITY CHECK

یا

BLOCK SUM CHECK

⊛ فرض کنیم m را به بلوکهای ۵ بیتی تبدیل کرده و بصورت ماتریس - آرایه - دوبعدی فرض می‌کنیم.

$$m = \left[\begin{array}{ccccc|c} 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ \hline 1 & 0 & 0 & 1 & 1 & ①* \end{array}\right]$$

+ در سطر و ستونها می‌توان از هرکدام از طرح های زوج یا فرد استفاده کرد.
برای سطر پایین که بدست آمده است، باز هم بیت توازن تشکیل می‌دهیم، این بیت را با * مشخص کرده‌ایم.

+ با فرض تعداد سطر L و تعداد ستون V در پیام اصلی تعداد بیت های اضافه شده برابر است با $L+V+1$ (سطر ↙ ، ستون ↓ ، * ↓)

⊛ نسبت بیت های اضافه شده :

$$RAT_{Io} = \frac{L+V+1}{(L+1)(V+1)}$$

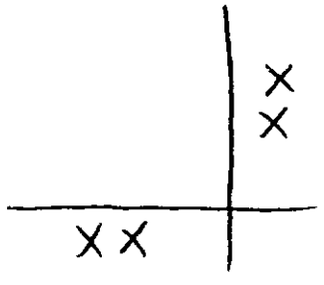
فرض کنیم بیت * خراب شده : (از یک به صفر)

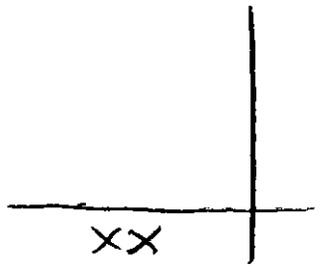
1	0	0	1	0	✓ 0
0	0*	0	0	0	× 1
1	0	0	1	0	✓ 0
1	1	0	1	1	✓ 0
1	0	0	1	1	✓ 1
✓	×	✓	✓	✓	

(1 بیت خطا)

محل تلاقی دو خطا - X - در سطر و ستون، خطای رخ داده است و قابل تصحیح است.

⊕ فرض کنیم دو بیت خطا رخ داده است :





خطا قابل تشخیص اما غیر قابل تصحیح است

⊕ خطای سه بیتی همواره قابل تشخیص است.

⊕ خطای چهار بیتی قابل تشخیص است مگر در حالتی که چهار بیت چهارگوشه یک مستطیل باشند

⊕ خطای پنج و شش و هفت قابل تشخیص است.

⊕ خطای ۸ بیتی قابل تشخیص است مگر در حالتی ۸ بیت چهارگوشهٔ دو مستطیل باشند.

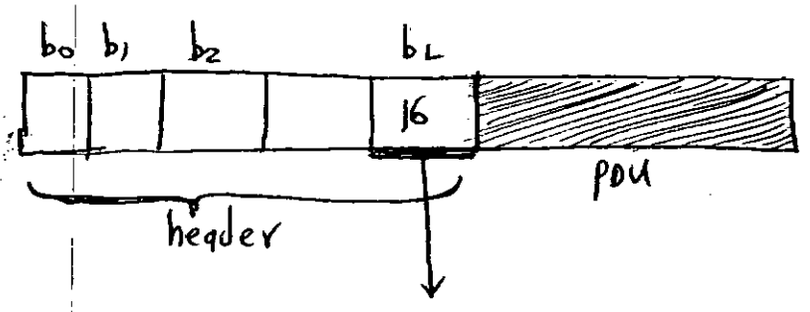
⊛ تمرین : تعداد حالتهای غیر قابل تشخیص در حالتهای ۸ و ۴ بیتی را در این مثال بدست آورید.

(E.e.2)

INTERNET CHECKSUM روش تشخیص خطای

(۳۵)

* لایهٔ اینترنت :



شکل (۱۸۱) INTERNET CHECKSUM

header	b0	b1	b2	———	bL-1	bL
	16	16	16		16	16

منطق روش :

$$x \equiv b_0 + b_1 + \text{———} + b_{L-1} \pmod{(2^{16}-1)}$$

جمع به پیمانهٔ $2^{16}-1$

$$\Rightarrow \quad 0 \leqslant x < 2^{16}-1$$

$$b_L = -x \pmod{2^{16}-1}$$

* در مبدأ فریم b_L به header افزوده می‌شود.

تشخیص خطا در طرف گیرنده :

$$y = b_0 + b_1 + \text{———} + b_{L-1} + b_L$$

y را چک می‌کند

$$\begin{cases} y = 0 & \text{خطا نداریم} \\ y \neq 0 & \text{خطا داریم} \end{cases}$$

⊛ سؤال: تا چند بیت خطا را می‌توان با روش INTERNET CHECKSUM تشخیص داد؟

(تا 16 بیت)

$2^{16}-1$ = 1111 ——— 1

مضارب آن = 111111 ——— 10

= 1111 ——— 100

= 1111 ——— 1000

---

4 کد چند جمله‌ای:

POLYNOMIAL CODE

↓ یا

CYCLIC REDUNDANCY CHECK

↓

CRC

⊛ فرض کنیم پیامی m بیتی داریم.

⊛ می‌خواهیم n بیت 0 به سمت راست m اضافه کنیم.

⊛ مولد کد: $n+1$ بیت که سمت چپ‌ترین آن‌ها 1 است.

⊛ عملیات خاصی انجام می‌دهیم به نحوی که $m0$ را بر مولد کد تقسیم می‌کنیم و در این عملیات به جای تفریق از XOR بیت به بیت استفاده می‌کنیم. باقیمانده‌ی نهایی R را به m اضافه کرده (به انتهای) و روی خط می‌فرستیم.

⊛ گیرنده mR را بر G تقسیم می‌کند (مولد کد) — همان تقسیم بالا —
اگر باقیمانده عمل صفر نبود خطایی رخ داده و اگر $R=0$ باشد خطایی رخ نداده و یا قابل تشخیص نیست.

⊕ توضیح: «همنهشتی»

$a \overset{n}{\equiv} b \Rightarrow a = b + kn \Rightarrow$

$a \mid n$, k خارج‌قسمت، b باقی‌مانده.

$\left\{ \begin{array}{l} a_1 = a+1 \\ a_2 = a+2 \\ \vdots \\ a_{n-1} = a+n-1 \end{array} \right\} \overset{n}{\not\equiv} b$

$a_n = a+n \overset{n}{\equiv\!\equiv} b$

$a_n \mid n$, k'

---

b

$17 \equiv 12 \pmod 5$

$17 \to 14 \to 11 \to 8 \to 5$

$17 \equiv 12 \equiv 7 \equiv 2 \equiv -3 \pmod 5$

$12 \not\equiv 5 \pmod 5$

$27 \to 24 \to 21 \to -6$

$27 \equiv 24 \equiv 21 \equiv -6 \pmod 5$

$a_1 + a_{n-1} \mid n$, h

$c \neq b$

RING

⊕ حلقه‌ی $\mathbb{Z}_n$:

$< \mathbb{Z}_n , +_n , \bullet_n >$

$+_n$: انجمنی ـ همانی ـ وارون پذیری ـ جابه‌جایی

$\bullet_n$: انجمنی

$(+, \bullet)$: جابه‌جایی، توزیع پذیری

⊛ اگر n اول باشد حلقه‌ی $\mathbb{Z}_n$ یک گروه آبلی و نیز یک میدان است.

⊛ $GF(n)$: به میدان Z_n درصورت اول بودن n، گالوا فیلد یا «میدان گالوای n» می گویند.

برای $GF(n)$:

مربنا جمع داریم

۰ = خنثی جمع

۱ = خنثی ضرب

⊕ تعریف میدان جدید :

$$a(x) = a_0 + a_1 x + a_2 x^2 + \text{———} + a_{n-1} x^{n-1}$$

جمله‌های با توان حداکثر $n-1$.

a_i ها از $GF(2)$ انتخاب می‌کنیم.

مثال : $n=3$:

$$a(x) = a_0 + a_1 x + a_2 x^2$$

اعضا :

۸ عضو

0
1
x
$1+x$
$1+x^2$
x^2
x^2+x
x^2+x+1

تعریف : جمع + میدان چندجمله‌ای‌ها:

$$a(x) + b(x)$$

+ را از میدان $GF(2)$ و مانند آن انجام می‌دهیم.

+ در میدان چند جمله‌ایها در واقع عمل XOR است. (در GF(2))

$$0+0 \overset{2}{\equiv} 0$$
$$0+1 \overset{2}{\equiv} 1 \quad \leftarrow XOR$$
$$1+0 \overset{2}{\equiv} 1$$
$$1+1 \overset{2}{\equiv} 0$$

* تفریق هم در این میدان به مانند XOR و به مانند جمع است. (در GF(2))

$$0-0 \overset{2}{\equiv} 0$$
$$0-1 \overset{2}{\equiv} 1 \quad \leftarrow XOR$$
$$1-0 \overset{2}{\equiv} 1$$
$$1-1 \overset{2}{\equiv} 1$$

⊙ عمل ضرب در میدان چند جمله‌ایها:

$$a(x) \times b(x)$$

$$x^2(x^2+x+1) = x^4+x^3+x^2$$

می‌بینیم که این چند جمله‌ای خارج از میدان شد، پس باید آن را بر چند جمله‌ای خاصی تقسیم کنیم:

* چند جمله‌ای کاهش ناپذیر از درجه‌ی n:

$g(x) = x^n + \dots$

«مانند عدد اول در حلقه Z_n که آن را به میدان تبدیل می‌کرد.»

◎ مثال برای $n=3$ ، $a_i \in GF(2)$:

$$x^4+x^3+x^2 \quad | \quad g(x)=x^3+x+1$$
$$x^4+x^2+1 \quad | \quad x+1$$
$$x^3 \oplus x$$
$$[-1=1] \quad x^3+x+1$$
$$-1 = \oplus 1$$

$$x^4+x^3+x^2 \triangleq 1 \bmod (x^3+x+1)$$

به میدان حاصل از چند جمله‌ای و اعمال جمع [XOR] و ضرب خاص، میدان گالوای 2^n می‌گویند.

◎ $GF(q^n)$:

: $a_0 x + \text{———} + a_{n-1}x^{n-}$

: $a_i \in GF(q)$

: $+$: $+_q$

: $0 = 0$ پس /

⊕ مثال : $GF(2^3)$:

: $a_0 + a_1 x + a_2 x^2$

: $(a_0, a_1, a_2) \in [0,1]$

$+$: XOR

: X : X پس /$(1+x^2)$ Modolo

نسبت دادن چند جمله‌ای های $GF(2^3)$ به رشته‌های سه بیتی منحصر بفرد.

	a_2	a_1	a_0
0	0	0	0
1	0	0	1
x	0	1	0
$1+x$	0	1	1
x^2	1	0	0
x^2+1	1	0	1
x^2+x	1	1	0
x^2+x+1	1	1	1
	x^2	x^1	x^0

(*) می‌توان برای جمع از رشته بیت‌ها استفاده کرد.

$$x^2 + (x^2+x+1) = 1+x$$

$$100 + 111 = 211 \equiv 011$$

نتیجه: به جای جمع و تفریق می‌توان از XOR بیت به بیت برای رشته‌های متناظر استفاده کرد. نتیجه نمایش رشته بیت جمع دوجمله‌ای است.

(⊗) **ضرب** دو میدان $GF(m^n)$

$x^2(x^2+x+1)$

مرحله اول:

$$\begin{array}{r} 100 \\ 111 \\ \hline 100 \\ 100 \\ \hline 100 \\ \hline 11100 \end{array} \sim x^4+x^3+x^2$$

x^2	100
x^2+x+1	111
$x^4+x^3+x^2$	11100

مرحله دوم:

$$\begin{array}{l|l} x^4+x^3+x^2 & x^3+x+1 \\ x^4+x^2+x & \hline (x+1)_q \\ \hline x^3+x & \\ \downarrow & \\ x^3+x+1 & \\ \hline -1 & \\ \downarrow & \\ (+1)_R & \end{array}$$

$g(x) \simeq G$

$$\begin{array}{r|l} 11100 & 1011 \\ 1011 & \hline (11)_q \\ \hline 01010 & \\ 1011 & \\ \hline (0001) & \end{array}$$

R

$R = +1$

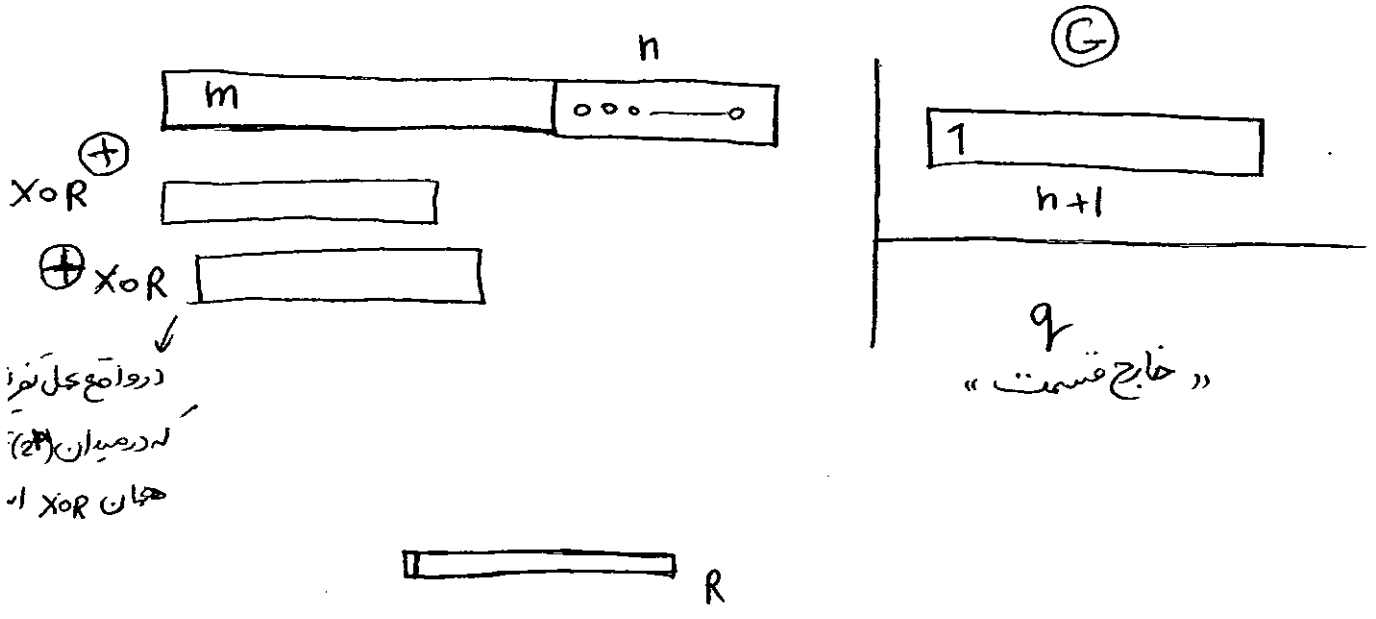
$R = 1$ بیت

$q = 1+x$

$q = 11$ بیت

کد CRC :

نشان می‌دهیم که عملیات CRC به مانند عملیات تقسیم (مرحله‌ی دوم عمل ضرب در میدان $GF(2^m)$) است.



صحت‌سنجی روش CRC :

m = پیام
n = n تا صفر

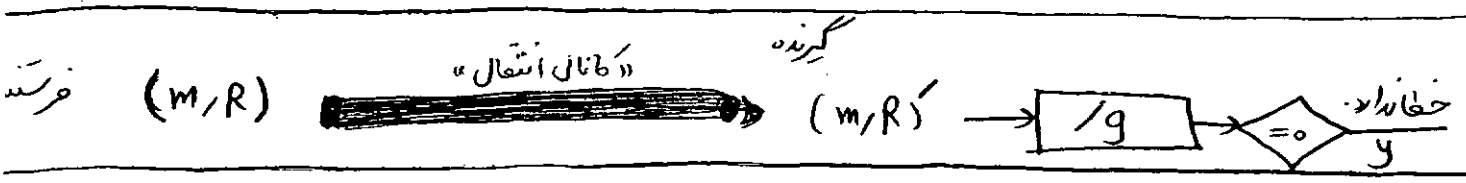
$$mn = qG + R$$

$$mn - R = qG$$

$$mn + R = qG$$

(مانند)

$$\Rightarrow \boxed{mR = qG}$$



$$M(x) \cdot x^n \equiv mn$$

$$M(x) \cdot x^n \mid G(x)$$

$$M(x) \cdot x^n - R = M(x) \cdot x^n + R$$

$R(x)$ → باقی مانده

« رشته بیت ارسالی » $\boxed{M(x) \cdot x^n + R}$

⊗ مثال ساده :

اگر یک بیت خطا : $e(x) = 1$

$M(x) \cdot x^n + R + 1 \longrightarrow R' = 1 \neq 0 \Rightarrow$ error

⊕ اگر خطا صفری از $g(x)$ باشد گیرنده دیگر نمی تواند خطا را تشخیص دهد.

⬅ کلیه خطاها با طول n تشخیص داده می شوند.

(تعداد n صفر سمت راست)

⊛ در مورد خطاهای بیش از n بیت نمی توان نظر داد.

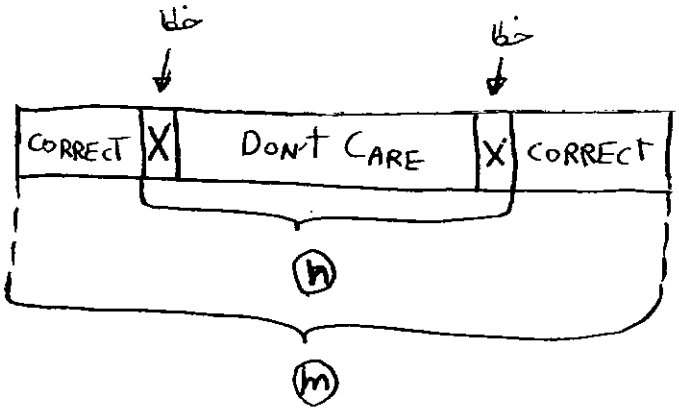
8 خطای BURST به طول n :

کلیه خطاهای به طول کمتر از n بیت (اول و آخر خطا)

⊛ در پروتکل های ETHERNET معمولاً n را 16 یا 32 می گیرند.

Appendix

« نکاتی درباره خطای BURST به طول n »



⊛ خطای بالا برای رشته m بیتی، خطای BURST به طول n است.

⊛ مکان بیت اول و آخر BURST دلخواه است.

⊛ بیت اول و آخر BURST حتماً خراب و بیت های بقیه اهمیتی ندارند.

⊛ بقیه بیت های m که خارج از BURST هستند نیز ممکن است صحیح باشند.

⊛ علت استفاده فراوان از کد CRC چند چیز است - پیاده سازی بسیار آسان آن است.

● حال به بررسی نحوه پیاده سازی کد CRC می پردازیم.

"CRC IMPLEMENTATION"

مثال: $GF(2^n)$

$M(x) = x^7 + x^6 + x^5 + x^2 + x$ ← $M(x)$

$m = 11100110$

$x^3 \cdot M(x) = x^{10} + x^9 + x^8 + x^5 + x^4$ ← ضرب $M(x)$ در x^3

$m' = 11100110000$

$x^4 \cdot M(x) = x^{11} + x^{10} + x^9 + x^6 + x^5$ ← ضرب $M(x)$ در x^4

$m'' = 111001100000$

$M(x) \cdot P(x) = M(x) \cdot (x^4 + x^3 + 1) = M(x) \cdot x^4 + M(x) \cdot x^3 + M(x)$

$\Rightarrow M(x) \cdot P(x) = x^{11} + 2x^{10} + 2x^9 + x^8 + x^7 + 2x^6 + 3x^5 + x^4 + x^2 + x$

کاهش ضرایب به $GF(2)$

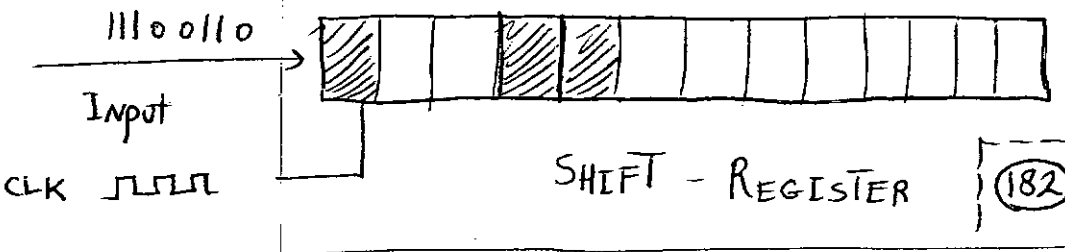
$M(x) \cdot P(x) = x^{11} + x^8 + x^7 + x^5 + x^4 + x^2 + x$

$M(x) \cdot P(x) = m + m' + m''$ جمع رشته های متناظر

```
   11100110
XOR 11100110000
    111001100000
    ------------
    100110110110  →  x^11 + x^8 + ____ + x
```

⊛ پس عمل جمع به راحتی توسط XOR انجام می شود.

⊛ برای عمل ضرب از یک شیفت رجیستر استفاده می کنیم.

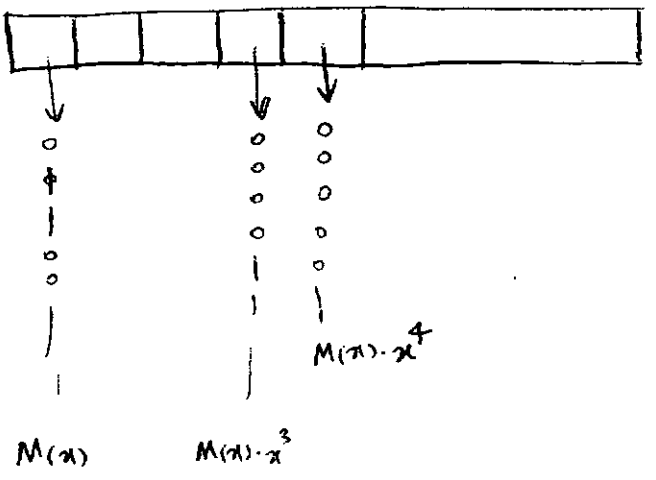


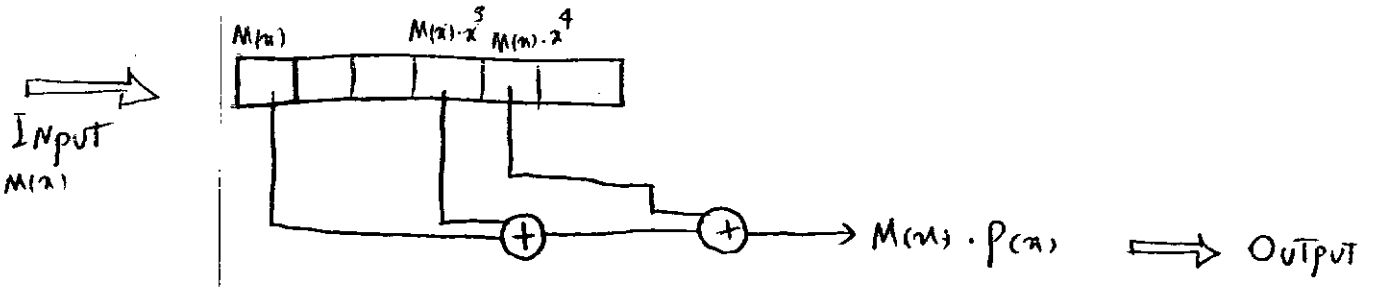
شکل (182)

⊛ ابتدا تمام بیت‌های رجیستر را صفر قرار می‌دهیم.

← رشته‌ی حاصل از بیت اول $M(x)$ است.

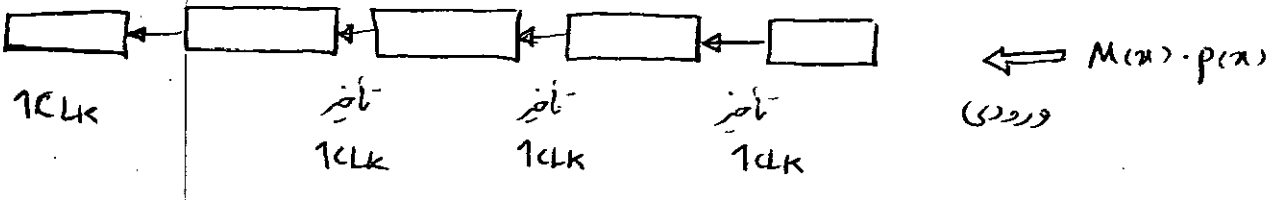
← رشته‌ی حاصل از بیت سوم $M(x)\cdot x^2$ است و همین ترتیب.

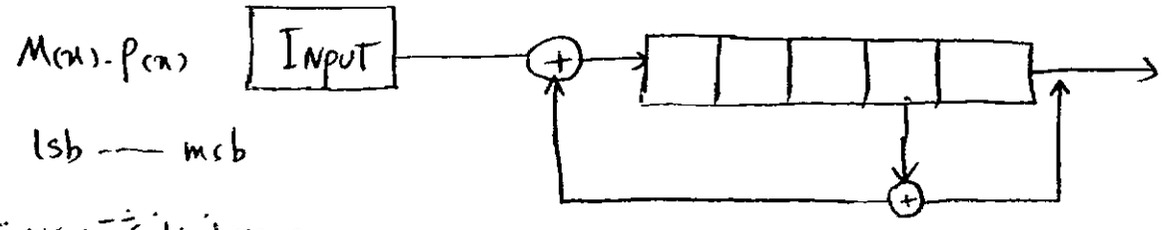




شکل (183) مدار ضرب کننده $M(x)\times P(x)$

⊛ برای عمل تقسیم از مدار بالا در حالت معکوس زمانی استفاده می‌کنیم (با کمی اصلاح و تغییر)





شکل (184) مدار تقسیم در $GF(2^p)$

* خارج‌قسمت از سمت راست (از پرارزشترین بیت) خارج می‌شود.
* باقیمانده در داخل شیفت رجیستر باقی می‌ماند.

تمرین ۳: با استفاده از مدارهای مذکور کد CRC بسازید با شرایط زیر:

E.e.3

$GF(2^4)$

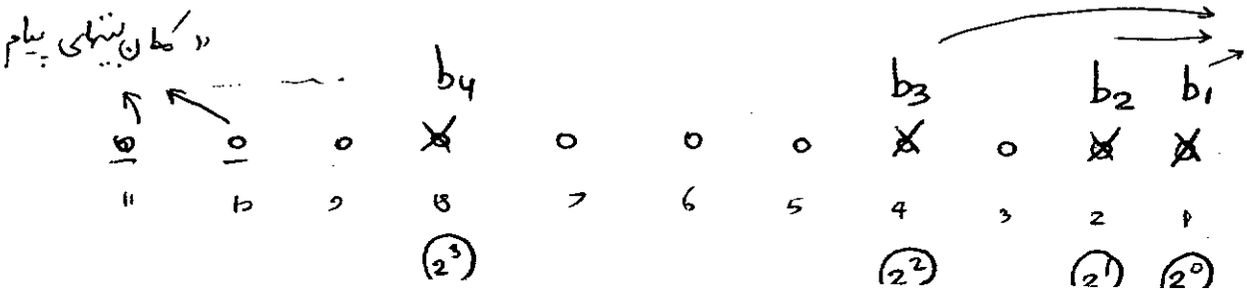
مولد کد $g(x) = x^4 + x^3 + x$

مثال خاص $M(x) = 1110011_0$

---

5 کد همینگ: HAMMING'S CODE

* کد تشخیص و تصحیح خطا.
* فرض کنیم پیام m مورد نظر است.
* در این روش مکانهای توان ۲، مکان رزرو شده‌اند. که کد در آنها جای می‌گیرد نه پیام.



فرض کنید پیام m به شکل زیر باشد:

$m = 1001101$

c(7,4,3)

پیام m بیت دارد که این k بیت پاریتی است $m = 2^k - k - 1$

این پیام باید در مکان‌های زیر قرار گیرد.

b_{11}	b_{10}	b_9	b_8	b_7	b_6	b_5	b_4	b_3	b_2	b_1	
1	0	0	—	1	1	0	—	1	—	—	← مکان‌های رزرو شده برای کد

حال مکان 1 های پیام را به دست می آوریم:

و دودویی آنها را XOR می‌کنیم.

$b_3 \leftarrow 3$	=	0011
$b_6 \leftarrow 6$	=	0110
$b_7 \leftarrow 7$	=	0111
$b_{11} \leftarrow 11$	=	1011
		1001 → Code which Must be located in Reserved locations.

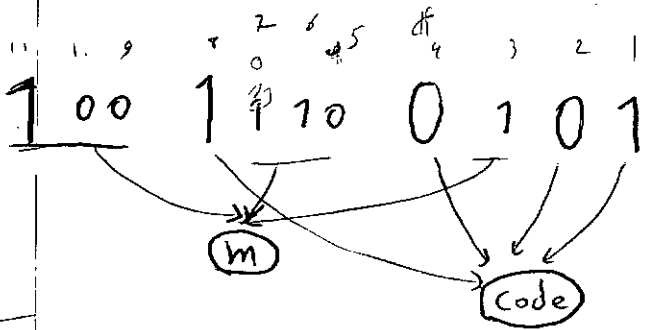
(XOR)

عدد بدست آمده را در مکان‌های رزرو شده قرار می‌دهیم و پیام را ارسال می‌کنیم.

1 0 0 1 1 1 0 0 1 0 1

m و c :

کد و پیام (بصورت درهم آمیخته)



طرف گیرنده:

مکان 1ها را بدست می آوریم و باز هم دودویی آنها را با هم XOR می‌کنیم.

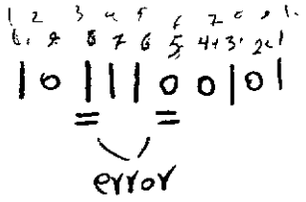
اگر حاصل صفر شد یعنی خطایی نداریم! (چرا؟)

اگر حاصل صفر نشد یعنی خطایی رخ داده است و اگر یک بیت خطا رخ دهد حاصل همان بیت خراب است.

⊛ مثالی از حالتی که دو بیت خطا رخ داده است:

فرض کنیم دو بیت در مکانهای ۶ و ۹ قرار دارند:

دریافتی داریم:



عملیات:

$$\begin{aligned} 1 &= 0001 \\ 3 &= 0011 \\ 7 &= 0111 \\ 8 &= 1000 \\ 9 &= 1001 \\ 11 &= 1011 \\ \hline & 1111 \end{aligned}$$

→ error

+ پس خطا قابل تشخیص است اما قابل تصحیح نیست.

---

⑤ کد همینگ:

1 بیت خطا را تصحیح می کند.

2 بیت خطا را تشخیص می دهد.

بیش از 2 بیت خطا ← ممکن است و معمولاً تشخیص نمی دهد.

⇐ نتیجه:

فاصلهٔ کلمات کد همینگ (فاصلهٔ همینگ!) ۳ است.

---

⑤ محاسبات کد:

پیام‌های ۴ بیتی:

فرستنده:

$$M \equiv (b_3, b_5, b_6, b_7)$$

کد کلمات زوج شده:

$$\begin{cases} b_1 = b_3 \oplus b_5 \oplus b_7 \\ b_2 = b_3 \oplus b_6 \oplus b_7 \\ b_4 = b_5 \oplus b_6 \oplus b_7 \end{cases}$$

⊕ در طرف گیرنده:

$b_1 \oplus b_3 \oplus b_5 \oplus b_7 = e_0$

$b_2 \oplus b_3 \oplus b_6 \oplus b_7 = e_1$

$b_4 \oplus b_5 \oplus b_6 \oplus b_7 = e_2$

پیام دریافتی

$e_6 e_1 e_2$ = 0 عدم خطا

= n مکان خطا برای یک بیت خطا

← با عوض کردن جای b_3 و b_4 می توان کد را در سمت راست پیام قرار داد:

HAMMING CODE

M	C
4 bit	3bit

$c_7 . c_6 . c_5 . c_4$ پیام

$c_3 . c_2 . c_1$ کد

$$c_1 = c_4 \oplus c_5 \oplus c_7$$
$$c_2 = c_4 \oplus c_6 \oplus c_7$$
$$c_3 = c_5 \oplus c_6 \oplus c_7$$

«مدار مولد کد»

⊕ دستگاه معادلات خطی کد همینگ:

$$[c_7 \ c_6 \ c_5 \ c_4] \begin{bmatrix} 1000 & 111 \\ 0100 & 110 \\ 0010 & 101 \\ 0001 & 011 \end{bmatrix} = [c_7 c_6 c_5 c_4 \cdot c_3 c_2 c_1]$$

(Unit | logic)

$$M_{1\times4} \times Filter_{4\times7} = Code\ Word_{1\times7}$$

⊕ به این کد، یک Code(7,4) می گویند به این معنی که پیام ۴ بیت و کلمه کد ۷ بیت است.

کد خطی : LINEAR CODE

تعریف : کدی که توسط ضرب پیام اصلی در یک ماتریس خاص قابل حصول باشد.

- کد هیمنگ یک کد خطی است.
- در کد هیمنگ (7,4) ، 3/4 سربار کد داریم.
- با اضافه کردن بیتهای پیام سربار کد کمتر می‌شود. مثلاً در مورد پیام ۱۱ بیتی به ۴ بیت کد هیمنگ نیاز داریم که سربار کمتری شود. 6/11 .
- اما با اضافه کردن بیتها احتمال تعداد خطای بالا، بیشتر می‌شود و کارآیی کد کم می‌شود.

دستگاه معادلات خطی در طرف گیرنده :

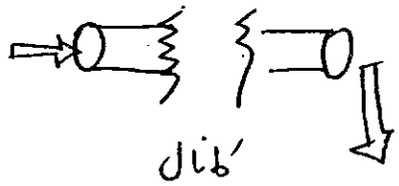
$$[c_7 \text{ —— } c_1] \left[\begin{array}{ccc|c} 1&1&1 & \\ 1&1&0 & \text{logic} \\ 0&1&1 & \\ \hline 1&0&0 & \\ 0&1&0 & \text{UNit} \\ 0&0&1 & \end{array}\right] = [e_1\ e_2\ e_3]$$

$\text{Code}_{1\times7}$　　$\text{Parity check}_{7\times3}$　　error check 1×3

خلاصهٔ کد هیمنگ :

$M \cdot G = MC$

پیام　ماتریس مولد کد　«کدو پیام دریم آمیخته»



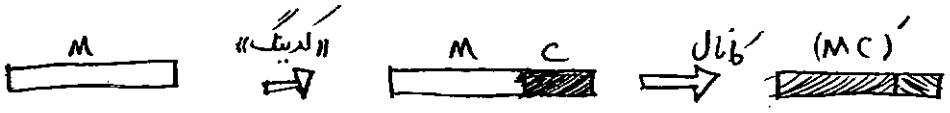
$MC \cdot PC = \text{Error Matrix}$

کد　Parity check

انواع کدها:

(A) کدهای قطعه‌ای: BLock Codes:

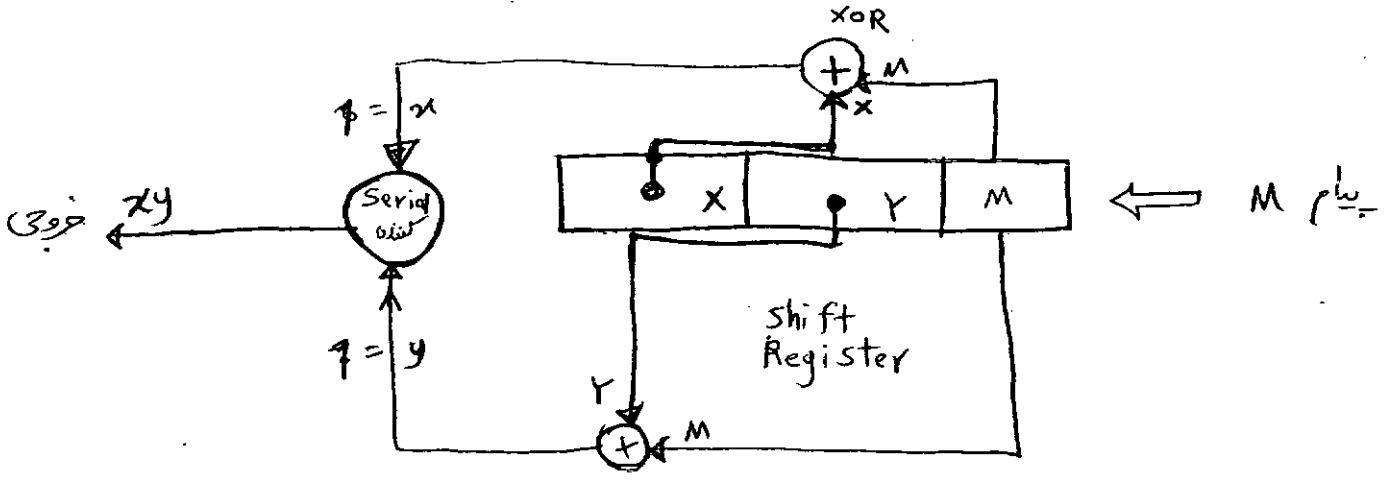
به پیام اصلی قطعه‌ای افزوده می‌شود تا توسط آن و پیام، گیرنده تشخیص یا تصحیح خطا کند.



(B) کدهای CONVOLUTION «کانولوشن»

تشخیص خطا نه تنها به پیام وارده بلکه به پیام‌های قبلی آن هم مربوط است.
در واقع در این نوع کد، مدارها حافظه‌دار هستند.
و در آنها معمولاً از رجیسترها استفاده می‌شود.

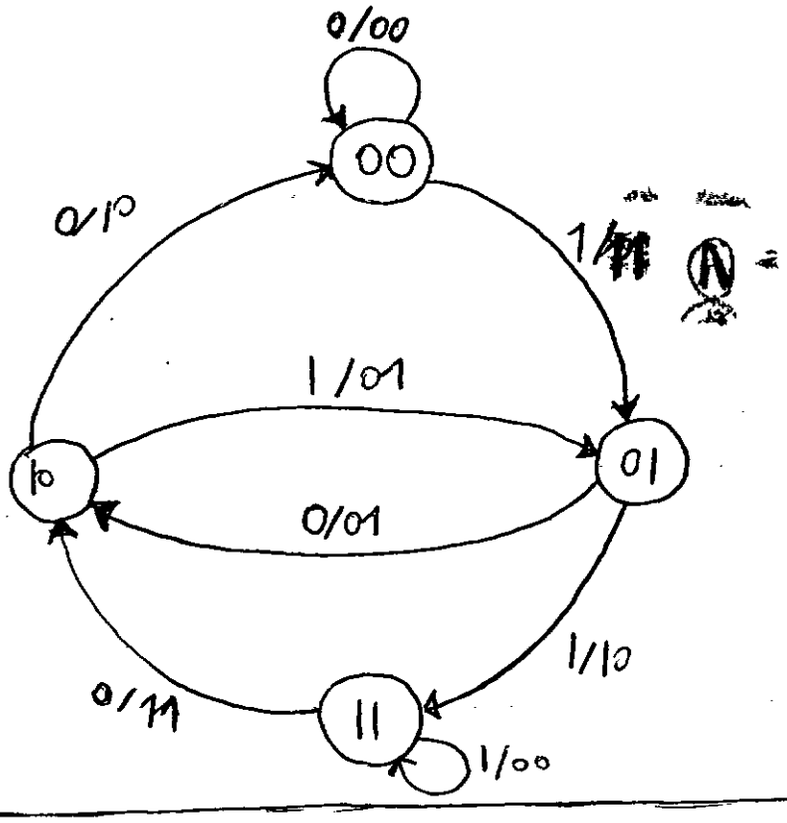
⊕ مثالی از کدهای CONVOLUTION:



شکل (185)

⊛ صادر ترتیبی چهارحالته بالا مولد کد است.

مدار ترتیبی را رسم می‌کنیم: STATE DIAGYAM



شکل (186)
مدار چهار حالته
کد کانولوشن
دارای نمودار
حالت روبرو است

و همچنین جدول حالات:

حالت فعلی	00	01	10	11
ورودی	1/0	1/0	1/0	1/0
خروجی	11/00	10/01	01/10	00/11
حالت بعدی	01/	11/10	01/00	11/10

جدول شمارهٔ (۶)

هر پیام کدی با طول دو برابر دارد که در طرف گیرنده به راحتی قابل تبدیل به پیام اصلی می‌باشد.

(*) دو بیت دو بیت جدا می‌کنیم و با توجه به حالت فعلی و حالت قبلی بیت اصلی پیام را به دست می‌آوریم.

11 10 11 01 01 01
↓ ↓ ↓
1 1 _ _ _ _

⊕ تمرینهای فصل سوم : (سری دوم)

32, 33, 35, 38, 42, 43, 44, 51, 54

⊕ ادامهٔ کد CONVOLUTION :

00 = A
01 = B
10 = C
11 = D

« مدار چهار حالتهٔ لدنیگ »

m = 1 1 0 1 0 1

code = 11 10 11 01 01 01 = DCDBBB

گذردن یک پیام نمونه.

⊕ در طرف گیرنده حل دیکدینگ با توجه به نمودار حالات انجام می‌شود. (حالت اولیهٔ A در نظر گرفته می‌شود.)

⊕ تا وقتی که خطایی رخ ندهد مشکلی نداریم.

⊕ حال به بررسی موارد نمونه‌ای از بروز خطا بپردازیم :

⊕ فرض کنیم به‌جای DCDBBB پیام خراب شدهٔ DDDBBB به گیرنده برسد.

code = 11 10 11 01 01 01 (error)

C = 11 11 11 01 01 01 (error)

⊕ در اینجا امکان تشخیص خطا هست ولی امکان تصحیح آن نیست.

⊛

از لایهٔ فیزیکی، پیام من گذر کرد
گویا خطای Burst ای در آن پیام بوده
یارم نکرده تصحیح! حتی نداده تشخیص!!
ای لایهٔ Data Link، به آه و داد من رس

یارم پیام را دید، احوال ما بتر کرد؛
که طولش از n یار، بسیار بیش بوده
افسوس از این CRC، نفرین بر این Trellis
پیام M دوباره، برای یار بفرست

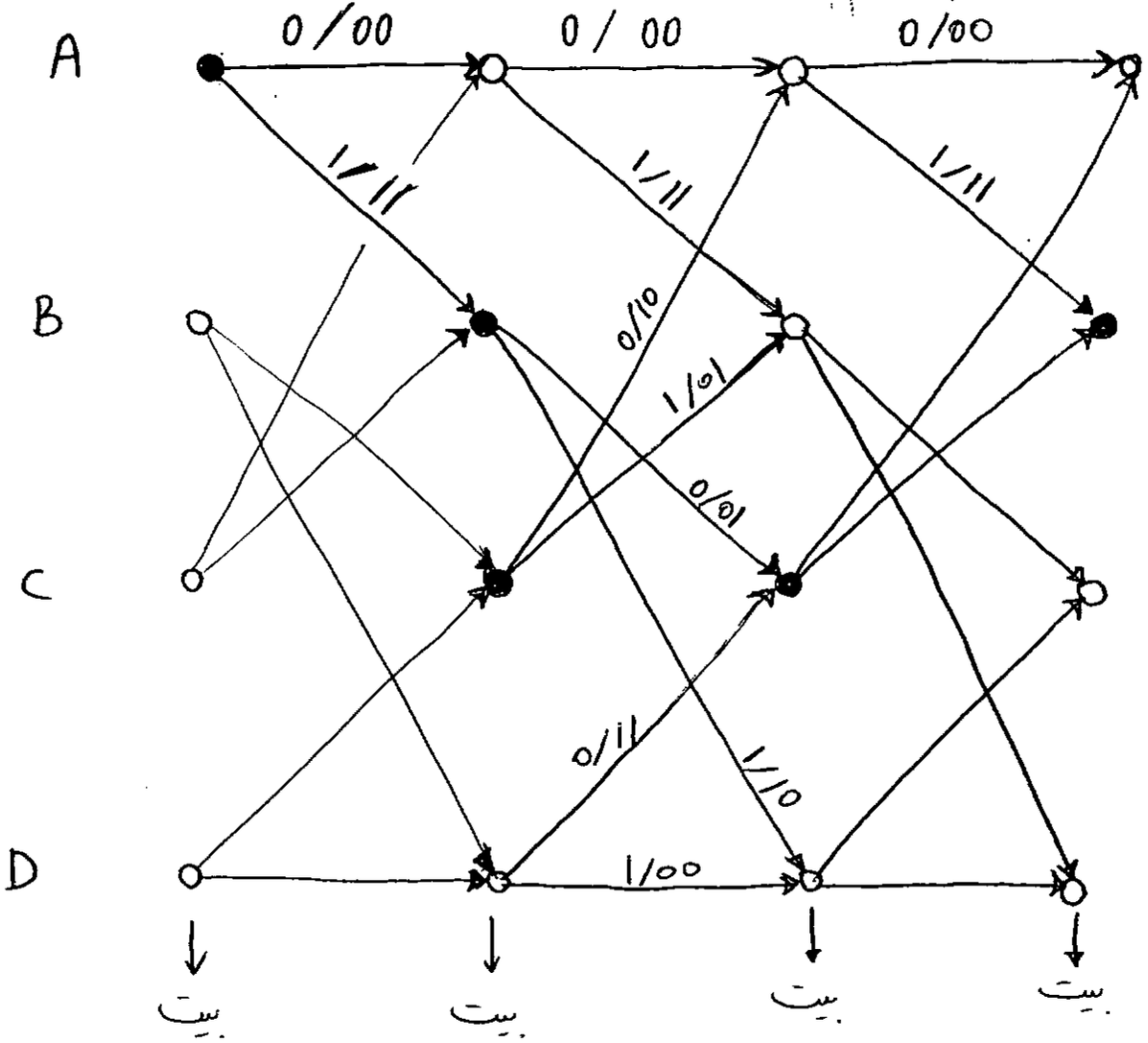
۸۴/۲/۵

⊕ نمودار ترلیس : TRELLIS DIAGRAM :

با فرض اینکه حالت اولیه‌ها A باشد.

○ : بیت

● : بیتهای رشته‌پیام مثال 11101



شکل (187) نمودار TRELLIS

با فرض اینکه 1 بیت خطا داشته باشیم :

می‌توان از بین مسیرها، مسیری که فاصلهٔ همری دارد را انتخاب کرد.

که با این طریق تصحیح خطا هم انجام می‌شود.

کد CONVOLUTION یک کد code (2,1) است.

⊕ با این روش کد DDDBBB به کد DCDBBB تصحیح می شود.

⊛ درصورت بروز دو بیت خطا:

تعدادی از پیغام ها خراب کد برداری می شوند ولی بقیه را می توان درست بازیابی کرد.

❓ به عنوان تمرین فرض کنید:

الف) دو بیت روی یک زوج بیت خطا داشته باشند.

ب) دو بیت روی زوج بیت های مختلف باشند.

و در هر دو حالت کلاردکد برداری توسط نمودار تریلیس را بررسی کنید.

⑨ Extension: اگر مدار چهار حالته را به مدار ۸ حالته ای تبدیل کنیم (شیفت چپ ۳ بیتی)، خطاهای به طول بیشتری را می توانند تشخیص و تصحیح شوند.

⊕ نکته در مورد خطای BURST

خطای $Burst_1$: n حالت

خطای $Burst_2$: n-1 حالت

خطای $Burst_3$: $2 \times (n-2)$ حالت ($n-2$ × 2)

خطای $Burst_4$: $4(n-3)$ ($n-3$ × 4)

شکل (188) خطای BURST

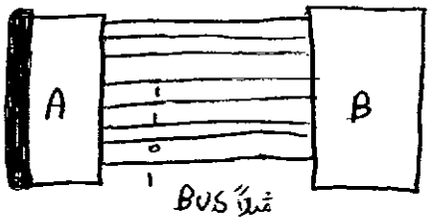
⊕ خطای BURST به طول n خطایی است که اول و آخر آن خطاست و مابین این دو خطا n-2 بیت خطا یا صحیح وجود دارد.

⑤ انتقال موازی و متوالی :

PARALLEL & SERIAL TRANSMISSION

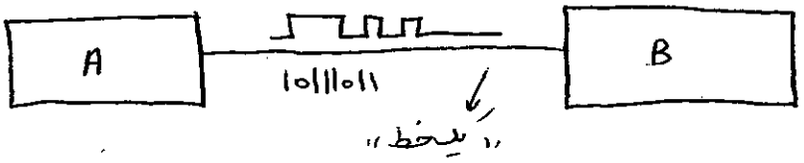
★ در ارتباطاتی که طرفین فاصلهٔ زیادی دارند، انتقال موازی هزینهٔ بالایی می‌برد که نتیجتاً از انتقال سریال استفاده می‌شود.

شکل (189) ارتباط Parallel موازی



PARALLEL

شکل (190) ارتباط SERIAL سریال



SERIAL

★ در صحبت شبکه‌های کامپیوتری عموماً از انتقال سریال استفاده می‌شود.

انواع انتقال سریال :

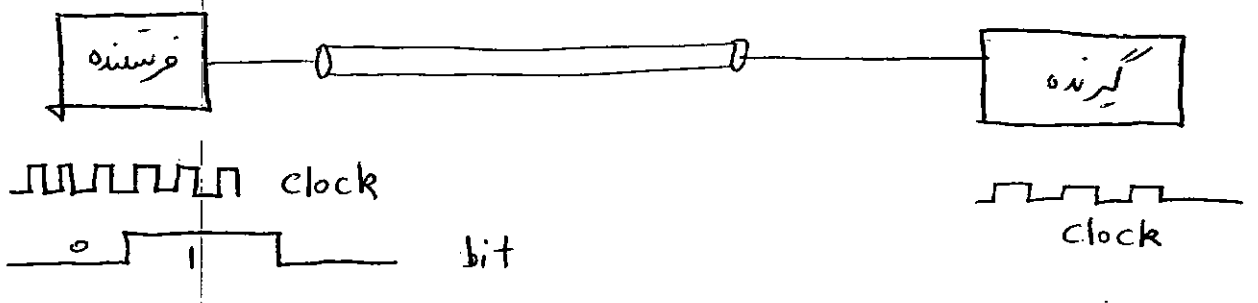
(A) SYNCHRONOUS "هنگام یا سنکرون"

طرفین دارای پالس کلاک محلی هستند.

$T_b = N \times T_c$

- T_b: طول هر بیت
- N: معمولاً 2^n
- T_c: طول پالس کلاک محلی

(B) ASYNCHRONOUS ناهنگام یا آسنکرون



$T_b = N \times T_c$

شکل (۱۹۱) ارتباط سریال

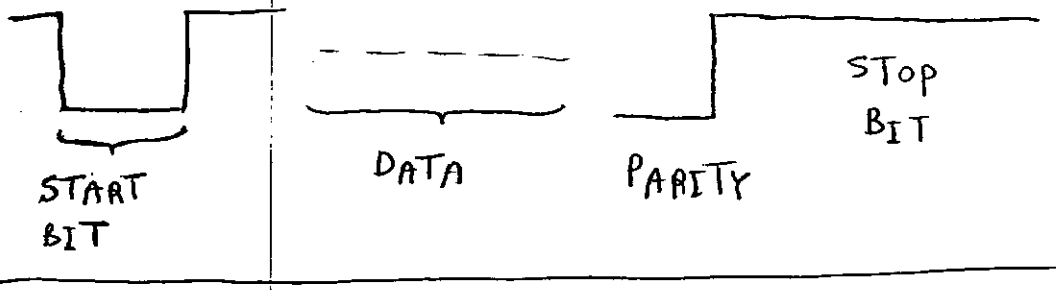
* در انتقال آسنکرون فرض بر این است که رابطهٔ $T_b = N \times T_c$ دقیق نیست و به شکل $T_b \simeq N \times T_c$ ظاهر می‌شود.

طریقهٔ دریافت اطلاعات در ارتباط سریال آسنکرون : ASYNCHRONOUS

⊕ فرستنده در ابتدا بیتی (به طول یک بیت) صفر می‌فرستد که آن را بیت شروع می‌گویند. بعد از بیت شروع تعدادی بیت (توافق شده) اطلاعات و پس از آن بیت توازن (زوج یا فرد) قرار می‌گیرد.

⊕ در انتها خط در ولتاژ بالا قرار می‌گیرد که طول آن ۱ یا ۱٫۵ یا ۲ برابر طول بیت (بنا بر توافق) است که آن را بیت خاتمه گویند.

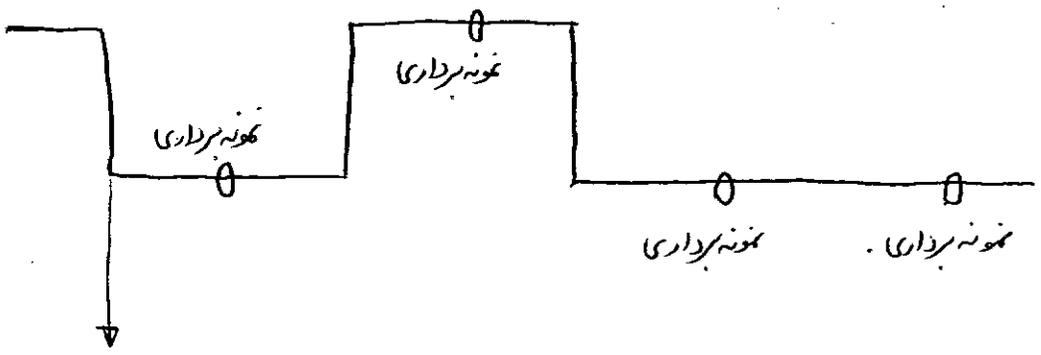
⊕ در ادامه اگر اطلاعات نداشته باشیم خط در ولتاژ بالا می‌ماند و اگر اطلاعات داریم START BIT فرستاده می‌شود.



شکل (۱۹۲) بیتهای مورد توافق در ارتباط آسنکرون

* این نوع ارتباط برای جاهایی که اطلاعات کم و پراکنده‌اند مناسب است (ارتباط ترمینال با کامپیوتر)

⊕ طریقهٔ عمل کردن گیرنده :



شکل (193)
طریقه تفسیر داده های خط گیرنده در ارتباط آسنکرون

⊛ با دیدن لبهٔ پایین رونده ، $N/2$ برابر پریود کلاک محلی (گیرنده) [و پس از N کلاک] نمونه برداری صورت می گیرد.

محدودیت
CONSTRAINT

$Tb \cong N \times Tc$: $Tb < N \times Tc$: $T_b > N \times Tc$:

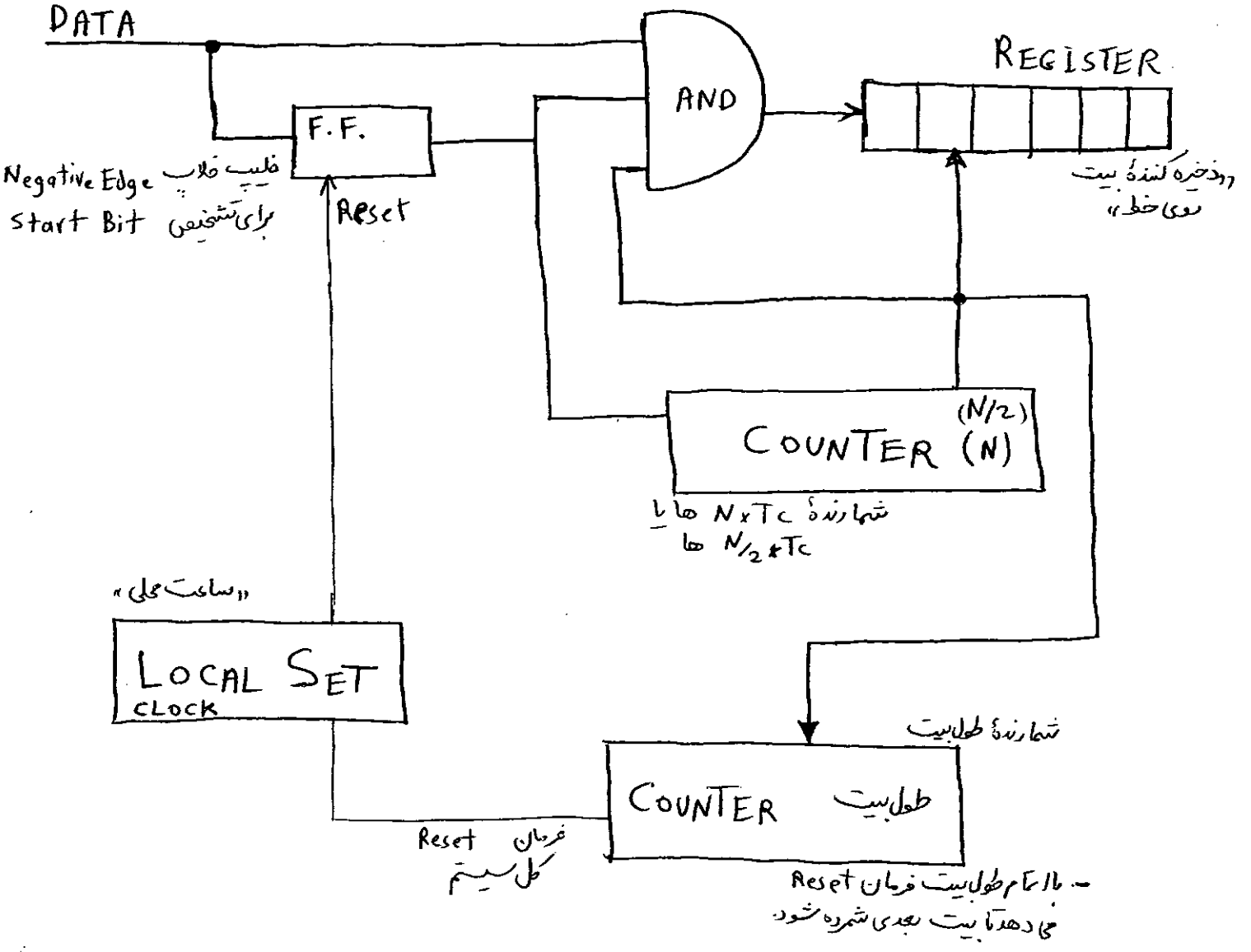
⊛ اگر T_b کوچکتر از $N \times Tc$ باشد پس از چند نمونه برداری ، نمونه برداری خارج می افتد و امکان اشتباه بالا می رود. در مورد حالت دیگر هم همین مشکل را داریم. (لبه های پالس ها در فرکانسهای بالا نیز نیستند)

⊛ برای حل این مشکل تعداد بیتها را محدود می کنند. (مثلاً بین START و STOP ، ۵ بیت)

⊖ در این مثال برای انتقال ۵ بیت ۳ بیت اضافه هم ارسال شد.

5 bit	Data
1 bit	STAAT
1 bit	STOP
1 bit	Parity

⊕ مدار انتقال آسنکرون دریافت کننده :



شکل (۱۹۶) مدار دریافت کنندهٔ آسنکرون

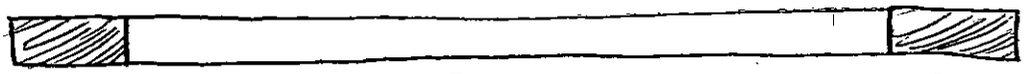
در اینجا اطلاعات خط (Stop ، Start و Data) همگی به ترتیب در رجیستر مدار ذخیره می شوند.

⊕ انتقال سنکرون : SYNCHRONOUS TRANSMISSION:

« دونوع دارد : ۱- بیت محور ۲- کاراکتر محور »

ⓐ انتقال bit_oriented « بیت محور »

مجموعه‌ای از بیتها که بین دو flag شروع و خاتمه قرار دارند.



01111110 HDLC protocol

01111110 HDLC protocol

شکل (۱۹۵) انتقال بیت محور

(؟) ⊛ اگر الگوی flag در درون داده‌ها باشد چه کنیم ؟!!

BIT STUFFING :

(برای رفع ابهام تعبیر داده ها به عنوان الگوی flag)

⊕ هرجا در درون داده‌ها ۵ تا ۱ پشت سر هم بود یک صفر به انتهای آنها اضافه می‌شود (پروتکل HDLC)
در طرف گیرنده هرجا ۵ تا ۱ پشت سر هم بود صفر بعد از آنها برداشته می‌شود. و اگر صفری بعد از آنها نبود، به عنوان الگو تلقی می‌شود.

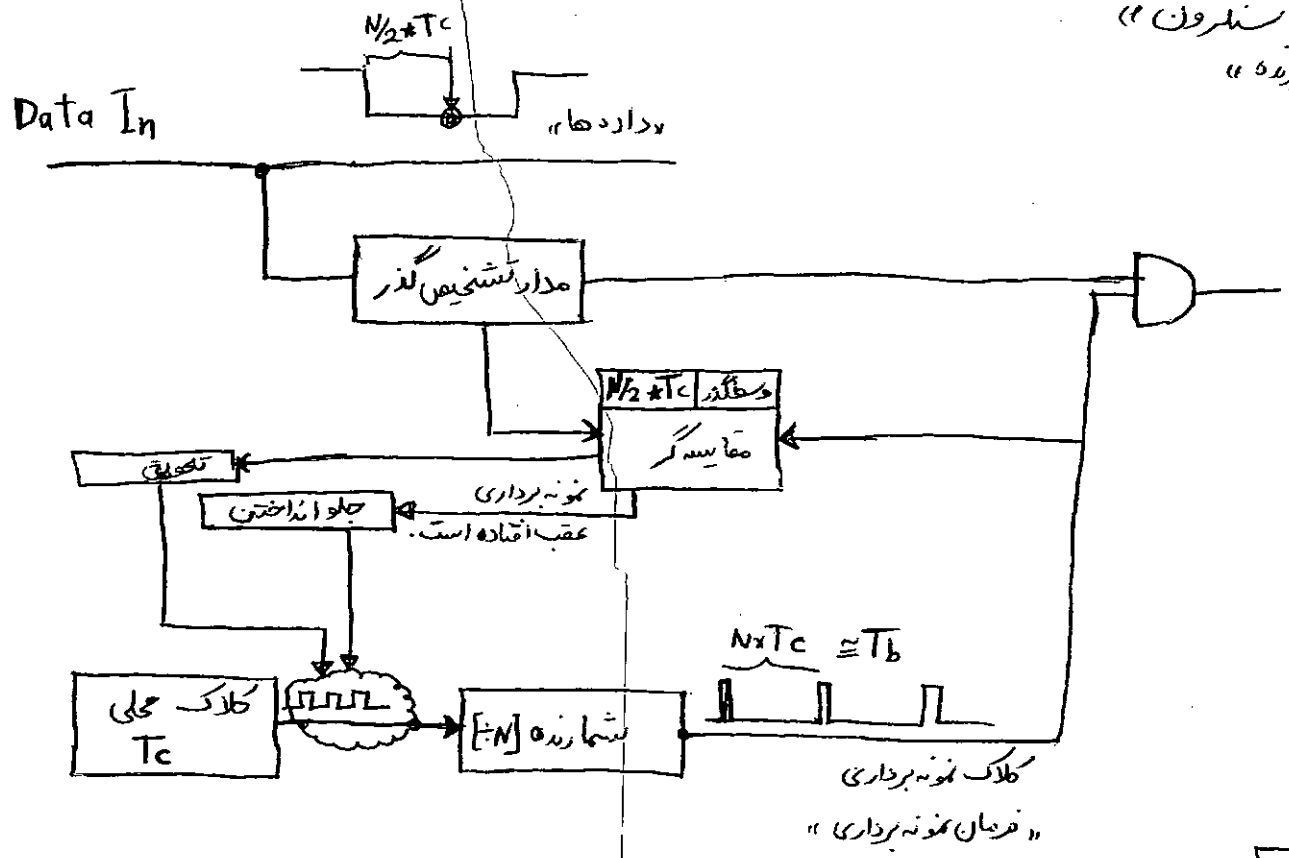
ⓑ انتقال CHARACTER ORIENTED « کاراکتر محور »

مضربی از طول ۸ انتقال داده می‌شود.

در شروع کاراکتر SYN و در انتها کاراکتر EOM (END OF MESSAGE) قرار می‌گیرد. برای جلوگیری از مشکل تکرار EOM در وسط بیت در انتها قبل از EOM یک EOL قرار می‌دهند و باز هم برای جلوگیری از بقیه مشکل‌ها هرجا در وسط پیام EOL یا EOM پشت سر هم داشتیم با شیء آن را [illegible]

⊕ نمودار عملکرد مدار هنگام سازی [SYNCHRONIZATION]

«انتقال سنکرون»
«مدار گیرنده»

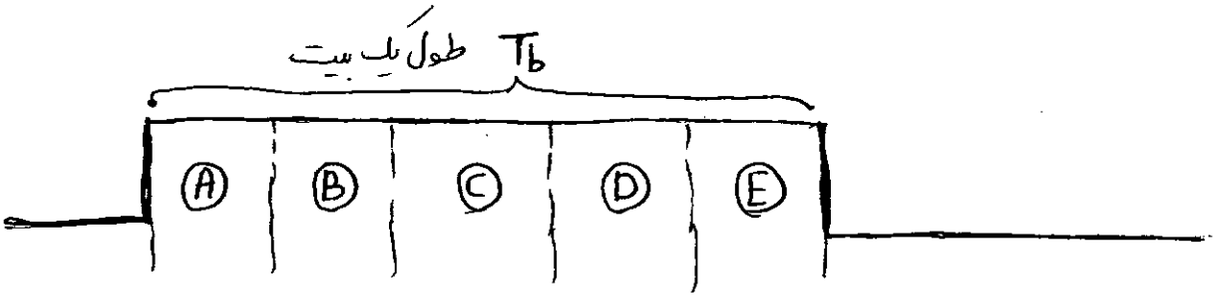


شکل (196)

* فرض براین است که $T_b \cong N \times T_c$
* کلاک محلی $T_c =$
* مدار مقایسه گر وسط گذرها را با $\frac{N \times T_c}{2}$ مقایسه می کند تا بفهمد که نمونه برداری کجای پالس انجام می شود.
* هنگام سازی توسط همین مقایسه انجام می شود.
* مدار جلو انداز ورودی شمارنده را دستکاری می کند، به نحوی که چند پالس به آن اضافه می کند تا خروجی شمارنده زودتر فرمان نمونه برداری صادر کند.
* مدار تعویق (POSTPONE) هم مانند مدار جلو انداز است، عملکرد آن این گونه است که تعداد پالس های ورودی به شمارنده را کمتر می کند. در نتیجه فرمان نمونه برداری دیرتر صادر می شود.
* با این دو مدار عمل هنگام سازی در هر گذر انجام می شود. نتیجتاً تعداد زیادی بیت پشت سر هم می توانند مبادله شوند.
* در صورت نبود گذر (تغییر سطح و لتاژ) امکان احتیاج مانند مدار آشکارون است که با اولین گذر تصحیح می شود.

⊕ وجود گذر به عمل هگام سازی کمک می کند. در نتیجه روش کدینگ خط در هگام سازی تأثیر مستقیم دارد

⊕ تقسیم طول بیت به نواحی مختلف جهت تسهیل عمل هگام سازی.



شکل (۱۹۶)

if	A	3Tc	عقب بیانداز
if	B	Tc	" "
if	C	0	—
if	D	Tc	جلو بیانداز
if	E	3Tc	" "

⟸ عملکرد مدار در حالتی که نمونه برداری در هر یک از نواحی A تا D رخ داده باشد.

★ تعداد نواحی برحسب شرایط تعیین می شود.

" زمانهای مهم در انتقال سنکرون "

⊕ ACQUISITION TIME :
زمان بدست آوردن هگامی. (برحسب تعداد بیت)

⊕ TRACKING TIME :
زمان ماندگاری هگامی در صورت ثابت بودن وتاثر در خط. (برحسب تعداد بیت)

" راهی است راه عشق که پیچش کناره نیست
آنجا حزن آدمی که جان بسپارد هیچ چاره نیست "

مقایسه انتقال سنکرون و آسنکرون :

① Redundancy : میزان افزونگی داده‌ها.

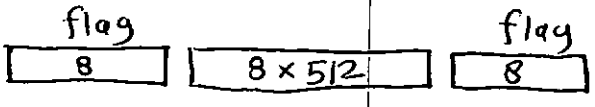
⊕ آسنکرون : REDUNDANCY زیاد است.

Start	Data	Parity	Stop
1	7	1	2

7 Data.
4 Redundant.

$\Rightarrow$ Reffective = $\frac{7}{11} R_{Line}$

⊕ سنکرون : REDUNDANCY کمتر است.



8×512 = Data
2×8 = Redundant.

$Reff \simeq 9.99 \times 10^{3} \times R_{Line}$ ← مثال

EXPENCE CRITERIA

② هزینه :

⊕ روش آسنکرون هزینهٔ کمتری دارد. (مدارش ساده است)

⊕ روش سنکرون هزینهٔ بالاتر و مدار پیچیده‌ای دارد.

APPLICATIONS:

③ کاربرد.

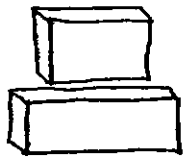
⊕ روش آسنکرون برای اطلاعات پراکنده خوب است. (مثلاً ترمینال به کامپیوتر)

⊕ روش سنکرون برای اطلاعات حجیم مناسب است. (کامپیوتر به کامپیوتر)

⊕ آخرین مبحث لایه فیزیکی:

واسط لایه فیزیکی،

و استانداردهای آن



PC

یا

تجهیزات پایانه داده

(DTE)

PHYSICAL LAYER INTERFACE

(رابط) (interfaces)

DTE: وسایلی که به شبکه متصل می‌شوند

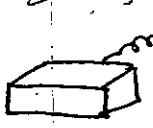
داده‌ها را تولید یا مصرف می‌کنند

DCE: داده‌ها را بین آنالوگ و دیجیتال تبدیل می‌کند

و سیگنال‌ها را برای ارسال در محیط فیزیکی آماده می‌کند

«استاندارد»

RS232.C



MODEM

تجهیزات انتقال داده (DCE)

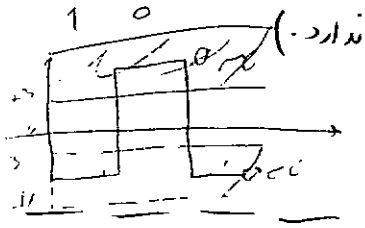
V.24 / EIA 232

شکل ۱۹۸

واسط لایه فیزیکی

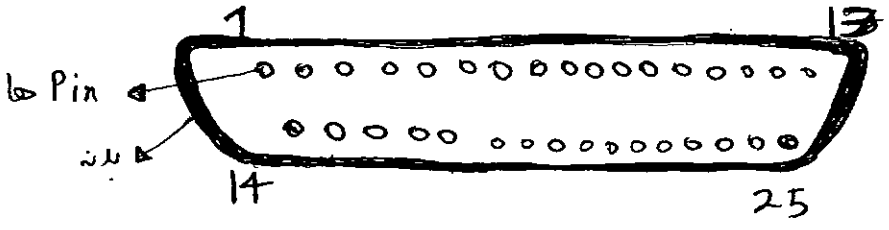
⊕ استاندارد Interface باعث می‌شود که طراحی مودم‌ها و ساخت آن‌ها استاندارد شود.

استاندارد RS232.C/V.24



⊕ این استاندارد بین DTE و مودم است. (پرکاربردترین استاندارد)

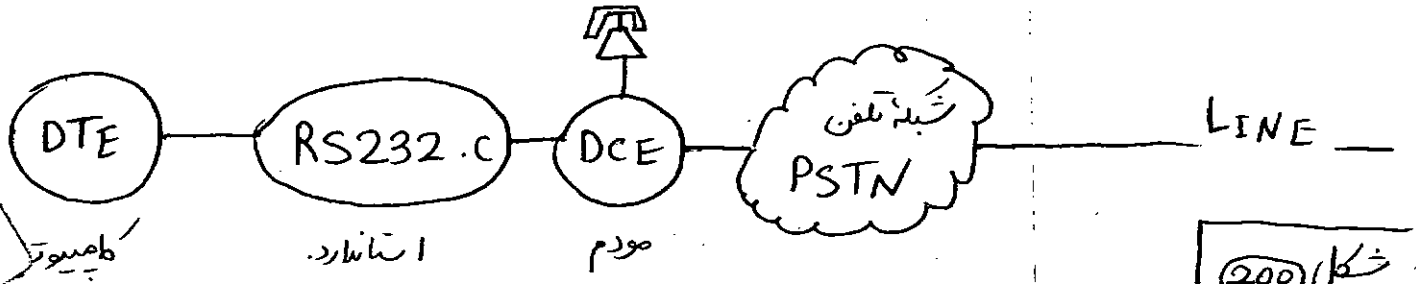
⊕ روی یک D-type Connector برقرار می‌شود.



[Port متصل‌کننده دارای ۲۵ پایه (نوع D)]

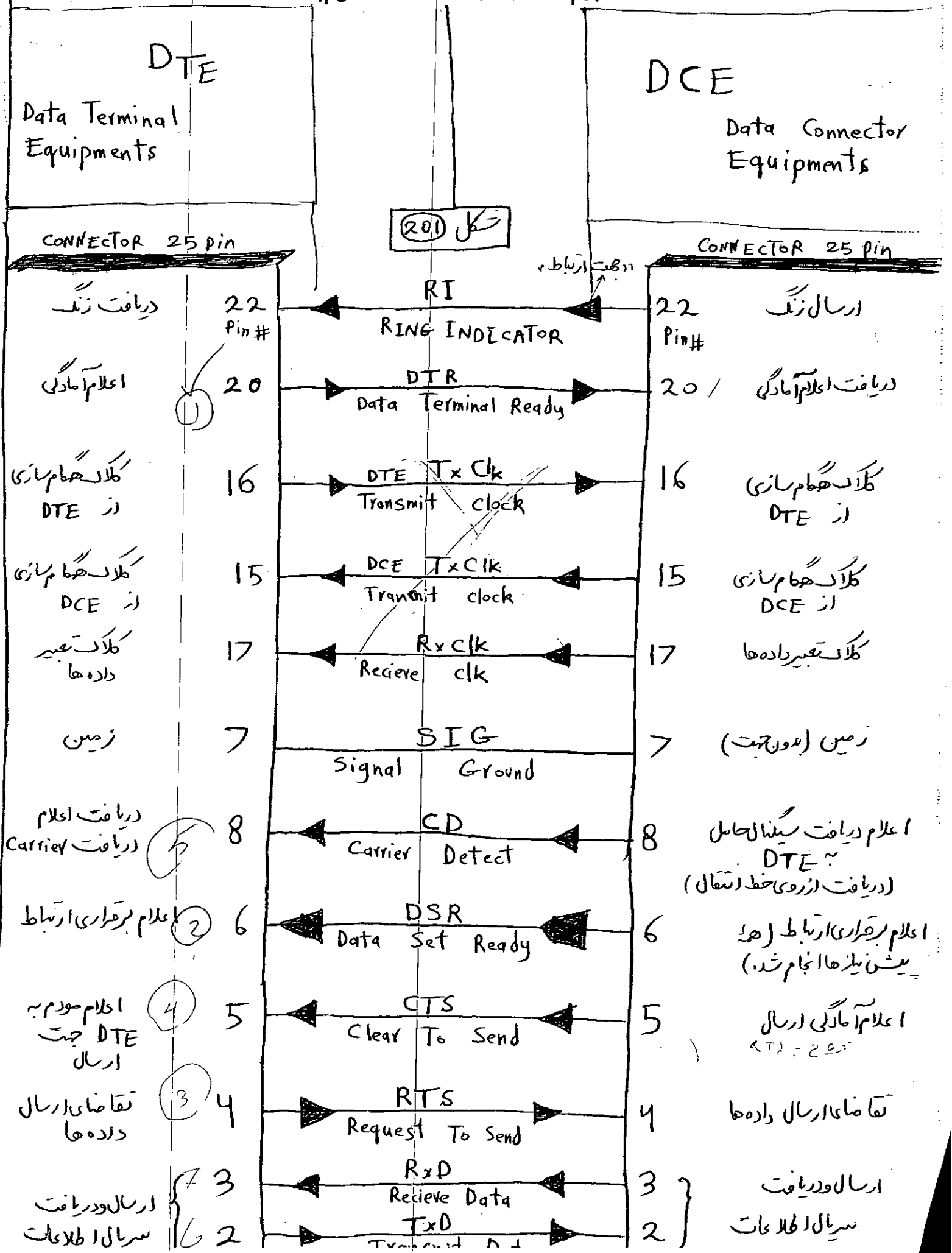
D-type CONNECTOR

شکل (۱۹۹)



شکل (۲۰۰)

پیاده سازی استاندارد RS232.C روی یک D-TYPE CONNECTOR 25PIN



شکل (20)

RI : (#22) ارسال سیگنال زنگ از DCE به DTE

DTR : (#20) اعلام آمادگی DTE به DCE جهت ارتباط

Tx clk DCE : (#15) کلاک هماهنگ سازی که DCE به DTE می فرستد تا DTE براساس آن بفرستد

Tx clk DTE : (#15) کلاک هماهنگ سازی که DTE به DCE می فرستد تا DCE اطلاعات را درست تفسیر کند.

Rx clk : (#17) کلاک هماهنگ سازی که از DCE به DTE جهت اطلاعات از DCE به DTE است.

SIG : (#7) زمین مشترک، بدون جهت است.

CD : (#8) اعلام DCE به DTE هنگامیکه سیگنال بوق TONE حامل از روی خط به DCE می رسد

DSR : (#6) اعلام DCE به DTE هنگامیکه تمام پیش نیازهای ارتباط برقرارند.

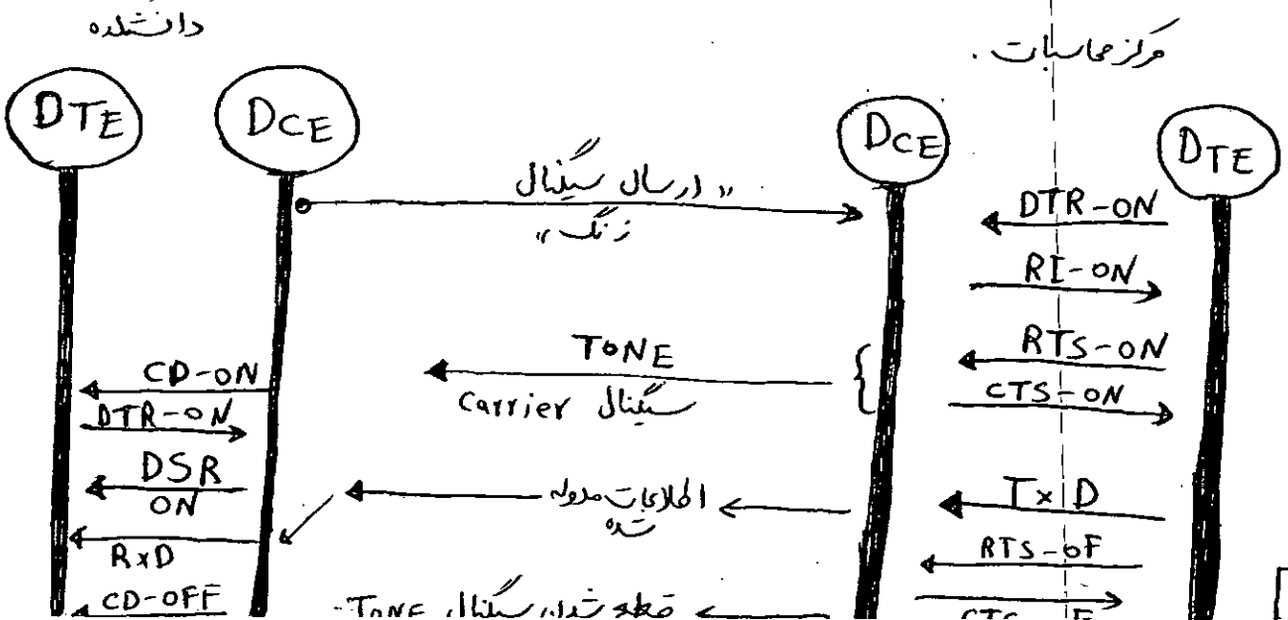
CTS : (#5) اعلام DCE به DTE وقتی که DCE آزاد است تا اطلاعات را بفرستد.

RTS : (#4) تقاضا برای ارسال داده ها (از DTE به DCE)

RxD : (#3) پایه دریافت اطلاعات، از DCE به DTE

TxD : (#2) پایه ارسال اطلاعات، از DTE به DCE

مثال: برقراری ارتباط دوطرف با مودم از طریق استاندارد RS232.C



شکل (202)

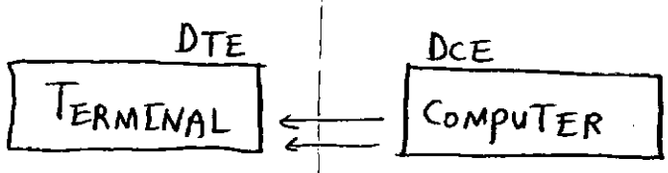
CASE STUDY

دو کامپیوتر را که یکی هم دارم. چگونه با استاندارد RS232 این دو را هم متصل کنیم؟

+① راه حل اول: استفاده از یک مودم بین این دو.

!* هزینه زیادی دارد.

+② راه حل دوم:

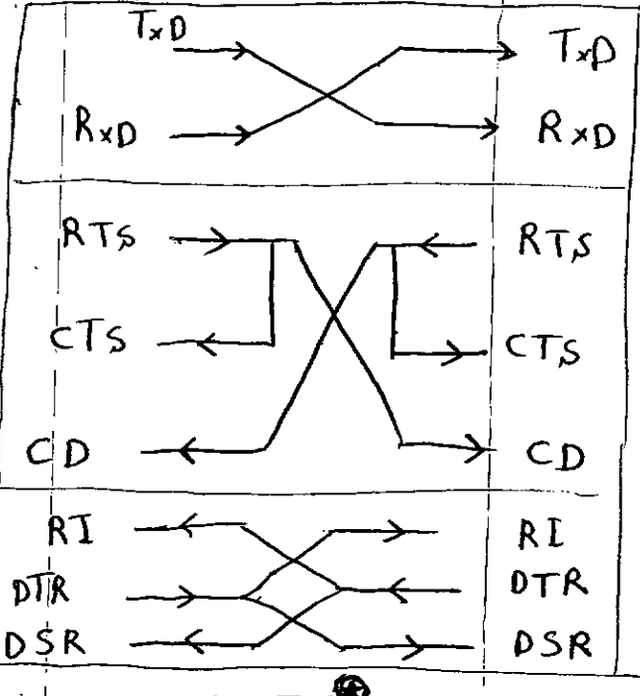


برای کامپیوتر Enterface را بصورت DCE و برای ترمینال DTE تعریف می‌کنیم.

!* برای کوتاه مدت خوب است ولی فقط یک کاربرد دارد و برای دیگر کاربردها باید دوباره تعریف را عوض کنیم.

+③ راه حل سوم: NULL DATA

تعویض سیم بندی به شکل زیر:



شکل (203)

* بین پایه‌های کلاک تجهیزاتی مورد نیاز است که خطوط دیگر کلاک را تهیه کنند.

!!!!!! نکاتی درباره لایه فیزیکی

الف) MODULATION

مدولاسیون.

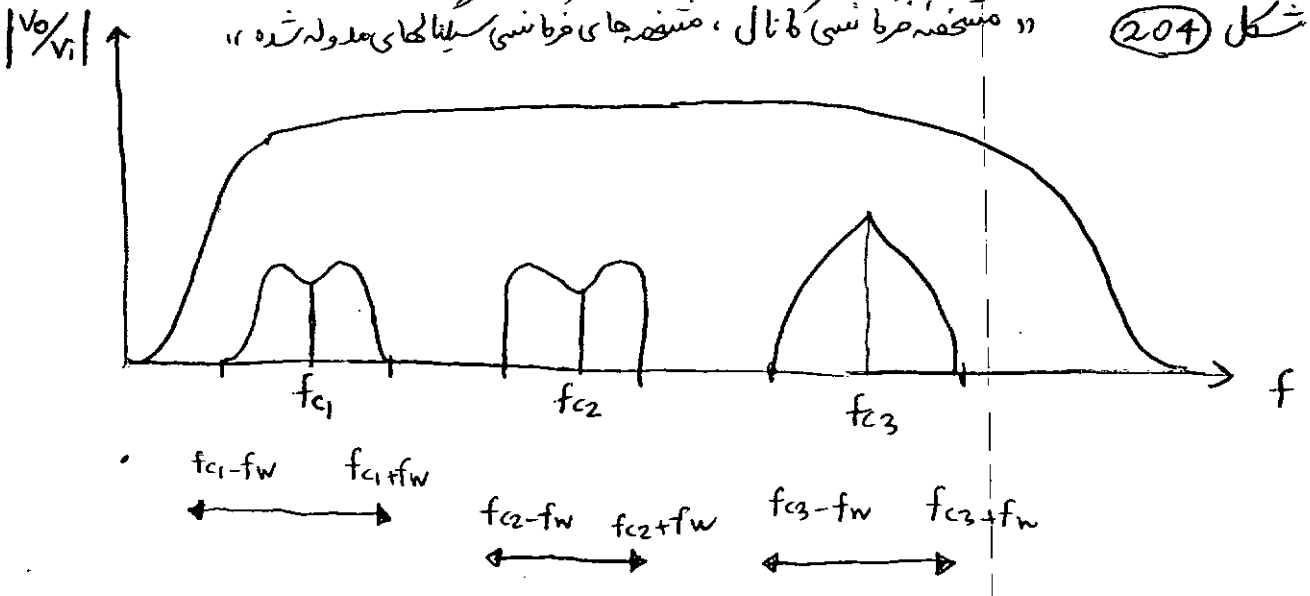
با این تکنیک ما می‌توانیم سیگنال را در فرکانس مناسب برای انتقال مدل کنیم.

خط ارتباطی را به سیگنال آشکار می‌دهد.

* کانالی داریم با پهنای باند زیاد.

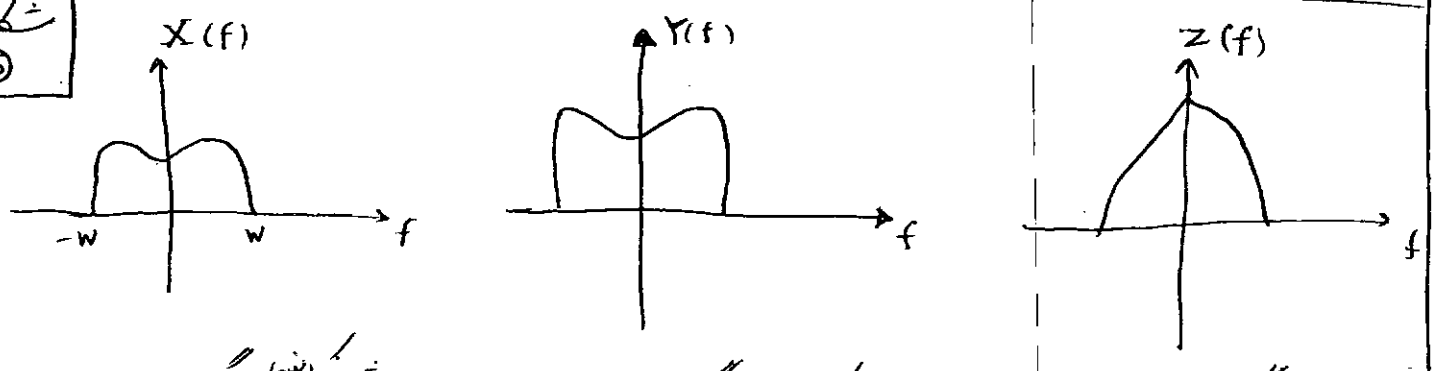
می‌خواهیم سیگنالهای متعددی را روی این کانال بگذاریم بدون اینکه با هم تداخل کنند.

شکل (204) «مشخصه فرکانسی کانال، مشخصه‌های فرکانسی سیگنالهای مدوله شده»



* برای سهولت فرض کرده‌ایم هر سه سیگنال دارای محدوده فرکانسی یکسان (2fw) هستند.

شکل (205)



مشخصه فرکانسی سیگنال x(t) (با پهنای باند) — سیگنال حامل fc1

مشخصه فرکانسی سیگنال y(t) — سیگنال حامل fc2

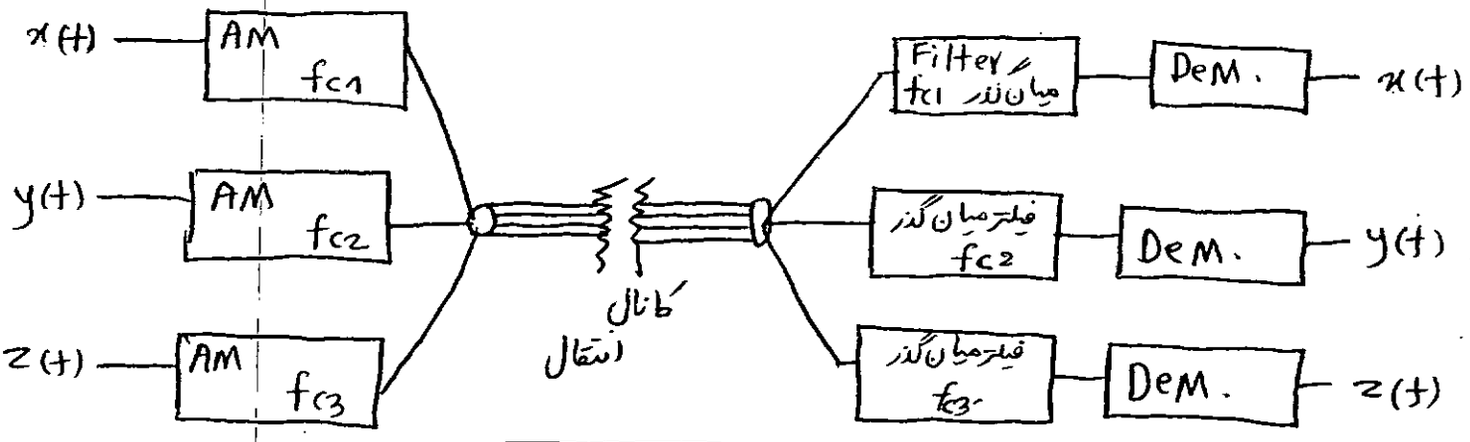
سیگنال z(t) — سیگنال حامل fc3

* با فیلترهای گوناگون می‌توان (رمعکد این سیگنالها را جدا و بازیابی کرد.

* (رفضای پیرامون) ما (رحال حتی سیگنالهای مدوله شده‌ای فراوان وجود دارند.

⊕ طرح کلی برای ارسال سه سیگنال x، y و z در یک کانال:

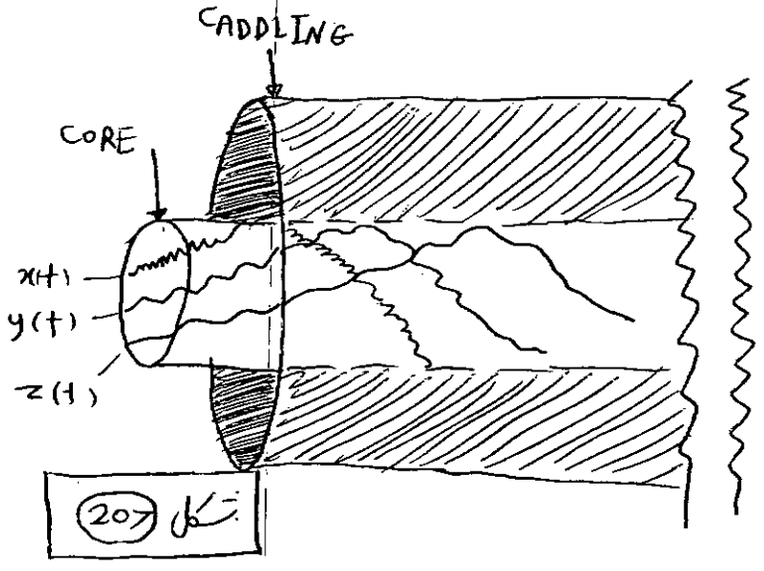
FREQUENCY DIVISION MULTIPLEXING (FDM)



شکل (206) طرح کلی FDM

ب) فیبر نوری

FIBER OPTIC



شکل (207)

ارسال سه سیگنال بصورت همزمان در فیبر نوری

* هر سیگنال با طول موج مخصوص خودش

WAVE LENGTH DIVISION MULTIPLEXING

W.D.M

⊛ اعمال FDM و WDM در لایه فیزیکی انجام می شوند.

⇦ نتیجه: برای انجام این اعمال، لایه های فیزیکی طرفین باید این روشها را SUPPORT کنند.

ج) مالتی پلکسینگ تقسیم زمانی

TIME DIVISION MULTIPLEXING

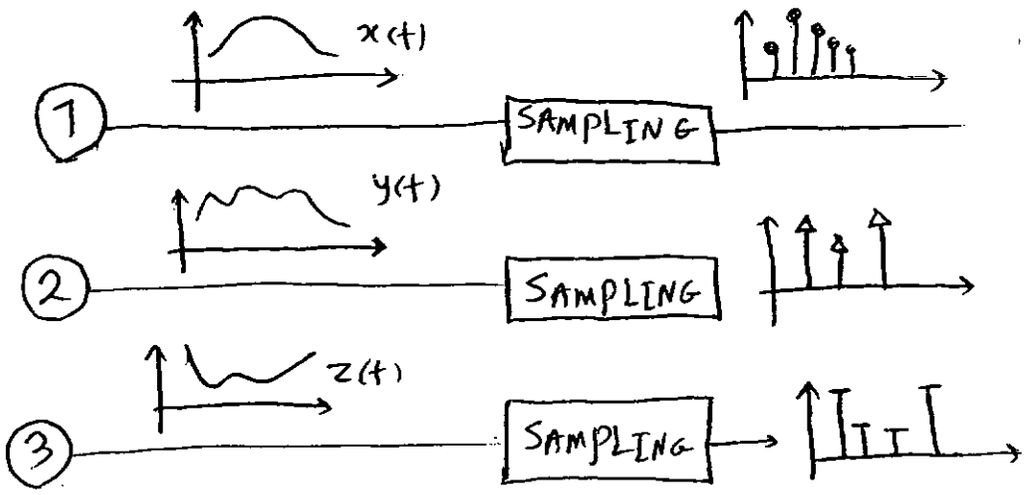
T.D.M

کنترلت نرخ ارسال

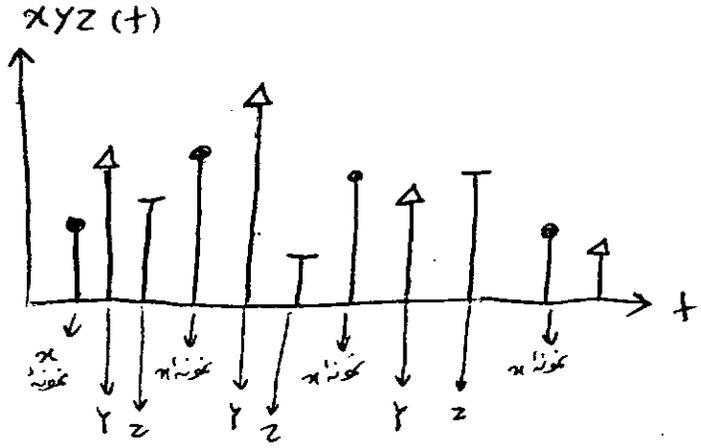
داده ای ارسالی از نرخ داده ای

آنالوگ (a)

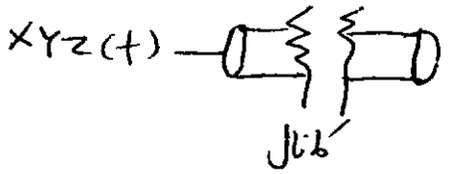
ANALOG



شکل (208)
در نمونه برداری از سه سیگنال



شکل (209)
« نمونه ای از سه سیگنال »

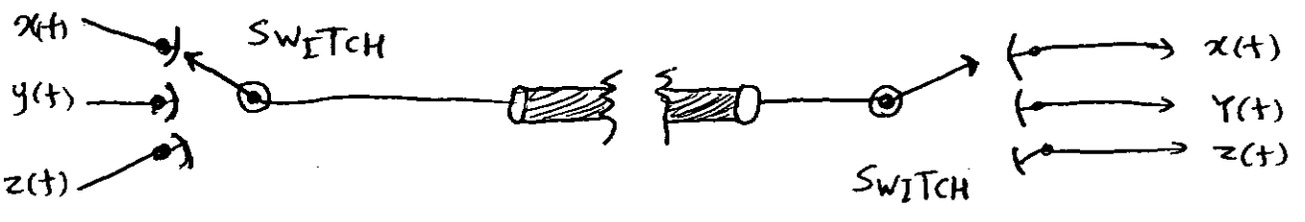


* تلفیقی از سه نمونه را روی خطی فرستیم.

* با این کار در هر لحظه هر یک از باند در اختیار یکی از سیگنالها قرار می گیرد.

---

* تسهیل اجمال با الا با استفاده از یک سوییچ. (شکل (210))



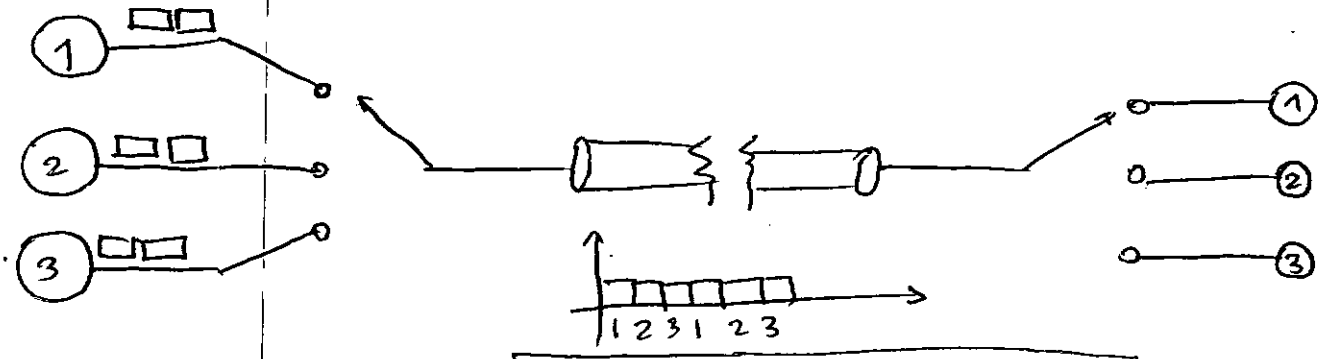
* سوییچ هم عمل نمونه برداری و هم عمل سریم کردن نمونه ها را انجام می دهد و در مقصد عکس این عمل انجام می شود.

ⓑ دیجیتال انتقال دیجیتال برای منابع آنالوگ:

DIGITAL :

شکل (211)

ⓒ انتقال برای منابع دیجیتال:



شکل (212) T.D.M برای منابع دیجیتال

ⓓ محل TDM در لایهٔ فیزیکی انجام می‌شود، هرچند می‌توان در لایه‌های بالاتر هم آن را انجام داد.

مقایسهٔ پهنای باند:

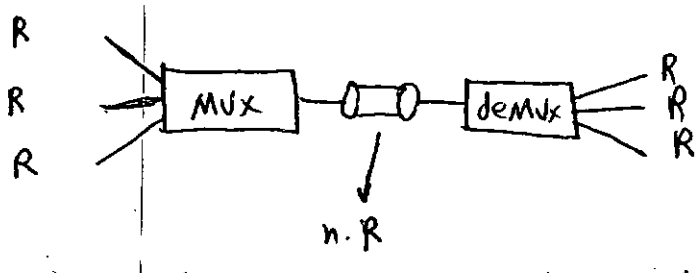
ⓐ F.D.M :

برای انتقال n سیگنال:

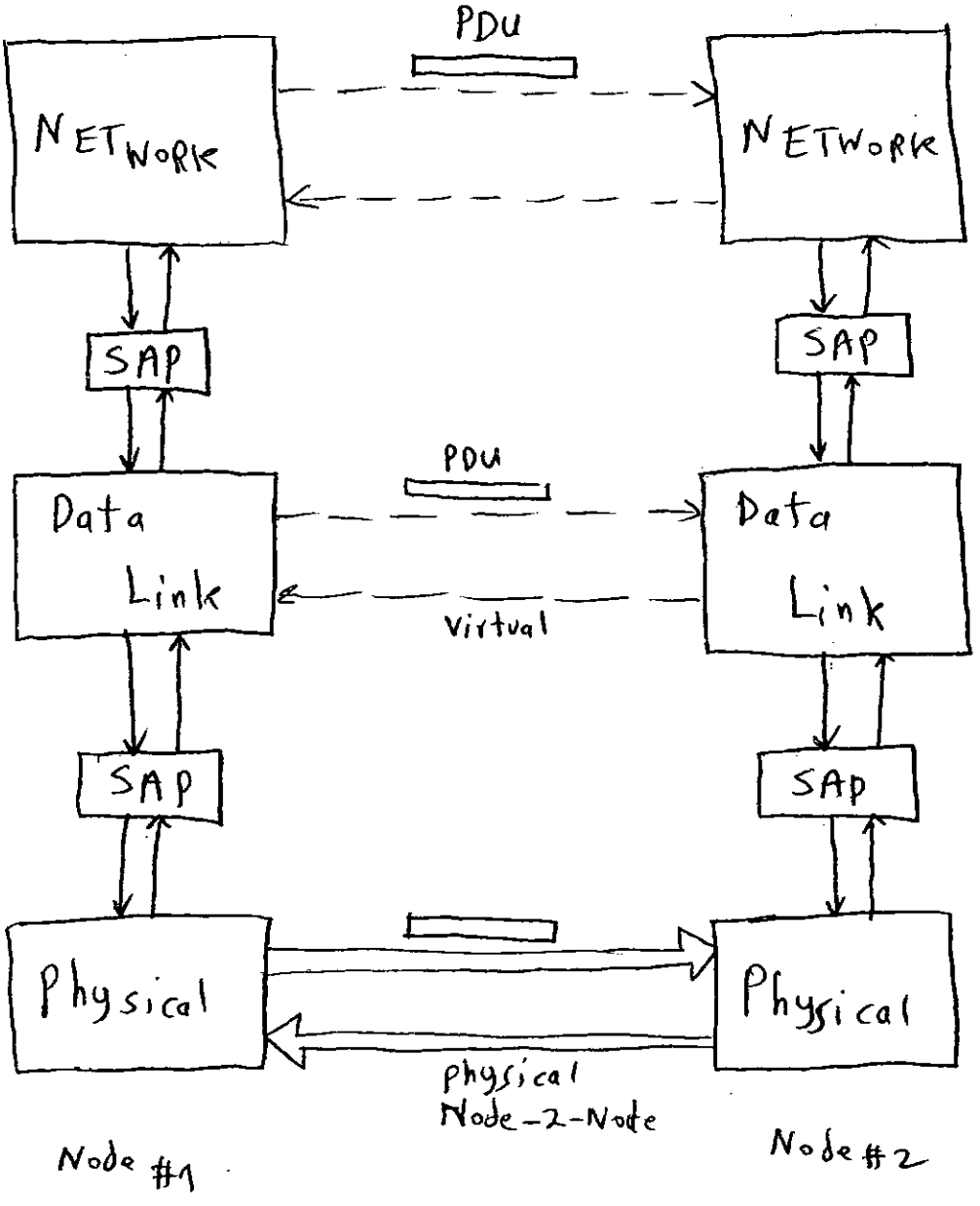
پهنای باند $n.B$ لازم است.

که B = پهنای فرکانس هر سیگنال

ⓑ T.D.M :

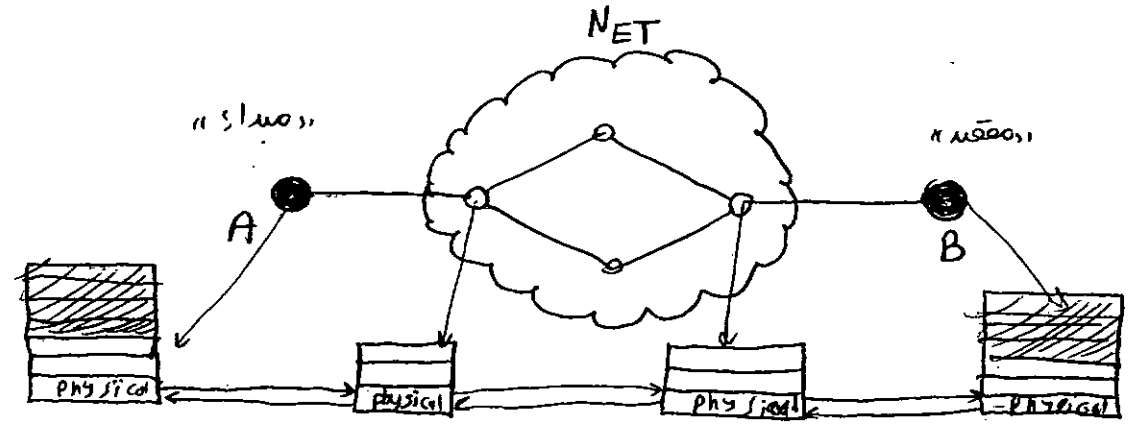


نرخ بیت و پهنای باند کانال n برابر نرخ بیت و پهنای باند هر سیگنال باند باشد.



* سرویسهای لایهٔ فیزیکی را بررسی کردیم !!

« شکل (213) سه لایهٔ زیرین مدل OSI »



(*) « ارتباط نود به نود در سه لایهٔ پایین و ارتباط پایان به پایان در چهار لایهٔ بالا » (شکل (214))

مثال: ارتباط تلفنی:

نیازها: REQUIREMENTS:

۱- ارتباط دوطرفه مدنظر است.

۲- تأخیر نداشته باشیم. ارتباط REAL TIME باشد.

۳- نویز خیلی زیاد نداشته باشیم.

نیازهای لایه‌های بالاتر که باید توسط لایه‌های پایین SUPPORT شود.

مثال:

نیازهای لایهٔ کاربرد: APPLICATION LAYER REQUIREMENTS:

۱- بزرگی دلخواه پیام

2- توالی پیام‌ها

3- زمان

4- کنترل جریان

5- امنیت انتقال

⊕ مدل سرویس SERVICE MODEL

«جنبه‌های مختلف برای طبقه‌بندی سرویس‌ها»

(A)
* 1- Connection-Less
* 2- CONNECTION ORIENTED

(B)
* 1- قطعه‌ای BLOCK
* 2- رشته‌ای STREAM

(C)

* ۱- نرخ بیت ثابت CONSTANT BIT RATE

* 2- نرخ بیت متغیر VARIABLE BIT RATE

(D)

کیفیت سرویس QUALITY SERVICE

- تأخیر Delay
- حمل درست پیام
- امکان از دست رفتن پیام
- .....

■ از دید لایهٔ Data Link کار لایهٔ فیزیکی فقط حمل اطلاعات است.
جزییات ربطی به لایهٔ Data Link ندارد.

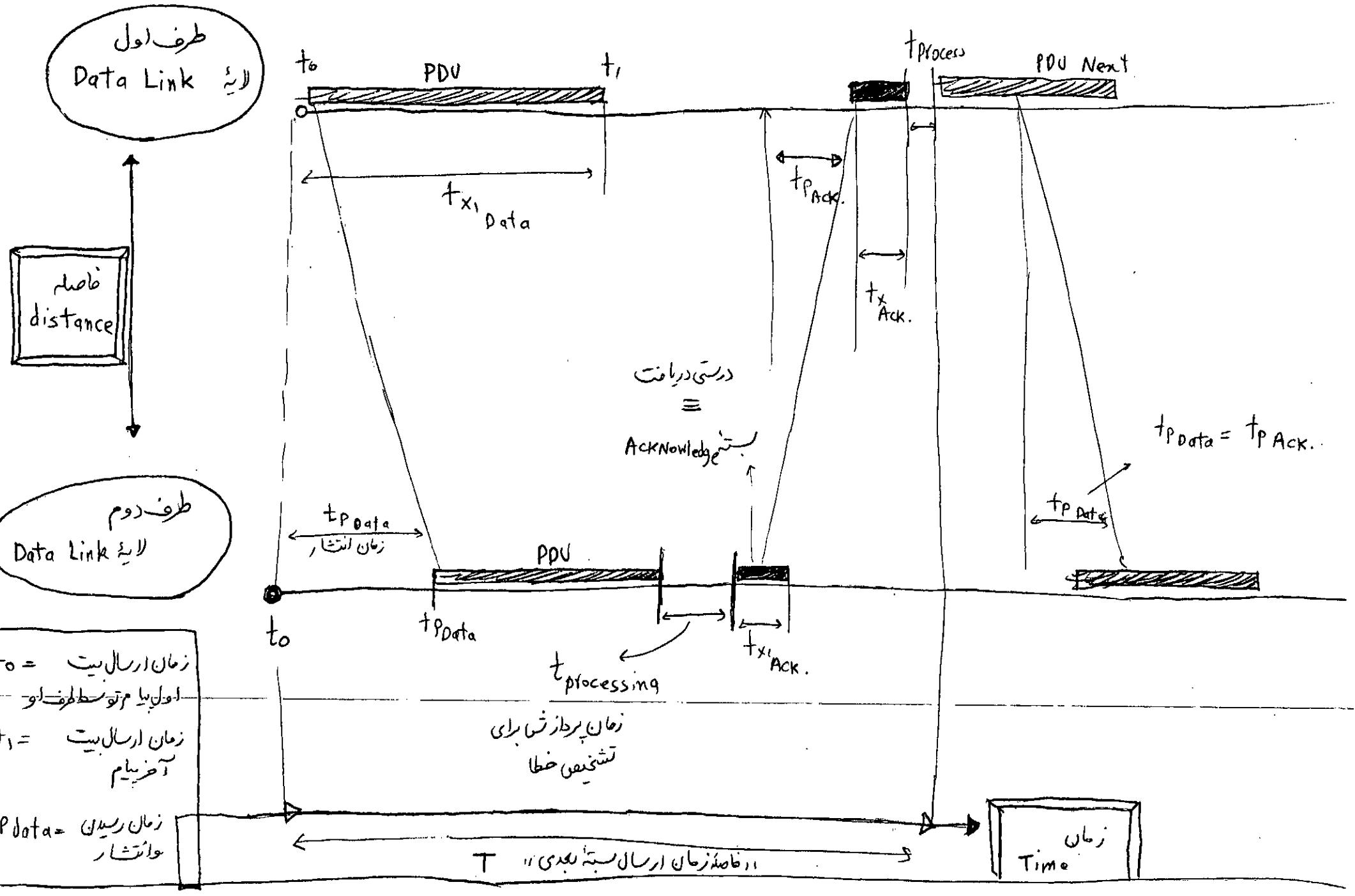
⊕ وظایف لایهٔ Data Link :

① کنترل خطای نود به نود : ERROR CONTROLING

⊛ کنترل خطا در لایهٔ Transport هم انجام می‌شود منتها بصورت END 2 END .

⊕ کنترل خطا { Feed Back تشخیص خطا. / Forward تصحیح خطا. }

- بین هر دو نود مجاور احتمال تشخیص خطا انجام می‌شود.
- در شبکه‌های کامپیوتری عمل تصحیح خطا در لایهٔ Data Link است درحالیکه در برخی سیستم‌های مخابراتی در لایهٔ فیزیکی بحث می‌شود.
- هر نود بافری دارد که آخرین پیام برای ارسال مجدد در آن نگاه داری می‌شود.


طرف اول
لایهٔ Data Link
طرف دوم
لایهٔ Data Link
فاصله
distance
t_0
t_1
PDU
$t_{x_1\ Data}$
$t_{process}$
PDU Next
$t_{p\ Ack.}$
$t_{x\ Ack.}$
درستی دریافت
≡
بستهٔ AckNowledge
$t_{p\ Data} = t_{p\ Ack.}$
$t_{p\ Data}$
زمان انتشار
PDU
$t_{p\ Data}$
$t_{processing}$
$t_{x_1\ Ack.}$
t_0
زمان پردازش برای تشخیص خطا
t_0 = زمان ارسال بیت اول یا مرتبط طرف او
t_1 = زمان ارسال بیت آخر پیام
$t_{p\ Data}$ = زمان رسیدن موانتشار
T = فاصلهٔ زمان ارسال بستهٔ بعدی
زمان
Time

محاسبه زمان بین دو ارسال «در فرستنده»

$$T = t_{x,\,data} + t_{prop.\,data} + t_{proc.} + t_{x,\,ack.} + t_{Prop.\,Ack} + t_{proc.}$$

- $t_{x,\,data}$: زمان ارسال پیام
- $t_{prop.\,data}$: زمان انتشار پیام
- $t_{proc.}$: زمان پردازش ← قابل مرفظر
- $t_{x,\,ack.}$: زمان ارسال Ack.
- $t_{Prop.\,Ack}$: زمان انتشار Ack. ← قابل مرفظر
- $t_{proc.}$: زمان پردازش ← قابل مرفظر

$$\boxed{T \simeq t_{x_{DATA}} + 2\,T_{PROP.}}$$

$T =$ فاصله بین دو ارسال متوالی

$t_x = L_{Data} / R_{Data}$ «زمان ارسال»

$t_{prop.} = d / v$ «زمان انتشار» (زمان انتشار)

با توجه با زمان T می توان عملیات را بصورت زیر تصحیح کرد.

فرستنده:

در t_0 پیام را می فرستد،
پس از گذشت زمان T، اگر بسته ای نرسید دوباره
پیام را قبلی را می فرستد. (تشخیص خطا)

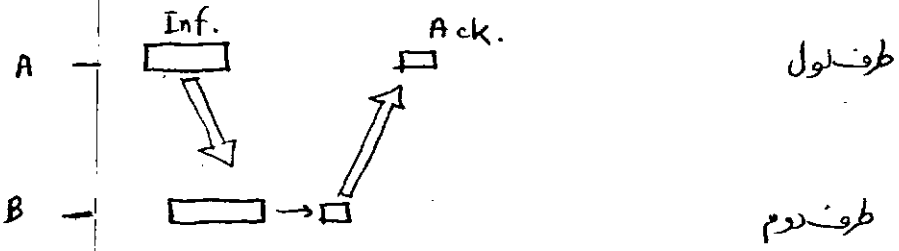
گیرنده:

پس از دریافت پیام درصورت خطا یک AckNowledge
می فرستد و در غیر اینصورت چیزی نمی فرستد!!

⊕ پروتکل ARQ :

AUTOMATIC RETRANSMISSION REQUEST.

- همان ایده مطرح شده در مثال قبلی.
- Automatic بودن به این مفاست که ماشین کنترل خطا می‌کند. (برخلاف ارتباط ترمینال با کامپیوتر که انسان تشخیص دهنده خطا است، از روی کاراکتری که کامپیوتر Echo می‌کند.)



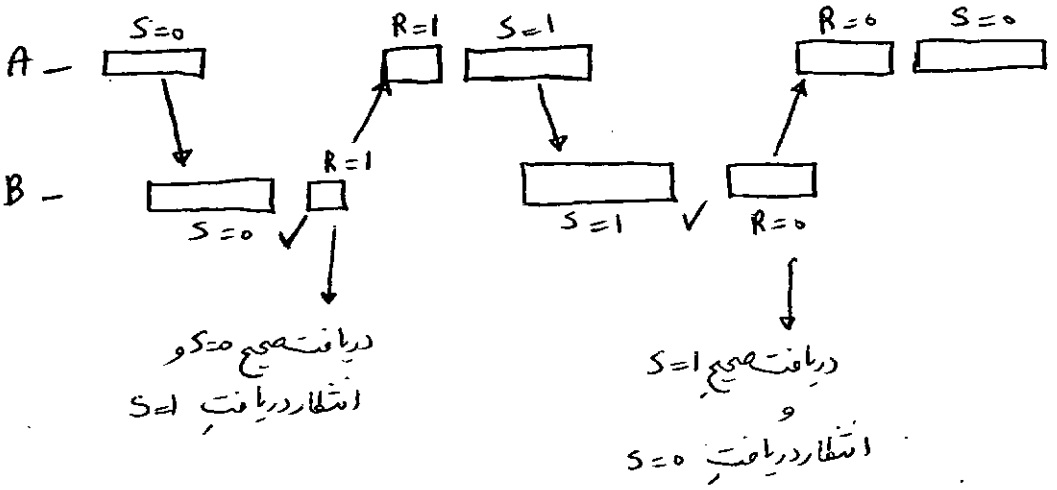
- به بستهٔ اصلی بستهٔ Information می‌گوییم.
- به بستهٔ درستی خطا Acknowledgement می‌گوییم.
- بسته های Information را با دو شماره مشخص می‌کنیم که فرستنده اگر فریمی را دوباره فرستاده گیرنده از روی شمارهٔ آن بفهمد که این بسته تکراری است.
- بدین منظور در فرستنده شمارنده S و در گیرنده شمارندهٔ R قرار می‌گیرد.
- S شمارهٔ بسته های اطلاعات است.
- R شمارهٔ بسته های Acknowledgement است.
- S در ابتدا صفر است. هر بسته که می‌خواهد برود شمارهٔ S را با خود می‌گیرد.
- R در صورت درستی پیام شمارهٔ بسته ای از اطلاعات را با خود می‌گیرد که انتظار دارد آن روی خط برود.

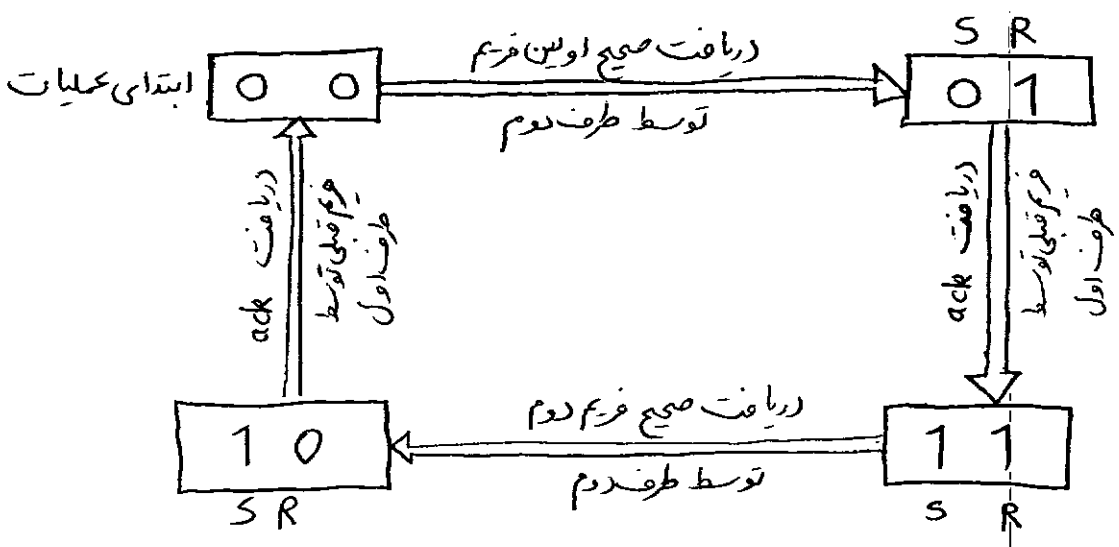
IP یار چند است؟

در کانال بود که پرسید Packet !

چکسی بود صدا زد صد و نود و دو دات صد و هشتاد و پنج دات هفت دات دو؟

بیتها یم کو؟ باید امشب بروم توی کانال، پشت سیم ها LAN ای است!!





– به این پروتکل Stop and Wait می گویند. (یا IDLE IRQ)

– پروتکل Idle IRQ استفادهٔ خوبی از خط نمی کند.

$$T = t_x + 2t_p$$

– اگر فاصله زیاد باشد t_p زیادی شود.

– اگر بستهٔ اطلاعات طولانی باشد t_x زیادی شود.

– حالتی را در نظر بگیریم که t_x زیاد و t_p کم باشد.

⊕ تصحیح پروتکل (Idle ARQ)

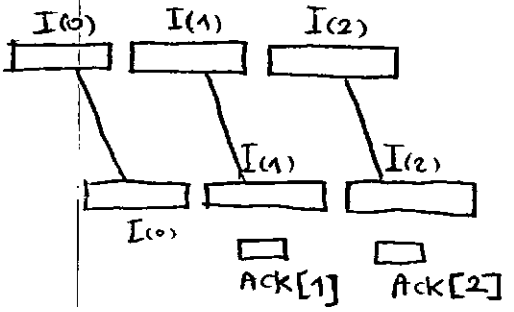
- فرستنده درصورت دریافت نکردن Ack. یک فریم ENQ (Enquiring) به طرف دیگر می‌فرستد تا بفهمد که Ack. از این راه خراب شده یا واقعاً اطلاعات خراب رسیده‌اند. (این کار اگر یکبار بیشتر رخ داد می‌توان سلسله به قرارداد ـ تشخیص خطای واقعی داد)

- به این پروتکل CHECK POINTING می‌گویند.

⊕ پروتکل CONTINIOUS ARQ :

- بسته‌های شماره‌گذاری شده پشت سرهم روی خط گذاشته می‌شود و برای هر یک Ack. با شمارهٔ مخصوص دریافت می‌شود.

- برای حالتی مناسب است که منبع اطلاعاتی داریم که اطلاعات فراوانی را پشت سرهم روی خط قرار دهد.



⊛ Ack[1] ≡ بستهٔ I[0] دریافت شده و منتظر بستهٔ I[1] هستیم.

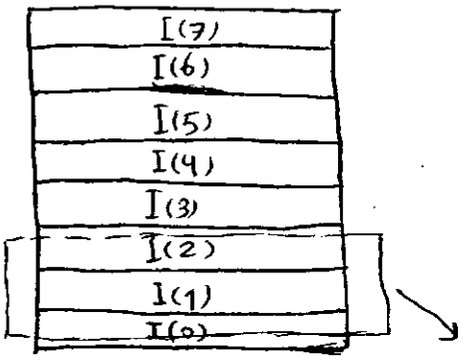
? سؤال: چند بسته را پشت سرهم بفرستیم؟

⊞ SLIDING WINDOW

پنجرهٔ لغزان

Ⓦ - اندازهٔ پنجره تعداد فریم‌هایی است که پشت سرهم فرستاده می‌شود و سپس منتظر Ack ها می‌مانیم.

- فرستنده باقری دارد که اطلاعات را در آن نگاه داری می‌کند.



+ با دریافت Ack ، فریم مربوطه را حذف می‌کنیم و پنجره حاوی عضو دیگری می‌شود که می‌توان آن را روی خط فرستاد.

+ فعلاً حالتی را بررسی می‌کنیم که خطایی رخ نداده است.

⊕ مکانیزم‌های پاسخ به خطا :

Ⓐ مکانیزم Go-Back N :

- در صورت تشخیص **خطا** یک Negative Ack با شمارهٔ فریم خطا می‌فرستیم.
- تا وقتی که فریم خطا بصورت درست دریافت نشود ، NAck با همان شماره دایماً فرستاده می‌شود.
- در طرف فرستنده، در صورت دریافت NAck[n]، پیام n پیام بعدی است و بقیهٔ پیام‌های پنجره [پیام‌های بعدی] هم فرستاده می‌شوند.
- دریافت‌کننده باقی نیاز است که شمارهٔ آخرین پیام صحیح دریافت شده را در خود نگاه می‌دارد.
- تنها حُسن این مکانیزم این است که در طرف گیرنده توالی اطلاعات حفظ می‌شود.
- R از صفر شروع می‌شود و در صورت بروز خطا تا دریافت پیام صحیح همان فریم مقدارش اضافه نمی‌شود و در صورت دریافت صحیح همان پیام ، دوباره یکی یکی مقدارش اضافه می‌شود.

- از طرف فرستنده بافرهایی داریم که به اندازهٔ پنجره است.

(?) سؤال: اگر یکی از Ack ها خراب شود چه کار کنیم؟

- با نگاه کردن به بقیهٔ Ack ها می‌فهمیم که آیا خطایی رخ داده است یا خیر؟

+ اگر دریافت‌های بعدی Ack باشد یعنی همهٔ قبلی‌ها درست رسیده‌اند.

+ اگر دریافت‌های بعدی NAK(n) باشد یعنی از n به بعد را باید دوباره بفرستد.

⊕ مکانیزم‌های پاسخ به خطا:

Ⓑ SELECTIVE reTRANMISSION «انتقال انتخابی»

- با هر فریم خرابی مستقلاً برخورد می‌شود.

- برای هر فریم تایمری داریم که پس از نامعتبر شدن زمان، باید آن فریم را دوباره بفرستیم.

⊛ حُسن این روش در این است که درصورت بروز خطا، به جای n فریم فقط یک فریم فرستاده می‌شود!

↓ معایب:

1- به تعداد فریم‌ها تایمر نیاز است.

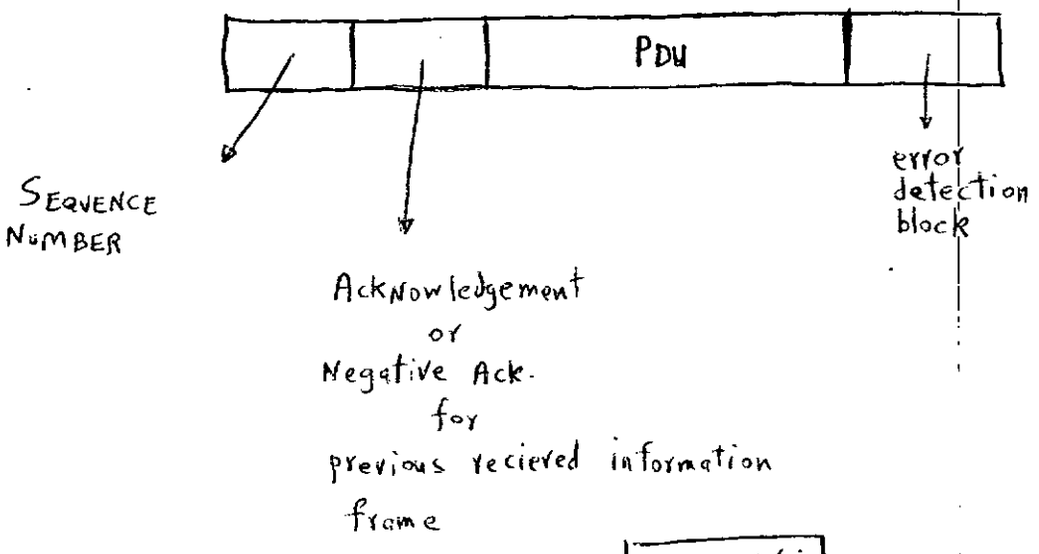
2- توالی از بین رفته است.

3- برای فریم‌های دریافتی در گیرنده برای حفظ توالی بافری داریم که همیشه مستمت پایین بافر (سریع و متصل) را به لایهٔ بالاتر می‌فرستیم.

نتیجه: Sel. reTran. از خط بهتر استفاده می‌کند ولی توالی از بین می‌رود.

"پروتکل Continuous ARQ دوطرفه"

+ اگر ارتباط طرفین دوطرفه باشد، یعنی کلی فقط گیرنده نباشد، Ack. ها در درون همان بسته‌های اطلاعاتی فرستاده می‌شوند.



شکل (22)

⊕ به این روش PIGGY BACKING می‌گویند.

⊕ اگر در هنگام فرستادن Ack فریم اطلاعاتی برای ارسال موجود نیست، مدت زمان خاصی منتظر دریافت فریم از لایه‌های بالاتر می‌مانیم و در صورت عدم دریافت خود Ack. را جداگانه می‌فرستیم!!

خلاصه‌ی پروتکل‌های ARQ

1- Idle RQ یا Stop & Wait

تصحیح: CHECK POINTING

2- CONTINUOUS RQ

a- Go Back N

b- Selective ReTransmission

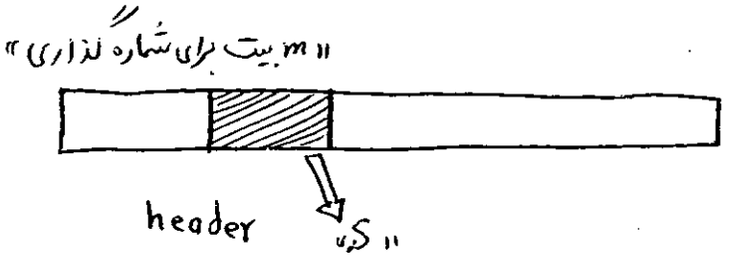
رجیسترها برای بستهٔ اطلاعاتی و Ack. «طرف اول» S_1 R_1

رجیسترها برای بستهٔ اطلاعاتی و Ack. «طرف دوم» S_2 R_2

ادامهٔ بررسی پروتکلهای CONTINUOUS RQ:

★ فرض کنیم برای تخصیص شماره به هر فریم اطلاعاتی m بیت در header آن تخصیص یافته.

- بدین ترتیب شمارهٔ فریم‌ها از ۰ تا 2^m-1 است.
- پس 2^m فریم قابل شماره‌گذاری هستند.



PDU

شکل

محاسبات روش Go Back N

$t_o \cong t_{x,\,data} + 2t_{Prop.}$

⊕ در مدت t_o چند فریم می‌توان انتقال داد؟

t_o / t_x = «تعداد فریم‌ها»

⊕ اندازهٔ پنجره باید کوچکتر از این مقدار باشد. (در صورت بزرگتر بودن دیگر معنایی ندارد.)

$$W \leqslant t_o / t_x$$

⊕ تعداد فریم‌ها $= 2^m$

در نتیجه

$$W \leqslant 2^m - 1$$

⊛ رابطه $W \leqslant 2^m$ مشکل زا است چون در صورت نرسیدن Ack[0] باید I[0] را دوباره بفرستیم و در صورت وجود اطلاعات جدید باز هم باید I[0] بفرستیم که این دو حالت قابل تفکیک نیستند. در نتیجه باید اندازهٔ پنجره یکی کمتر از تعداد فریم‌های قابل شماره‌گذاری باشد.

محاسبات روش Selective :

- گیرنده و فرستنده بافرهایی دارند که به عملیات ما کمک می‌کنند.
فرستنده تا Ack. دریافت نکند یکی از مربوطه‌ها را در بافر خودش خالی نمی‌کند.
- اندازهٔ پنجره باید کمتر از نصف تعداد فریم‌ها باشد. (چون بافرها باید پر شوند)
- با فرض اینکه Ack. (در مرتبهٔ اول) خراب می‌شود. (بدترین حالت)

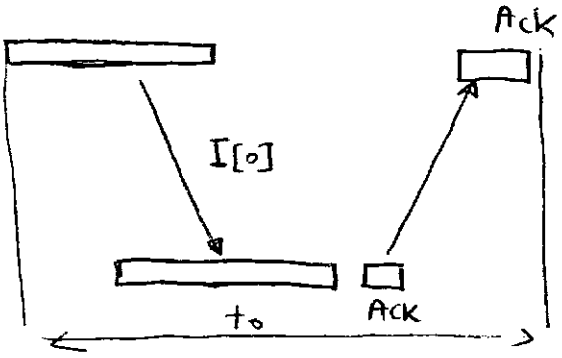
در نتیجه:

$$2W \leqslant 2^m$$

$$\Rightarrow \boxed{W \leqslant 2^{m-1}}$$

⊕ بررسی کاربردهای سه پروتکل ARQ :

(I) IDLE RQ PROTOCOL :



شکل (223)

Idle RQ

① تعریف : میزان بهره وری خط : η (اتا)

$$\eta = t_x / t_0 = \frac{\text{زمان ارسال بسته اطلاعاتی}}{\text{کل زمان در اختیار}}$$

$$\eta = \frac{n_f / R}{t_0} = \frac{\text{تعداد بیتهای فریم}}{\text{نرخ بیت}} \Big/ \text{کل زمان}$$

t_x = زمان ارسال فریم

t_0 = کل زمان

n_f = تعداد بیتهای فریم اطلاعاتی

R = نرخ بیت

+ الف - میزان بهره وری خط در حالت بدون خطا :

$$t_0 = t_{x,data} + 2t_{prop.} + t_{x,ack} + t_{proc.ack} + t_{proc.data}$$

$t_{tx,\,data} = n_f / R$

$t_{tx,\,Ack.} = n_a / R$

$t_{prop.} = d / V$

n_a = تعداد بیت‌های Ack.

- زمان‌های پردازش کیان فرض شده‌اند.

$$t_0 = n_f/R + n_a/R + 2d/V + 2t_{Proc.}$$

نتیجه:

$$\eta = \frac{\frac{n_f}{R}}{\frac{n_f}{R} + \frac{n_a}{R} + 2d/V + 2t_{proc.}}$$

$$\boxed{\eta = \frac{n_f}{n_f + n_a + R(2t_{prop.} + 2t_{proc.})}}$$

میزان بهره‌وری از خط در حالت بدون خطا - پروتکل Idle RQ

⊕ تعریف دیگر از بهره‌وری خط:

$$\eta = \frac{\text{نرخ بیت مؤثر}}{\text{نرخ بیت خط}}$$

$$\boxed{\eta = \frac{R_{eff.}}{R_{CHANNEL}}}$$

---

(۴) محاسبه بهره‌وری خط از تعریف دوم:

$$R_{eff.} = \frac{n_f}{t_0} = \text{نرخ بیت مؤثر}$$

$$\Rightarrow \eta = \frac{n_f}{t_0} / R = \frac{n_f}{R} / t_0$$

شبکه ۱ — جلسه نهم

(۴) محاسبه بهره‌وری خط با احتساب بیتهای اضافه در header و trailer هر فریم:

$$\eta^{\circ}_{Idle\text{-}RQ} = \frac{R^{\circ}_{eff.}}{R} = \frac{nf - n_o}{nf + na + 2R(t_{prop.} + t_{proc.})}$$

⊕ بهره‌وری خط در حالت بدون خطا Idle-RQ

n_o = « بیتهای اضافی (در هر فریم) »

nf = « تعداد بیتهای فریم »

⊕ بررسی فاکتورها:

$$\eta^{\circ} = \frac{1 - n_o/nf}{1 + \frac{na}{nf} + \frac{2R(t_{prop.} + t_{proc.})}{nf}}$$

- الف - $\frac{n_o}{nf}$: با زیاد شدن نسبت بیتهای اضافه به بیتهای فریم بهره‌وری کم می‌شود.

- ب - t_{prop} : با زیاد شدن فاصله طرفین بهره‌وری کم می‌شود.

- ج - R : با زیاد شدن نرخ بیت بهره‌وری کم می‌شود!

زیرا در این صورت در زمان کوتاهی فریم روی خط می‌رود و بقیه زمانها بهره بیشتری پیدا می‌کنند.

★ توجیه ب و ج - $2\frac{t_{prop}}{t_n}$: با زیاد شدن این نسبت بهره‌وری کم می‌شود.

→ $2\,t_{prop}.R$: با زیاد شدن این عبارت بهره‌وری کم می‌شود.

یعنی با نرخ × فاصله

n_o/n_f ↑	↓ η
$t_{prop.}$ ↑	↓ η
R ↑	↓ η
$\frac{t_{prop.}}{t_n}$ ↑	↓ η
$t_{prop.} \cdot R$ ↑	↓ η

فاصله × پهنای باند

---

⑤ مثال عددی :

IDLE-RQ

مفروضات
$\begin{cases} n_f = 1024 \text{ bytes} \\ n_a = 8 \text{ bytes} \\ n_o = 8 \text{ bytes} \end{cases}$

$t_{proc.} \cong 0$ صرفنظر از زمان پردازش

مفروضات در سه حالت مختلف

$d = \begin{cases} 1500 \text{ km} \\ 15000 \text{ km} \\ 150000 \text{ km} \end{cases}$ $R = \begin{cases} 30 \text{ kbps} \\ 1.5 \text{ mbps} \\ 45 \text{ mbps} \end{cases}$

محاسبه زمان انتشار

$t_{prop.} = d/v = \frac{1500 \times 10^3 \text{ m}}{3 \times 10^8 \text{ m/s}} \Rightarrow \boxed{t_{prop.} = 5 \times 10^{-3} \text{ s}}$

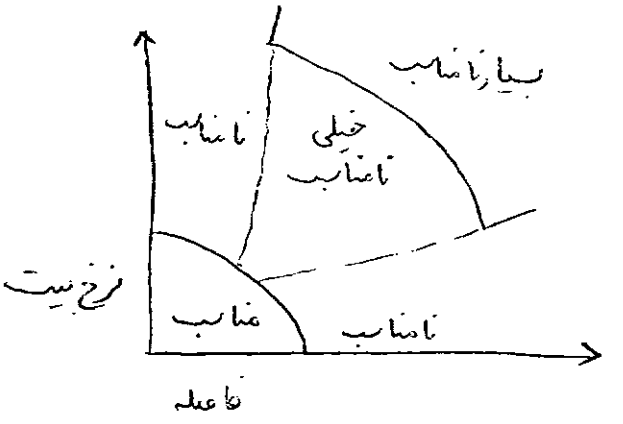
مقادیر η برای حالات مختلف.

« بهره وری خط »

$d/t_{prop.}$ \ R	30 kbps	1.5 mbps	45 mbps
$1500^{km}/5\times10^{-3}s$	0.95	0.35	1.77×10^{-2}
$15000^{km}/5\times10^{-2}s$	0.72	0.0514	1.8×10^{-3}
$150000^{km}/5\times10^{-1}s$	0.21	0.00539	1.81×10^{-4}

نتیجه:

پروتکل Idle-RQ برای فواصل کم و پهنای باند یا نرخ بیت کم مناسب و برای فاصله زیاد و نرخ بیت بالا بسیار نامناسب است.



Trade-off بهره‌وری خط در پروتکل Idle-RQ در حالت بدون خطا

شکل (224)

⊕ ب - بهره‌وری خط در حالت بروز خطا در پروتکل IDLE-RQ

⊕ زمان انقضای Timer را $t_{t\text{-}out}$ می‌نامیم.

$$t_{t\text{-}out} \geq t_0$$

⊛ زمان انقضا حتماً باید بیشتر از t_0 باشد چون در این فاصله فرستنده منتظر Ack است.

- اگر $i-1$ بار تایمر منقضی شود و در مرتبهٔ iام ارسال داشته باشیم:

$$t_{کل} = (i-1)\,t_{t\text{-}out} + t_0$$

$$\Rightarrow \quad \eta = t_x / t_{کل}$$

- خطا پدیده‌ای است آماری و بدین جهت مقدار متوسط کل t را حساب می‌کنیم.

$$\Rightarrow \quad \eta = t_x / E[(i-1)\,t_{t\text{-}out} + t_0] \longrightarrow \text{مقدار متوسط آماری}$$

۴ محاسبه مقدار متوسط زمان کل :

p = احتمال خطای بیتی

$1-p$ = احتمال درست رسیدن بیت

⇒ احتمال درست رسیدن کل فریم = $(1-p)^{nf}$

اگر p بسیار کم باشد ⇒ احتمال درست رسیدن = $1-p^{nf} = 1-p_f$

⇒ احتمال دریافت خطادار پیام یا فریم = $1-(1-p)^{nf}$

درصورت کم بودن p ⇒ احتمال خطای فریم = $1-(1-p^{nf})$

⇒ احتمال خطای فریم = $p^{nf} = p_f$

جایگزینی p_f بجای p^{nf} برای سهولت

$p^{nf} = p_f$ احتمال دریافت صحیح

$1-p^{nf} = 1-p_f$ احتمال دریافت خطادار

* درصورت غلط بودن تقریب از رابطه‌ی اصلی

(احتمال خطا) $p_f = 1-(1-p)^{nf}$

$1-p_f = (1-p)^{nf}$

استفاده می کنیم

p_f = احتمال دریافت خطای فریم

$1-p_f$ = احتمال دریافت صحیح فریم

بار ارسال / تعداد ارسال برای رسیدن صحیح یک بسته	زمان (زمان کل برای ارسال صحیح در تعداد دفعات ارسال)	احتمال (احتمال وقوع تعداد دفعات)
times (n)	Time	Likelihood
1 بدون خطا	t_0	$1-P_f$
2 یکبار خطا	$t_0 + t_{t.out}$	$P_f(1-P_f)$
3	$t_0 + 2t_{.out}$	$P_f^2(1-P_f)$
i	$t_0 + (i-1)t_{.tout}$	$P_f^{i-1}(1-P_f)$

⊛ فرض: $t_0 = t_{.tout}$ زمان انقضای تایمر = زمان t_0

$$E[t] = t_0(1-p_f) + 2t_0 p_f(1-p_f) + \cdots + it_0 p_f^{i-1}(1-p_f)$$

⊕ به عنوان تمرین این مقدار متوسط را به دست آورید.

$$E[t] = \sum_{k=1}^{i} it_0 p_f^{i-1}(1-p_f) = t_0 \sum_{K=1}^{i} i(p_f^{i-1} - p_f^{i})$$

$$= t_0 \sum_{k=1}^{i} i p_f^{i-1} - t_0 \sum_{k=1}^{i} i p_f^{i}$$

⊕ ادامه‌ی محاسبه‌ی $E[t]$ برای پروتکل Idle_RQ در حالت خطادار:

$$E[t] = \sum_{K=1}^{i} K t_0 p_f^{K-1}(1-p_f) = t_0(1-p_f)\sum_{K=1}^{i} K p_f^{K-1}$$

t: مقدار مترقب

⊛ با فرض $t_0 = t_{timeout}$.

- در سال‌های واقعی i محدودیت دارد، یعنی درصورت زیادبودن i به‌سبب گذر قطعی خط فرآیند موقتاً خاتمه می‌یابد.

+ در اینجا ما فرض می‌کنیم i تا بی نهایت برود. (برای تسهیل محاسبات)

$$\Rightarrow E[t] = (1-p_f)t_0\left[1 + p_f + \cdots + p_f^{i-1} + \cdots\right]$$

$$+ (1-p_f)t_{out}\left[p_f + 2p_f^2 + 3p_f^3 + \cdots + ip_f^i + \cdots\right]$$

$$\Rightarrow E[t] = (1-p_f)t_0\left[\frac{1}{1-p_f}\right] \longleftarrow \text{مجموع تصاعد هندسی}$$

$$+ (1-p_f)t_{out}\left[\frac{p_f}{1-p_f} + \frac{p_f^2}{1-p_f} + \frac{p_f^3}{1-p_f} + \cdots\right]$$

$$\Rightarrow E(t) = t_0 + t_{out}\left[p_f + p_f^2 + p_f + \cdots\right]$$

t_0: اول — $p_f + p_f^2 + p_f + \cdots$: دوم

⊛ درصورت مساوی‌نبودن t_0 و t_{out}:

$$E(t) = t_0 + t_{out} \times \frac{p_f}{1-p_f}$$

⊛ درصورت مساوی‌بودن $t = t_{out}$:

$$E(t) = \frac{1}{1-p_f} \times t_0$$

$1-P_f$ = احتمال درست رسیدن بسته

$$\Rightarrow \quad E[t] = \frac{t_0}{1-P_f}$$

$$\Rightarrow \quad \eta^\circ = \frac{\frac{n_f - n_0}{t_0}}{R}$$

بدون خطا

$$\eta = \eta^\circ \times (1-P_f)$$ خطادار

N_r = تعداد متوسط دفعات ارسال

$$E[t] = N_r \cdot t_0 \quad \Rightarrow \quad N_r = \frac{1}{1-P_f}$$

تعداد متوسط دفعات ارسال

احتمال خطای فریم $= P_f$	$\eta = \eta^\circ / N_R$ بهره‌وری در حالت خطادار.

p = احتمال خطای بیتی	تقریب‌ها $(1-p)^{nf} \simeq 1 - p \cdot nf$ pهای کوچک
$1-p$ = احتمال درست رسیدن بیتی	
$(1-p)^{nf} \simeq 1-P_f$ = احتمال درست رسیدن بسته	$(1-p)^{nf} \simeq e^{-pnf}$ pهای کوچک و nfهای بزرگ

$$P_f = C \cdot P \cdot nf$$

- nf: تعداد بیتهای فریم
- P: خطای بیتی
- C: ضریبی که ناشی از عدم تشخیص خطا در الگوریتمهای تشخیص خطاست.

P_f = احتمال خطای فریم

⊕ برای روشهای مختلف $\frac{1}{10} \leqslant C \leqslant \frac{1}{2}$

II CONTINUOUS RQ

الف) حالت بدون خطا:

- بجای صبرکردن در زمان t_0، W تا فریم می فرستیم، نتیجتاً بهره وری در اینجا W برابر است زیرا در زمان های برابر t_0، W برابر حالت مشابه در Idle-RQ اطلاعات مفید فرستاده می شود.

$$\eta^{\circ}_{CONT.} = \eta^{\circ}_{idle} * W$$

حالت بدون خطا

$$W_{MAX} = t_o / t_x$$

$$\eta^{\circ}_{\text{حداکثر}} = \eta^{\circ}_{idle} = \frac{W_{max} \cdot \frac{nf-no}{R}}{t_o} = \frac{\frac{t_o}{t_x} \times \frac{nf-no}{R}}{t_o}$$

$$= \frac{nf-no}{Rx} = \frac{nf-no}{nf} \Rightarrow$$

$$\eta^{\circ}_{\substack{\text{حداکثر} \\ \text{بدون خطا} \\ \text{CONT.}}} = 1 - no/nf$$

در واقع در این جا به بهره وری کامل نزدیک شده ایم و تنها عامل بازدارنده بیتهای اضافی است

ب) حالت خطادار : مکانیزم Go Back N :

+ می توان از مباحث به کار رفته در حالت خطادار Idle-RQ استفاده کرد.
+ فرض بر این است که اندازه پنجره بیشینه است.

⊛ $W = W_{MAX} = t_o / t_x$

GB.N

تعداد دفعات ارسال	1	2	3	تعداد ارسال یک فریم	I
N ارسال فریم	1	1+W	1+2W	تعداد ارسال کل فریم ها	1+(I-1)W
احتمال	1-Pf	Pf(1-Pf)	$P_f^2(1-P_f)$	احتمال	$P_f^{I-1}(1-P_f)$

$$E[t] = E[N_T] \times t_x$$

مقدار مترقب زمان

«محاسبهٔ مقدار مترقب تعداد دفعات ارسال» (کل فریم‌ها)

$$N_T = 1 \times (1-p_f) + (1+W)\, p_f(1-p_f) + \text{——}$$

$$= (1-p_f)\left[1 + p_f + p_f^2 + \text{——}\right]$$

$$+W p_f(1-p_f) + 2W p_f^2 (1-p_f) + \text{——}$$

$$= 1 + (1-p_f) W \sum_{i=1}^{\infty} i p_f^i$$

$$= 1 + W(1-p_f) \times \frac{(p_f)}{(1-p_f)^2}$$

$$\Rightarrow \boxed{N_T = 1 + \frac{W p_f}{1-p_f}}$$

$$\eta = \frac{1}{N_T}\left(\eta^{\circ}_{\substack{\text{Max} \\ \text{CONT.}}}\right) = \frac{1}{N_T}\left(1 - \frac{n_o}{n_f}\right)$$

$$\eta = \frac{1}{N_T}\left(\frac{n_f - n_o}{n_f}\right) = \frac{1-p_f}{1+(W-1)p_f}\left(\frac{n_f - n_o}{n_f}\right)$$

$$\Rightarrow \boxed{\eta = \frac{(1-p_f)(n_f - n_o)}{\left(p_f(W-1)+1\right)(n_f)}}$$

کنترل خطا در Go Back N با وجود خطا

تمرین کلاسی E-e.4 : محاسبهٔ η در حالتی که اندازهٔ پنجره کوچکتر از مکان تعیین است

ج) خطادار: مکانیزم SELECTIVE Retransmission

S.R.

ارسال فریم	1	2		i
N	1	2		i
احتمال	$(1-p_f)$	$p_f(1-p_f)$		$p_f^{i-1}(1-p_f$

محاسبهٔ مقدار مترقب تعداد دفعات ارسال:

$$N_r = (1-p_f) + 2p_f(1-p_f) + \text{---} + i\,p_f^{i-1}(1-p_f) \text{---}$$

$$= (1-p_f)\left[1 + 2p_f + \text{---} + i\,p_f^{i-1}\right]$$

$$\Rightarrow \quad N_r = \frac{1}{(1-p_f)}$$

$$\eta_{S.R} = (1-p_f)\left(1 - \frac{n_o}{n_f}\right)$$

بهره وری خط در حالتی که خطا داریم. Selective Retr.

تمرین کلاسی: محاسبهٔ بهره وری خط در حالتی که اندازهٔ پنجره ماکزیمم نباشد.

⊕ مقایسهٔ روشهای مختلف ARQ

مثال:

$$w_{max} = \frac{t_o}{t_x} = \frac{t_x + 2t_{prop.} + 2t_{proc.} + t_{ack.}}{t_x}$$

$$w_{max} \simeq \frac{t_x + 2t_{prop.}}{t_x}$$

$$w_{max} \simeq 1 + 2\, t_{prop.}/t_x$$ اندازهٔ تقریبی پنجره

a از پارامترهای خط است و برابر زمان انتشار به زمان ارسال است.

$$a = t_{prop.}/t_x$$

$$\Rightarrow \boxed{w \leq 1 + 2a}$$

$$t_x = \frac{L}{R} \qquad t_{prop.} = d/V$$

$$t_{prop.}/t_x = \frac{d/V}{L/R} = \frac{dR}{LV}$$

$$a \sim \frac{dR}{LV}$$ ⇐ پارامترهای اثرگذار بر پارامتر @

$W \sim a \Rightarrow$

$d \uparrow$	$W_{max} \uparrow$
$L \uparrow$	$W_{max} \downarrow$
$R \uparrow$	$W_{max} \uparrow$

$V \uparrow \; W_{max} \downarrow$

مثال عددی - + اندازه پنجره +

W_{MAX}	R = 30 Kbps	R = 1.5 Mbps	R = 45 Mbps
d = 1500 km $t_{prop.}$ = 5 ms	2	4	57
d = 15000 km $t_{prop.}$ = 50 ms	2	20	551
d = 150000 km $t_{prop.}$ = 500 ms	6	185	5495

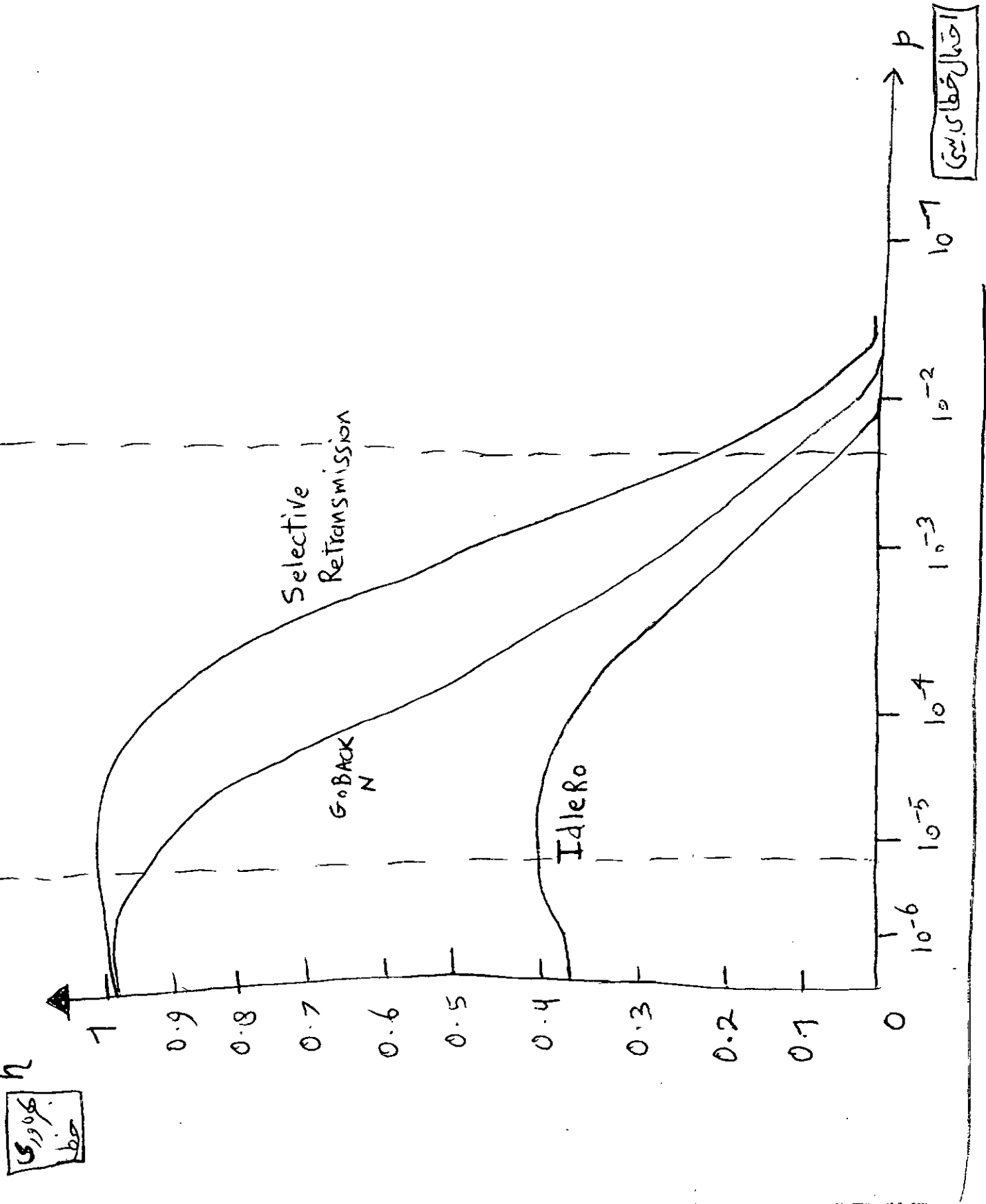
⊕ نتیجه: در جاهایی که اندازه پنجره ماکزیمم خیلی زیاد است، پروتکل Continuous بهره وری بسیار بالایی دارد.

توجیه:

⊕ در جاهایی که اندازه پنجره ماکزیمم کم است Idle RQ خوب است.

توجیه:

[مقایسهٔ پروتکلهای ARQ]

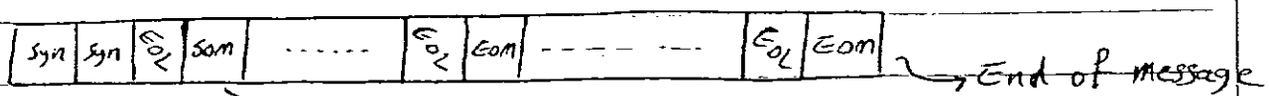


الف) احتمال خطای بسیار کم:

CONTINUOUS RQ بهره‌وری تقریباً صددرصدی دارد.

IDLE RQ بهره‌وری کمی دارد.

۲- Character oriented :



- مقدری از طول 8 انتقال داده می شود.

- در شروع کاراکتر Syn و در انتها EOM (End of Message) برای مرزبندی گیرنده برای جلوگیری از شکل تکرار EOM

جلسه نوزدهم: 84,2,7

کنترل خطا:

ARQ: Automatic Retransmission reQuest

پروتکلی برای فرستادن دوباره اطلاعات

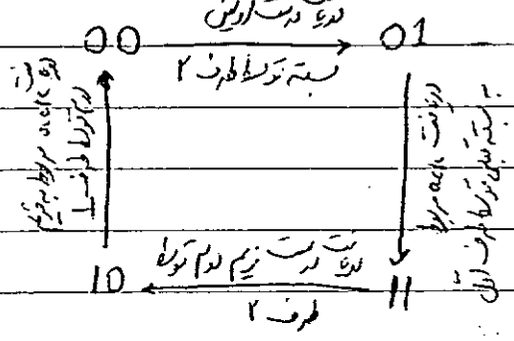
چون ماشین انجام کار می کند لذا عبارت Automatic می آید.

* فرض می کنیم در اثر اطلاعات غلط دریافت شد آن ACK صحیح بسته ای نرسید.

$$T_{timer} \geq t_x + 2t_p$$

* برای اینکه گیرنده بفهمد کدام بسته فرستاده شده دقیقا نمی داند اسم که در نظر می گیرند بسته که را مشخص می کنیم. گیرنده هم می داند - R در نظر می گیریم که متقابلا اولیه حول آن صفر است.

در ابتدا فرستنده بسته با شماره I[0] را می‌فرستد. گیرنده اگر بسته درست دریافت شد یک واحد اضافه می‌شود (=1) سپس ACK[1] را برای فرستنده می‌فرستد. پس از دریافت این بسته فرستنده بسته بعدی با شماره I[1] را می‌فرستد (ابتدا یک واحد اضافه می‌شود) پس از دریافت درست گیرنده، R صفر می‌شود و ACK[0] را می‌فرستد. پس از دریافت فرستنده S صفر می‌شود و بسته I[0] را می‌فرستد...



* پروتکل گفته شده را Stop_and_Wait یا (idle-RQ) می‌نامند.

این پروتکل استفاده موثری از خط نمی‌کند چون زمان انتظار زیادی صرف می‌شود!

$T = t_x + 2t_p$

اگر فاصله زیاد باشد t_p زیاد است.

و اگر بسته اطلاعات حاوی تعداد بیت زیادی باشد t_x زیاد است.

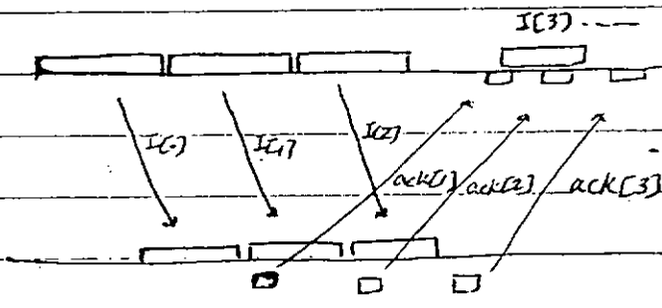
الکون فرض می‌کنیم فاصله و تعداد بیت کم زیاد است. در این حالت در صورتیکه گیرنده Ack دریافت نشود یک بسته بسیار کوچک بنام NAK می‌فرستد که گیرنده بفهمد! چه اطلاعاتی خراب رسیده بود یا خیر.

به این روش Check pointing می‌گویند.

ARQ
- Stop-and-wait (idle RQ) ← توضیح داده شد
- Continuous ARQ

Continuous ARQ:

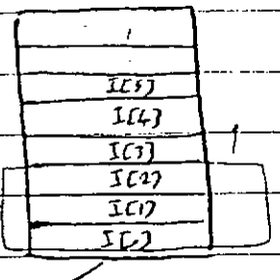
در این روش بسته ای شماره گذاری شده پشت سر هم در خط فرستاده می‌شود بدون اینکه Ack ها شماره مخصوص به خود داشته باشند می‌شود.



* سوالی که مطرح می‌شود اینه چند بسته را پشت سر هم بفرستیم؟

Sliding Window (پنجره لغزان) ..... اندازه پنجره = W

* به فرض اینکه از Ack ها دریافت شود پنجره یک واحد به سمت بالا می‌رود.



* حال پنجره مثل این است که پس از دریافت Ack آخرین اطلاعات با فریم بعدی پر شود و پنجره یک واحد به سمت پایین می‌آید.

موضوع: ..... تاریخ: .....

* در اندازه بسته می توانیم اطلاعاتی را نیز بفرستیم که Aek آن را دریافت نکرده ایم.

* تا کنون صرف از ایجاد خطا نداشته ایم!

Go-Back-N
Selective Retransmission
} Continuous RQ

حال فرض می کنیم که در دیاگرام قبل، I[1] غلط دریافت شده باشد.

Go Back N :

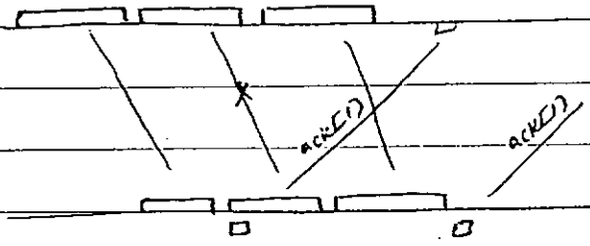
بسته NAK[i] را می فرستد به همین ترتیب چون I[1] خراب شده، NAK[2] را برای I[2] می فرستد. در طرف فرستنده در صورت دریافت NAK[n] یا n و بعد از آن همراه با بقیه پیام ها را می فرستد (از اندازه حول بخبر) فرستاده می شود.

- نقاط ضعف این مکانیزم این است که در طرف گیرنده تمام پیام ها رعایت شده است.

R : شماره آخرین فریم درست دریافتی را ذخیره می کنیم.

حال فرض می کنیم که ACK[1] درست آمده. بسته I[2] غلط است آمده و NAK[2] خراب شده و NAK[3] بعدی هم خراب شود تا اینکه NACK[1] بعدی درست برسد از همین رو می فهمیم که فقط I[2] درست دریافت نشده است.

* Selective Transmission : در این روش فقط فریم‌هایی که خطا دارند ارسال می‌شوند.



عیب این روش نسبت به روش قبل این است که وقتی یک بسته خراب می‌شود یا بسته‌ای گم می‌شود، در Go back N همه بسته‌ها را مجدداً ارسال می‌کرد ولی در این روش فقط یک بسته ارسال می‌شود.

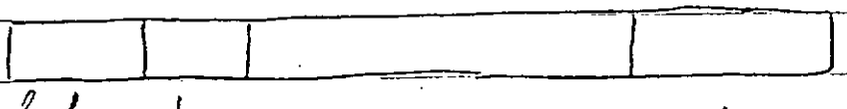
عیبش این است که طرف گیرنده باید بافر بزرگی برای نگهداری فریم‌های خارج از توالی داشته باشد. همین طور به تعداد فریم‌ها بافر نیاز داریم.

* مثلاً بسته 1 درست آمده است. بسته 2 دریافت نشده و بسته 3 هم درست دریافت شده هنگامی که بسته 2 دریافت شود او 2 تا بافر فضای می‌خواهد که باید در نظر گرفته شود.

* این روش از خطا استفاده بهتری می‌کند ولی کمی از این می‌رود.

* کنترل خطا ARQ و Continuous دو طرفه:

در این حالت، هر دو طرف می‌توانند هم‌زمان فرستنده و گیرنده عمل کنند.



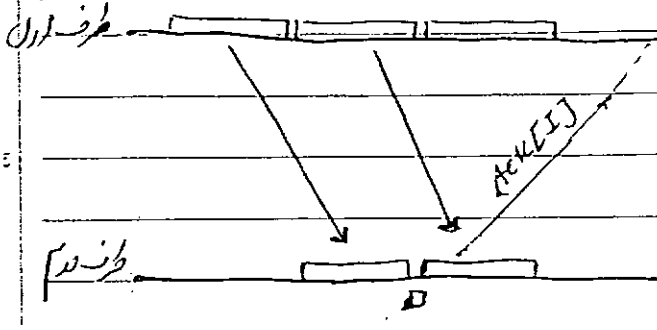
* این روش Piggy Backing نام دارد.

موضوع: ..... تاریخ: .....

جلسه بیست: 84/2/10

t_0

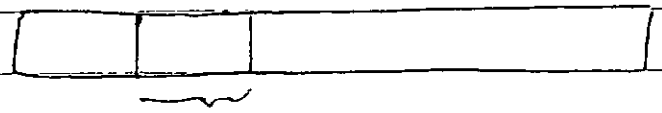
Go-Back-N



$$t_0 = t_{x,max} + 2t_{prop} + t_{proc,data} + t_{proc,ack} + t_{x,ack}$$

$$t_0 \cong t_{x,data} + 2t_{prop}$$

حداکثر تعداد فریم: $w \leq \frac{t_0}{t_x}$



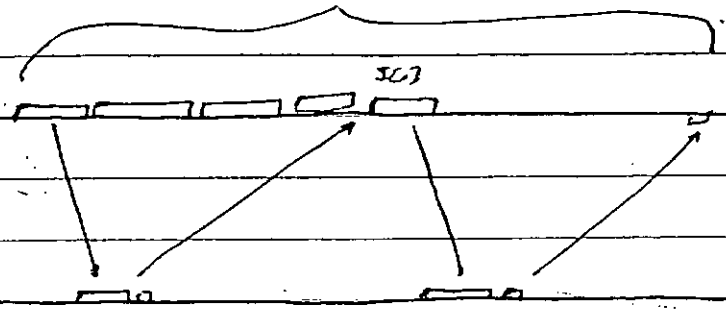
m بیت = 2^m حالت، فضای شماره

اگر تعداد فریم‌ها، 2^m یا بیشتر باشد، در پذیرنده چه ACK بیاید چه نیاید فرستنده آن را دریافت کرده است. لذا فرستنده می‌تواند پنجره اطلاعات جدید را بفرستد.

بنابراین تعداد پنجره حداکثر $2^m - 1$ تا باشد

$$w \leq 2^m - 1$$

Selective-Repeat:



* اگر فرض کنیم ACK خراب شود و حداکثر همان فرایند را دنبال کنیم:

$$w \leq 2^{m-1}$$

$$w \leq \frac{2^m}{2}$$

موضوع: ..... تاریخ: .....

Idle RQ

η میزان بهره وری لینک:

کل زمانی که خط را صرف ارسال داده ها کرده است / کل زمانی که خط را صرف ارسال داده ها کرده است + زمان اتلاف بدون خطا

$$\eta = \frac{t_x}{t_0} = \frac{\frac{n_f}{R}}{t_0}$$

t_x زمانی که خط را صرف ارسال داده ها کرده است

t_0 کل زمانی که خط را صرف ارسال داده ها کرده است + زمان اتلاف بدون خطا

n_f تعداد بیت های فریم

R نرخ بیت

$$t_0 = t_{x,data} + 2t_{prop} + t_{x,data} + t_{proc} + t_{x,ack} + t_{proc}$$

t_0 زمان لازم برای ارسال یک فریم

$$t_{x,ack} = \frac{n_a}{R}$$

n_a تعداد بیت های ack

$$t_{x,data} = \frac{n_f}{R}$$

n_0 بیت های اضافی

$$t_{prop} = \frac{d}{v}$$

$$t_0 = \frac{n_f}{R} + \frac{n_a}{R} + 2\frac{d}{v} + 2t_{proc}$$

$$\Rightarrow \eta = \frac{t_x}{t_0} = \frac{n_f/R}{n_f/R + n_a/R + \frac{2d}{v} + 2t_{proc}}$$

$$= \frac{n_f}{n_f + n_a + (2t_{prop} + 2t_{proc})R}$$

تعریف دیگر:

$$\eta = \frac{R_{eff}}{R}$$

R_{eff} نرخ بیت مؤثر

$$R_{eff} = \frac{n_f}{t_0} = \frac{n_f}{\frac{n_f}{R} + \frac{n_a}{R} + 2t_{prop} + 2t_{proc}} = \frac{R n_f}{n_f + n_a + (2t_{prop} + 2t_{proc}) * R}$$

$$\eta = \frac{R_{eff}}{R} = \frac{n_f}{n_f + n_a + (2t_{prop} + 2t_{proc}) * R}$$

$$\eta_{idle-RQ} = \frac{R^{\circ}_{eff}}{R} = \frac{n_f - n_0}{n_f + n_a + (2t_{prop} + 2t_{proc}) * R}$$

$$= \frac{1 - \frac{n_0}{n_f}}{1 + \frac{n_a}{n_f} + \frac{(2t_{prop} + 2t_{proc})R}{n_f}}$$

t_x^{-1}

$\frac{n_0}{n_f} \uparrow$ $\eta \downarrow$ (?)

$t_{prop} \uparrow$ $\eta \downarrow$

$R \uparrow$ $\eta \downarrow$

$\frac{2t_{prop}}{t_x} \uparrow$ $\eta \downarrow$

$t_{prop} \times R \uparrow$ $\eta \downarrow$

مثال:

d { 1500 km, 15000 km, 150000 km } , $n_0 = 8$ Byte , $n_a = 8$ Byte , $n_f = 1024$ Byte ÷ ؟

R { 30 Kbps, 1.5 Mbps, 45 Mbps }

P η

$t_{prop} = \frac{d}{v} = 5\times10^{-3}\ s$

" = " = 5×10^{-2}

" = " = 5×10^{-1}

d, t_{prop} \ R	30 Kbps	1.5 Mbps	45 Mbps
1500 Km, 5×10^{-3} s	0.95	0.35	1.77×10^{-2}
15000 Km, 5×10^{-2} s	0.72	0.0514	1.8×10^{-3}
150000 Km, 5×10^{-1} s	0.21	0.00	1.81×10^{-4}

(وجود)

در صورت بروز خطا:

یک تایمر هم در نظر می‌گیریم. (t_{out})

$t_{out} \geq t_0$

فرستنده:

زمان ارسال فریم i = $(i-1)\, t_{out} + t_0$

$$\eta = \frac{t_x}{E[t]} = \frac{\eta f/R}{E[t]}$$

احتمال درست دریافت شدن یک بیت $= (1-P)$ ← احتمال خطای بیتی P_s

احتمال دریافت بدون خطا $= (1-p)^{n_f} = 1 - p_{n_f} = 1 - P_f$

(اگر p کوچک باشد)

احتمال دریافت حداقل با یک خطا $= 1 - (1-p_{n_f}) = p_{n_f} = P_f$

$$\text{یا} \begin{cases} P_f = 1-(1-p)^{n_f} \\ 1-P_f = (1-p)^{n_f} \end{cases}$$

تعداد ارسال	زمان	احتمال
۱	t_0	$1-P_f$
۲	$t_{out}+t_0$	$P_f(1-P_f)$
۳	$2t_{out}+t_0$	$P_f^2(1-P_f)$
⋮	⋮	⋮
i	$(i-1)t_{out}+t_0$	$P_f^{i-1}(1-P_f)$

$$\eta = \frac{t_x}{E[t]} \quad , \quad E[t] = t_0 \times (1-P_f) + (t_{out}+t_0) P_f (1-P_f) + \cdots$$

$$+ (i-1)t_{out} + t_0) P_f^{i-1} (1-P_f) + \cdots$$

جلسه هفتم - دوشنبه ۱۲/۲/۹۸

ادامه کنترل ص قبل

$$= (1-P_f)\,t_0\left[1 + P_f + P_f^2 + \cdots + P_f^{(i-1)} + \cdots\right] + \frac{1}{1-P_f}$$

$$(1-P_f)\,t_{out}\left[P_f + 2P_f^2 + 3P_f^3 + \cdots + (i-1)P_f^{i-1} + \cdots\right]$$

$$= t_0 + (1-P_f)\cdot t_{out}\cdot\frac{P_f}{(1-P_f)^2} = t_0 + \sum_{i=1}^{\infty} i\,t_{out}(1-P_f)P_f^{\,i}$$

A

لرتب

$$A = P_f + P_f^2 + P_f^3 + P_f^4 + \cdots + (i-1)P_f^{i-1} + \cdots \quad \frac{P_f}{1-P_f}$$

$$+ P_f^2 + P_f^3 + P_f^4 + \cdots + (i-1)P_f^{i-1} + \cdots \quad \frac{P_f^2}{1-P_f}$$

$$+ P_f^3 + P_f^4 + \cdots + (i-1)P_f^{i-1} + \cdots \quad \frac{P_f^3}{1-P_f}$$

$$+ P_f^4 + \cdots + (i-1)P_f^{i-1} + \cdots \quad \frac{P_f^4}{1-P_f}$$

$$= \frac{P_f}{1-P_f} + \frac{P_f^2}{1-P_f} + \frac{P_f^3}{1-P_f} + \cdots = \frac{1}{-P_f+1}\times P_f \times \frac{1}{1-P_f} = \frac{P_f}{(1-P_f)^2}$$

(؟) $\Rightarrow$ $\boxed{E[t] = t_0 + t_{out}\cdot\frac{P_f}{1-P_f}}$

$t_0 = t_{out} \Rightarrow \frac{t_0(1-P_f) + t_0 P_f}{1-P_f} = \frac{t_0}{1-P_f}$

شرط بهره‌وری از خط در حالت خطا دارد

$$\eta = \frac{R_{eff}}{R} = \frac{\frac{n_f - n_0}{E[t]}}{R} = \frac{\frac{n_f - n_0}{t_0/(1-P_f)}}{R} = \frac{(1-P_f)\frac{n_f - n_0}{t_0}}{R} = (1-P_f)\,\eta'$$

$$\begin{cases} E[t] = \dfrac{t_0}{1-P_f} \\ E[t] = N_r \cdot t_0 \end{cases} \Rightarrow N_r = \frac{1}{1-P_f}$$

تعداد متوسط ارسال ←

$$\Rightarrow \eta = \frac{\eta^*}{N_r}$$

استنتاج: $(1-\rho)^{n_f} \cong 1-\rho n_f$ ، $1-P_f = (1-\rho)^{n_f}$ (تقریب)

$$P_f = 1-(1-\rho)^{n_f} \qquad 1-(1-\rho)^{n_f} \cong \rho n_f$$

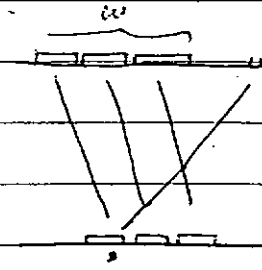
$$P_f \cong n_f \rho$$

از تخمین دیگر هم می‌توانیم استفاده کنیم: $(1-\rho)^{n_f} \cong e^{-\rho n_f}$ اگر n_f بزرگ

احتمال خراب شدن فریم $P_f = C_0 \cdot \rho \cdot n_f \qquad 1/10 < C < 1/3$

یک بهره‌بردار از خطوط مالتی‌پلکس مربوطه ~ idle RQ بوده است.

Continuous R.Q :



$$\eta^*_{c\text{-}RQ} = \frac{w\left(1-\frac{n_0}{n_f}\right)}{1+\frac{n_a}{n_f}+\frac{(t_{proc}+t_{layer})R}{n_f}}$$

$$= w \cdot \eta^*_{idle_RQ} = \frac{w\,(n_f - n_0)}{R}$$

هرچه W را بیشتر کنیم بهره از خط بیشتر می‌شود و $W_{max} = \frac{t_0}{t_x}$

$\eta^{\circ}_{C-RQ} = W_{max} \cdot \eta^{\circ}_{idle\,RQ} = \frac{W_{max} \cdot \frac{n_f - n_0}{t_0}}{R} = \frac{\frac{1}{t_x} \cdot \frac{n_f - n_0}{t_0}}{R}$ (دوباره)

$= \frac{\frac{1}{n_f/R} \cdot \frac{n_f - n_0}{1}}{R} = \frac{n_f - n_0}{n_f} = 1 - \frac{n_0}{n_f}$

n_0 مقداریست که اضافی می‌فرستیم (مثل header, trailer, ...) و بستگی به طول n_f دارد.

η_{C-RQ} زیاد است $\Leftarrow$ n_0 کوچک باشد $\Leftarrow$ بازده زیاد می‌شود.

Go back N

Continuous RQ

Sel Retransmission

اکنون می‌خواهیم حالت خطا را نیز وارد بحث و بررسی قرار دهیم.

برای سادگی فرض کنید $W = W_{max} = \frac{t_0}{t_x}$

$\eta^{\circ} = \left(1 - \frac{n_0}{n_f}\right)$

Go Back N:	شماره ارسال	N	احتمال
	1	1	$1-P_f$
	2	$1+W$	$P_f(1-P_f)$
	3	$1+2W$	$P_f^2(1-P_f)$
	⋮		
	i	$1+(i-1)W$	$P_f^{(i-1)}(1-P_f)$

$$N_r = 1\times(1-P_f) + (1+w)P_f(1-P_f) + (1+2w)P_f^2(1-P_f) + \cdots + (1+(i-1)w)P_f^{i-1}(1-P_f) + \cdots$$

$$= (1-P_f) + P_f(1-P_f) + P_f^2(1-P_f) + \cdots + P_f^{i-1}(1-P_f) + \cdots + wP_f(1-P_f) + 2wP_f^2(1-P_f) + 3wP_f^3(1-P_f) + \cdots + (i-1)wP_f^{(i-1)}(1-P_f) + \cdots$$

$$= 1 + (1-P_f)w\underbrace{\sum_{i=1}^{\infty} iP_f^i}_{P_f/(1-P_f)^2} = \frac{1+(w-1)P_f}{(1-P_f)}$$

$$\eta_{\text{کانال}} = \frac{1}{N_r}\left(1-\frac{n_0}{n_f}\right) = \frac{(1-P_f)}{1+(w-1)P_f}\cdot\left(1-\frac{n_0}{n_f}\right)$$

تمرین: محاسبه $\eta_{C.R.Q\ \text{Go Back N}}$ وقتی $w < w_{max}$ است.

Sel Rete :

شماره ارسال	N	احتمال
1	1	$(1-P_f)$
2	2	$P_f(1-P_f)$
3	3	$P_f^2(1-P_f)$
⋮	⋮	⋮
i		

$$N_r = (1-P_f) + 2P_f(1-P_f) + 3P_f^2(1-P_f) + \cdots$$

$$= (1-P_f)\left[1+2P_f+3P_f^2+\cdots\right] = (1-P_f)\frac{1}{(1-P_f)^2} = \frac{1}{(1-P_f)}$$

$$\eta = \frac{\eta_{max}}{N_t} = \frac{(1-\frac{n_0}{n_f})}{\frac{1}{(1-P_f)}} = (1-P_f)(1-\frac{n_0}{n_f})$$

برای حالتی که $W < W_{max}$ صدق کند.

$$\begin{cases} \eta_{Go\ Back\ N} = \dfrac{(1-P_f)(1-\frac{n_0}{n_f})}{1+(W-1)P_f}, & W = W_{max} \\ \eta_{Sel-Rep} = (1-P_f)(1-\frac{n_0}{n_f}), & W = W_{max} \\ \eta_{Idle-RQ} \end{cases}$$

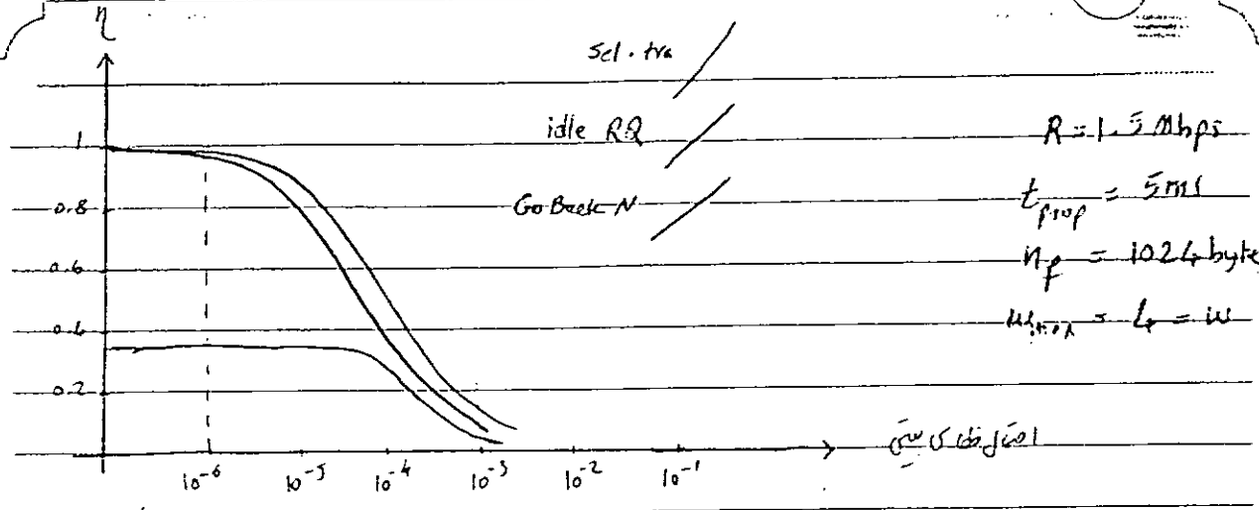
$W_S = 1$

$$\frac{t_0}{t_X} = W = \frac{t_X + 2t_{prop} + t_{X,ack} + t_{proc}}{t_X} \simeq \frac{t_X + 2t_{prop}}{t_X} = 1 + 2\underbrace{\frac{t_{prop}}{t_X}}_{a}$$

$$= 1 + 2a \quad ; \quad a = \frac{t_{prop}}{t_X}$$

W_{max}	$R = 30Kbps$ $t_X = n_f/R$	$R = 1.5Mbps$	$R = 45Mbps$
$d = 1500^{km}$, $t_{prop} = 5^{ms}$	2	2.2	57
$15000km$, $50ms$	2	20	551
$150000km$, $500ms$	6	185	4575

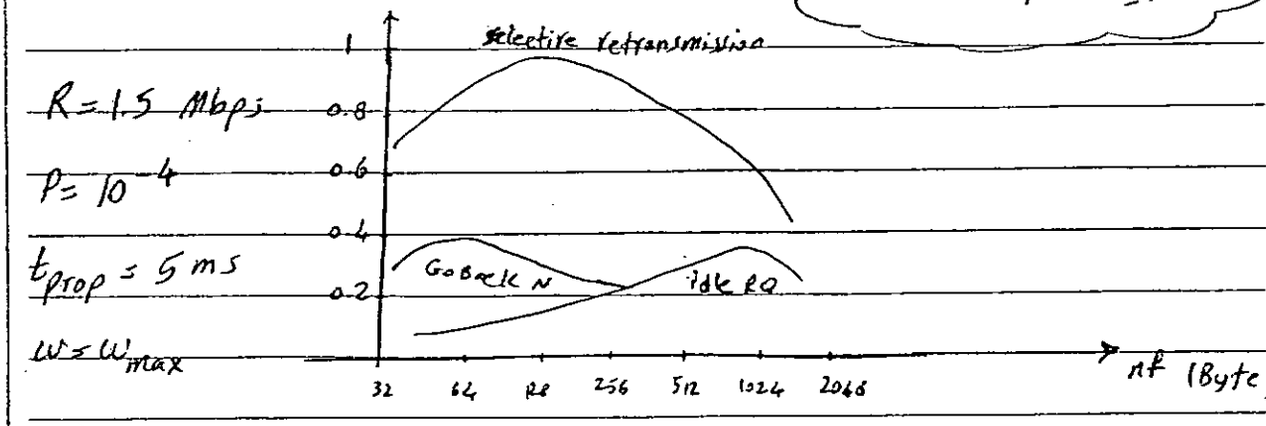
* $R \uparrow$ $t_i \downarrow$ $a \uparrow$ $W_{max} \uparrow$ *



* جایی که خطا خیلی زیاد می‌شود روش خوبی نداریم. (هیچ‌کدام خوب نیستند!)

* عملکرد Sel Tran مشابه Go Back N است چون بازارسال فقط فریم‌های خراب ارسال می‌شود. ولی هزینه بیشتر مثل buffering و Timering و ... .

جلسه بیست و یکم: ۱۴۰۲/۰۲/۱۷



* برای هر P یک n_f بهترین (Optimum) داریم.

* در تعداد بیت کم، تعداد بیت زیاد عمل مقایسه را انجام داده‌ایم.

کنترل خطا:

framing:

در انتقال آسنکرون مثلاً Frame Check Sum ، 2 Flag و ... را لایه Data link بر می‌کند.

کنترل جریان: اگر گیرنده ظرفیت محدودی برای دریافت داده داشته باشد، فرستنده نمی‌تواند به طور نامحدود و با هر سرعتی برای آن، داده بفرستد.

مثلاً در انتقال Go Back N اندازه پنجره‌ای داشته‌ایم وقتی که به اندازه آن داده فرستادیم صبر می‌کنیم تا گیرنده Ack بدهد لذا خود پنجره، مکانیزمی برای کنترل جریان است

۱- مکانیزم پنجره

۲- X-ON/X-off فرستادن داده X-ON:

فرستادن داده X-off:

* هر موقع گیرنده نتواند اطلاعات را دریافت کند، X-off را می‌فرستد.

** اگر گیرنده هنگامی که عنوان شد، بخواهد X-off را بفرستد ممکن است در طی فرستادن گیرنده چند frame را بفرستد. لذا بهتر است که قبل از پر شدن بافر، Xoff را بفرستد.

موضوع: ..... تاریخ: .....

چون زمانی بایت Xoff را ارسال شده $2T_p$ زمان قبل از پر شدن بافر

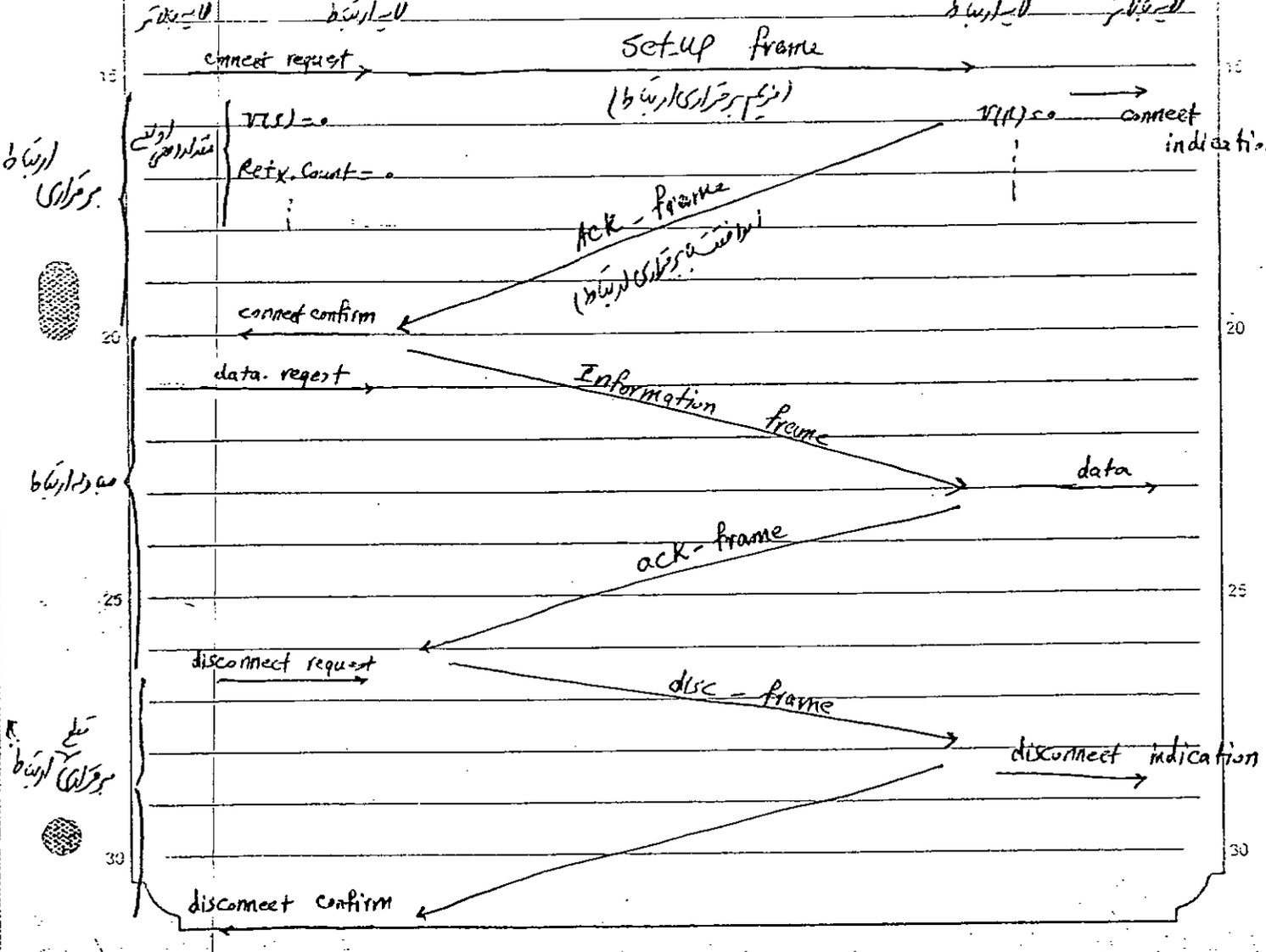
لذا:

تعداد بافرهای خالی که مانده به بیتی که Xoff را ارسال شده $= \frac{2T_p}{T_x}$ (؟)

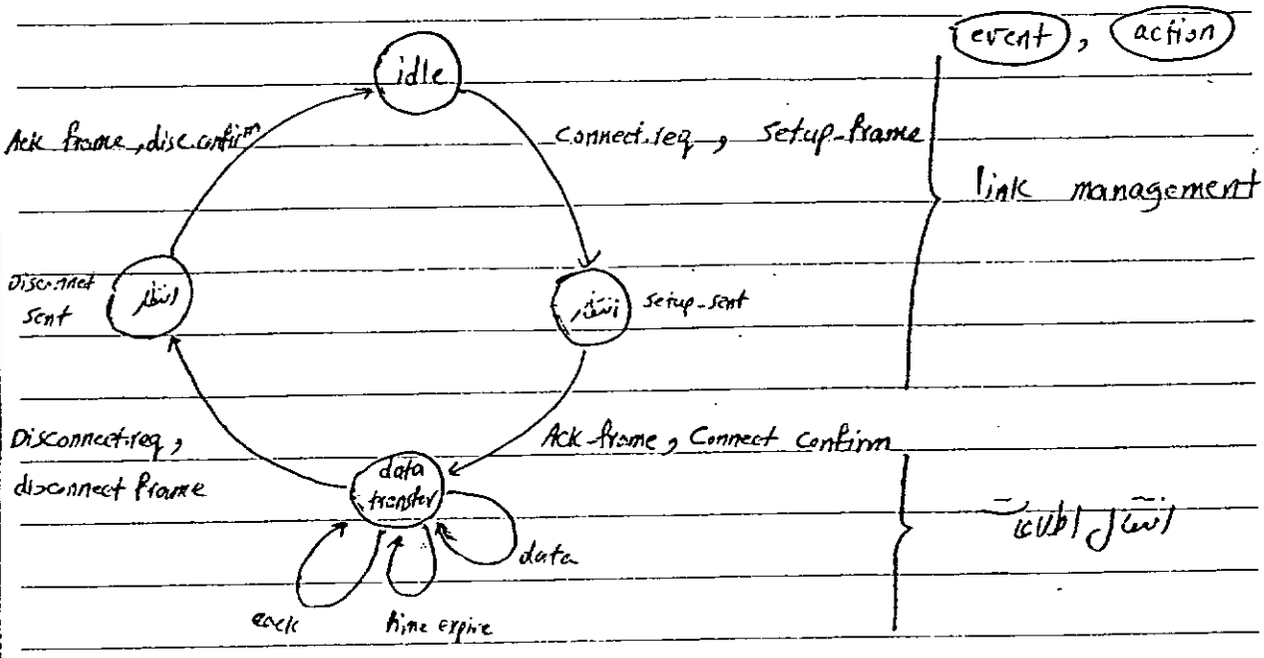
: Link Management

Network
DataLink
physical

Network
DataLink
physical

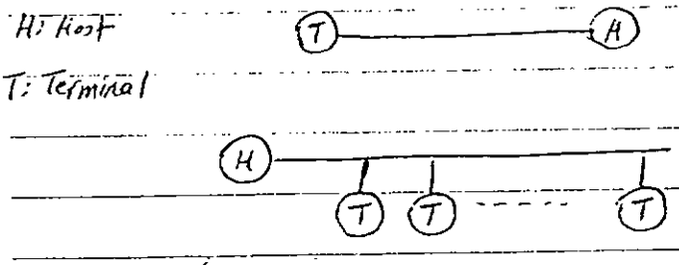


* برای برقراری ارتباط و قطع ارتباط را جزء وظایف لایه Data link بحث می‌کنیم.



- توپولوژی ..... Topology
- دوگانی ..... Duplexity
- انضباط خط : Line Dicipline

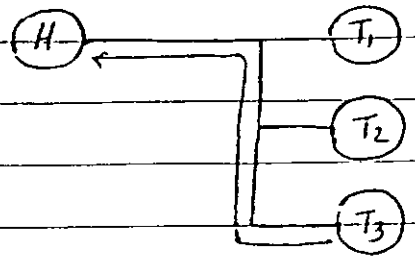
جهت طرح در link management



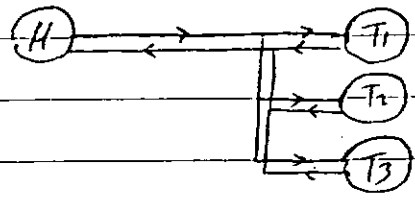
point-to-point
multi-point
topology

Duplexing

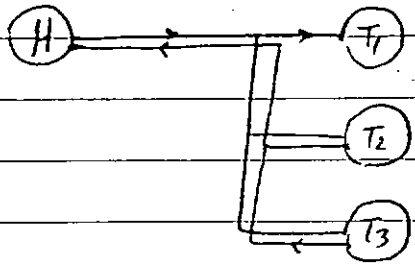
- Simplex ← یک طرف فقط (همیشه) می‌فرستد و طرف دیگر فقط می‌گیرد.
- half duplex ← یک طرف می‌فرستد، و طرف دیگر می‌گیرد. (فرستنده و گیرنده جایشان را باهم عوض می‌کنند)
- full duplex ← توان واحد، هر دو طرف می‌فرستند.



multi point , half duplex –



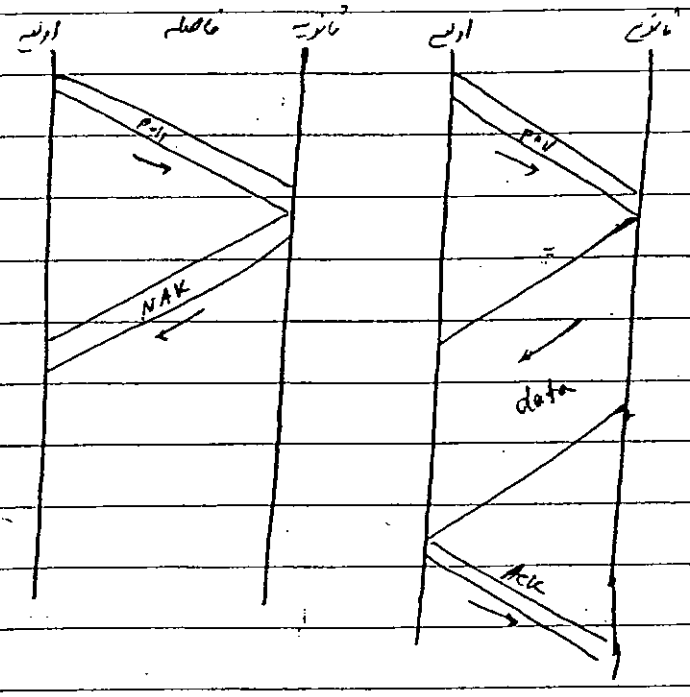
multi point , full duplex –



multi - multi duplex

full ← H

Half ← T



اضافه خط :

* مکانیزم Poll (مراجعه و اخذ داده)

① Primary نظام دهنده

② Secondary نظام پذیرنده

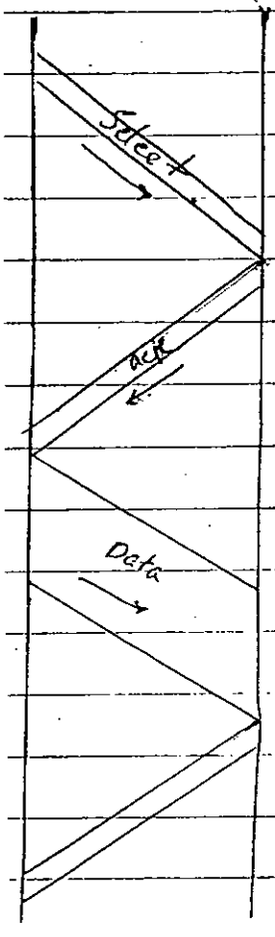
$T_1 = t_{x,poll} + t_{prop,poll} + t_{proc} + t_{x,nak} + t_{prop,nak} + t_{proc}$

$T_2 = t_{x,poll} + t_{prop,poll} + t_{proc} + t_{x,data} + t_{prop,data} + t_{proc} + t_{x,ack}$

$+ t_{prop,ack} + t_{proc} \Rightarrow T_2 = T_1 + t_{x,data} + t_{prop,data} + t_{pr}$

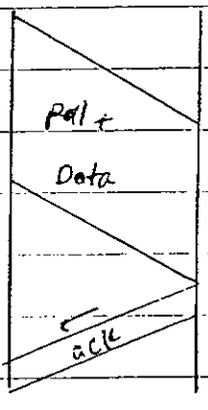
$t_{x,nak} \ll t_{x,data}$ ، $t_{prop,nack} = t_{prop,data}$

$t_{proc} \simeq 0$ $\quad T_2' = T_1 + t_{x,data}$



مکانیزم Select : اولین برای کم کردن ترافیک

Fast Select



$T_2 = T_{x,scl} + T_{x,data} + t_{prop,data} + t_{proc}$

$+ t_{x,ack} + t_{prop,data} + t_{proc}$

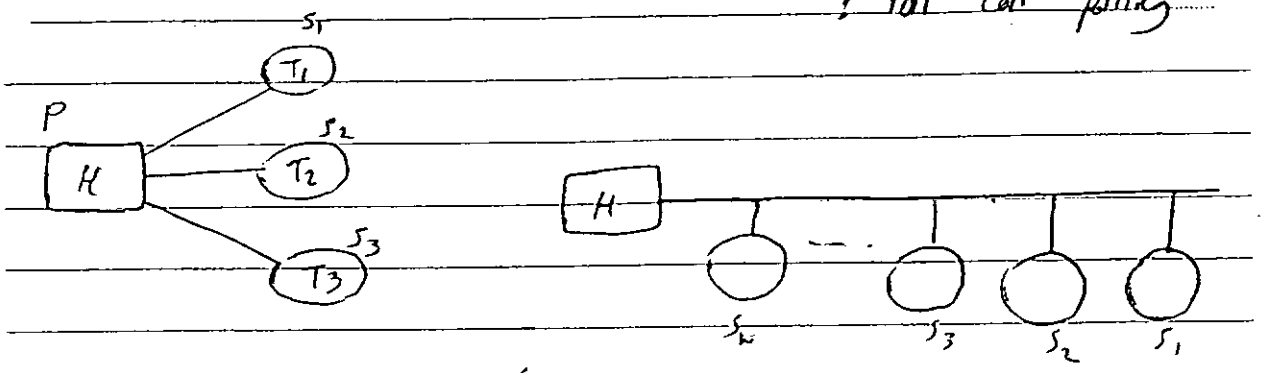
$T_2 \simeq T_1 + 2t_{prop}$

تا اسلاید بعدی: ۱۹، ۲، ۴۸

roll call polling - } poll

hub polling - }

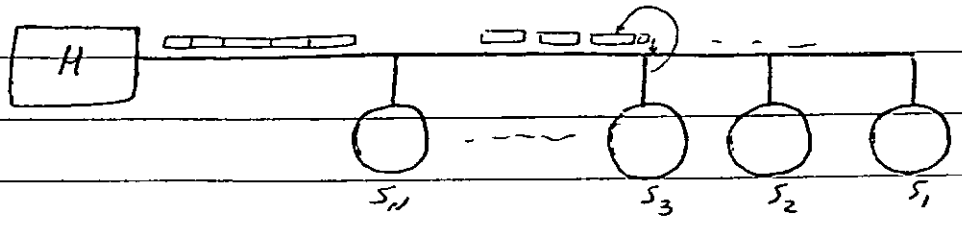
Roll call polling :



ابتدا primary ، secondary را شماره گذاری می کنند و به ترتیب به آنها سرویس دهی می کند .

hub polling : * به توپولوژی ستاره ای نیز hub می گویند .

ابتدا یک بسته Poll برای اولین ثانویه فرستاده می شود . اگر اطلاعات را نیز مستند همراه آن یک بسته Poll به آدرس ثانویه قبلی می فرستد بسته قبلی اگر اطلاعاتی داشت همراه آن اطلاعات قبلی فرستاده همراه با یک بسته Poll برای ثانویه قبلی و همین طور الی آخر .



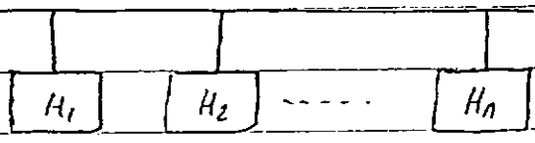
* دراین روش ترافیک زیادی را خرج می کنیم .

تمرین : یک multi point configuration داریم . n ثانویه داریم . secondary کدام ارتباط است . همه ثانویه ها داده اطلاعات هستند طول اطلاعاتی که می فرستند برابر است با L میزان بهره وری این روش را بدست آورید .

اطلاعات لایه 2 است

لذا لازم نیست Point-to-Point : آدرس دهی نیازی نداریم ، ولی در Multi point به آدرس دهی احتیاج داریم ! (آدرس ثانویه لازم است تعیین گردد ، ولی معمولاً در پروتکل ها آدرس ثانویه (مقصد) ترافیک داده می شود که آدرس ثانویه تعیین شود (در هر دو حالت Secondary)

* اگر به جای ترمینال ها ، کامپیوتر داشته باشیم لازم است تغییراتی اعمال کنیم . از جمله اینکه آدرس مبدا و مقصد باید تعیین شود .



پروتکل HDLC (High-Level Data Link Contol) :

- لایه 2 OSI کار می کند .
- سنکرون
- bit-oriented
- connection oriented
- مبادله اطلاعات به صورت full duplex و half duplex قابل انجام شدن است .

Station

Primary Station : Command

Secondary Station : Command

Combined Station

موضوع: ..... تاریخ: .....

دو نوع پیکربندی خط در نظر گرفته می‌شود:

- unbalanced Configuration: یک اولیه، یک یا بیشتر ثانویه، point to point یا multi point
- balanced: Combined، point-to-point

حالت‌های عملیاتی:

۱- Normal Response Mode: یک اولیه و چند ثانویه، تابع تحت فرمان اولیه
۲- Asynchronous Balanced Mode: balanced configuration
۳- Asynchronous Response Mode: پیکربندی unbalanced، ثانویه هم می‌تواند تا حدی کارهای اولیه را انجام دهد

در حالت ۲: هر کدام از Node ها Combined هستند، عنوان اولیه عمل کرده و دیگری به عنوان ثانویه.

* حالت 3 فوق به حالت معمولی و extended تقسیم می‌شود.

* هر سه حالت فوق به حالت معمولی و extended تقسیم می‌شوند.

Flag	frame check sum	Information	Control	آدرس	Flag
۸ بیت	۱۶ یا ۳۲ بیت *		۸ یا ۱۶ بیت *	۸ یا ۱۶ بیت	۸ بیت

آدرس مربوط به ثانویه است چه مبدا باشد چه مقصد و یک طرف ارتباط اولیه است.

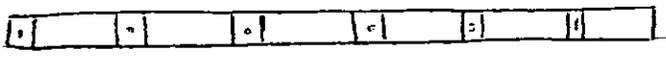
* ۱ عددی بیشتر مربوط به حالت extended است.

* کارای مربوط به bit stuffing را اینجا هم صدت می‌زند.

ن ← این صفرها حذف می‌شود. اگر ۱۱۱۱۱۱۱۰ داشته باشیم نشان دهنده این است که خطا قطع داشته است.

* اگر آدرس ۱۱۱۱۱۱۱۱ باشد یعنی حالت Broad Cast (برای همه شنونده‌ها می‌فرستد).

اگر بخواهیم آدرس را ثبت کنیم:

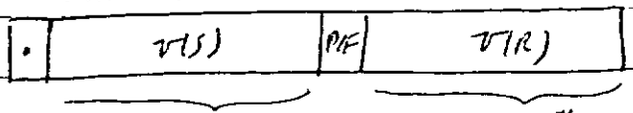


ابتدای هر آدرس وقتی صفر باشد یعنی یک آدرس دیگر هم داریم و ادامه دارد. وقتی ۱ باشد نشان دهنده آخرین آدرس است.

انواع فریم ها:
- information frame
- supervising frame ← فقط برای information (دارد) ← مثل Aek (ارزش)
- unnumberred frame ← برای کنترل ارتباط ← ممکن است داده هم داشته باشد.

1)

Information frame



V(R) شمارش Counter R

V(S) شمارش Counter S

۳ بیت ۳ بیت

Primary → Secondary Poll 1/0

Secondary → Primary Final 1/0

P اگر ۱ باشد یعنی اولیه است و می‌خواهد که اطلاعاتی هم عقب‌دار اطلاعاتی که می‌گیرد بفرستد. در غیر اینصورت اطلاعاتی نیست.

F ← شنونده اولیه می‌گوید که این بسته آخرین بسته است یا خیر.

* اندازه پنجره در این حالت $2^3 - 1 = 7$ خواهد بود.

* در حالت extended اندازه پنجره $2^7 - 1 = 127$ خواهد بود.

2)

Supervisory Frame

1	0	S	P/F	N(R)
		۲ بیت		۳ بیت
		۶ بیت (extended)		۷ بیت (extended)

← با ۲ بیت کنترلی که در این فیلد وجود دارد فریم Supervisory را مشخص می‌کنیم.

3) Unnumberred frame

1	1	M	P/F	M
		۲ بیت		۳ بیت
		۶ بیت (extended)		۷ بیت (extended)

در حالت extended 2^{6+7} حالت مختلف خواهیم داشت.

جلسه بیست و چهارم: ۹۷/۲/۲۴

فیلد information:

طول این فیلد به پروتکل مورد نظر بستگی دارد.

فیلد frame check sum: در حالت عادی ۱۶ بیتی و در حالت extended، 32 بیت است.

* در این CRC استفاده می‌شود:

در حالت عادی $\rightarrow g(x) = x^{16} + x^{12} + x^5 + 1$

$g(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$ ← در حالت extended

فریم Supervisory :

تقسیم به ۲ بیت هست که ۴ تا نوع فریم داریم.

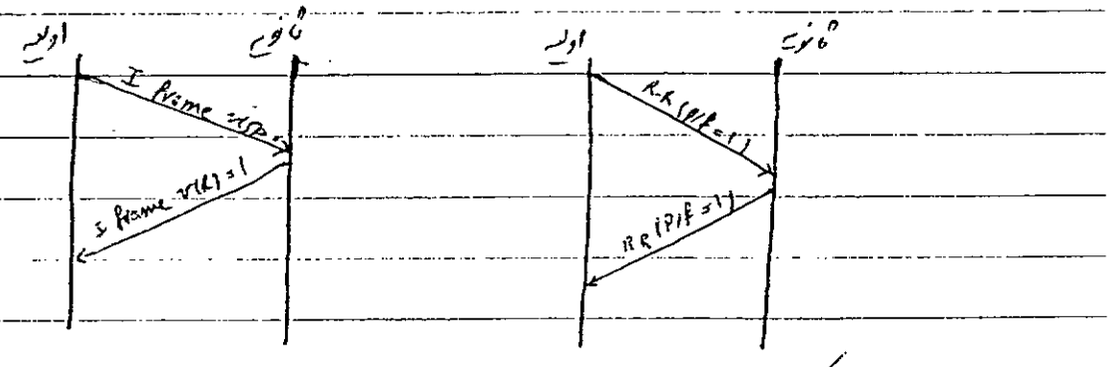
RR = (Recieved_Ready) (Xon . ack) اگر اطلاعات بصورت مجزا وجود نداشته باشد

RNR = (Recived_Not Ready) (Xoff . ack)

Negative ack = (Reject) REJ

که هنگامی که از پروتکل GoBackN استفاده می‌کنیم.

SREJ = (Slective Reject) Negative Ack که از روش SelectiveRetr استفاده می‌کنیم.



* از بین فریم های گفته شده فریم unnumbered تنوع بیشتری دارد

فریم unnumbered :

- فریم هایی که برای set کردن mode ارتباط :

① Set normal response mode (SNRM) Command

② " " " extended (SNRME) "

⑧ Set Async Response mode (SARM) command

⑨ " " " " etended (SARME) "

⑩ Set Async Balanced Mode (SABM) "

⑪ " " " " extended (SABME) "

این دو فرمان برای تنظیم مدار ارتباطی بودند.

⑫ Unnumbered Acknowledgment (UA) (Combined برای پاسخ مثبت) response

⑬ Set Initialization Mode (SIM) برای ستاندارد های پیشرفته تر command

⑭ Disconnected Mode (DM) response

⑮ Request Initialization Mode (RIM) (برای تقاضای ستاندارد های اولیه) response

⑯ Request Disconnect (RD) (برای قطع ارتباط) response

⑰ Disconnect (DISC) command

فرمان های برای ارسال اطلاعات کنترلی استفاده می شوند:

نوع unnumbered برای فریم اطلاعات است.

⑱ Unnumbered Information (UI) command

⑲ Unnumbered Poll (UP) (اولین بسته برای گرفتن اطلاعات از گیرنده) command

فرمان های Recovery

⑳ Frame Reject (FRMR) response

اگر فریم دریافتی خطا داشت

فیلد کنترل اشتباه شده

شماره فریم اشتباه شده است

فریم باید حذف اطلاعات داشته باشد و دوباره ارسال

⑯ reset (RSET) Command

کلیه شمارنده‌ها R و S را reset می‌کنند. پارامترها init می‌کنند.

فرمت‌های غیرتهی

⑰ exchange identification (XID) برای مشخص کردن پارامترها Command/response

اندازه پنجره - پروتکل مورد استفاده ...

⑱ test Command/response

جهت برای ایستگاه ارتباط برقرار است.

مثال برای ۲ کامپیوتر: Combined

ایستگاه Data Link ..... Data Link ایستگاه

connect req. → SABM → connect indication

UA ← connect confirm

I frame → data indication

RR

disconnect request → DISC → confirm disconnect

UA ← Disconnect Confirm

موضوع: ..... تاریخ: .....

(تحویل دوشنبه هفته آینده)

تمرین: کلیه ماشین‌های دیتاگرام (ارسال شده) را بررسی کنید. (مفاهیم Action، آدرس‌دهی و ... را توضیح دهید)

HDLC	Framing کنترل خطا همچنان مدیریت Link

PPP (Point-to-Point Protocol)

این پروتکل زیر مجموعه‌ی TCP/IP است.

مورد استفاده در بعضی از اتصال کامپیوترهای خانگی (معمولاً از طریق مودم) است.

مورد استفاده بعدی اتصال بین 2 تا Router است.

پروتکل لایه‌ی پایین‌تر IP است. برای آن عمل encapsulation را بررسی کنید.

پروتکل لایه‌ی بالاتر را مشخص می‌کند.

Flag	آدرس	Control		SCP	Flag	CRC	Flag
01111110	11111111	11.....					01111110

Broadcast — معمولاً 11..... — معرفی از 8 — 16 یا 32 بیت

این پروتکل بصورت Connectionless است.

طول اطلاعات مضربی از 8 است (Charater oriented).

برای عمل bit stuffing برای ترافیم انجام دهیم، برای جبران عمل از Flag، کاراکتر Escape می‌گذاریم.

- لایه‌های بالاتری می‌تواند: IP، IPX، LCP، ... باشد.

LCP: در PPP، فاز برقراری ارتباط نداریم. برای برقراری ارتباط از این پروتکل استفاده می‌کنیم.

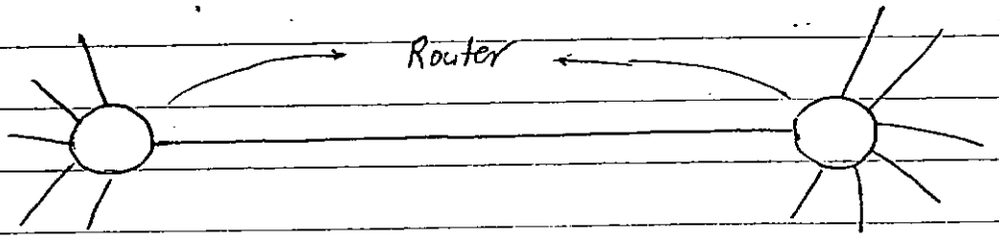
برقراری ارتباط - قطع ارتباط - مذاکره - تصدیق اصالت (authentication)

برای بررسی تصدیق اصالت از پروتکل PAP (Password Authentication Protocol) استفاده می‌شود.

پروتکل دیگری که LCP برای تصدیق اصالت استفاده می‌کند chap است.
(Challange-Handshake Authentication Protocol)

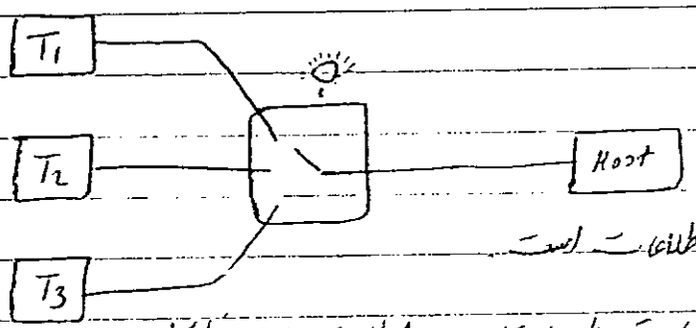
تصدیق اصالت ← برای مطمئن شدن از اینکه طرف مقابل، خودش است یا الله اکبر!!

NCP: طراحی شده است برای set کردن پارامترهای پروتکل‌های لایه‌های بالاتر.



پروتکل PPP قابلیت اتصال Multi Link را دارد.
(Router اطلاعات چندین Link را با هم ترکیب می‌کند و روی یک خط می‌فرستد.)

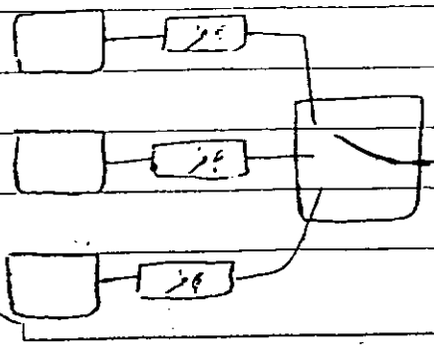
تاریخچه‌ای از Multilink:



* روشن بودن لامپ ← به منزله انتقال اطلاعات است.

هر ترمینال که می‌خواهد اطلاعات را بفرستد ← طور دستی باید switch را روی خط قرار دهد.
خود.

- این روش خیلی ابتدایی بود لذا تغییر یافت. روش دیگر ارائه شد.



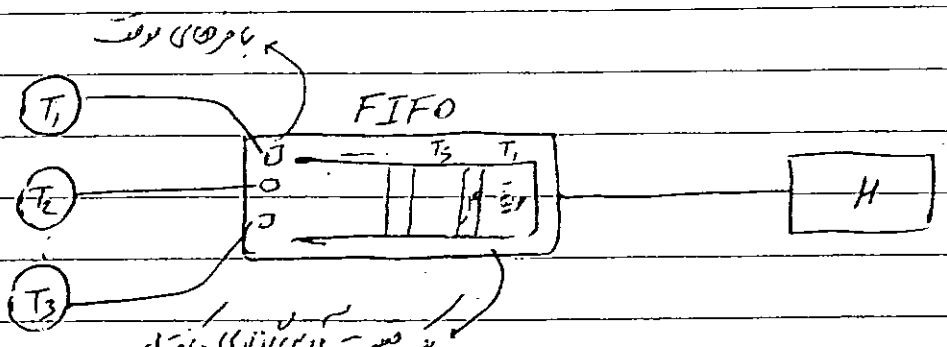
* صفحه جعبه اطلاعاتی را از ترمینال دریافت می‌کند (از نظر ترمینال اطلاعات فرستاده شده است.)
* جعبه به صورت هوشمند عمل می‌کند.
* پروتکل | جعبه باید upgrade شود.
جعبه‌ها باید با یکدیگر ارتباط برقرار

موضوع: ..... تاریخ: .....

جلسه بیست و پنجم: ۲۶/۲/۹۸

تحویل: ۲/۳/۹۸

تمرینهای فصل ۵: 5، 8، 15، 18، 21، 23، 27، 33،

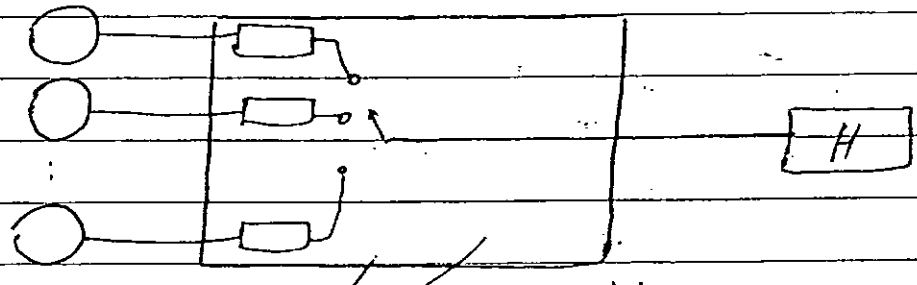


یک قسمت آدرس گذاری کنترل هم وجود دارد.

اطلاعات در اینجا نیز به چند دارند تا بدرا اطلاعات مشخص شوند.

این هدر به آدرس را مشخص می‌کند. توسط آدرس گذار گذاشته می‌شود.

معایب: حجم اطلاعات مشخص نیست (محدود نیست) و ممکن است ترتیبی کل بافر را اشغال کند.

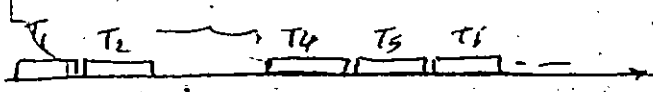


* اگر بافری اطلاعات نداشت، سوئیچ همان اندازه روی آن توقف می‌کند و روی بقیه.

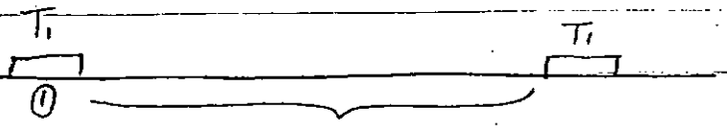
- این روش ترتیبی مانند مالتی پلکس زمانی TDM است.

- این روش Synchronous Multiplexing می‌گویند.

- سوئیچ مالتی پلکسر مشترون گویند.



حالتی را در نظر بگیرید که فقط T_1 اولویت دارد:



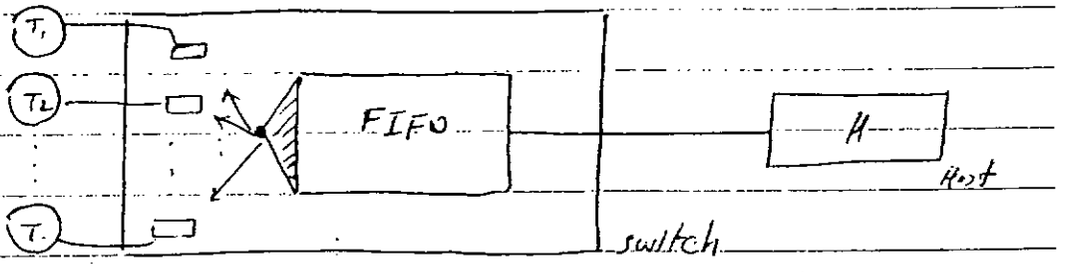
معایب: ۱- از خطا استفاده خوبی نشده است.

۲- ترمینال T_1 تأخیر زیادی را تحمل می‌کند.

برای حل کردن این مشکل از ایده‌ی حالتی پلکسینگ استفاده شده است.

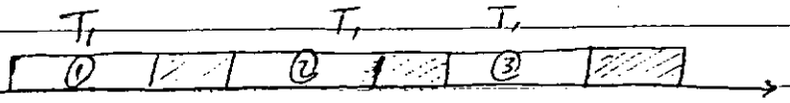
حالتی پلکسینگ استرون:

به شکل توجه کنید با این تفاوت که اگر ترمینال اولویت نداشته باشد از خط استفاده نمی‌شود و محدودیت حجم محدود شده است زیرا سوئیچ روی هاب فرمهای موقت ثبت می‌کند.



اگر حالتی همه ترمینال‌ها اطلاعاتی داشته باشند، روش پلکسینگ بهتر است زیرا هدر ندارد.

اگر حالتی فقط T_1 اطلاعات داشته باشد این روش بهره‌ور بسیار بالاتری از روش پلکسینگ دارد.



- هرچه طول قطعات کمتر باشد بهره‌وری کمتر است.

موضوع : ..... تاریخ : .....

بهره وری خط ← طول تقطعات یا فریم ها :

$L_1 > L_2 \Rightarrow \eta_1 > \eta_2$

نتیجه : اگر طول فریم ها زیاد شود بهره وری زیاد می شود (اما یک مشکل داریم)

اگر مقدار زیادی پیام کوچک داشته باشد برای ارسال آن باید یک فریم بزرگ بفرستد

و این کار باعث کم شدن بهره وری می شود.

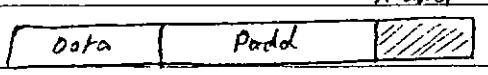
- با مطالعه ترافیک شبکه باید طول بهینه ای برای فریم ها انتخاب شود.

مالتی پلکسینگ آماری : Statistical Multiplexing

در این طرح طول فریم ها بین دو مقدار کمینه و بیشینه در نظر گرفته می شود.

الف - اگر فریمی با طول بین Min و Max بود، برایش هیچ کاری نمی کنیم و ارسال می کنیم.

ب - اگر طول آن از Min کمتر باشد :



ج - اگر طول آن از Max بیشتر بود، اطلاعات را قطعه قطعه می کنیم.

* اندازه : از معیارهای آماری استفاده می کنیم :

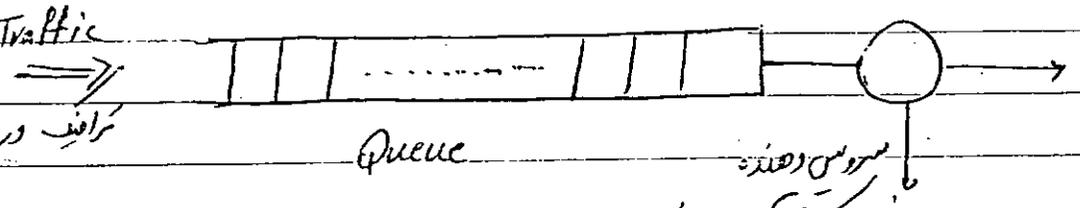
" اندازه یا فرم

- اگر کم باشد ممکن است پر شود

- اگر زیاد باشد تا فریم ها قبلی زیاد می شوند !!

تحلیل اندازه با فرم:

مدل: مدل زیر را جهت تحلیل اندازه مناسب میان گیر بکار می بریم.



* نظم صف: هر کس اول آمد، اول می رود.
* ترافیک ورودی: نرخ متوسط ورودی
* تعداد سرویس دهنده: در اینجا یک سرویس دهنده داریم.
* زمان سرویس: زمانی که سرویس دادن (خروج) طول می کشد (متوسط)
* طول صف: ممکن است محدودیت داشته باشد.
* همه عوامل گفته شده در تحلیل تأثیر دارند.

فرضیات مدل سازی:

۱- ترافیک ورودی از توزیع آماری پواسن تبعیت می کند (متغیر متوسط λ)

$$P_n(t) = \frac{(\lambda t)^n}{n!} e^{-\lambda t}$$

۲- نظم خط = FIFO

۳- تعداد سرویس دهنده‌ها = ۱

۴- زمان فرایت متناسب با قانون فرایم که از توزیع پواسن تبعیت می‌کند:

$$P_n(S) = (\mu S)^n / n! \cdot e^{-\mu S}$$

* به این مدل، مدل M/M/1 می‌گویند.

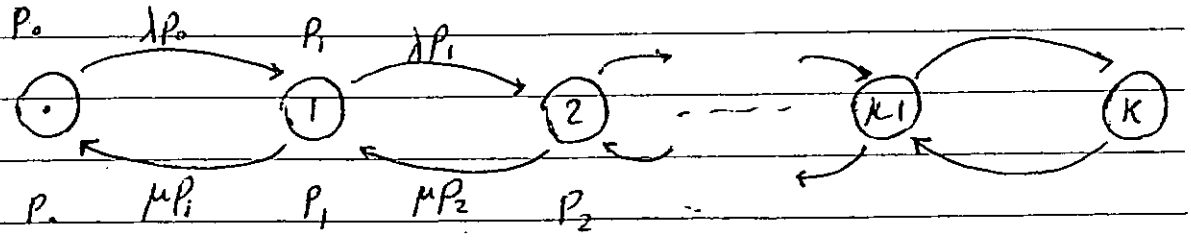
حالت‌های مختلف صف

۱- حالت ۰: هیچ پیامی در صف نیست.

۲- حالت ۱: ۱ پیام در صف است.

⋮

حالت n: n پیام در صف است.



وضعیت تعادل سیستم: وضعیتی که احتمال گذرهای دو حالت مجاور با یکدیگر یکسان است.

$$\lambda P_0 = P_1 \mu$$

$$\lambda P_1 = P_2 \mu$$

$$\lambda P_2 = P_3 \mu$$

$$\frac{P_n}{P_{n+1}} = \frac{\lambda}{\mu}$$

⋮

از این به بعد، تمام محاسبات را در وضعیت تعادل در نظر می‌گیریم.

* برابست $\frac{1}{\mu}$ میزان ترافیک گیرند

نرخ ترافیک $\rho = \lambda/\mu$

نتیجه:

$P_1 = P_0 \times \lambda/\mu = P_0 \rho$

$P_2 = \rho P_1 = \rho^2 P_0$ $\Rightarrow$ $P_i = \rho^i P_0$

$P_3 = \rho P_2 = \rho^2 P_1 = \rho^3 P_0$

تحلیل:

$\sum_{k=0}^{\infty} P_k = 1 \Rightarrow \sum_{k=0}^{\infty} \rho^k P_0 = 1 \Rightarrow P_0 \sum_{k=0}^{\infty} \rho^k = 1$

نکته: اگر $\rho > 1$ باشد این معادله جواب ندارد! لذا $P_0 = 0$

نتیجه: برای اینکه صف‌ها واقعی باشند ρ را کمتر از یک در نظر می‌گیریم.

$P_0 = \frac{1}{\sum_{k=0}^{\infty} \rho^k} \Rightarrow P_0 = \frac{1}{\frac{1}{1-\rho}} \Rightarrow$

احتمال خالی بودن بافر $P_0 = 1 - \rho$

نتیجه نهایی:

نرخ ترافیک $\rho = \lambda/\mu$

احتمال خالی بودن $P_0 = 1 - \rho$ (۱)

احتمال k تایی $P_k = (1-\rho)\rho^k$

موضوع: ..... تاریخ: .....

نتیجه: احتمال صف با بیشتر شدن مقدار پیام از کم به سمت زیاد مقدار بیشتر می‌شود (به شرط $\rho < 1$)

محاسبهٔ طول متوسط صف:

$$N = \sum_{i=0}^{\infty} i \cdot P_i$$ مقدار متوسط تعداد صف

$$N = \sum_{i=0}^{\infty} i(1-\rho)\rho^i = (1-\rho)\sum_{i=0}^{\infty} i\rho^i$$

$$N = (1-\rho)\cdot \frac{\rho}{(1-\rho)^2} \Rightarrow$$

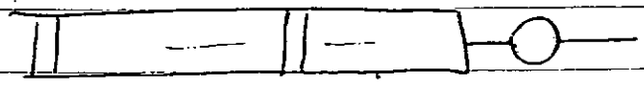
طول متوسط صف $$N = \frac{\rho}{1-\rho}$$

صف خلوت $\rho \to 0 \quad N \to 0$

صف شلوغ $\rho \to 1 \quad N \to \infty$

جلسه بیست و ششم: ۸۴/۲/۲۸

اکنون می خواهیم پارامتر T (تأخیر) را مورد بررسی قرار دهیم.



فرض می کنیم بسته ای وارد شدن موقع سرویس شدن را

زمان تأخیر T می نامیم. بنابراین زمان $\lambda.T$ بسته وارد خواهد شد.

N: تعداد متوسط بسته هایی که در سیستم

$$N_0 = \lambda.T \Rightarrow T = \frac{N}{\lambda} = \frac{\rho/1-\rho}{\lambda} \Rightarrow \frac{\rho}{1-\rho} \times \frac{1}{\lambda}$$

$$= \frac{\lambda/\mu}{1-\lambda/\mu} \times 1/\lambda = \frac{1}{\mu-\lambda}$$

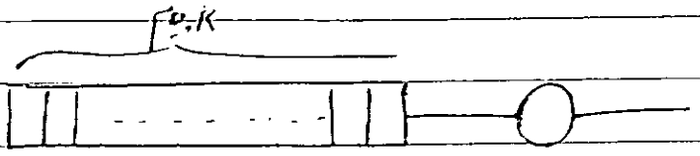
اگر $\mu > \lambda$ ... $T > 0$ ← بسته ها با تأخیر خارج می شوند.

اگر $\mu > \lambda$ ، $\mu \to \lambda$ ، $\mu - \lambda \to 0$ ⇐ $T \to \infty$ $(\rho = \frac{\lambda}{\mu} \to 1)$

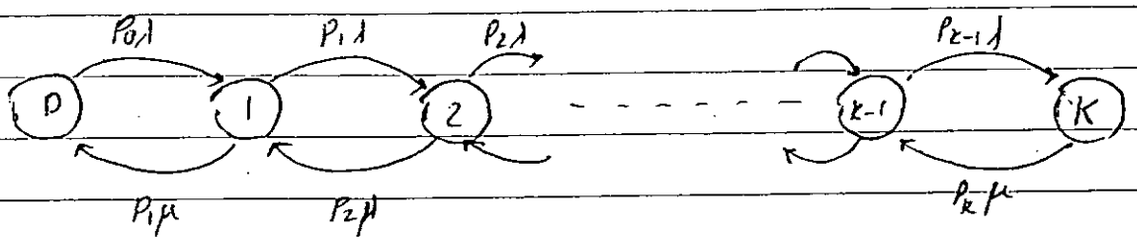
$\mu > \lambda$ امکان پذیر نیست. (T متوسط زمانی می گیرد و به عکس صف دچار اختلال می شویم.)

* در عمل طول صف همیشه محدود است لذا اگر بررسی می کنیم عالقی را به K تعداد مکان حافظه داده شده.

در سیستم K نفر ظرفیت:



فرآیند تغییر جمعیت به صورت پیوسته و در یک واحد زمان در حالت می‌رود:



P_0 P_1 P_2 ..... P_{k-1} P_k

وضعیت تعادل:

$$P_0\lambda = P_1\mu \qquad P_1 = P_0 \times \lambda/\mu = P_0\rho$$

$$P_1\lambda = P_2\mu$$

$$\sum_{i=0}^{k} P_i = 1$$

$$P_2\lambda = P_3\mu$$

$$P_{k-1}\lambda = P_k\mu \qquad P_k = P_0\rho^k$$

$$(P_0 + P_0\rho + P_0\rho^2 + \cdots + P_0\rho^k) = 1$$

$$\Rightarrow P_0 =$$

میزان متوسط تراکنش از دست رفته $= \lambda \times P_k$

$P_k = P_{loss}$ احتمال اتلاف

$$P_{loss} = P_k = P_0 \times \rho^k = \frac{(1-\rho)\rho^k}{1-\rho^{k+1}}$$

حالتی که صف بی‌نهایت بود و $\rho < 1$ باشد:

$$E(T) = \frac{E[N]}{\lambda}$$

حالتی که صف K نفر و $\rho \neq 1$ باشد:

$$E(T) = \frac{E[N]}{\lambda[1 - P_{full}]}$$

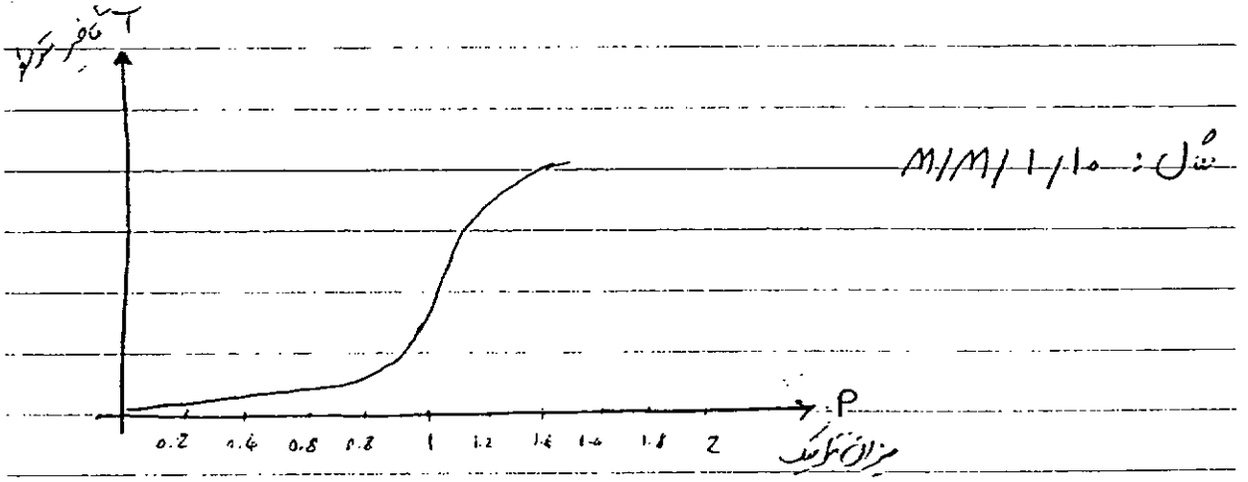
این سیستم $M/M/1/K$ نام دارد.

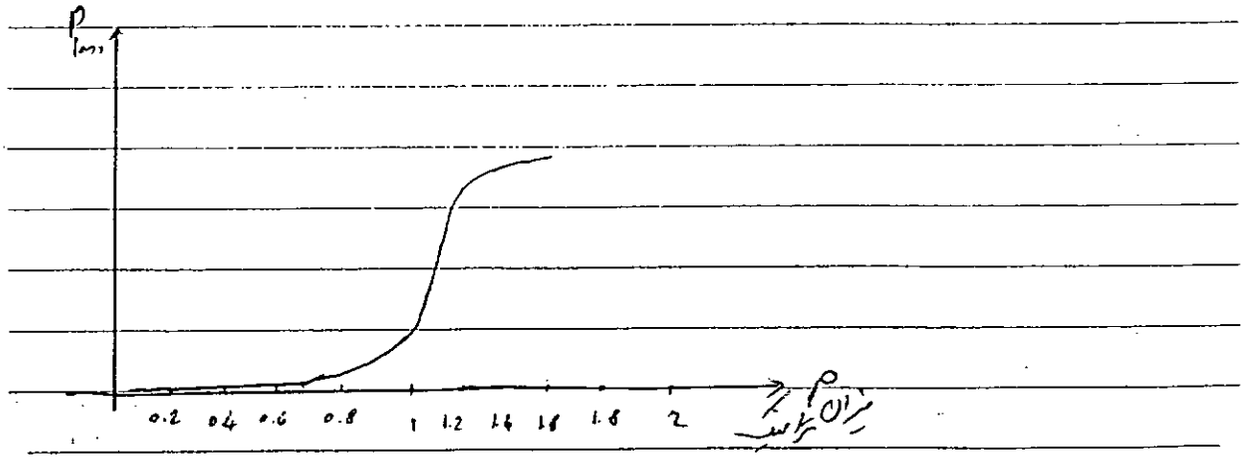
$$E(N) = \frac{\rho}{1-\rho} - \frac{(K+1)\rho^{K+1}}{1-\rho^{K+1}}$$

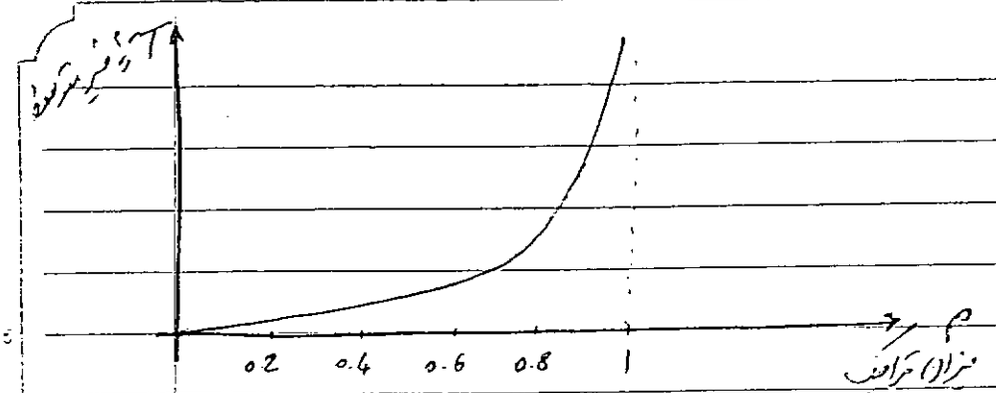
تمرین: برای K نفر $E(N)$ را اثبات کنید.

اولین ρ که $\rho = 1$ می‌شود چرا؟ چون از این به بعد دیگر نمی‌تواند سرویس دهی شود، از دست می‌رود به همین دلیل.

مثال: $M/M/1/10$



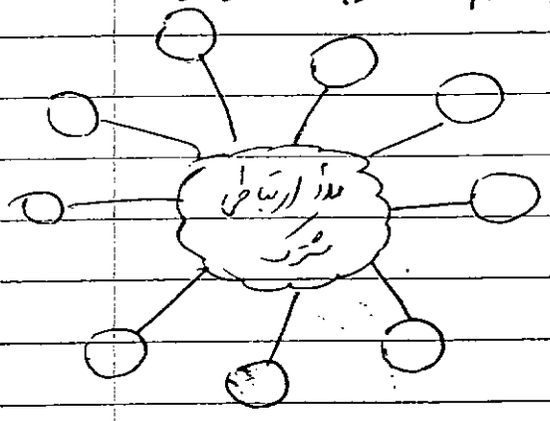




پایان فصل 5

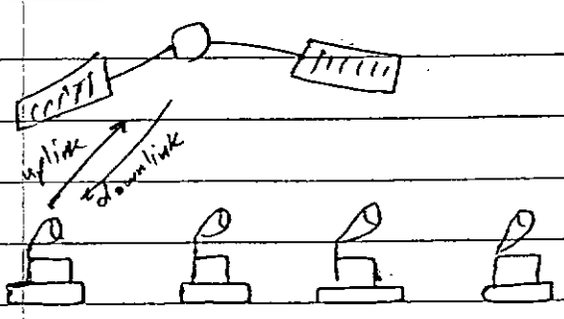
Connection medium

اگون با این موضوع حی پردازیم: چند کامپیوتر هم متصل داریم کدام‌ترین بسته را ارسال کنند.

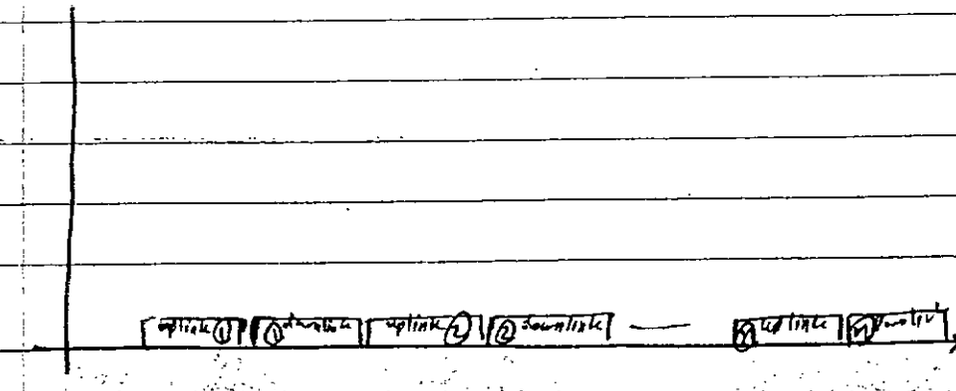


* در اینجا تکنولوژی broadcast پیاده می‌کنیم!!!

مثال: ارتباطات ماهواره‌ای



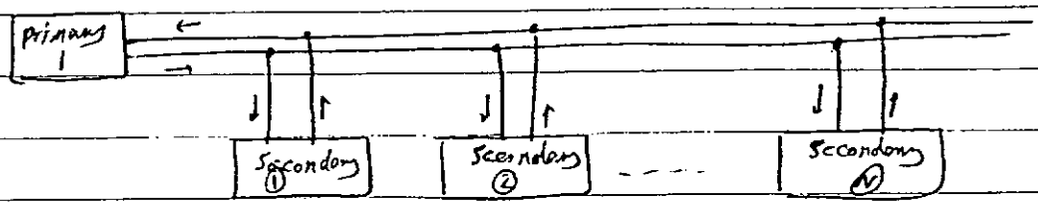
uplink ⟺ inbound
downlink ⟺ outbound



(ادامه دارد)

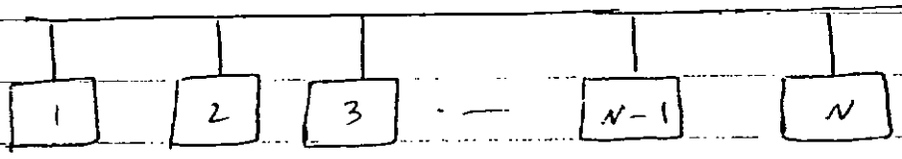
در اینجا برای ارتباط هر ایستگاه زمینی با ماهواره و بالعکس، پهنای باند خاصی را تخصیص می‌دهیم که این روش همان روش تقسیم فرکانس است. (FDM)

مثال ۲:



این لایه، مربوط به بحث فنی نیز بوده چون کامپیوترها هم شامل هستند. (از مکانیزم های مختلف polling برای نظم دهی استفاده می‌شود.)

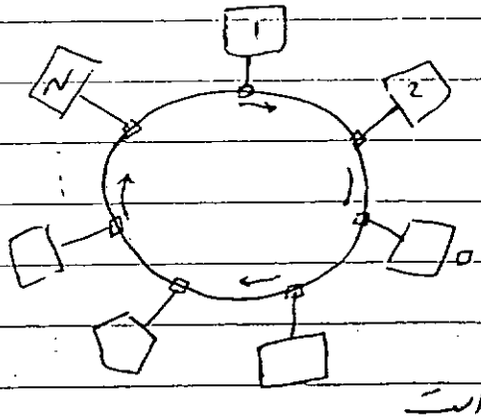
مثال ۳:



در اینجا برخلاف حالت قبل، نظم خط نداریم حین ارسال هر کامپیوتری می‌تواند به هر سمتی، اطلاعات بفرستد

* پروتکلی برای نظم دهی دسترسی به خطوط ارتباطی مشترک استفاده می‌شود پروتکل Medium Access است. ← همان ایده‌ی Broad Cast که ابتدا گفته شده!!!

مثال ۴:



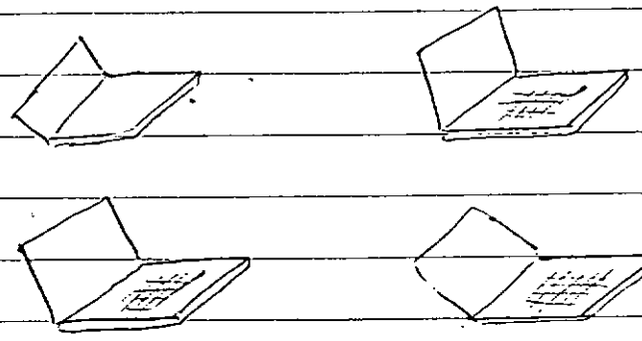
* سعی token دست هر کسی بود می‌تواند اطلاعات را ارسال کند.
* داده‌ها بسته‌ها در یک جهت حرکت می‌کنند.
* کلیه کامپیوترها در این هم می‌توانند پیام را به روی خط ارسال دریافت کنند. (نوعی BroadCasting)

موضوع: ..... تاریخ: .....

جلسه بیست و هفتم: (۲۱/۲/۸۴)

مثال 5:

ارتباط چند کامپیوتر - نظریه ارتباطی که می خواهند با هم تبادل اطلاعات داشته باشند.



* اگر از قبل کانال را جدا کرده باشیم مشکلی نداریم ولی اگر این کار را نکرده باشیم باید مدیریتی انجام دهیم. مثال چنین حالتی در شبکه های wireless LAN دیده می شود.

تکنیک های اشتراک ملا ارتباطی

(پیشنهادی خاصی برای مورد نظر گرفته می شود)

Channelization
کانال بندی

Dynamic (کنترل و رسیدگی به مدیریم مجدد)

scheduling
برنامه ریزی شده

Random Access
دسترسی تصادفی

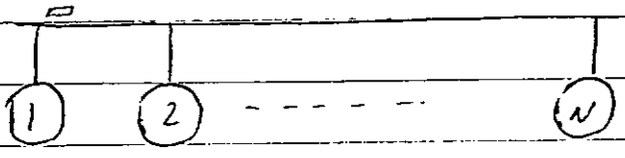
* در LAN از نوع Random Access استفاده می شود.

* لایه ای از عملیات لایه Data Link ، Medium Access استفاده می شود.

۱- کنترل خطا ۲- کنترل جریان ۳ framing

Medium Access -۵ Link management ٤

* لذا امروزه روش Random Access در LAN بیشتر بحث می‌شود، در اینجا این بحث را بررسی می‌کنیم!!



فرض اینکه فقط یک ثره روی خط اطلاعات می‌گذارد: t_x = مدت بودن فریم روی خط

$t_x + t_p + t_p = t_x + 2t_p$

$\eta = \frac{t_x}{t_x + 2t_p} = \frac{1}{1 + 2t_p/t_x}$

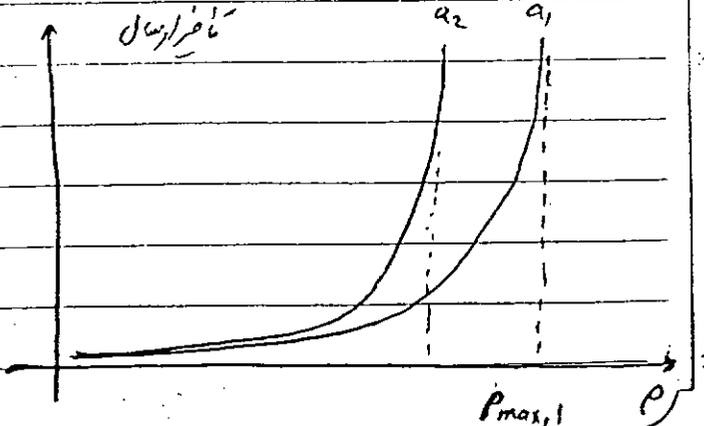
t_p = زمان لازم برای انتشار یک بیت روی خط $= \frac{d}{v}$

t_x = زمان لازم برای ارسال فریم روی خط $= \frac{n}{R}$

$a = \frac{t_p}{t_x}$

$a \uparrow \quad \frac{t_p}{t_x} \uparrow \quad \eta \downarrow$

$a \downarrow \quad \frac{t_p}{t_x} \downarrow \quad \eta \uparrow$



* هرچه ترافیک بیشتر می‌شود تأخیر ارسال بصورت نمایی زیاد می‌شود (چون با تداخل) پس از یک حدی فریم‌ها را دوباره نفرستیم!

وسی پروتکل a بیشتر است لذا $\frac{tp}{tx_2} > \frac{tp}{tx_1}$ لذا tx_2 بیشتر از tx_1 است.

این به معنی این است که طول خط بیشتر است و این باعث می شود که اگر تونل های بالای بخواهند اطلاعات را ارسال کنند فکر می کند که خط آزاد است در صورتی که چنین نیست و این باعث تداخل بیشتر و دیرتر خبر زدن می شود!

ویژگی های مطلوب پروتکل دسترسی به رسانه:

① efficiency ← از امکانات خط به درستی استفاده کند.

② fairness

③ Reliability

④ scalability

⑤ انواع ترافیک را به درستی تحمل کند.

⑥ کیفیت سرویس QoS

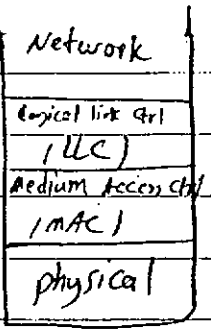
IEEE 802 : ← یک استاندارد برای کنترل دسترسی به خط

کارت شبکه Network Interface Card (NIC)

۱- تبدیل داده به سیگنال

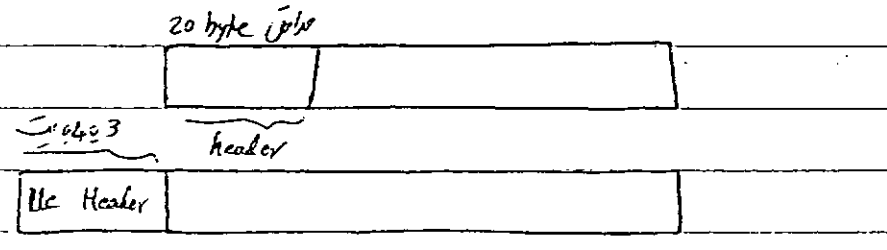
۲- عملیات لایه فیزیکی

۳- پروتکل MAC (Medium Access Control)



data link

MAC
- IEEE 802.3 → Ethernet
- IEEE 802.5 → Ring
- IEEE 802.11 → wireless LAN



① سرویس LLC type 1 :

(unacknowledged connectionless service)

مثال: پروتکل زیرساخت HDLC در unnumbered

② سرویس LLC type 2 :

(Reliable connection oriented service)

- ایجاد connection
- Ack
- مثال: HDLC در ABM

- سرویس لایه ۴ و ۳ و ۲ (مثل transport) سرویس قبل اضافه و تبدیل این سرویس متفاوت است

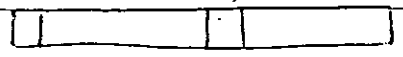
موضوع: ..... تاریخ: .....

④ سرویس LLC type 3 :

acknowledged connectionless service

مثـ: بعضی کنترلرها در HDLC توسط خطای بودن شمارنده فریم

(۴) بیت در LLC header این بیت مشخص میکند: command / response

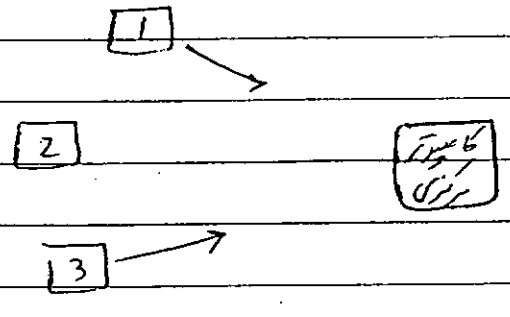


۸ بیت ۸ بیت ۱ یا ۲ بیت

عملیات Sequencing در این سرویس توسط خارج از این پروتکل انجام میشود.

MAC :

پروتکل ALOHA : مربوط به سال 1970 در دانشگاه هاوایی در آن کار شد.



۱- ارسال پیام

۲- منتظر Ack

۳- ـــــــ

خطای ناشی از فرستنده

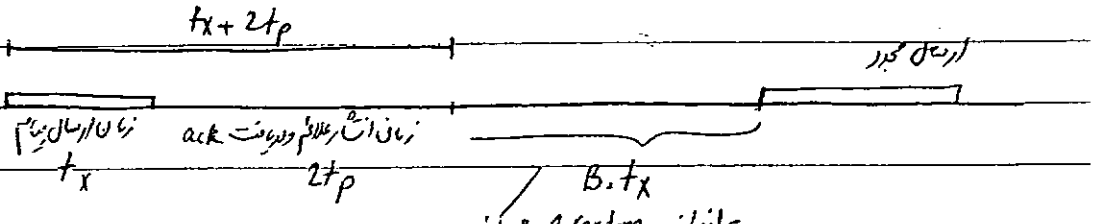
خطای ناشی از دریافت علائم

اصلاح می‌کنیم

① ارسال پیام ← محض آزاد شدن آن

② بعد از منقضی شدن تایمر توقع دریافت ack

- دریافت ack ← مبادله انجام شده
- عدم دریافت ack ← اندازه random صبر می‌کند و سپس مجدد ارسال می‌کند



اندازه random صبر می‌کند.
Back off Time

* دراین پروتکل، هنگام ارسال پیام، گوش نمی‌دهند.

* اکنون فرض می‌کنیم به طور متوسط G تا پیام می‌فرستیم و به طور متوسط S تا پیام به مقصد می‌رسند.

$G \geq S$

توزیع پوآسن را در نظر می‌گیریم:

$Prob(\text{در } 2t_x \text{ ارسال داشته باشیم } K) = \frac{(2G)^K}{K!} e^{-2G}$

* فرض می‌کنیم پیامی داریم که در بازه $t_0 + t_x$ تا t_0 زمان می‌برد. اگر هر پیامی در این محدوده زمانی با پیام ما تداخل داشته باشد پیام ما خراب می‌شود. لذا در محدوده زمانی $t_0 - t_x$ تا $t_0 + t_x$ فرض نباید شروع به ارسال شدن کند!

$2t_x$

تا الله اکبر

$S = G \times P$

$= G \times Prob($ عدم تصادم $)$

اگر دوبیت $2t_x$ فاصله شده پیام دیگر ارسال گردد، تصادم حاصل می‌شود.

$\Rightarrow S = G \times Prob($ ارسال پیام دوبیت $2t_x) = G \times \left(\frac{(2G)^0 \times e^{-2G}}{0!}\right)$

$= G e^{-2G}$

$\Rightarrow \frac{S}{G} = e^{-2G} \quad , p$

جلسه بیست و هشتم: ۸۴/۳/۲

$\frac{G}{S} = e^{2G}$ ← تعداد متوسط دفعات ارسال

ارسال مجدد

t_0 — $t_0 + t_x$ — $2t_p$ — $B \cdot t_x$

B به صورت آزاد

$T = (t_x + 2t_p + B \cdot t_x)(e^{2G} - 1) + t_x + 2t_p$

زمان متوسط سپری شده برای ارسال یک فریم

فرضهایی که کرده بودیم ۱- t_x یکسان

۲- فاصله تصادفی یک سان فرض شد

۳- توزیع پواسن را برای مدل کردن استفاده کردیم.

* برای اینکه B چه بدهیم، احتیاج به کمی تأمل داریم.

* اگر تداخلی ایجاد نشود هرکدام به تعداد خاصی تأخیر می‌کنند $B \in \{1, 2, \dots, K\}$
و این مقدار به صورت یکنواخت است:

$\bar{B} = \frac{K+1}{2}$ مقدار متوسط B $\Rightarrow$ مرتبه قبل

$T_1 = (t_x + 2t_p + \frac{K+1}{2} \cdot t_x)(e^{2G} - 1) + t_x + 2t_p$

$T_2 = \dots\dots\dots\dots\dots\dots + t_p$

زمان متوسط سپری شده برای آنکه فریم به مقصد برسد.

Slotted Aloha :

در این روش اندازه‌های خاصی، هرکدام که اطلاعات داشته باشند می‌فرستند (نه هرزمان دلخواه!)

- زمان خود را به اندازه‌های خاصی تقسیم می‌کنیم و در هر اسلات اگر اطلاعاتی وجود داشته باشد می‌گوید که آن را بفرستد.

این تغییر باعث می‌شود که تداخل به صورتی اتفاق می‌افتد که پیام‌های بیشتر از ۱ در اسلات خاص آماده شده‌اند و در آن لحظه بخواهند اطلاعات را بفرستند.

$S = G \times P = G \times \text{Prob}\{\text{عدم collission}\}$

$= G \times \text{Prob}\{\text{در مدت } t_x \text{ فریم‌های دیگری آماده ارسال نشده باشد}\} = G \times e^{-G}$

$\text{Prob}\{t_x \text{ ارسال در مدت } k\} = \frac{G^k}{k!} e^{-G}$ ، $k = 0, 1, 2, \dots$

$\text{Prob}\{t_x \text{ ارسال در مدت } 0\} = e^{-G}$

* ادامه در صفحه بعد! *

$$\frac{S}{G} = e^{-G} \qquad \frac{G}{S} = e^{G}$$

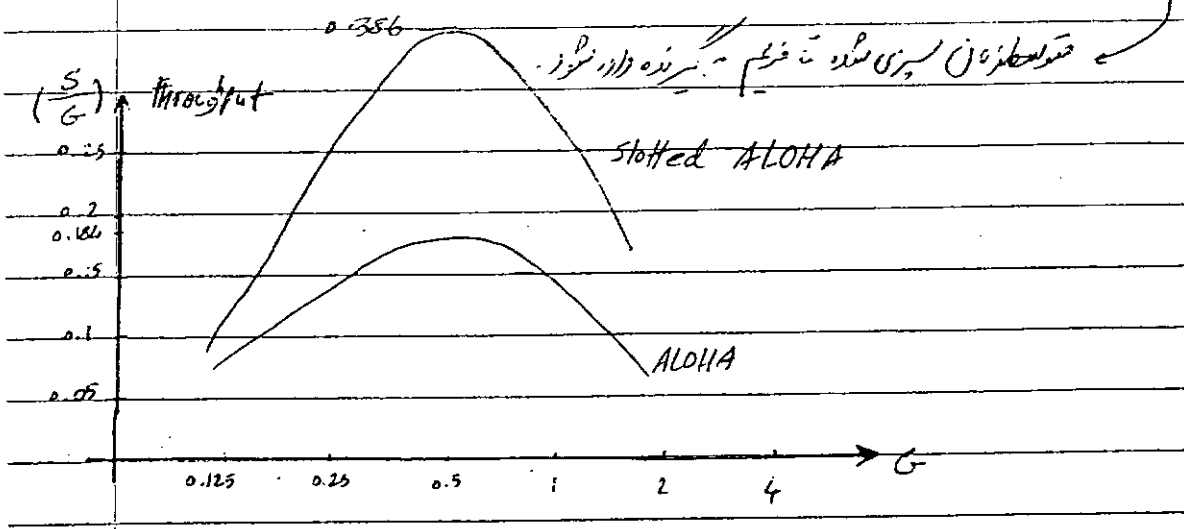
$2t_p$ $\quad$ $B.t_x$

$T =$ متوسط زمان سپری شده برای ارسال یک فریم

$$T_1 = (t_x + 2t_p + B.t_x)(e^{G} - 1) + t_x + 2t_p$$

$$T_2 = \text{---------- } + t_p$$

متوسط زمان سپری شده تا فریم به گیرنده وارد شود.



S/G : تعداد فریم های به مقصد رسیده به کل فریم ها.

CSMA : (Carrier Sensing Multiple Access)

ابتدا خطا را sense کن اگر خالی بود یواشکی بفرست صبرکن ! و اگر پر بود (آزاد) ...

* هر node برای اینکه بفهمد خط مشغول است یا نه حداکثر به اندازه t_p باید گوش دهد.

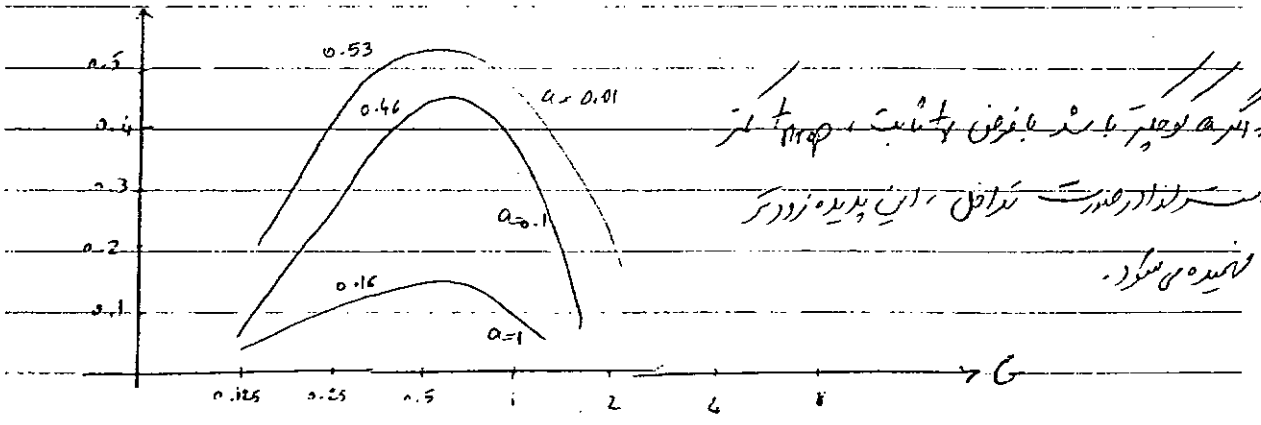
1- Persistant → اگر آزاد بود ارسال می‌کند. در صورت مشغول بودن مجدداً خط را چک می‌کند تا خط آزاد گردد. به محض آزاد شدن خط ارسال می‌کند. در صورت تداخل به زمانی به صورت تصادفی Back off می‌گردد. بعد از پایان این زمان یکی عملیات دوباره تکرار می‌شود (خط sense می‌شود).

2- non Persistant → اگر خط آزاد بود ارسال می‌گردد. در غیر اینصورت به مدتی را به صورت random صبر می‌کند و مجدداً خطا sense می‌کند. در صورت تداخل مثل بالا عمل می‌شود.

3- P Persistant → خطا دائماً sense می‌شود تا آزاد شود. ←
با احتمال P ارسال می‌کند
با احتمال 1-P به مدت t_{prop} صبر می‌کند و مجدداً خطا را sense می‌کند. ← در صورت مشغول بودن مجدداً sense می‌کند
* در صورت تداخل باشد بالا عمل می‌کند.

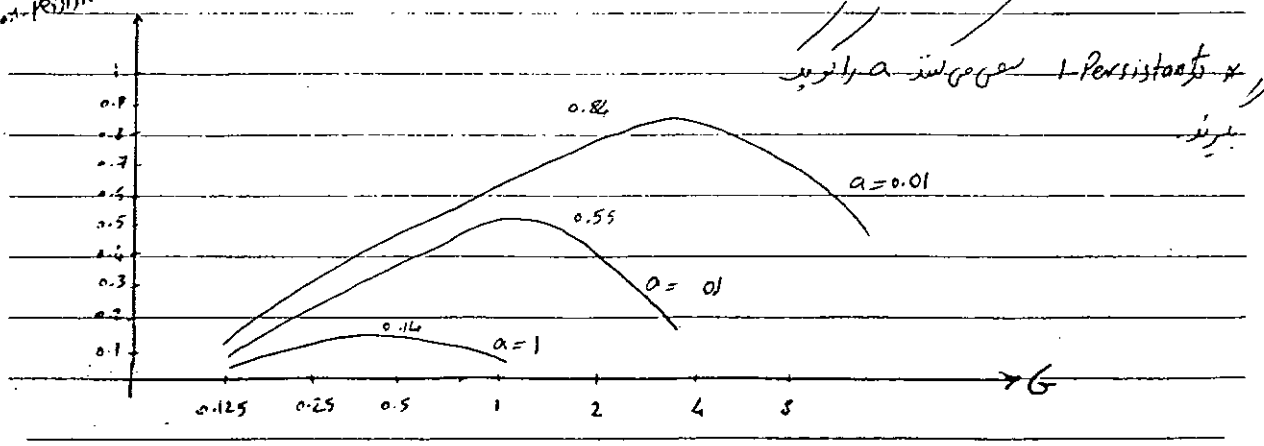
در پروتکل‌های لایه شبکه پارامتر $a = \frac{t_{prop}}{t_x}$ خیلی مهم است.

1-Persistant



* اگر a کوچکتر باشد با فرض ثابت بودن t_{prop} یعنی t_x بزرگ است پس در صورت تداخل، این پدیده زودتر تشخیص داده می‌شود.

Non-Persistant



* در 1-Persistant سعی می‌شد a را کوچک بگیرند.

موضوع: ..... تاریخ: .....

CSMA-CD (Carrier Sensing Multiple Access - Collision Detect) :

قبل از ارسال هر فریم، ابتدا خط sense می‌شود تا مطمئن شویم که در حالت بیکار است. در صورت آزاد بودن خط، ارسال صورت می‌گیرد.

$t_p \ll t_x$

فضای بین اطلاعات برای خطی فرستنده جهت اطمینان از خالی بودن خط را sense می‌کند. بعد از $2t_p$ اگر متوجه شود که پیام خراب شده، ارسال اطلاعات را متوقف می‌کند. یک زمان تصادفی Back off می‌کند برای بار دوم خط را sense می‌کند. اگر بعد از $2t_p$ تداخل نبود، خط را در اختیار خود است و ارسال خود را ادامه می‌دهد.

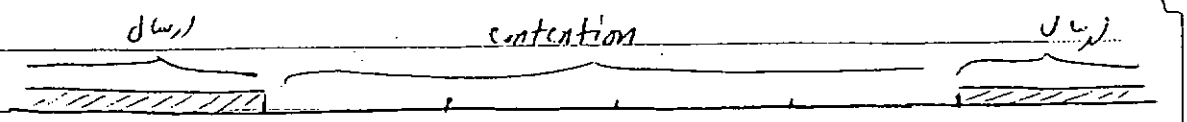
idle	بیکار	حالتی که کامپیوتر ارسال نمی‌کند
listen	گوش	
send	ارسال	حالتی که کامپیوتر در حال ارسال می‌کند

جلسه بیست و نهم: ۹۷/۳/۲۴

حالت Contention حالتی است که :

خط در حالت آزاد می‌شود و این هنگام نزدیک 3 کامپیوتر آماده ارسال اطلاعات هستند که هر سه نیز تشخیص می‌دهند خط آزاد است و شروع به ارسال می‌کنند بعد از $2t_p$ متوجه این تداخل می‌شوند. دوباره هر یک از آن‌ها به صورت تصادفی اقدام به ارسال اطلاعات می‌کنند. احتمال دارد این عمل انجام خواهد شد تداخلی نداشته باشیم. این حالت که کامپیوترها برای ارسال اطلاعات رقابت می‌کنند را contention

صفحه بعد



برای 1persistant

mini-slot : $2\,t_{prop}$

$P_{success}$: احتمال آنکه خاب طور موفقیت آمیزی در اختیار یکی از سیستم ها قرار گیرد و ارسال انجام شود در یک mini-slot

$= P \cdot (1-P)^{N-1} \cdot \binom{N}{1}$

(موفقیت یک کامپیوتر) (عدم موفقیت بقیه)

این طور فرض کردیم که احتمال در اختیار گرفتن خط برای تمام گره ها یکی و برابر باشد.

$P_{success}^{max} = N \cdot \frac{1}{N} \left(1-\frac{1}{N}\right)^{N-1} = \left(1-\frac{1}{N}\right)^{N-1}$

$(P=\frac{1}{N})$

$\lim_{N\to\infty} P_{success}^{max} = \frac{1}{e}$

* سوال: تعداد متوسط mini slot لازم است تا خط طور موفقیت آمیز در اختیار گرفته شود چقدر است؟

$Prob[\text{در } j \text{ امین mini slot برای در اختیار گرفتن خط پیروز شود}] = (1-P_{success})^{j-1} \cdot P_{success}$

(موفقیت در mini slot آخر) (عدم موفقیت در $j-1$ mini slot اول)

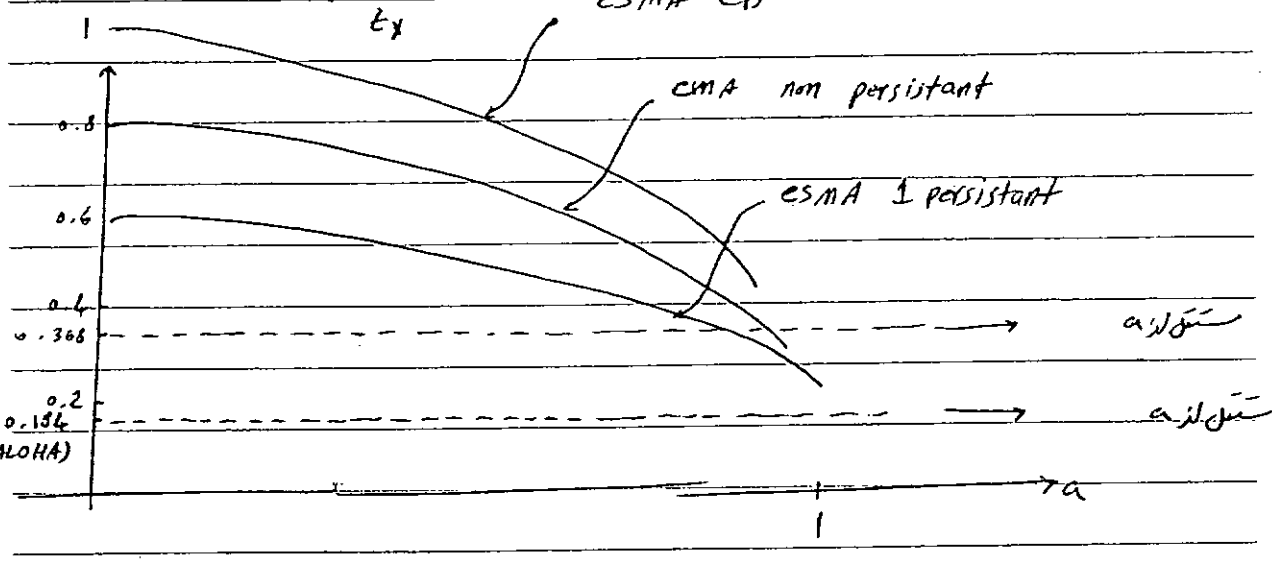
تعداد متوسط mini slot $= \sum_{j=1}^{\infty} j \cdot Prob[\text{پیروز شدن در } j \text{ امین mini slot}] = \sum_{j=1}^{\infty} j (1-P_{success})^{j-1} P_{success}$

$= \sum_{j=1}^{N} j (1-P_{success}^{max})^{j-1} P_{success}^{max} \xrightarrow{N\to\infty} \frac{1}{P_{success}^{max}} = \frac{1}{1/e} = e \cong 2.7$

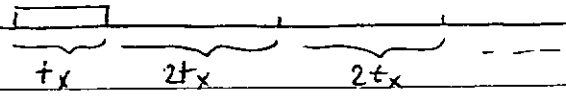
(بدترین حالت ممکن)

نمی t_{prop} زیاد شد (فاصله زیاد شد) باعث می‌شود زمانبندی minislot زیاد شود

$$a = \frac{t_{prop}}{t_x}$$



* هرچقدر a بزرگتر است (t_{prop} زیاد) آن‌گاه وضعیت رقابت داریم.



* این محاسبات را بطور دقیق از کتاب مطالعه کنید. (برای طراحی هم لازم خواهید داشت!!!)

دسته‌بندی به سه دسته کلی

کانال بندی

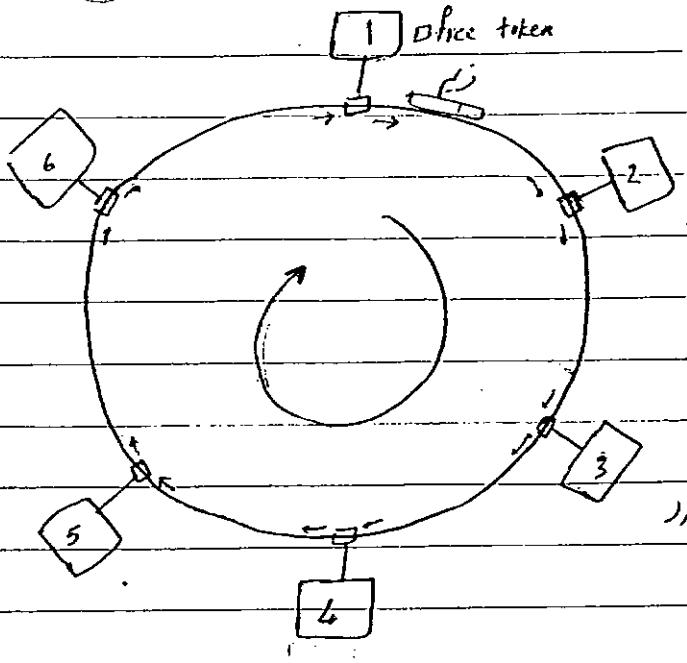
دسترسی اتفاقی

Scheduling
برنامه‌ریزی

random access
دسترسی رقابتی

موضوع: ..... تاریخ: .....

Token Passing Ring



* token است که می تواند اطلاعات را جابجا کند.

* اگر فریم اضافی نفرستند، دیگر token تبدیل می شود.

* فریم به هر گره که رسید و آدرس او بود پیام را برمی دارد و فریم اضافه را برای بعدی می فرستد.

* گیرنده بعد از اینکه اطلاعات را دریافت کرد باید به طریقی به فرستنده بفهماند که ۲ روش دارد:

۱- اطلاعات را از روی خط برمی دارد که صحت دریافت همه اطلاعات ack می فرستد.

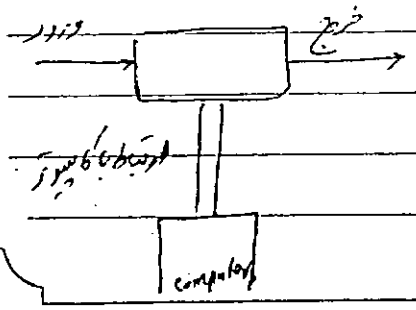
۲- Jet بخشی روی فریم به معنای دریافت اطلاعات و اجازه آنکه فریم به مبدأ برگردد. (کردن)

* مبدأ بعد از اینکه همه اطلاعات فرستاده شده token را در حالت آزاد به بعدی می دهد.

* باید بین اطلاعات و token تمایزی وجود داشته باشد.

← در جایی اطلاعات اگر token و bus وجود داشت معلوم می شد که اطلاعات است و نه token.

یک Patter است.



listen mode

(transmit mode) Send mode

موضوع: ..... تاریخ: .....

- حالت listen بیشتر مواردی که فروشی است اگر کسی بخواهد مربوط به فروش باشد کامپیوتر گوش را برای فروش بررسی کرده.

چون احتیاج به Process دارد لذا تأخیری هم دارد این است. هرچه تعداد node بیشتر باشد تأخیر هم بیشتر خواهد بود.

- حالت transmit ارتباط بین مواردی که فروشی تقاضا است.

(الزمه خط دیتا منتقل می‌کند و از خط دیگری نمی‌رسد)

free token را اختیار نمی‌کند مگر اینکه بخواهد transmit کند.

تأخیرها: t_{prop}: تأخیر انتشار علائم الزمه سیم به گیرنده t_x: تأخیر ارسال

t_I: تأخیر اینترفیس

τ = جمع تأخیرهای انتشار = $\frac{\text{طول خط}}{\text{سرعت انتشار}} = \frac{D}{V}$ یا $\frac{m.d}{V}$

$D = m.d$ ← با فرض تعدادی المان عبور کامپیوتر

طول فیزیکی (برحسب متر)

فرض کنید تعداد b bit ارسال شود ← $t_I = \frac{b}{R}$ = تأخیر اینترفیس

یا

b (bit) یا t_x (sec) = تأخیر اینترفیس

کل تأخیر انتقال = Mt_I یا Mb/R

$\tau' = Ring\ Latency = \tau + Mb/R$

تأخیر انتشار علائم در حلقه

* گیرنده بعد از busy token می‌فهمد فریم ارسال شده اطلاعات است و بعد از free token می‌فهمد نوبت اوست که ارسال کند.

; token reinsertion

Multi token

Single token ← بازدهی ← header فریم همان busy token است بلافاصله free token را می‌فرستد بعدی.

Single Packet ← بدترین ← بعد از فرستادن کل فریم free token را برای بعدی می‌فرستد

مثال: $M = 9$ و $a = 3.2$

الان می‌خواهیم بررسی کنیم که در چه حدی از روش‌های بالا را استفاده کنیم.

فرض‌ها:

- هر گاه نیاز داشته باشد اطلاعات برای ارسال است.
- هر ایستگاه بعد از free token دریافتی اگر بخواهد یک فریم اطلاعاتی با طول l بیت برای فرستد

موضوع: ..... تاریخ: .....

ρ_{max} = بیشترین بهره‌وری خط حداکثر — Multi token

$t_x = \frac{L}{R}$ — زمان ارسال فریم — طول فریم

- فاصله گیرنده و فرستنده یکسان است.

τ' — زمان انتشار علائم بین دو ایستگاه مجاور

$\frac{\tau'}{m}$

چون m تا گیرنده داریم لذا $m t_x$ زمان صرف شده برای اینکه اطلاعات روی خط گذاشته شود.

$$\rho_{max} = \frac{t_x + t_x + \cdots + t_x}{\underbrace{t_x + t_x + \cdots + t_x}_{m} + \tau'} = \frac{m t_x}{m t_x + \tau'}$$

$$= \frac{1}{1 + \frac{\tau'}{m . t_x}}$$

$$= \frac{1}{1 + \frac{a'}{m}}$$

$$a = \frac{t_{prop}}{t_x}$$

$$a' = \frac{\tau'}{t_x}$$

تعریف کتاب

بیشترین بهره‌وری (برای یک گیرنده)

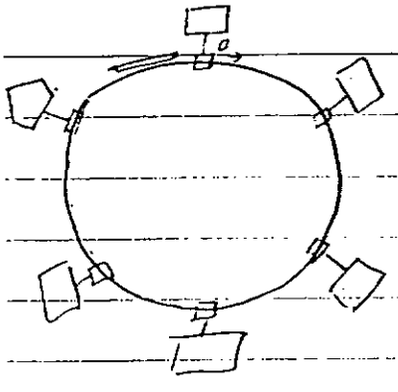
$$\rho_{max} = \frac{t_x}{t_x + \frac{\tau'}{m}}$$

Single Token :

$$\rho_{max} = \frac{t_x}{\tau' + \tau'/M} = \frac{1}{a'(1+\frac{1}{M})}$$

* در این حالت در واقع ← محض برداشت از روی بیت (در رسنده) freetoken را ← بعدی می‌خوانیم

کل زمان سپری شده برابر $\tau' + \tau'/M$ است.



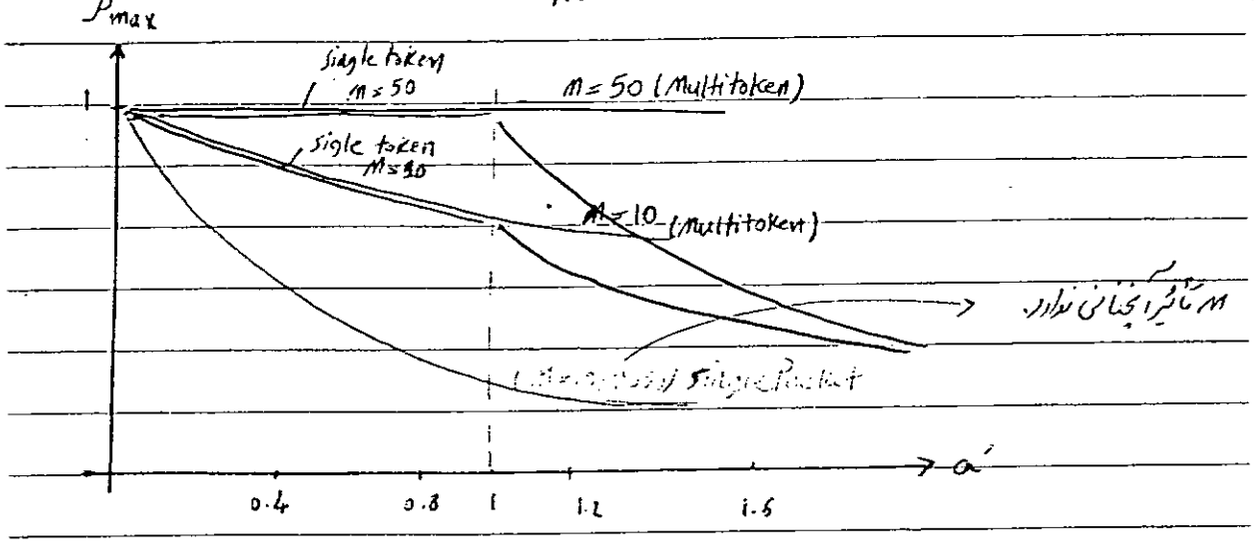
$$\rho_{max} = \frac{t_x}{t_x + \frac{\tau'}{M}} = \frac{1}{1 + \frac{a'}{M}}$$

در حالت کلی داریم :

$$\rho_{max} = \frac{1}{\max\{t_x, \tau'\} + \tau}$$

: Single Packet

$$\rho_{max} = \frac{t_x}{t_x + \tau' + \frac{\tau'}{m}} = \frac{1}{1 + a'(1 + \frac{1}{m})}$$



$t_x > \tau' \Rightarrow a' < 1$

هرگاه a' کمتر باشد بازده بیشتر بوده و آن گاه رابطه $\rho_{max} = \frac{1}{1 + a'(1 + \frac{1}{m})}$ (روش ۱ ترکیبی) نزدیک به ۱ می‌شود.

a' چیزی می‌شود و لذا سرعت $\frac{1}{a' + \frac{a'}{m}}$ در این حالت و رابطه Single Packet نزدیک می‌شود.

کانال بندی:

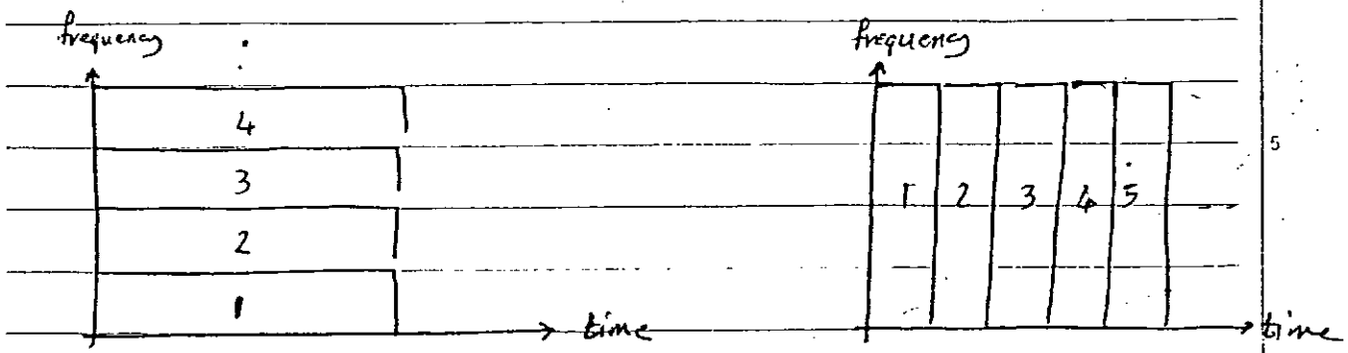
- کانال بندی
- دسترسی به مدیوم مشترک
  - برخورد ✓ دسترسی داده شده ✓

کانال بندی:

- امکانات کانال را تخصیص بندی می‌کنیم.

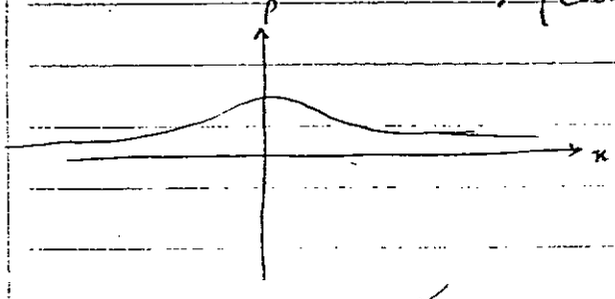
(Frequency Division Multiple Access) FDMA

TDMA (Time Division Multiple Access) :



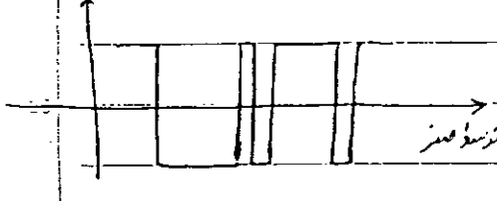
FDMA مربوط به لایه فیزیکی است ولی در مورد TDMA ممکن است در لایه های مختلفی انجام شود که آن را می آزماییم.

CDMA (Coded Division Multiple Access) :

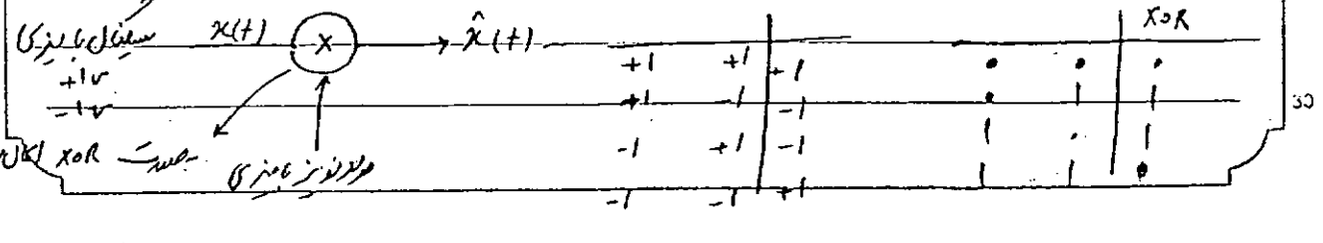


مقدار متوسط نویز برابر صفر است.

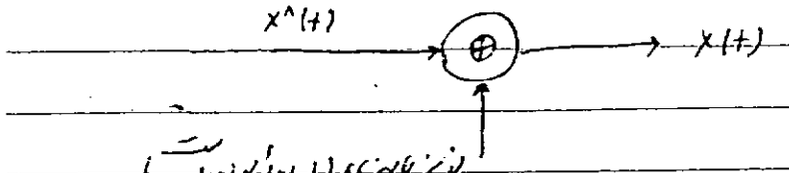
نویز بایزی : نویزی به طور تصادفی مقادیر ۰ و ۱ را می پذیرد. اگر محدوده ولتاژ را از ۰ تا ۵ در نظر بگیریم داریم که مقدار متوسط آن هم صفر خواهد بود.

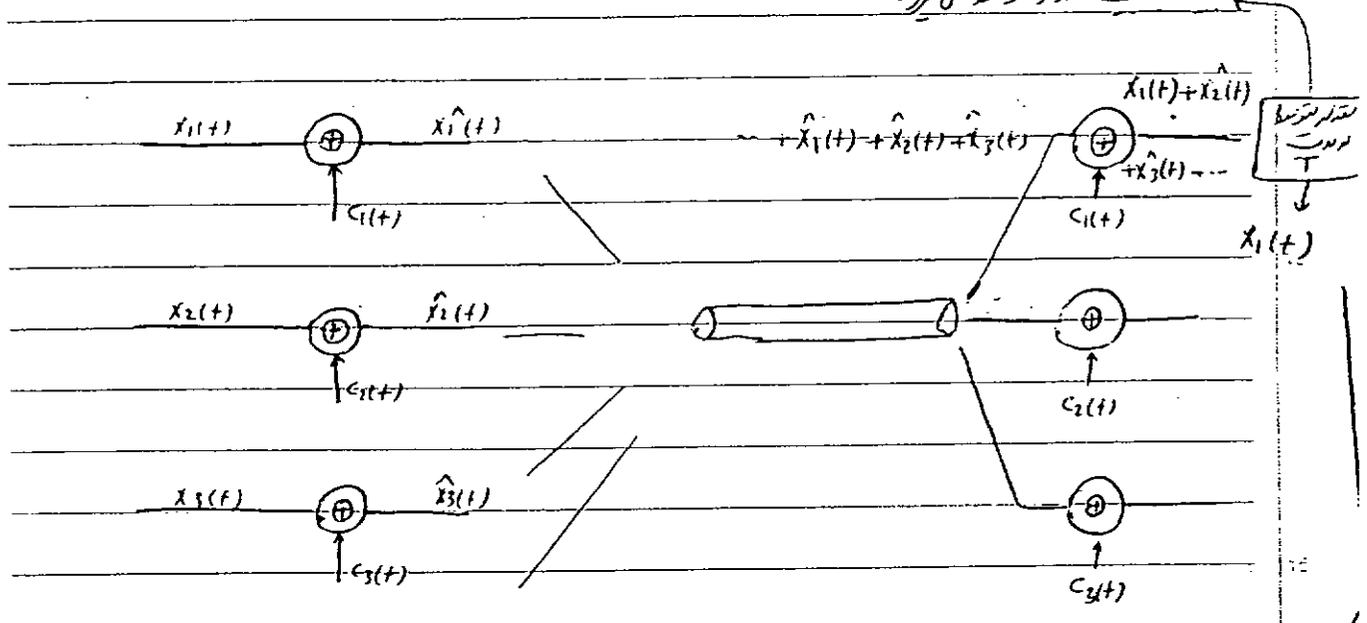


برای مقدار متوسط صفر، ظهور نویز



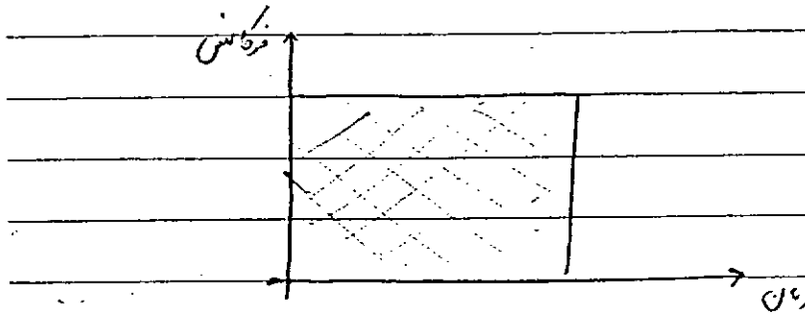
$X \oplus y = a \qquad a \oplus y = \quad X \oplus y \oplus y = X$





* نویزی که به Signal جمع شده نویز در نظر گرفته می‌شود که مقدار متوسط آن صفر خواهد بود.

* در مقدار متوسط گرفتن، مقدار متوسط سیگنال‌ها $x_1^{\wedge}(t) + x_2^{\wedge}(t)$ ... صفر خواهد شد.



* روش‌های دسترسی به کانال، همه‌ی بیت‌های بعدی را مورد استفاده قرار می‌دهد. مدولاسیون‌های TDMA و FDMA را با هم ترکیب می‌کند.

جلسه سی و پنجم : ۱۱ / ۳ / ۹۴

پروتکل IEEE 802.3 (Ethernet) :

جزو پروتکل های Random Access است و به صورت 1-persistance کار می کند.

CSMA - CD

R = 10 Mbps 512 bit = 64 Byte حداقل طول فریم

$$t_x = \frac{512}{10 \times 10^6} = 51.2\ \mu s$$

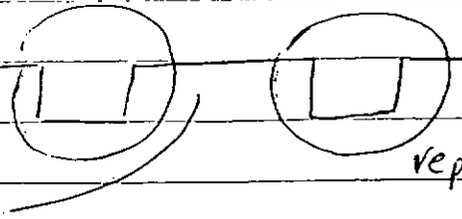
$$a = \frac{t_{prop}}{t_x}$$

$t_{prop} = 51.2\ \mu s \Leftarrow$ طول خط حداکثر 2500m می تواند باشد.

حداکثر طول خط 2500m

4 x 2500m = 10000m حداکثر ۴ سگمنت می توانیم داشته باشیم لذا

$$t_{prop} = \frac{10000\ m}{200,000 \times 10^3\ m/s} = 50\ \mu sec \simeq 51.2\ \mu s$$



موضوع: ..... تاریخ: .....

$2t_{prop}$ = mini slot

$K = \min(11, 10)$ $\quad B = \underbrace{10 \text{ تا } 2^K - 1}_{\text{بصورت دلخواه}}$

$t_{out} = B \cdot 2t_{prop}$

$n = 3$ $\quad K = \min(3, 10)$

$B = (0, 2^3 - 1) = \{0, 1, 2, 3, \dots, 7\}$

$\Rightarrow$ Back OFF $= B \cdot 2t_{prop}$

* اگر 16 بار ارسال مجدد نشد، بیش از آن به منزله قطع یا مشکل در خط ارتباطی است.

زمان از دست رفته = زمان تشخیص برخورد + $t_x + t_p$ + زمان منتظر برای فرصت برای برداشتن گرفتن خط

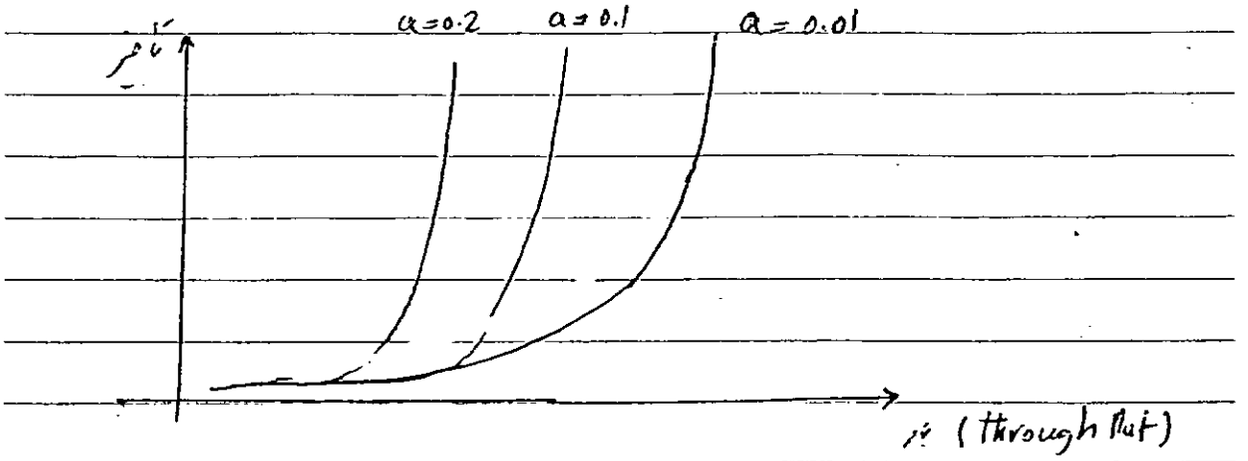
$= e \cdot 2t_{prop} + t_x + t_p$

بهره‌وری در خط مداوم $\Rightarrow \dfrac{t_x}{t_x + t_p + e \cdot 2t_{prop}}$ ، $a = \dfrac{t_{prop}}{t_x}$

سرعت در خط $\Rightarrow$ $\dfrac{1}{1 + \frac{t_{prop}}{t_x} + e \cdot \frac{2t_{prop}}{t_x}} = \dfrac{1}{1 + 6.44a}$

$\dfrac{1}{1 + (2e+1)\frac{t_{prop}}{t_x}}$

اگر a زیاد باشد بهره‌وری پایین می‌آید. توجیه این گونه بیان می‌شود که اگر a زیاد باشد احتمال تصادم کم خواهد بود ولی در صورت تصادم (خراب شدن) زمان زیادی از دست می‌دهیم.



با اضافه شدن بار ابتدا سرعت خطی زیاد می‌شود و در جایی دیگر سرعت کمی بالا می‌رود.

* احتمال collission در همه حالات یکسان است ولی مدت زمان آن بستگی به a دارد.

* هر چه a زیادتر باشد تعداد بارهایی که از شبکه می‌توان عبور داد کمتر می‌شود.

Pre amble	S.D	Destination add	Source addr	length	information	--
۷ بایت	۱ بایت	۲ یا ۶ بایت	۲ یا ۶ بایت	۲ بایت		

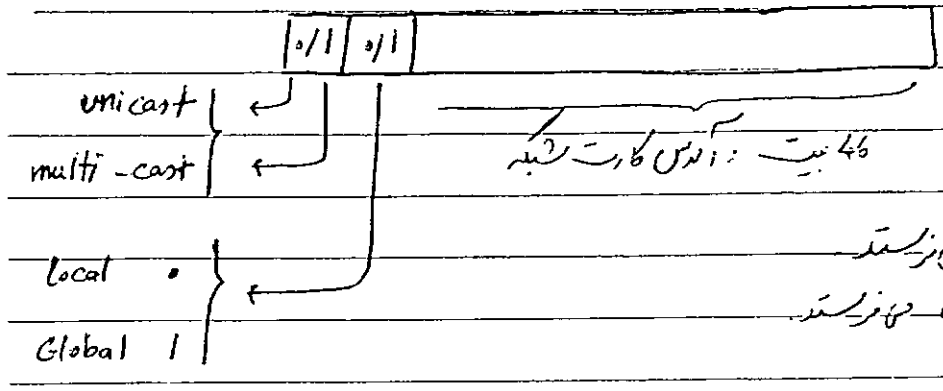
starting Delimiter

حرکت ماجاوله ۱۰۱۰۱۰۱۱

--	Pad	Frame checksum
		۳۲ بیت یا ۴ بایت

Pre amble برای همزمان کردن استفاده می‌شود.

destination Address یک آدرس ۴۸ بیت شبکه محلی است. نوع ۰ صفر بود



* آدرس کارت شبکه به صورت سخت افزاری در داخل کارت پیاده سازی شده است.

* برای حاصل شماره گذاری کارت بین سازنده ها توافق شده است:

CISCO 00000C ... (BCD)

3COM 02603C ... (BCD)

3 بیت

* length طول فریم را از D.S به بعد مشخص می کند. (خود DS نیست)

حداقل طول فریم 64 Byte

حداکثر طول فریم 1518 Byte

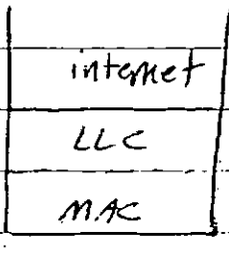
* اگر طول حوزه اطلاعات به اندازه ای نبود حداقل طول فریم رعایت شود، تعدادی صفر را برای کامل شدن فریم به عنوان Pad اضافه می کنیم.

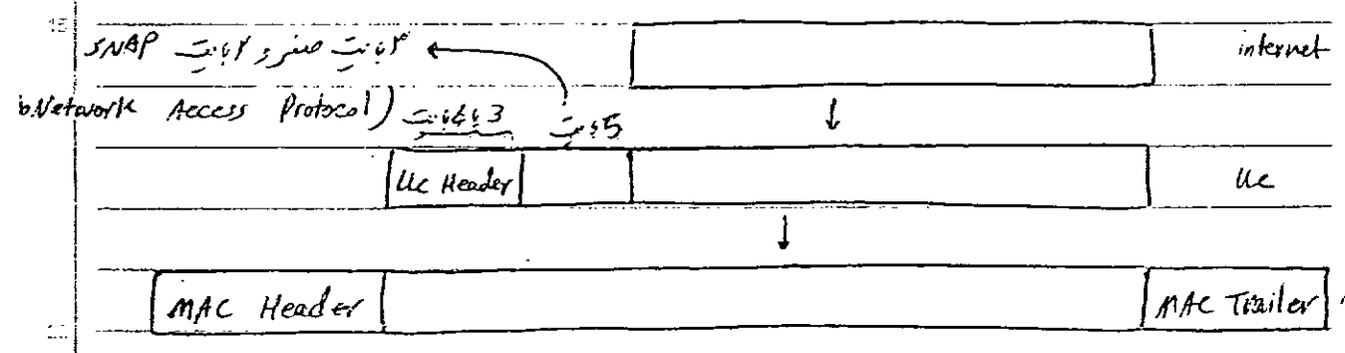
* در نسخه های قبل Ethernet تفاوت مختصری وجود دارد به جای length، type بود!! (پروتکل لایه بالا را مشخص می کرد) که مقدار آن از 0600 بزرگتر است.

* در واقع اگر مقدار مربوطه از 0600 بیشتر بود مربوط به نسخه های قدیمی و در غیر اینصورت نسخه جدید می باشد.

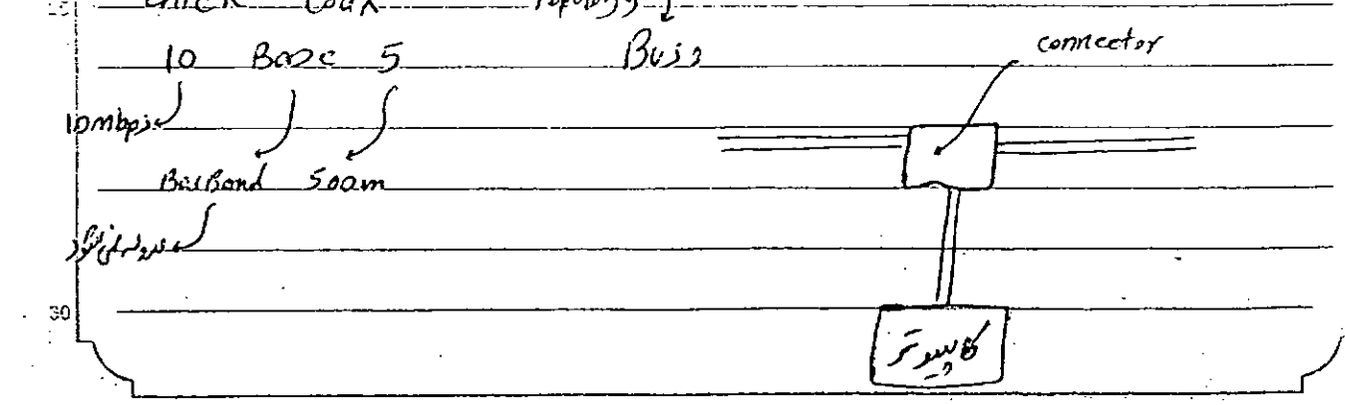
* در Ethernet مورد بحث ما (IEEE 802.3) ، لایه های زیرلایه MAC و LLC تعریف شده اند. (به طور استاندارد)

* در نسخه های قدیمی برای تشخیص طول فریم ، در انتهای فریم Preamble گذاشته می شود. و نیتش length را پیدا می کند.





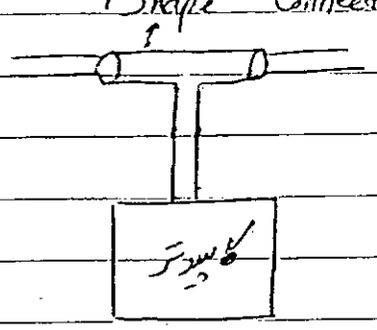
* لایه فیزیکی از Line Coding و نیز manchester استفاده می شود.



Thin Coax: topology : Bus

10 Base 2

Base Band 200m



Twisted Pair

10 Base T

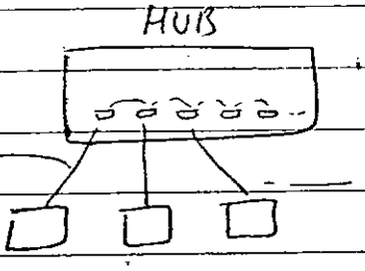
10 Mbps

Base Band twisted Pair

100m

topology : Star

تبدیل شد



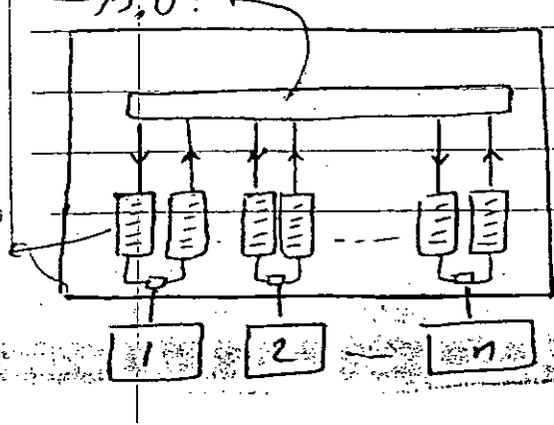
* توپولوژی آن شکل ظاهری ستاره است ولی فن الاتصال آن Bus است.

* HUB بیشتر از یک سیم مسی فعال است. اگر ترافیک اطلاعات صعب شود خود HUB Signal می‌فرستد و اطلاع می‌دهد.

* اگر بخواهیم به سرعت های بالاتر دسترسی پیدا کنیم باید از Edision کم کنیم.

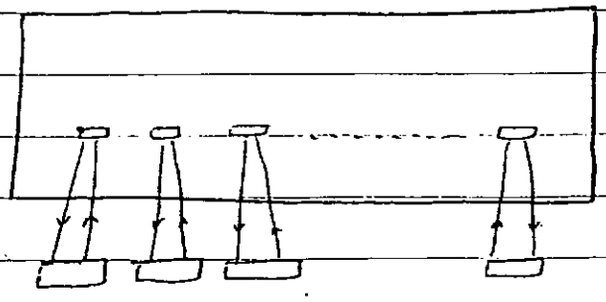
** Switch و Ethernet Switch :

این سوئیچ با کمک افزار خاصی از ایجاد Collision جلوگیری می‌کند. (کار صعب هوشمند و بسته پردازنده‌ای به آن تعبیه شده است صعب می‌گیرد)



طبقه بندی دوم ، ۱۶ و ۳۲ و ۸۴

Full-duplex Hub ،



دو ایتم خاص ، مدارش اطلاعات ندارم

* یک نوع کابل ، استفاده می شود عمدتا Twisted Pair است.

* کابل دیگری که استفاده می شود فیبر نوری است.

10 Base F ..... 2Km ..... Point-to-Point

Fiber ↙

Fast Ethernet (IEEE 802.3u) ،

نرخ بیت ۱۰۰ Mbps است.

$$a = \frac{t_{prop}}{t_x} \quad \leftarrow 4 \times 2300$$

مدارش حول فریم 64byte ، ۱۰ Mbps

$$t_x = \frac{64 \times 8}{10 \times 10^6} = 5.12\ \mu s$$

۱۰ Mbps ↙

لذا برای کم کردن a لوله پشتی ایجاد می کنیم .

موضوع: ..... تاریخ: .....

↑

الف - $R_t = 100\,Mbps$ $L = 640\,byte$ t_x ثابت بین هر دو مورد

ب - $L = 250\,m$ ↓ t_{prop} ↓ ✓

* دلیل را در حل سوال توجه کرده است. چون در هر دو حل تغییری ایجاد نمی‌کنیم.

* چون t_x را صورد 5.12μs برابر آورده‌ایم می‌خواهیم t_{prop} را هم حدود 5.12μs برابر آوریم تا نسبت را تغییر نداده باشیم.

کابل‌های مورد استفاده:

* 100 Base T4

twisted pair

configuration: star

لزوماً از hub استفاده می‌کنیم.

100m

Twisted pair UTP Category 3

unshielded

* تراشه‌ها روی 4 نوع سیم تقسیم می‌کنند.

* 100 Base T

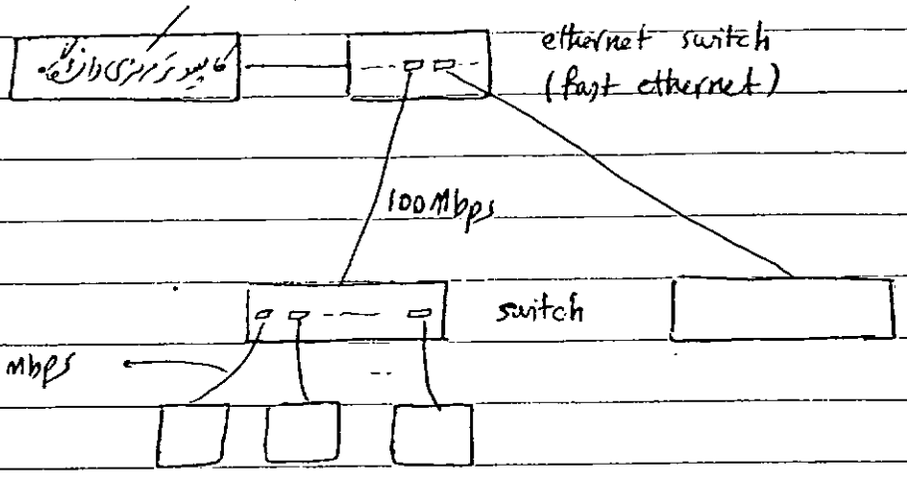
star, 100m, UTP Category 5

بهترین است!

100 Base F star, 2km, multi-mode

فیبر نوری

برای شبکه‌ی fast ethernet بیشتر از backbone استفاده می‌شود.



* تفاوت پروتکل ethernet, fast ethernet فقط در سرعت آن‌هاست.

Gigabit Ethernet (IEEE 802.z) ...

نرخ بیت برابر 10^3 Mbit/s است. ترجیحاً 1Gbps است.

در اینجا هم برای محاسبه‌ی a بیشتر بررسی می‌کنیم:

$l = 5000$ byte

$t_{prop} = 0.512\mu s \leftarrow l = 25m$

در این پروتکل راه‌حل کنترلی این است که طول فریم حداقل 512 byte باشد.

- در Packet bursting طول بیشتر فریم مورد توجه قرار می‌گیرد.

موضوع: ..... تاریخ: .....

کابلی:

1000 Base SX ، 1000 Base LX

فیبر نوری ارزشمندی — فیبر نوری

star — star

550m — 5km

1000 Base CX ، 1000 Base T

shielded مسی — twisted pair Category 5

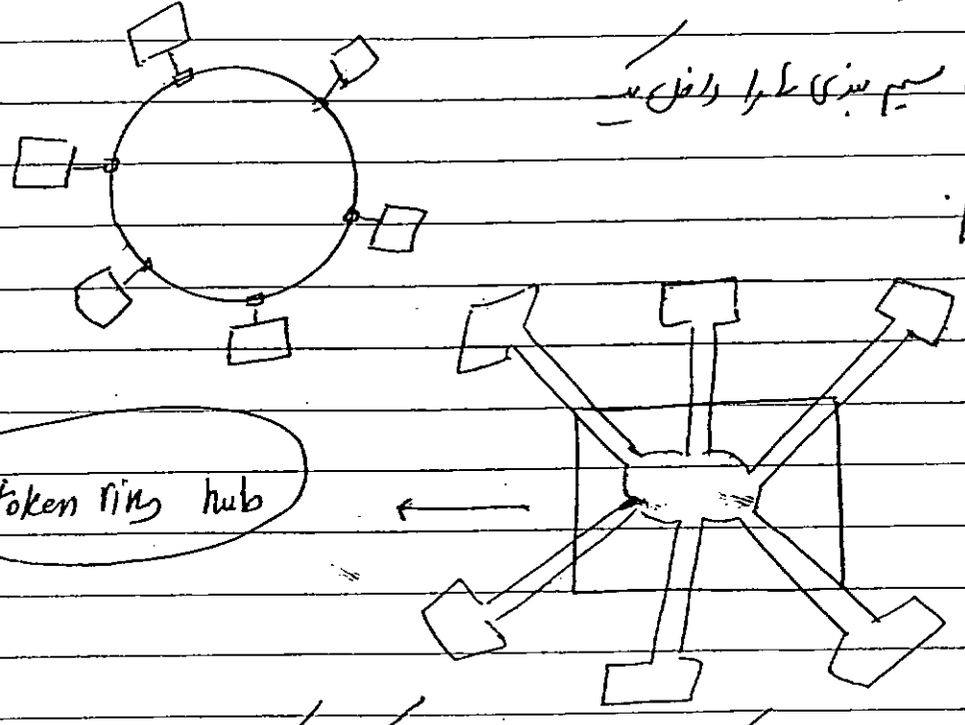
star — star

25m — 100m

: (token Ring) IEEE 802.5

مانند ethernet سیم بندی ها را رافی یه

حلقه تشکیل دهیم.



token ring hub

* اگر بخواهیم بین از کامپیوترهایی که وصل است را جدا کنیم با کشیدن فیش مربوطه

آن، این عمل شدنی است (با توجه به سافتر HUB)

۴ نرخ بیت کدام: 16 Mbps , 4 Mbps

حداکثر تعداد کامپیوترهایی که به hub وصل می‌شوند برابر 250 است.

الف) $R_1 = 4Mbps$, $M_1 = 20$, $d = 100\ m$, $b = 2.5\ bit$

$L = 400\ bit$, $L_{header} = 15\ byte$

$R_2 = 16Mbps$, $M_2 = 80$

① multi token

ا) $\rho_{max} = \frac{t_x}{t_x + \frac{\tau'}{M}}$

$\tau' = \tau + t_I = \frac{20 \times 100m}{2 \times 10^8 m/s} + \frac{20 \times 2.5}{4 \times 10^6 bit/s}$, $v = 200 \times 10^3 km/s$

$\tau'_{(b)} = \frac{4 \times 10^6 \times 20 \times 100}{2 \times 10^8} + 20 \times 2.5 = 40 + 50 = 90\ bit$

$\Rightarrow \tau_{sec} = \frac{90}{4 \times 10^6}\ bit/sec$

ب) $\tau' = \tau + t_I = \frac{80 \times 100m}{2 \times 10^8 m/s} + \frac{80 \times 2.5}{16 \times 10^6 bit/sec} \Rightarrow$

$(\tau')_{bit} = 16 \times 10^6 \times \frac{80 \times 100}{2 \times 10^8} + 80 \times 2.5 = 840\ bit$

$\Rightarrow \tau'_{sec} = \frac{840\ bit}{16 \times 10^6\ bit/s}$

موضوع: ..... تاريخ: .....

$$\Rightarrow \rho_{max} = \frac{400/R}{\frac{400}{R} + \frac{90}{20 \times R}} = \frac{400}{400 + \frac{90}{20}} \simeq 99\% \text{ ~ } 100\%$$

الف

$$\rho_{max} = \frac{400/R}{\frac{400}{R} + \frac{840}{R \times 8}} = \frac{400}{400 + 105} \simeq 99\%$$

ب

② Single token

$$\rho_{max} = \frac{t_x}{\max\{t_x, \tau'\} + \frac{\tau'}{M}}$$

الف) $$\rho_{max} = \frac{400/R}{\max\left\{\frac{400}{R}, \frac{90}{R} + \frac{120}{R}\right\} + \frac{90}{R \times 20}} = \frac{400}{400 + 6.5} \simeq 100\%$$

ب) $$\rho_{max} = \frac{400/R}{\max\left\{\frac{400}{R}, \frac{840}{R} + \frac{120}{R}\right\} + \frac{840}{R \times 80}} = \frac{400}{960 + 10.5} = 42\%$$

③ Single packet

$$\rho_{max} = \frac{t_x}{t_x + \tau' + \frac{\tau'}{M}}$$

الف) $$\rho_{max} = \frac{400/R}{\frac{400}{R} + \frac{90}{R} + \frac{90}{20R}} = \frac{400}{494.5} = 82\%$$

ب) $$\rho_{max} = \frac{400/R}{\frac{400}{R} + \frac{840}{R} + \frac{840}{20R}} = \frac{400}{1250.5} \simeq 32\%$$

نتیجه:

- روش multi token هم برای سرعت زیاد هم برای سرعت کم مناسب است.

- در single token وقتی سرعت بیت کم است و همین طور تعداد کامپیوتر، می‌تواند بهره‌وری بین ۱۰۰ تا ۸۵٪ باشد.

- روش single packet را خودت بررسی کن (به شکل)

طرح‌ها:

4Mbps IEEE 802.5 Single

4Mbps, IBM Single token

16Mbps IEEE 802.5 Multi token

16Mbps IBM

IEEE 802.5:

ثبت بیت‌های لوله‌ی مدار کنترل 10ms N=250 b=25 bit

SD	AC	FC	dest address	src address	information	FCS
۱ بایت	۱ بایت	۱ بایت	۶ یا ۲ بایت	۶ یا ۲ بایت		۴ بایت

ED	FS
۱ بایت	۱ بایت

موضوع: ..... تاریخ: .....

جلسه سی و سوم: ۱۸/۰۶/۹۸

تمرین‌های فصل ششم: 1، 2، 4، 5، 6، 7، 10، 18، 19، 21، 24، 29، 30، 33، 51، 52

از differential manch استفاده می‌شود. (Line coding)

SD: | J | K | · | J | K | · | · |

(start delimiter)

J: صفر منطقی

K: یک منطقی

AC: | P P P | T | M | R R R |

priority — Token — monitor — reserve

(T) token: 0 data / 1 token

* وقتی T یک است یعنی پشت سر token، ربات است یعنی busy

بیت M: زمانی از Station یک بیتی می‌گیرد همین فریم از خط خارج شود پاسش مدام در حلقه می‌چرخد. برای رفع این مشکل یک مانیتور روی خط قرار می‌دهند.

* ابتدا مقدار M برای فریم 0 است و یک Mstation آن را 1 می‌کند. اگر دوباره

بعد فریم را از حلقه حذف می‌کند.

سئوال: کدام کامپیوتر، زودتر است؟ اگر مانیتور خودش خراب شود، آن موقع چه کار باید کرد؟ ← طبق پروتکل بررسی شده است عیب‌یابی کنیم.

PPP: برای کامپیوتر اولویت قائل می‌شویم، کامپیوتری که دارای اولویت بالاتری است هنگامی که پیامی را مربوط به اولویت پایین‌تر است دریافت می‌کند، می‌تواند آن را متوقف و پیام خود را روی خط بفرستد.

* اگر token با اولویت P_1 باشد و اولویت ایستگاه P_2 باشد درصورتی می‌تواند از آن استفاده کند که $P_2 \geq P_1$ باشد.

* چندین کانال اولویت تعریف شده است.

FC: | FF | Z Z Z Z Z Z |

data 00 → نقطه به نقطه بین ایستگاه‌ها

Control 01 → " " برای عملیات کنترلی

* فیلد information رشته بزرگی از بیت‌ها است و استراتژی و تعداد کامپیوترهای مورد نظر تعیین شده است.

ED: | J K 1 J K 1 | I | E |

(ending Delimiter)

* فیلد I مشخص می‌کند که فریم آخری است یا نه.

Intermediate_frame bit

E : error bit

(اگر Line Coding خراب شود، FCS set می‌شود)

FS: | A | C | X | X | A | C | X | X |

↳ frame copied ← اگر A و C یک بیت 1 شوند

address recognized

* کامپیوتر فریم خود را ارسال می‌کند تا مبدأ برگردد. حال اطمینان پیدا می‌کند که گیرنده فریم را دریافت کرده است. توسط A تنظیم می‌شود (address recognized) این فریم را گیرنده set کرده

فریم خالی (توکن!) : | SD | AC | ED |

FDDI : (Fiber Distributed Data Interface)

M = 500 R = 100 Mbps

* Ring می‌تواند در محدوده 200 km کار کند.

* خروجی این پروتکل همان از لینک مطالعه شود.

Network		
LLC	IEEE 802.3	ethernet
	IEEE 802.5	token ring
	IEEE 802.11	wireless LAN
physical		

: (wireless LAN) IEEE 802.11

از CSMA / CA (Collision Avoidance) استفاده می‌کند. در 424

* چون هر کامپیوتر می‌خواهد با دیگری ارتباط برقرار کند، نمی‌تواند از ALOHA و CSMA-CD استفاده کرد.

(Collision Detection) ← چون در حالت بی‌سیم تشخیص خط کار مشکل است استفاده نمی‌شود

* هر یک از کامپیوترها روی خط گوش می‌دهند تا زمانی که خط آزاد است بتوانند ارسال کنند و وقتی اشغال است منتظر می‌مانند تا ارسال کنند. این امر باعث کندتر شدن سرعت می‌شود.

data

random

یک عدد تصادفی + یک زمان Back off = صبر

← باعث از بین رفتن Collision می‌شود.

یک مکانیزم handshaking نیز استفاده می‌کند.

② هر کس که خواهد ارسال کند ابتدا RTS (Request To send) را ارسال می‌کند و پیام CTS (Clear To Send) را دریافت می‌کند و سپس ← برای می‌فرستد

موضوع: ..... تاریخ: .....

اگر RTS و هم تداخل پیدا کند CTS فرستاده می‌شود و فرستنده موقت
اطلاعات را نفرستاده نگه داشته و دوباره می‌فرستد.

* چیزی که مربوط به سنگینی از طول کم فریم‌های RTS و CTS است و اصل
سرفرصت را خدشه‌دار می‌کند.

* در پروتکل‌های بی‌سیم بحث حساسیت به نویز خیلی مهم است و نمی‌توانیم
سرعت‌های خیلی بالا داشته باشیم.

* از کاوا استین (آنتن) شکل است. (قابل مشاهده است)

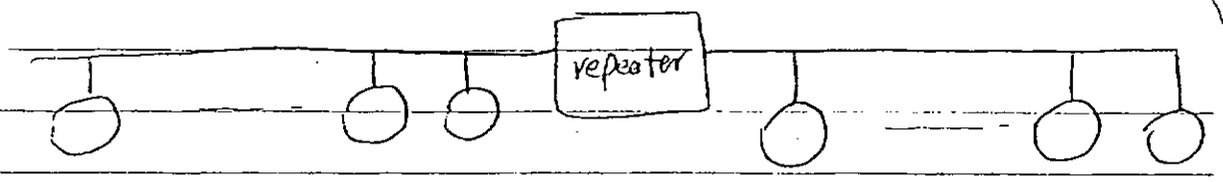
* محدودیت فرکانسی داریم که در کشورهای مختلف متفاوت است.

* بقیه مطالب را از کتاب مطالعه کنید.

(پل) Bridge:

فرض می‌کنیم 2 تا segment داریم. و داشتیم طول حداکثر segment برابر 1250m

اگر بخواهیم طول سیستم بزرگ‌تر شود که بیش از 500m می‌شود تضعیف
سیگنال زیاد می‌شود. اگر یک repeater داشته باشیم سیگنال را
تقویت می‌کند آن‌گاه مشکل حل خواهد شد ← repeater فقط
لایه فیزیکی تقویت می‌کند.



* شکل بالا یک LAN است که تعداد کامپیوترهای آن $N_1 + N_2$ است

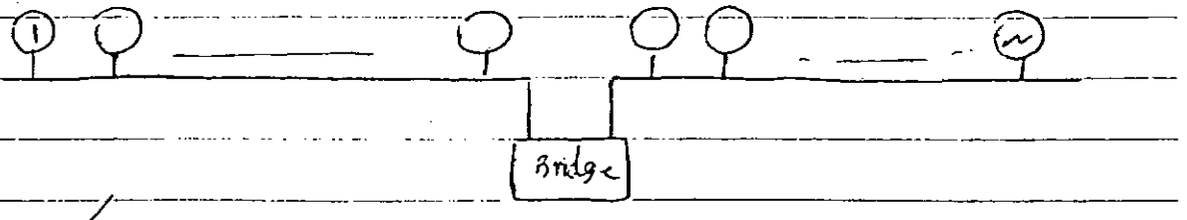
برای حالت هم	طول خط زیاد شده ← تداخل دیرتر متوجه می‌شود
	تعداد کامپیوتر زیاد شد ← تداخل بیشتر صورت می‌گیرد.

Repeater ← physical

Bridge ← Data link ← بررسی بیت‌ها

Router ← Network

جلسه سی و چهارم: ۸۴/۳/۲۱



* فرض کنید ① چیزی می‌خواهد بفرستد به ② . Bridge درست چپی نیست که آدرس کلیه کامپیوترهای پاراخیر را برای اینکه ریت لایه پایین (link) آدرس لایه MAC را می‌بیند. اگر بخواهد کاری انجام نمی‌دهد بنابراین باید به سمت راست عیناً انتقال دهد (چون اطلاعات لایه فیزیکی را هم داشته باشد).

موضوع: ..... تاریخ: .....

فرکانس در repeater :

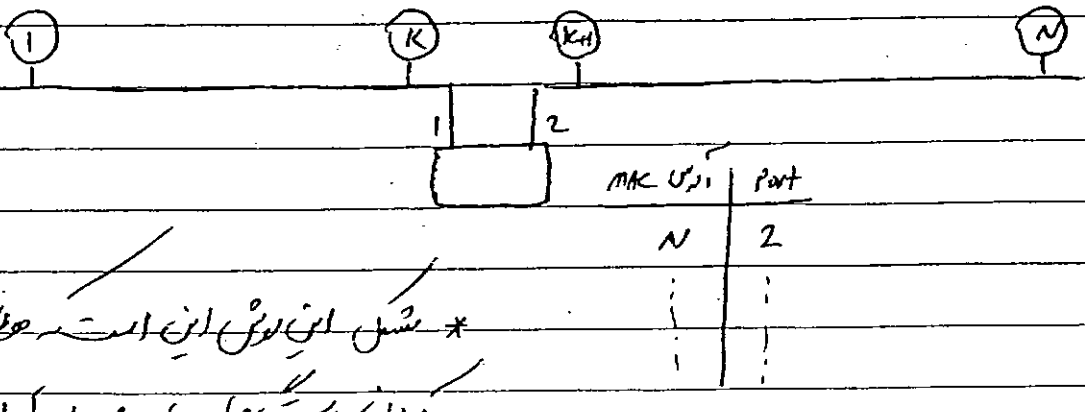
* در repeater کل ترافیک منتقل می‌شود. لذا ترافیک 2 طرف با هم جمع می‌شود $\rho = \rho_1 + \rho_2$

ولی در Bridge فرض کنیم P احتمال ترافیک طرف دیگر منتقل شود لذا داریم:

$$\rho_{\text{سمت چپ}} = \rho_1 + \rho_2 P \qquad \rho_{\text{سمت راست}} = \rho_2 + P\rho_1$$

نتیجه اینکه ترافیک با استفاده از Bridge کاهش یافته و احتمال collision هم می‌شود و در کل بهره‌وری از خط بالا می‌رود.

* برای اینکه Bridge بفهمد آدرس خاصی وجود دارد یا نه (در کدام طرف) مثلاً می‌آییم از یک جدول استفاده کنیم.

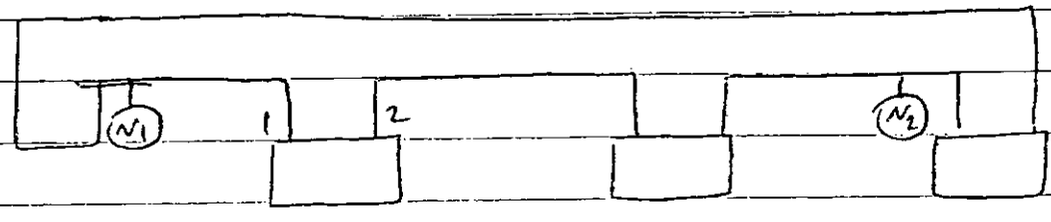


* مشکل این روش این است هر بار با عوض کردن دستگاه‌ها، باید جدول را هم عوض دهیم.

چند نقطه:

- عملکرد پل به یک فریم روی LAN بعدی چطور است.
- عملکرد یا کاربرد هر کامپیوتری روی کدام LAN.

برای رفع مشکل گفته شده از روش message forwarding استفاده می‌شود.



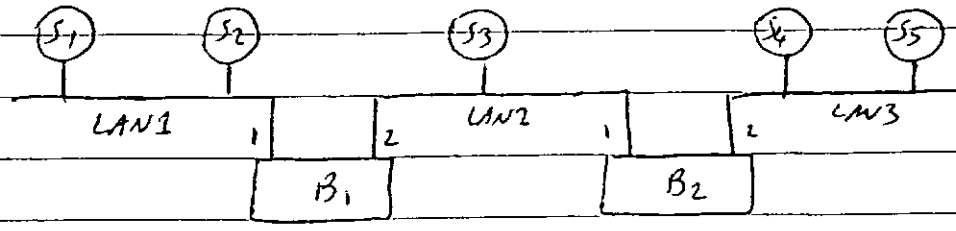
* در شکل مقابل اگر بخواهیم از N_1 به N_2 بفرستیم Bridge باید از کدام پورت عبور دهد (از ۲ طریق می‌شود این کار را کرد.) که اگر از هر ۲ طرف بفرستیم ترافیک زیاد می‌شود لذا مجبور را یکطرفه می‌کنیم.

* در این شبکه نباید loop یا حلقه وجود داشته باشد.

توضیح روش message forwarding

موضوع: ..... تاریخ: .....

مثال:



فرض می‌کنیم که S_1 شروع کند و S_2 پاسخ دهد.

MAC addr	port	MAC addr	port
S_1	1	S_1	1
S_3	2	S_3	1
S_4	2	S_4	2

$S_2 \leftarrow S_3$

$S_3 \leftarrow S_4$

پل 1 ، پیام گرفته شده از راست را به طرف چپ پخش نمی‌کند!

S_2	1

$S_2 \leftarrow S_1$

در این حالت B_2 کاری انجام نمی‌دهد.

* اگر حلقه وجود داشته باشد عمل کردن table بخلط می‌شود و همین عرض می‌شود به جای دیدن آدرس Node آن را از پل می‌گیرد و فکر می‌کند Bridge عمل می‌کند.

* در مثال بالا، اگر از LAN3 شروع کنیم هم Bridge ها اشتباه است و آدرس‌ها را با جدول حافظه نداشته باشند آن را اصلاح می‌کنند. ولی با اصلاح کامل ممکن تعدادی از اطلاعات گم شود.

اگر یکی از Node ها حذف شوند اینگونه عمل می‌شود:

برای هر آدرسی که در جدول وارد می‌کنیم یک timer در نظر می‌گیریم اگر تا مدت پایین زمان timer یک frame راجع به node مربوطه آن آدرس را refresh می‌کنیم در غیر اینصورت آن آدرس را حذف می‌کنیم.

* اگر تعداد زیادی LAN را با Bridge به هم وصل کنیم آنگاه با شکل پیچیده‌ای روبرو می‌شویم برای تشخیص loop باید چاره اندیشی کنیم: بنابراین از تکنیک درخت پوشا استفاده می‌کنیم

الگوریتم درخت پوشا (Spanning Tree Algorithm) :

هر LAN یک Node می‌شود و هر Bridge یک یال - گراف خاصی درخت پوشای مربوطه را پیدا می‌کنیم البته سعی می‌کنیم minimum spanning tree را پیدا کنیم - برای هر Bridge وزنی تعریف می‌کنیم که همان میزان ترافیک در آن LAN است.

- ابتدا یک یال - عنوان ریشه در نظر گرفته می‌شود (آن یالی که کمترین شماره شناسایی کوچک‌تر است)

- برای هر پل، به جز پل ریشه، پورتی را که از طریق آن با حداقل هزینه (نا فیر) به پل ریشه می‌توان وصل شد معین می‌کنیم. نام پورت را root port می‌نامیم.

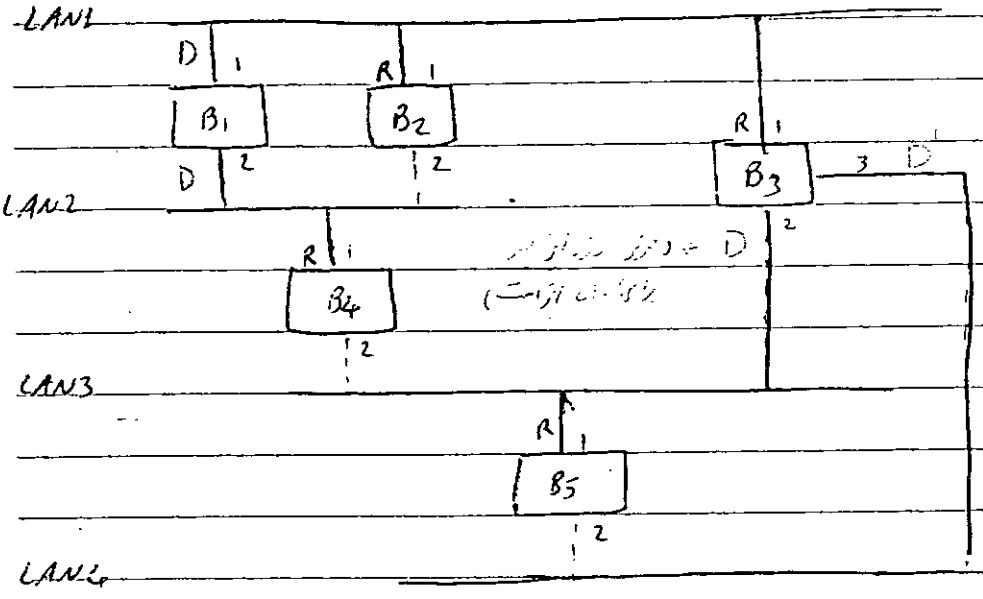
- برای هر LAN که داریم پورتی که از طریق آن با حداقل هزینه به پل ریشه وصل می‌شویم را معین می‌کنیم. نام این پورت را designated port می‌نامیم.

(پورت تخصیص)

موضوع: ..... تاریخ: .....

در حالت پایدار، با فعال کردن root port و designated port ها (و غیرفعال کردن بقیه پورت ها) وضعیت پل ها مربوط به درست می آید.

مثال:



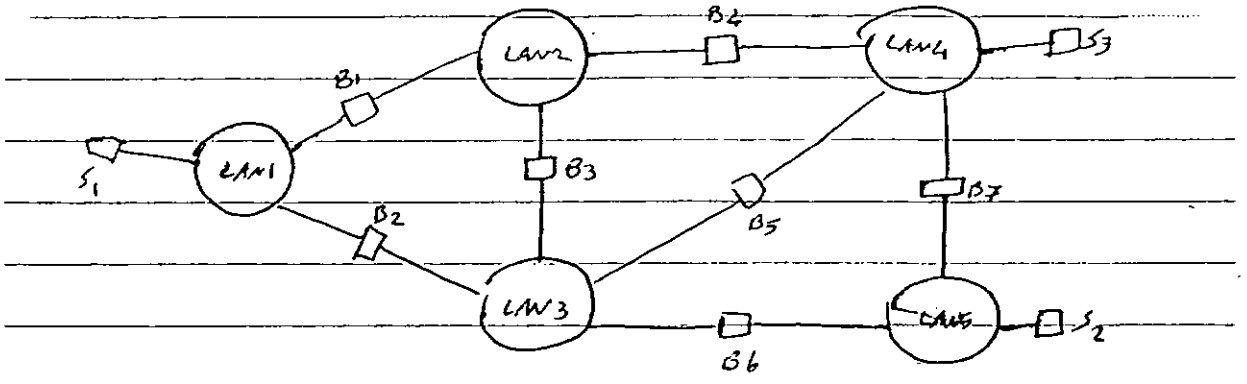
در این حالت B_3، B_2، B_4 غیرفعال شدن آنها را حذف نمی کنیم چون با عوض شدن حالت و رفتار سیستم ممکن است دوباره فعال شوند.

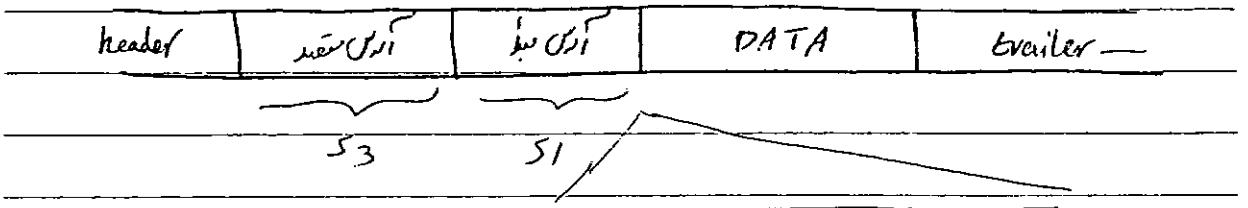
پل ها از نوع Configuration BPDU با یکدیگر مبادله می شود.

Ethernet LAN ← (transparent Bridge) شفاف

پل

TokenRing LAN ← (Source routing bridge) مسیریابی مبدأ





LAN1	B1			

مثال: $S_1 \rightarrow LAN1 \rightarrow B1 \rightarrow LAN2 \rightarrow B4 \rightarrow LAN4 \rightarrow S_3$

آدرس 12 بیتی — آدرس 4 بیتی

* نقطه ابتدا S_1 از کجا برای او زمان مسیر اطلاعات را فرستد؟

← از ۲ فریم به نام آن

1- single rout broadcast frame ← اندر حد آن مختص را دارد که مقصد بتواند که از زمان کمترین مسیر است.

2- all-routes " " "

* اول به مقصد اعلام تا این از نحوه مبدا را می کند. بعد از آن فریم دوم را می فرستد (و کار را انجام می دهد) S_3 به LAN4 می خواهد حرکت از Brides که به LAN4 وصل هستند آن را می بیند و کار می کند. S_1 — LAN1 — B_1 — LAN2 — B_4 — LAN4 — S_3

$S_3 \rightarrow LAN4 \rightarrow B5 \rightarrow LAN3 \rightarrow B2$ / $B3$ / $B6$

$B7$ — LAN5 — $B6$

لذا مسیر به زودترین رسید که در حد برای داشت می شود و سیستم آن راه مسیر حداقل را بلد است.

* اگر میزان خطای مدنظر نباشیم که از میزان گذشته نداشته باشیم، می‌توانیم از ترافیک مربوطه کم کنیم.

* مدل ناشناس عمده ترجمه عبارة Bridge است (Table) ولی مدل با میزان مبدأ عمده کار عبارة گره مبدأ است که باید میزان مدنظر را مشخص کند.

* اگر 2 تا شبکه محلی داشته باشیم از نوع Ethernet و Token Ring چگونه باهم وصل کنیم؟ ← نمی‌توانیم از مدل سوئی استفاده کنیم. چون ساختار Frame متفاوت است (با تغییر سری عملیات اضافی هم داشته باشیم. به آنها mixed media Bridge می‌گویند که به نوع تقسیم می‌شود.

Tranlation Bridge ← مدل فرم ها را انجام می‌دهد.

Source_route Transparent Bridge ← هر دو عملیات