

* 7. Hafta - Tatil

8. " - 16.11.2011 Vize 1 + Sınav Konuları Belirlenecek

12. " - 14.12.2011 Vize 2 + Rapor Teslimi

14. " - 28.12.2011 } Sınavlar
15. " - 04.01.2012 }

* Raporlar

↳ A4 kâğıdı (çarkalı - önü)
10 punto, tek satır aralık.

Konu
GRUP NO
ÖĞR. NO AD SOYAD

Vize 1	%20
Vize 2	%20
Sınav	%20
Final	%40

TELE COMMUNICATION
Uzak

→ iletişim için olmasa olmazlar.

- Ortam
- Verici
- Alıcı
- Mesaj
- Protokol (Kurallar Bütünü)

Doğru haberleşme için;

- Mesaj çıktığı gibi alıcıya ulaşmalı
- Mesaj doğru alıcıya ulaşmalı
- Mesaj zamanında ulaşmalı

→ Bilgisayar ağlarının faydaları?

- Veri paylaşımı
- Kaynak "
- Tasarruf

Veri - Data

Bilgi - Information (işlenmiş Veri)

→ Bilgisayar ağlarının olumsuz yönleri:

- Güvenlik

→ Bilgisayar Ağlarını Sınıflandırma

- PAN (Personal Area Network)
- LAN (Local Area Network)
- CAN (Campus " ")
- MAN (Metropolitan Area Network)
- WAN (Wide Area Network)

⊖ Kapsama alanı



⊕

(2)

15.01

- PAN: Kişisel verilerin paylaşıldığı ağ (telefon - bilg. arası)
- LAN: Yalıtılmış olarak 100m yarıçapında kurulan ağ.
(Aynı oda, aynı kat veya aynı bina)
- CAN: Duruma göre LAN yerine de kullanılabilir. Birkaç LAN birlesiminden oluşabilir (Birkaç km yarıçapı)
- MAN: Genellikle şehir sapında kurulur, CAN ile arasındaki fark servis sağlayıcılarının araya girmesidir.
(Kampüste sorumluluk okulundur, ama farklı kampüsler arasında araya bağlantıyı sağlayan başka bir şirket vs. girer.)
- WAN: Belli bir sınırı yoktur, ülkeler ve kıtalar arası olabilir, araya birkaç servis sağlayıcı girebilir.

→ İki bilgisayar ağını karşılaştırırsak nelere bakmamız gerekir?

- Hız (bps = bit/sec)
- Güvenlik
- Maliyet (Kurulum / işletme)
- Kapasite (Kullanıcı Sayısı)
- Hataları azaltma edebilirlik / Hata oranı
- Geliştirilme kapasitesi (Kullanılabilir protokol çeşitliliği)
- Cevap süresi

- STANDARTLARDE JURE

İnsanların birbiriyle görüşüp tartışıp fikir birliğine vararak oluşturduğu standartlar.

DE FACTO:

Değeri yeterli kendiliğinden oluşmuş standartlar.

→ Açık Standart: Herkes rahatca kullanabilir (Linux)

→ Kapalı Standart: Alt yapısını bilmezsiniz. Rahatca kullanabilmek için telif hakkını ödemek gerekir. (Windows)

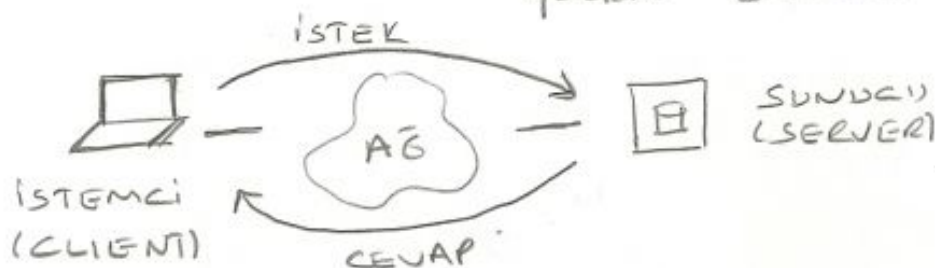
- ISO (International Standard Organization)
- IEEE
- ANSI
- ETSI
- ITU (International Telecommunication Union)

→ THROUGHPUT: Site düşen pay. (100 mbits hızla internetin 60 mbits - 55 mbits sara düşen paydır)

→ RESPONSE TIME: Herhangi bir yerdeki sunucu sistemine gönderilen istekğin ulaşma süresi

⊕
Sunucunun istek doğrultusunda yaptığı işlemlerin harcadığı süre

⊕
Oluşturulan cevabın geri dönmesi için geçen zaman.

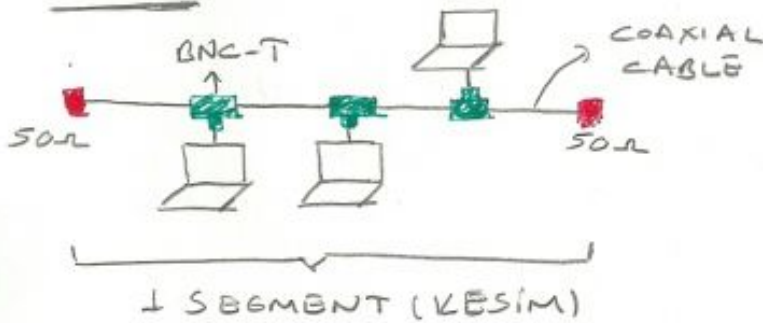


JITTER: Haberleşme paketlerinin gönderim süreleri arasında düzensizlik olması. (İletim süreleri arasındaki değişim)

- İlk mesaj 1 dk sonra, ikinci mesaj 10 dk sonra, 3. mesaj 3 dk sonra iletiliyorsa...

TOPOLOGY (Topoloji / İlinge)

1.) BUS (YOL):



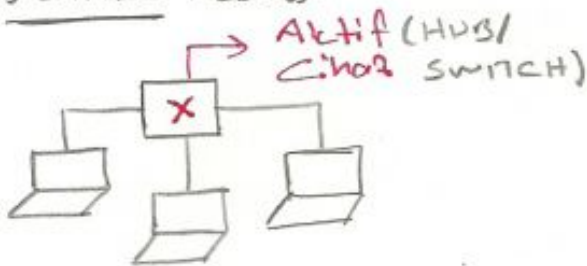
- (Elk. yok)
- (+) Aktif cihaz gerektirmez
- (+) Maliyeti düşüktür
- (+) Kurulumu kolaydır.

⊖ Hatta kesilme olursa sistem hiçbir şekilde çalışmaz (hassas)

⊖ 1 Segment arasında 185m sınırı vardır.

⊖ Çok fazla kullanıcı olduğunda efektif değil

2.) STAR (YILDIZ):

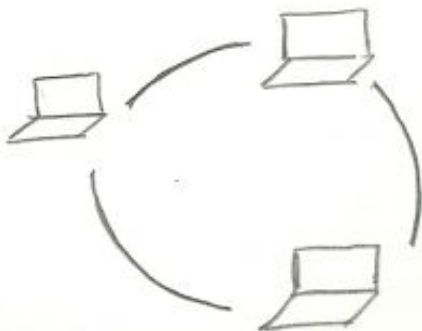


(+) Bilgisayarlardan birinin ağdan kopması diğer bilgisayarlar arasındaki bağlantıyı etkilemez

⊖ Elektrik kesilirse boku yersin.

⊖ Kablo maliyeti fazladır. (Her bilgisayar için kablolu)

3.) RING (HALKA):



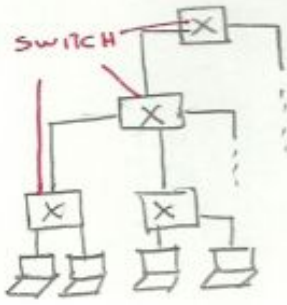
⊖ Bağlantılardan biri koparsa tüm sistem bozulur.

(+) Eğer 2. bir kablo çekilirse herhangi bir yerde kopukluk olduğunda;



Görüm böyle bulunur

4) TREE (AĞAC):

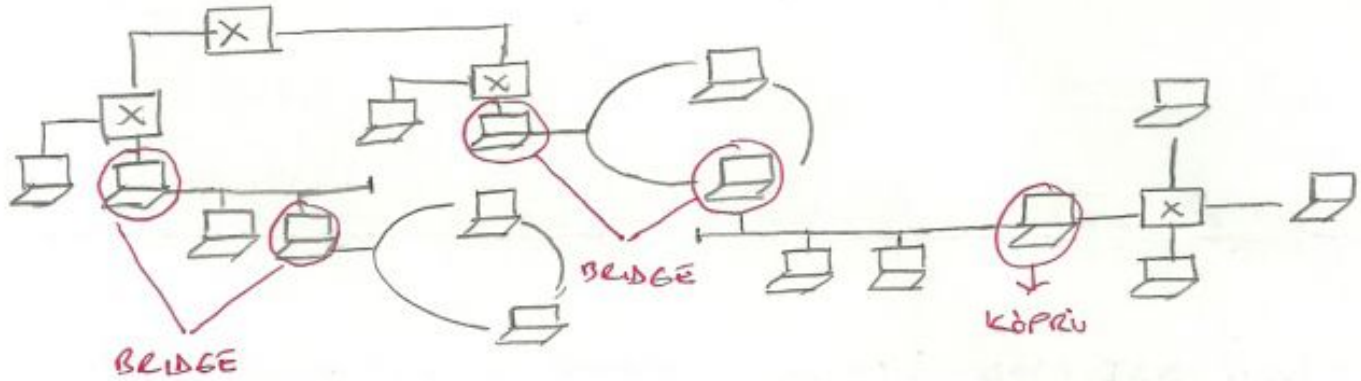


- (+) Hiyerarşik yapıda olduğu için yönetmesi kolaydır.
- (+) Farklı yollarlar arasındaki haberleşme birbirlerini etkilemez
- (-) Kökten daha doğru ayrıldıkça bağlantı hızı da köke göre azalır. (Teorik olarak)

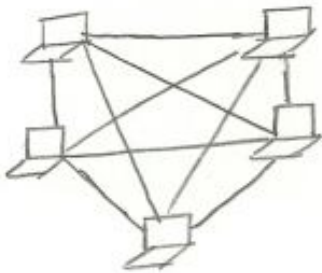
5.) HİBRİT (MEBLET):

→ Diğer 4'ünün karışımından oluşabilir.

- Geniş alan ağına aitken sistem içindeki hızdan daha az (genelde) hızla akar (Lab'da 100mbit/s WLAN'a aitken 50mbit/s)



6.) MESH (ÖRÖN): BİLGİSAYAR AĞLARI İÇİN UYGUN DEĞİL!



- Her bilgisayar, ağıdaki diğer bütün bilgisayarlara bağlıdır.
- (-) n bilgisayar bağlamak için her bilgisayara (n-1) tane bağlantı kartı takmak gerekir.
- (+) Bir bilgisayar bütün bağlantıları kopmadığı sürece diğer bütün bilgisayarlara bağlantı kurabilir.

- 4 temel
- 1 hibrit
- 1 uygun olmayan topoloji

İLETİM MODELLERİ

1. SIMPLEX:  V: Verici
A: Alıcı.

2. HALF DUPLEX:  L_2
 L_1

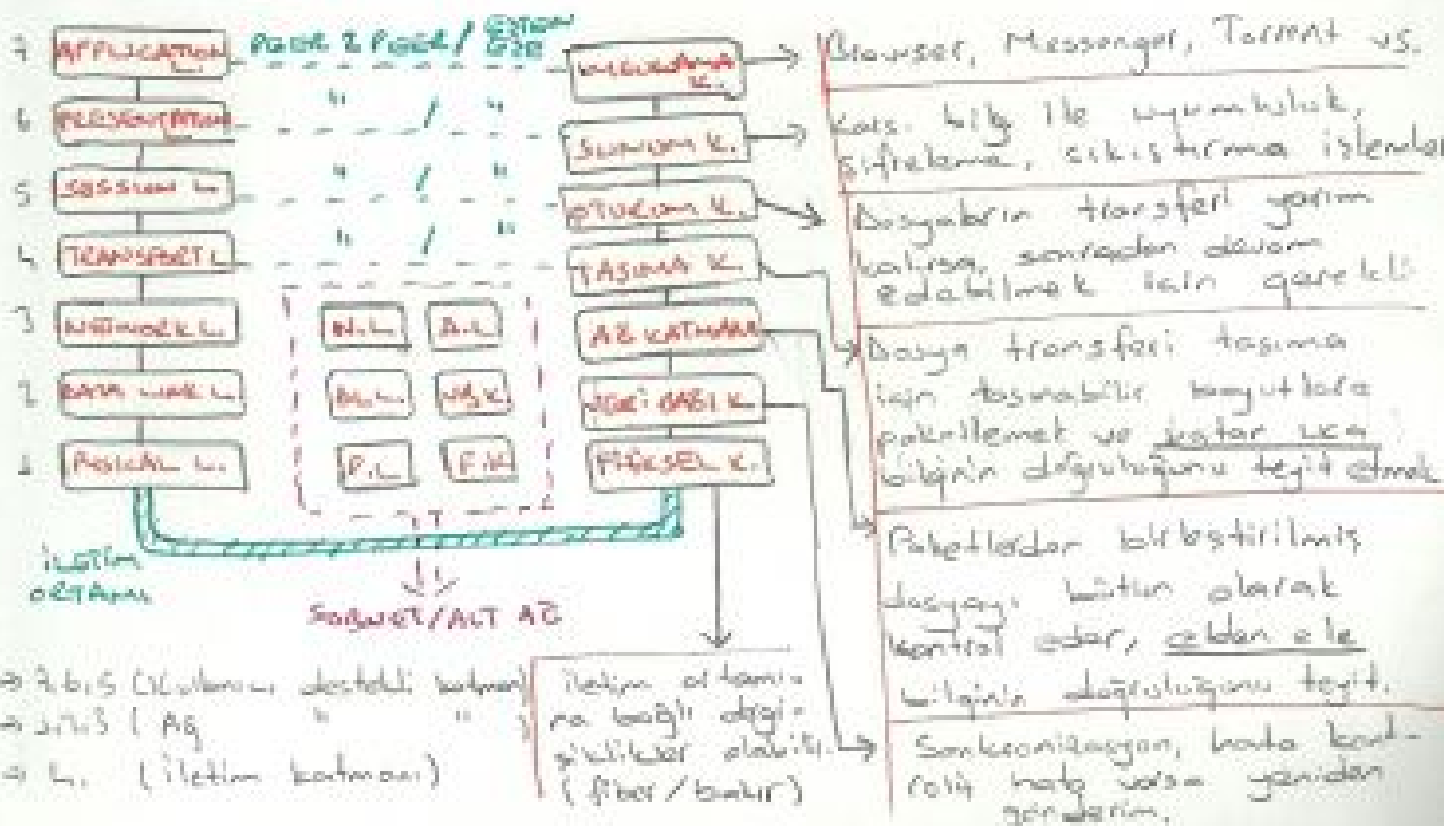
3. FULL DUPLEX: 

YAYINLAMA (BROADCAST) $\left\{ \begin{array}{l} \text{BROADCAST (HERKESE YAYINLAMA)} \\ \text{MULTICAST (ÇOKLA YAYINLAMA)} \\ \text{UNICAST (TEKE GÖNDERİM)} \end{array} \right.$

- SİMETRİK $\begin{array}{c} \xrightarrow{C_1} \\ \xleftarrow{C_2} \end{array}$ ($C_1 = C_2$) \rightarrow Giden, gelen kapasite eşit

- ASİMETRİK $\begin{array}{c} \xrightarrow{C_1} \\ \xleftarrow{C_2} \end{array}$ ($C_1 \neq C_2$) \rightarrow Genelde ekranda kullanılan bağlantı esastır.

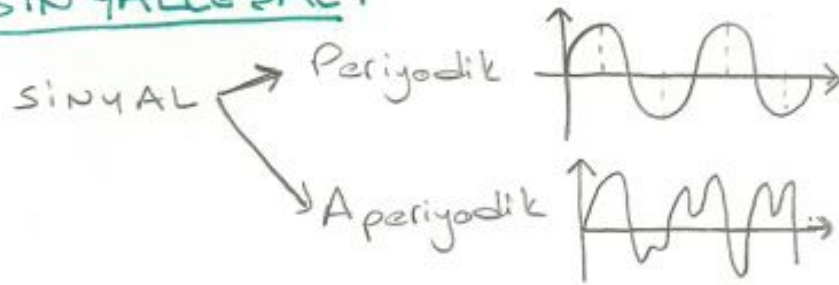
- ISO'NUN OSI MODELİ (Open Systems Interconnection) \rightarrow 7 Katmanlı.



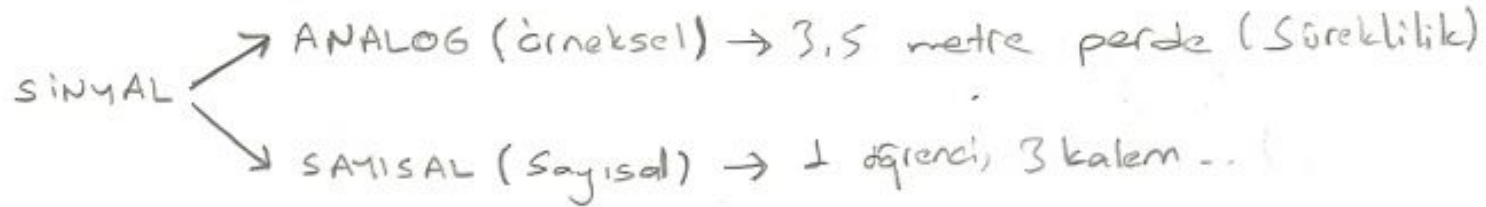
NOT: Uzak mesafelerde iletilen dosyalar başka bilgisayarlara ağlarını aracı olarak kullanır. [SUBNET]

- DATA LINK KATMAN seviyesinde IP adresi ile iletişim yapılabilir. (Aynı switch'e bağlı bilgisayarlar arası)

-SINYALLEŞME:

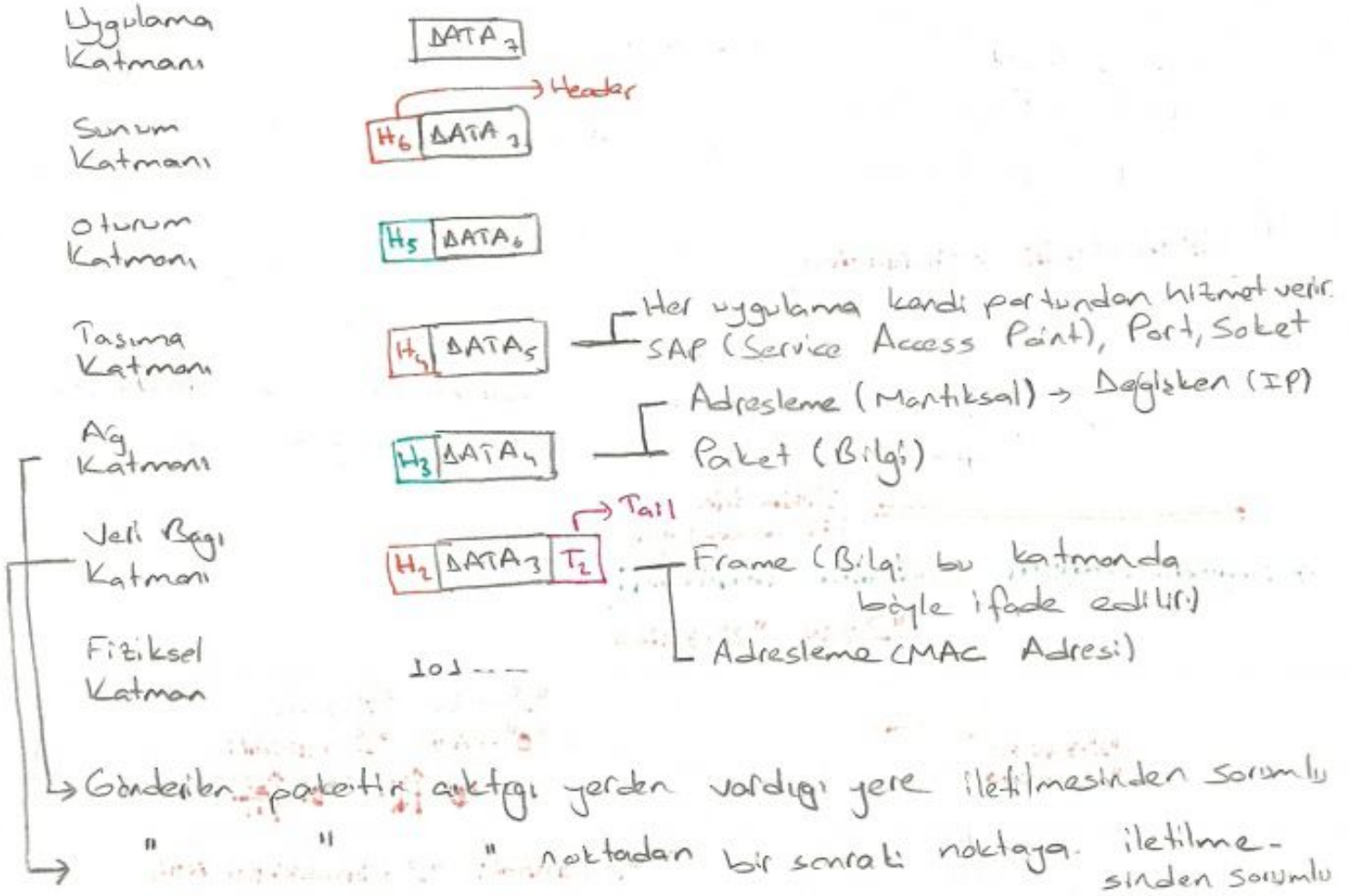


Muhtelif periyodik sinyallerin toplamı
(Fouriertrans)



→ Analog sinyali değiştirmek için; genlik, frekans ($1/\text{Periyot}$), faz değiştirilebilir.

→ Sayısal sinyallerde; bit interval ve bit per second önemlidir.



Bir Veri Hattı Üzerindeki Maksimum Veri İletim Hızı

1-) NYQUIST Formülü:

$$\text{Max Veri Hızı} = 2 \cdot (H) \cdot \log_2 \text{ bit/sec}$$

→ Ayırık Seviye Sayısı (Veriyi ifade etmek için kullanılacak farklı durum)
↓
+1 için +5V
-1 için 0V

→ Bant Genişliği (Hz)

→ Gürültüsüz hatlar için.

2-) SHANNON - HARTLEY Formülü:

$$\text{SNR} = 10 \cdot \log_{10} \frac{S}{N} \text{ dB} \rightarrow \text{Signal to Noise Ratio}$$

$$\text{Veri Hızı} = H \cdot \log_2 (1 + S/N) \text{ bit/sec}$$

İletim ortamının SNR değeri 24 dB ise 3kHz ile 4kHz arasında çalışan iletim sistemi için Max hız ve kaç ayırık seviye (v) olabileceğini hesaplayınız.

Örnek: Bant genişliği = 1 kHz = 1000 Hz

$$24 = 10 \cdot \log_{10} \frac{S}{N}$$

$$\frac{S}{N} = 251$$

$$1000 \cdot \log_2 (1+251) \approx 256 \rightarrow \underline{8000} \rightarrow \text{Max hız (bit/sec)}$$

$$8000 = 2000 \cdot \log_2 V$$

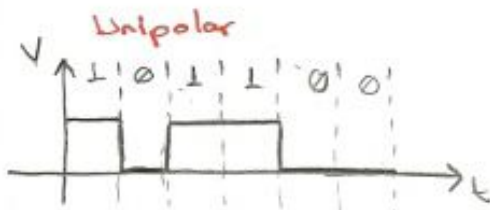
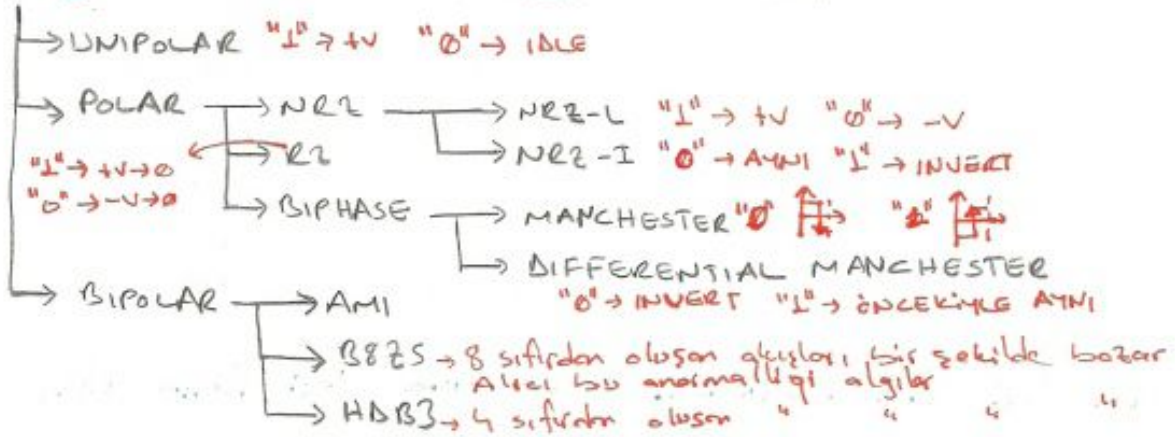
$$V = 4 \rightarrow \text{Ayrık seviye sayısı}$$

SİNYALLERİN KODLANMASI

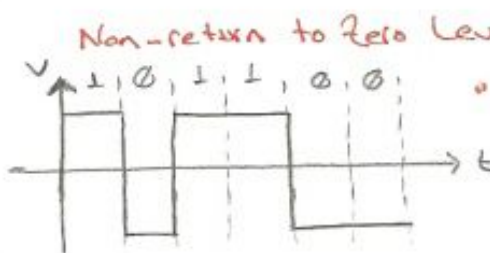
VERİCİ	ALICI
Sayısal	Sayısal
Sayısal	Analog
Analog	Sayısal
Analog	Analog

Bitler için
bir anlamı yok

→ Sayısal-Sayısal Kodlama Teknikleri:

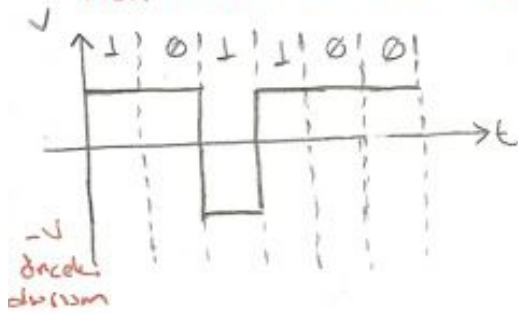


- DC bileşen sikintisi (hat üzerinde pozitif voltaj birikimi)
- Senkronizasyon (devamlı +5 volt gelirse bunun kaç bit olduğunu anlamak zor olur)



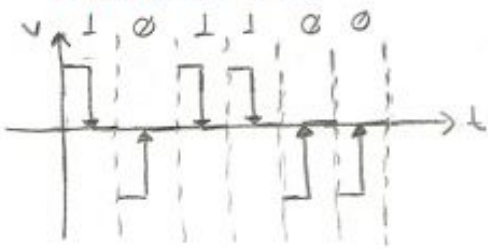
- DC bileşen sikintisi (hat üzerinde pozitif veya negatif voltaj birikimi)
- Senkronizasyon

Non-return to Zero-Inverted



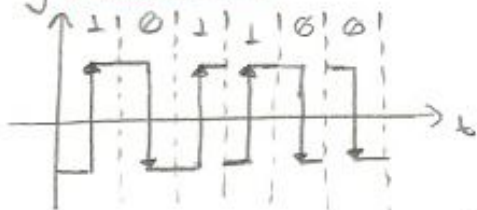
- DC bileşen sikintısı (Hepsi sıfırda voltaj birikimi)
- Senkronizasyon (hepsi sıfırda sorun)

Return-to-zero



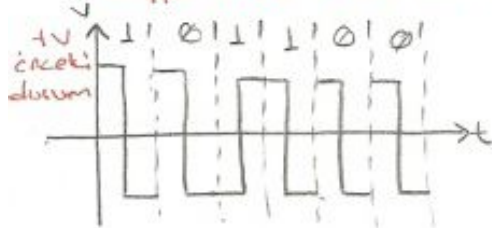
- + Senkronizasyon problemi: azdır bu
- DC bileşen sorunu (hepsi 1 veya 0 iken)
- Gerçek hayatta veri iletimi için mevcut frekansın ancak yarısı kullanılabiliyor.

Manchester

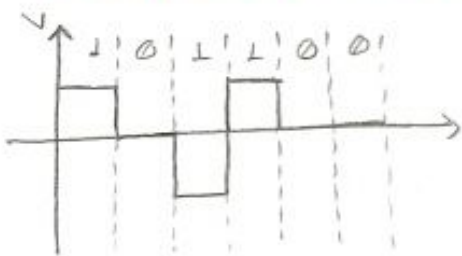


- + DC bileşen sorunu azdır bu
- + Senkronizasyon sorunu yok.

Differential Manchester

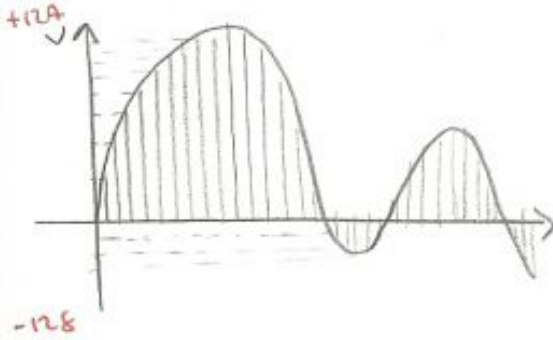


Alternate Mark Inversion (AMI)



- 0 geldimmi, birşey yok (Idle)
- 1 " polariteyi değiştir.
- (Tek sayıda 1 için pozitif yön, çift sayıda 1 için negatif yön.)
- Senkronizasyon sorunu (hepsi sıfır olursa)
- ↳ Bu problemleri hareketle B8ES ortaya atılıyor.

Analog - Sayısal Kodlama



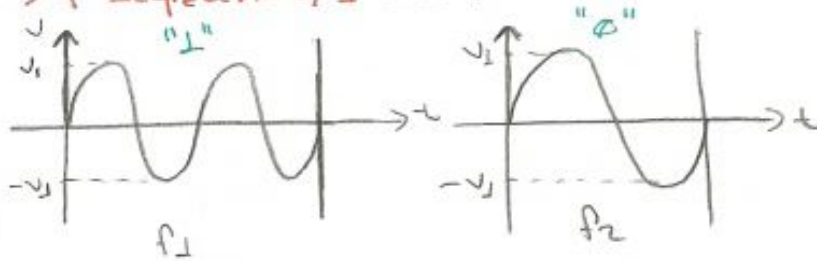
1. PAM (Pulse Amplitude Modulation)
2. PCM (Pulse Code Modulation)
 - Grafiği küçük parçalara böl
 - Belirli parçaların genişliğini ard arda diz.
3. Elde edilen rakamları binary'ye çevir.

Not: Analog işaretin bölünmesi parçaların frekansı işaretin maksimum frekansının en az 2 katı olmalı.

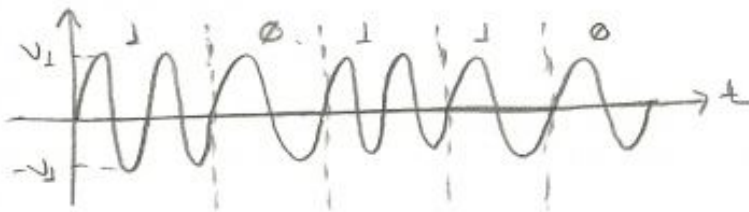
Sayısal - Analog Kodlama

- 1.) $f = (1/T)$ FREKANS
- 2.) V GERİLİM
- 3.) Φ FAZ

→ f Değişken - V, Φ Sabit

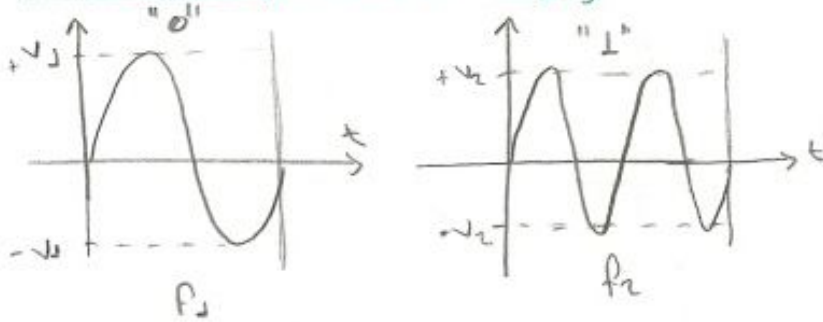


$$f_1 > f_2$$



FSK (Frequency Shift Keying)

1-) FSK (Frequency Shift Keying)

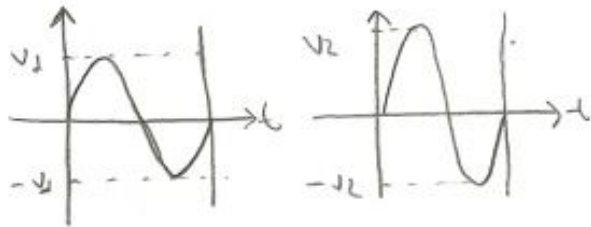


$$f_1 < f_2$$

$f \rightarrow$ Değişken

$V, \phi \rightarrow$ Sabit

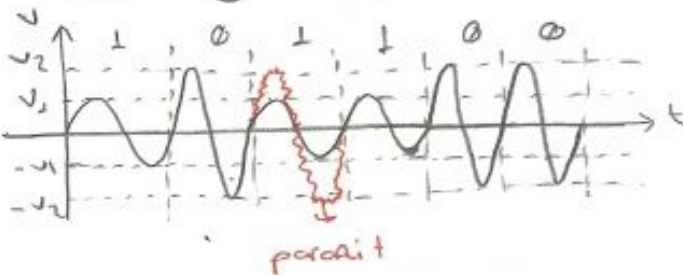
2-) ASK (Amplitude Shift Keying)



$V \rightarrow$ Değişken

$f, \phi \rightarrow$ Sabit

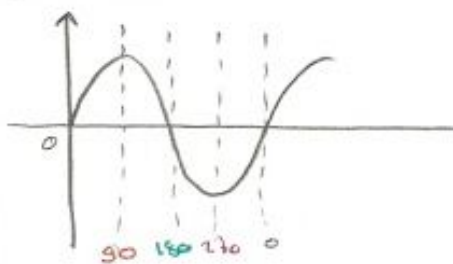
- Bu dünyadaki parazitlerden etkilenebilir.



$$\text{BAUD} = 6 \text{ Gbit/sec}$$

BAUD = Veri gönderilirken kullanılan farklı sinyal (dalga) sayısı. Bir baud ile kaç bit gönderebildiğin, hızını belirler

3-) PSK (Phase Shift Keying)

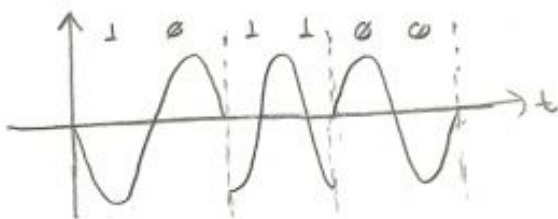


0 - 00
90 - 01
180 - 10
270 - 11

$\phi \rightarrow$ Değişken

$V, f \rightarrow$ Sabit

+ Bir seferde 2 yerine 4 ayrıık durum (0, 90, 180, 270) belirtilebilir. Buda tek seferde tek signalle $\log_2 4 = 2$ bit gönderebilmektir.

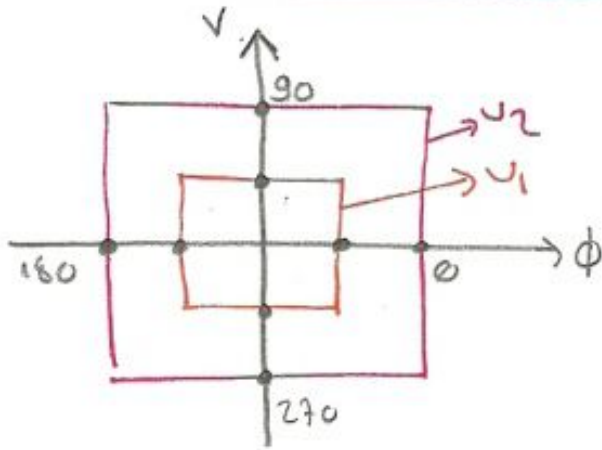


$$\text{BAUD} = 3 \text{ Gbit/sec}$$

Veri İletimi

② 13.45

QAM (Quadrature Amplitude Modulation)



$\Phi, \psi \rightarrow$ Değişken
 $f \rightarrow$ Sabit

8 farklı durum ($\log_2 8 = 3$ bit)

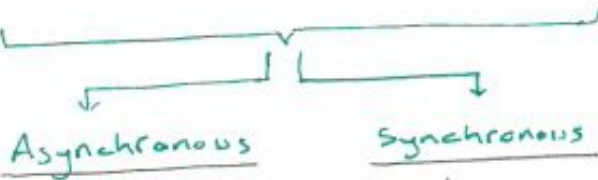
→ Bir seferde en fazla 8 bit gönderebiliriz
bu yöntemle. (1 baud ile 8 bit)

SERİ HABERLEŞME

- Adı seri olsa da birden fazla tel/bağlantıyla yapılabilir.
- Uzun mesafelerde seri haberleşme iyi.
- Az sayıda kablo var. (USB'de 4 tel)

PARALEL HABERLEŞME

- Kullanılan hat sayısı seriye göre çok daha fazladır. (Paralel portlarda 25 tane hat var, bunların 8 tanesi veri iletimini için kullanılır.)
- Belli bir sınırı var, mesafeyi çok uzak tutamazsın (max 25 ft. 7-8 metre)

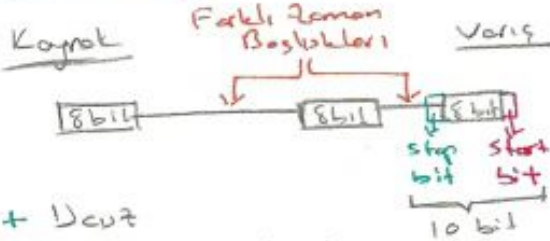


Asynchronous

Birimler arasındaki çalışmalar aynı zaman diliminde aynı sekilde değil

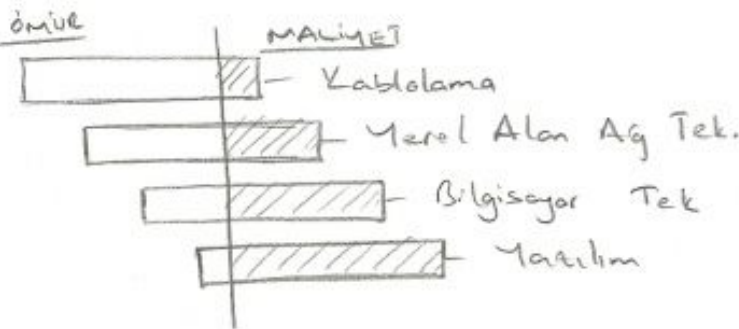
Synchronous

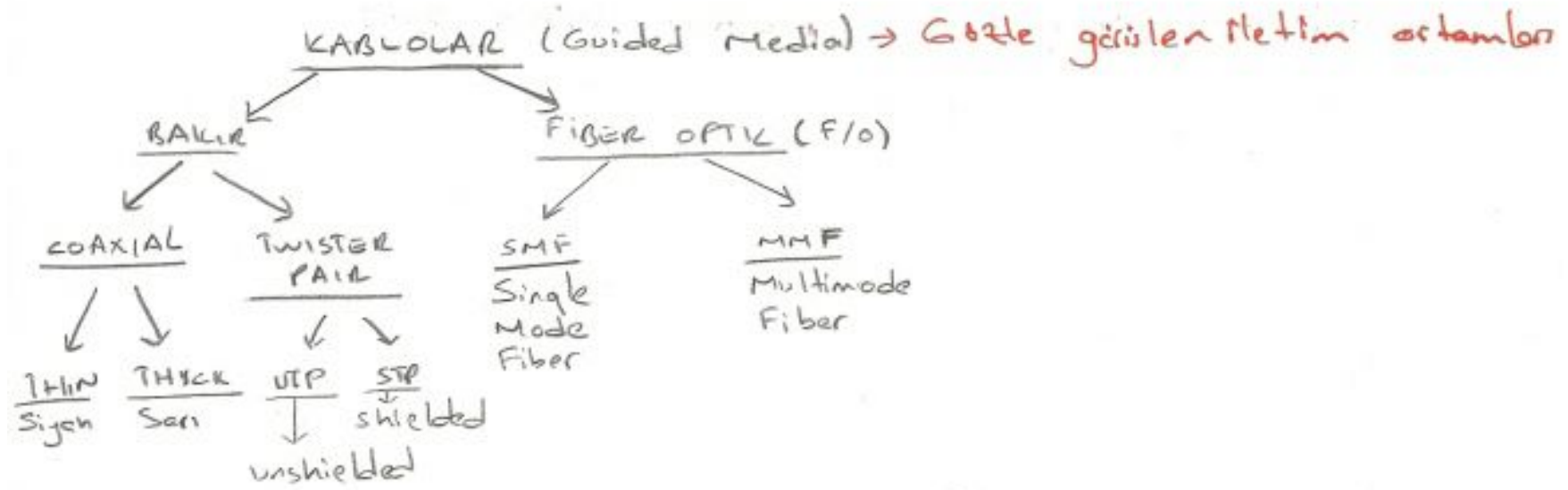
Birimler arasındaki çalışma aynı anda başlar, aynı anda biten

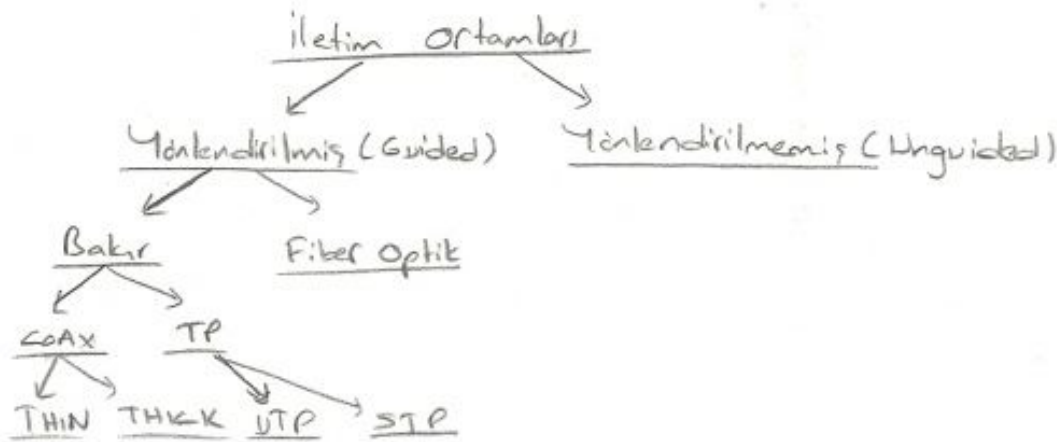


- + Düz
- + Çok karmaşık değil
- Yavaş

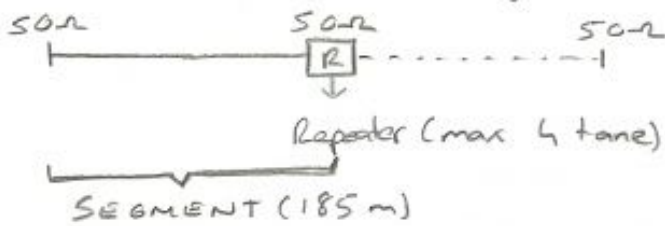
- + Daha hızlı
- Frame'lerin start, stop işaretleri vardır.







→ Bus topolojisinde ağ genişletmek (uzatmak) için uzatılacak her "aktif cihaz" bağlanır (maximum 4 tane)

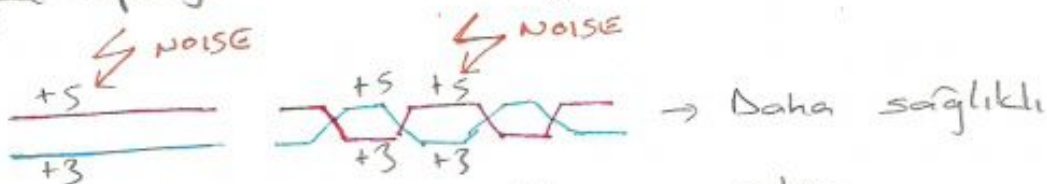


kalite düşer, mesafe problemi, dış etkenler, sinyal zayıflaması

→ Kalın kablo kullanıldığında (THICK ethernet kablo) 185m lik segment mesafesi 500 metreye çıkar.

Not: İstasyon sayısındaki artış, hat üzerindeki tek yönlü veri iletişimine yük olur. (kalite kaybı)

→ Ağac topolojisinde TP (burgulu) kablolar kullanılır. (Twisted Pair)



• UTP ve STP olarak ikiye ayrılır.

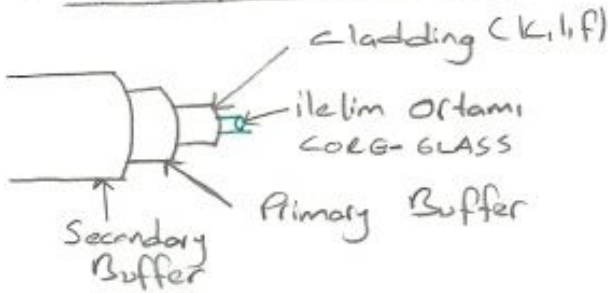
↓

Etrafı bakır örgüyle sarılmamış

Etrafı örgülenmiş

CAT-1	CAT-2	CAT-3	CAT-4	CAT-5	CAT-5e	CAT-6	CAT-6a	CAT-7
Ses	Data	Data	Data	Data	Data	Data	Data	Data
	4 KHz	16 KHz	20 KHz	100 KHz	>100 KHz	250 KHz	>250 KHz	600 KHz
				↓ 10/100 Mbps	↓ 100/1000 Mbps	↓ 10 Gbps	↓ 10 Gbps	

→ Fiberoptik Kablo:



	SMF	MMF	MMF	MMF	MMF
CORE	5-10µm	50µm	62.5µm	85µm	100µm
CLADDING	125µm	125µm	125µm	125µm	140µm
ATTENUATION (Zayıflama)	0.8db/km	3-4db/km	3-6db/km	5db/km	5-6db/km
Frekans	>1000 MHz	>400 MHz	160 MHz	200 MHz	10-100 MHz

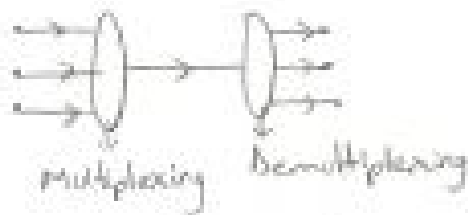
Kablosuz İletim Ortamları

Yönlendirilmemiş İletim Ortamları

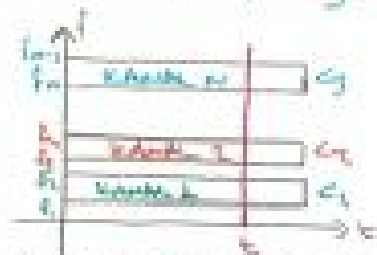
IR (INFRA RED) { ISM (INDUSTRIAL SCIENTIFIC MEDICAL-KULLANIMI ACIK)
RF (RADIO FREKANSI) { UHF/VHF

	UTP	STP	COAX	FO	RF	IR
FİYAT	UCUZ	ORTA	ORTA	Pahalı	ORTA	UCUZ
HIZ	5 Gbps	150 Mbps	1 Gbps	10 Gbps	10-50 Mbps	4 Mbps
SİNYAL ZAYIFLIĞI	YÜKSEK	YÜKSEK	YÜKSEK ORTA	Düşük	Düşük ORTA	Düşük ORTA
EMI	YÜKSEK	ORTA	ORTA	YOK	YÜKSEK	YÜKSEK ORTA
GÜVENLİK	Düşük	Düşük	Düşük	YÜKSEK	Düşük	Düşük
DÜĞÜM EKLEME	KOLAY	KOLAY	KOLAY	ZOR	KOLAY	KOLAY
MESAFE	KISA	KISA	ORTA	UZUN	↗	GÖK KISA

EMI: Bu ortamdan gelen elektromanyetik gürültünün iletim ortamına etkisi.

MULTIPLEXING

1. FDM (Frequency Division Multiplexing)



→ Herhangi bir t anında bütün kanallar yayındadır. Ate. hangi frekansa getirilirse o frekansa göre kanaldan yayın alınır.

→ Her kanalın yayın karakteristikliği birbirine eşittir. (Herisi eşit kapasiteye sahip)

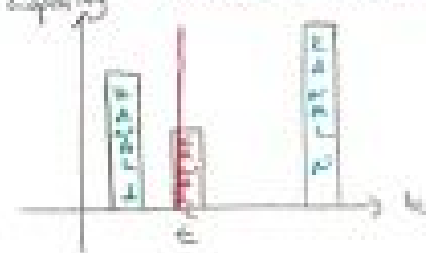
$$C \geq \sum_{i=1}^n C_i$$

→ Herhangi bir t anında 1 kanal yayında

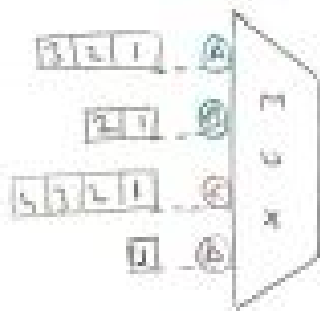
→ Yayın karakteristikleri aynı değildir. Kanalların kapasiteleri birbirinden farklı

$$C \geq \text{MAX } C_i$$

2. TDM (Time Division Multiplexing)



→ Sıra verilen zamanı ister kullan ister kullanma, her seferinde bir shuttle gönderilir, 0 an ne varsa gönderilir. Shuttle tamamen dolmak zorunda değil.



→ Zamanı geldiğinde boş kulan = tutur mantığı

- Toplam 16 SLOT kullanıldı
- 6 SLOT boş gitti



- Toplam 12 SLOT kullanıldı
- 2 SLOT boş gitti

→ Etkisizden inderen bilgisi taşınmalı, (Beyisler trafik, kimin nerede geldiği belli değil)

+ İlaflı, ayrılmazlık

- Fazla slotlar boş gitti
+ Bilgiler düzenli gidiyor.
(Ayrıştırma kolay)

HATA ALGILAMA VE DÜZELTME

- Data Link Layer'in görevidir. (frame'in bir noktadan diğer noktaya sorunsuz gitmesini sağlar)
- Verici bilgiyi yollarken frame'in sonuna tail koyulur. (Alıcının hata olup olmadığını anlayabilmesi için.) Hata olup olmadığının anlaşılması için ek bilgi göndermek gerekir.

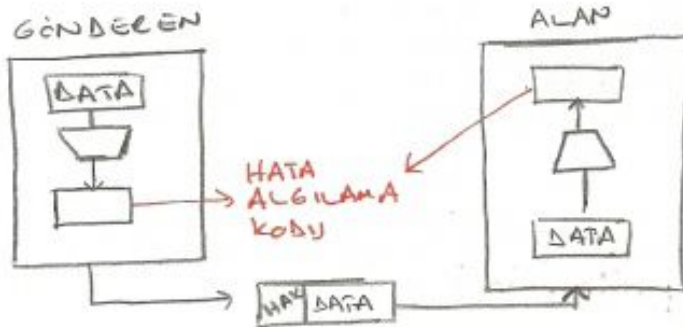
Hata Kaynakları

- Signal taşıflaması
- Gecikme (Bozuk senkronizasyon)
- Ortamdaki gürültü
 - Termal Gürültü
 - CrossTalk (Kıpraz Karusma)
 - Impulse Noise (Aşırı Gürültü)
 - Modülasyonlar Arası Gürültü (FDM vs.)

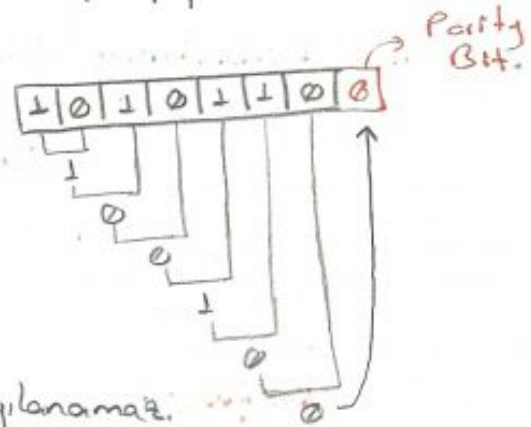
- TEK BİT HATA
- ÇOK BİT HATA → Verinin farklı yerlerinde olabilir.
- HATA PATLAMASI (BURST ERROR) → Birbirini takip eden bir blok boyunca oluşan hatalar

HATA ALGILAMA TEKNİKLERİ

→ Parity (VRC - Vertical Redundancy Check)



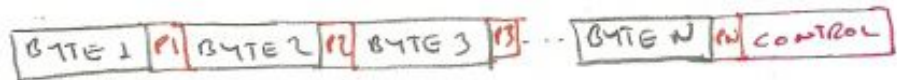
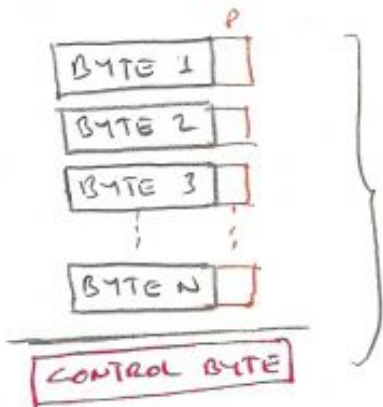
A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0



→ Data içerisindeki 1'lerin sayısı ile alakalıdır. Tek bitlik bir hata varsa ise farkı.

- Data'da çift sayıda hata varsa algılanamaz.

→ LRC (Longitudinal Redundancy Check) (iki bayutlu parity) Eline bayutna XOR alınıyor



1	0	0	1	1	1	0	0
1	1	1	0	0	0	1	0
0	1	0	1	0	1	0	1
0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	1

Aynı sonuç
hata yok

+ Aynı byte üzerinde çift hata olduğunda algılıyor.

- Çift sayıda, farklı iki byte'in aynı bitlerinde hata varsa hata algılanmaz.

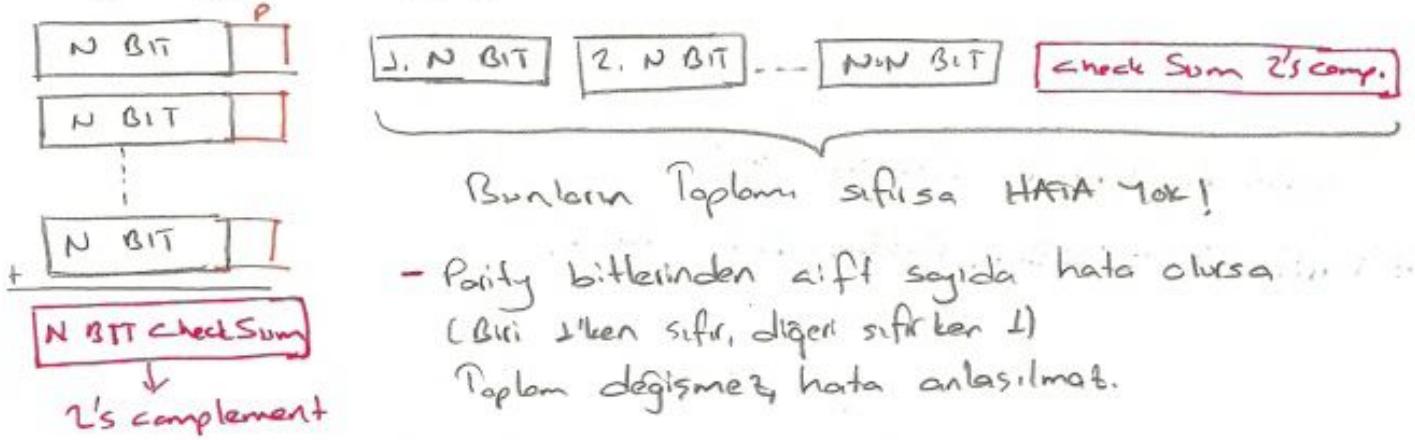
1	0	0	1	1	1	0	0
1	0	1	0	0	0	1	0
0	1	0	1	0	1	0	1
0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	1

Yeşil renkli olanlar hatalı bitler

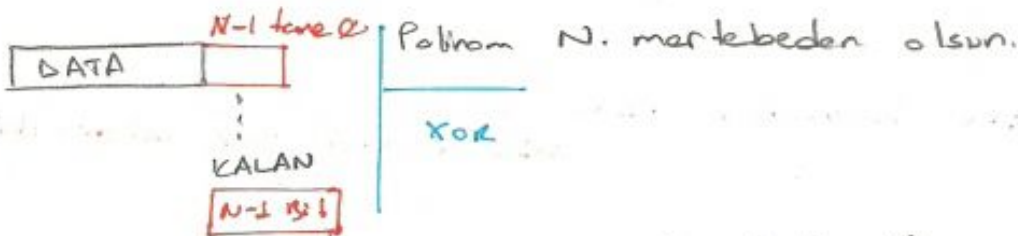
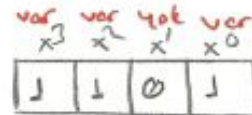
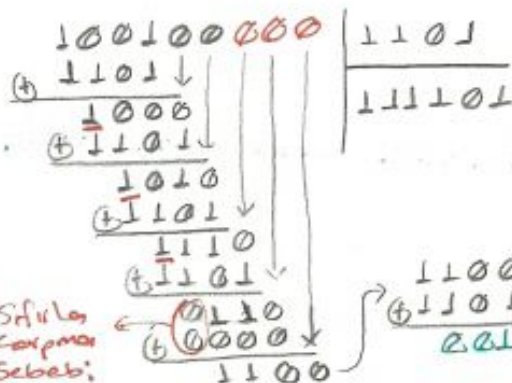
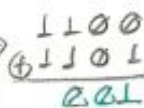
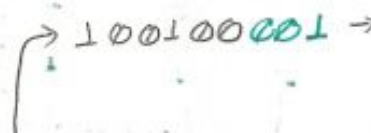
Hata yok gibi görünüyor
ama 4 tane hata var.

→ CHECK SUM

Veriyi segmentlere ayırıyor.



→ CRC (Cyclic Redundancy Code)

Görünbirene ayrılmayan polinomlar kullanılır.
(asal polinomlar)Asal polinoma bölme işlemi XOR kullanarak yapılır.
Kullanılacak polinomun mertebesine bağlı olarak kontrol
biti bretilir.→ Verinin sonuna bu kalan ifadeyi koyarak
gönderilir. Karşı taraf bu bölme işi
yaptığında Kalan = 0 ise HATA YOK!Ö/ $x^3 + x^2 + 1$ polinom olsun.
100100 gollanacak veriSoyunun sonuna
 $4-1=3$ tane 0 ekleSıfır
Carpma
Sebebi:Bu verinin
sonuna ekle
gönderBu 1101 ile
bölünürse sonuna
0 çıkacak

HATA YOK!

HAMMING CODE

→ Tek bit hatayı algılama ve düzeltme yapıyor.

m bit veri iletilecek (m ayrı bitte hata olabilir)

↓
m+1 ayrı durum

⊕
hatasız iletilme durumu

$$R = \log_2^{m+1} \rightarrow \text{hata algılama için kullanılacak (eklenecek) bit sayısı}$$

↓
Bu kontrol bitlerinde hatalı olabilir (4 durum daha)

$$R \geq \log_2 (m+R+1)$$

Ö/ m=7

$$R \geq \log_2 (8+R) \rightarrow R, 3'ten büyük olmalı. (4)$$

$$R=4, m=7, \text{hatasız olma durumu} = 1$$

$$4+7+1 = 12 \text{ ayrı durum (bit)}$$

	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0
0	0	0	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0
2	0	0	0	1	0	0	0	0
3	0	0	0	1	1	0	0	0
4	0	0	1	0	0	0	0	0
5	0	0	1	0	1	0	0	0
6	0	0	1	1	0	0	0	0
7	0	0	1	1	1	0	0	0
8	0	1	0	0	0	0	0	0
9	0	1	0	0	1	0	0	0
10	0	1	0	1	0	0	0	0
11	0	1	0	1	1	0	0	0

$$R_1 = a_7 \oplus a_6 \oplus a_5 \oplus a_4 \oplus a_3 \oplus a_2 \oplus a_1 \oplus a_0$$

$$R_2 = a_7 \oplus a_6 \oplus a_5 \oplus a_4 \oplus a_3 \oplus a_2 \oplus a_1 \oplus a_0$$

$$R_3 = a_7 \oplus a_6 \oplus a_5 \oplus a_4 \oplus a_3 \oplus a_2 \oplus a_1 \oplus a_0$$

$$R_4 = a_7 \oplus a_6 \oplus a_5 \oplus a_4 \oplus a_3 \oplus a_2 \oplus a_1 \oplus a_0$$

$$a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0, R_1, R_2, R_3, R_4$$

a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0	R_1	R_2	R_3	R_4
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

→ Eğer 2'inci kontrol bitinde hata varsa 1'inci bitte hata vardır.

$$a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0, R_1, R_2, R_3, R_4$$

0	1	1	0	0	0	1	1	1	0	1	1
---	---	---	---	---	---	---	---	---	---	---	---

→ Bu şekilde gönderiliyor.

$$R_1 = 0 \oplus 1 \oplus 1 \oplus 0 \oplus 0 \oplus 0 \oplus 1 \oplus 1 \oplus 0 = 1$$

$$R_2 = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 0 \oplus 0 \oplus 1 \oplus 0 = 0$$

$$R_3 = 0 \oplus 1 \oplus 0 \oplus 0 \oplus 0 \oplus 0 = 1$$

$$R_4 = 0 \oplus 1 \oplus 0 \oplus 1 \oplus 0 = 0$$

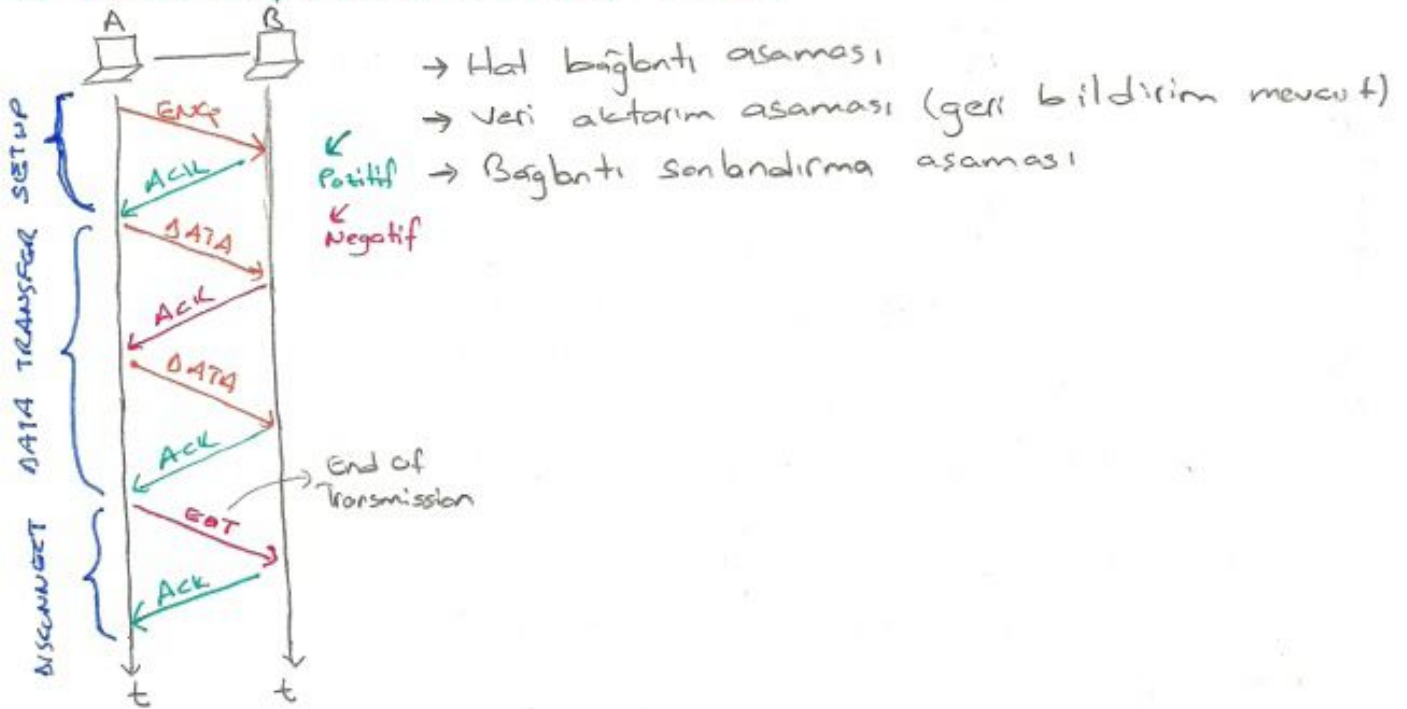
Not: Alıcı taraf R_1, R_2, R_3, R_4 'ü formüle göre hesaplayıp hepsini

0 buluyorsa HATA YOK!

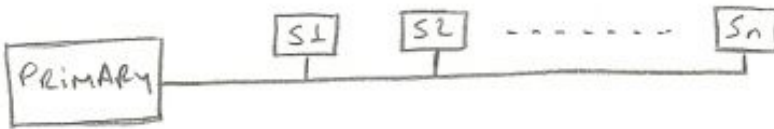
Hepsi 0 değilse $(R_1 R_2 R_3 R_4)_2 = (X)_{10}$ bulunan X rakamı hatanın kaçıncı byte'da olduğunu gösterir.

DATA LINK LAYER (Özellikleri)

1) ENQ/ACK (ENQUIRY/ACKNOWLEDGE)



2) POLL / SELECT (Multipoint)



→ İletimde bulunmak için primary ile iletişime geçmek gerekir.

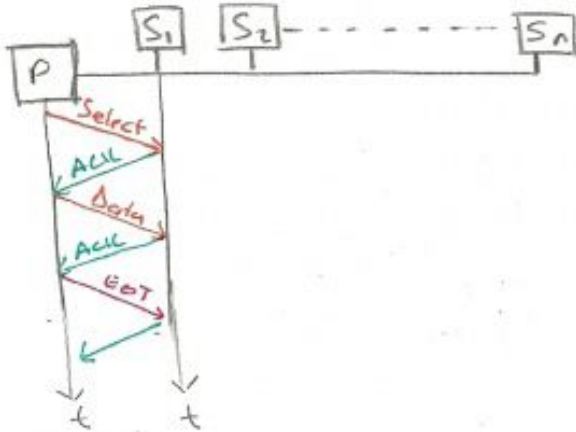
Primary'e mesaj gönderirken kendi adresi eklenmeli

→ Adres mekanizması olmalı (primary için)

Primary mesaj göndereceği zaman göndermek istediği adresi eklemeli.

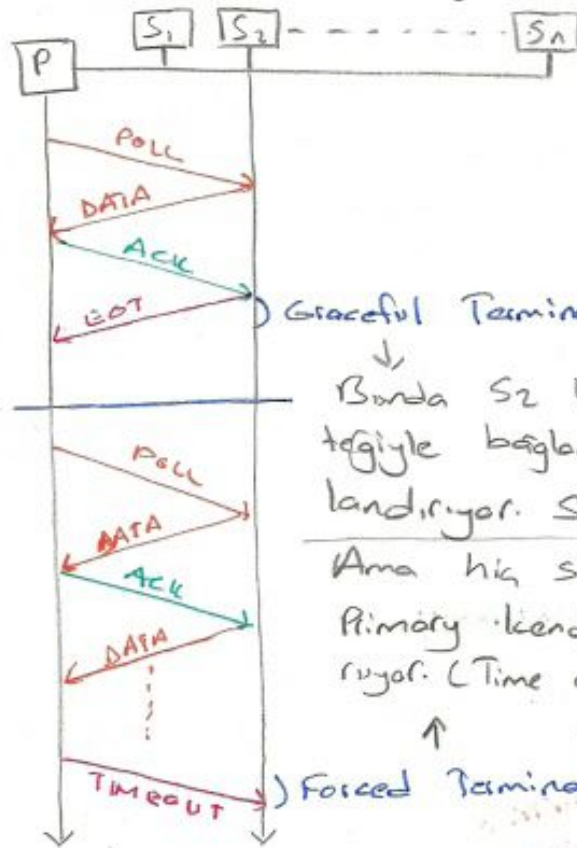
→ SELECT (P → S)

Primary'den Secondary'e data



→ PULL (S → P)

Secondary'den Primary'e data



Graceful Termination

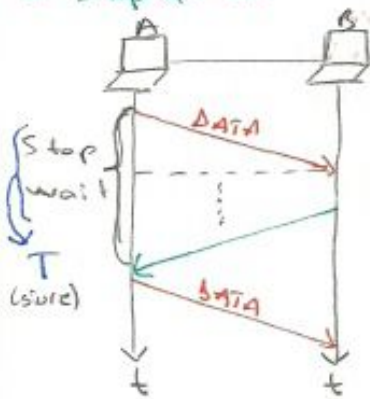
↓
Bunda S2 kendi isteğiyle bağlantıyı sonlandırıyor. Sorun yok. Ama hiç sızmasa, Primary kendi sistemiyor. (Time out)

↑
Forced Termination

Flow control (Akış Kontrolü)

→ Bağlantının sorunsuz sağlanması ve sorunsuz olarak sonlandırılması kabul ediliyor. Sadece veri aktarım katmanını ele alıyoruz.

- stop & wait



→ Bilgi gönderdikten sonra mutlaka olumlu/olumsuz bir geri besleme gelmesini bekliyoruz.

Utilization → geri dönüş zamanı

t frame → Verinin bilgisayardan iletim hattına çıkma zamanı

t prop → Verinin hat üzerindeki yokluk zamanı

t ack → Ack. için gerekli t frame'in hat üzerine çıkması için gereken zaman (kayıp taraf dahil.)

$$T = t_{\text{frame}} + t_{\text{prop}} + \underbrace{t_{\text{ack}}}_{\substack{\text{0 kabul} \\ \text{edilebilir}}} + t_{\text{prop}}$$

$$T = t_{\text{frame}} + 2 t_{\text{prop}}$$

$$\text{Utilization} = \frac{t_{\text{frame}}}{t_{\text{frame}} + 2 t_{\text{prop}}} \quad \left\{ \begin{array}{l} 1'den \text{ küçük her zaman (kısıtlı)} \\ \text{Bunu artırmak için çeşitli} \\ \text{teknikler var.} \end{array} \right.$$

$$a = \frac{t_{\text{prop}}}{t_{\text{frame}}} \quad \text{kabul edilirse;}$$

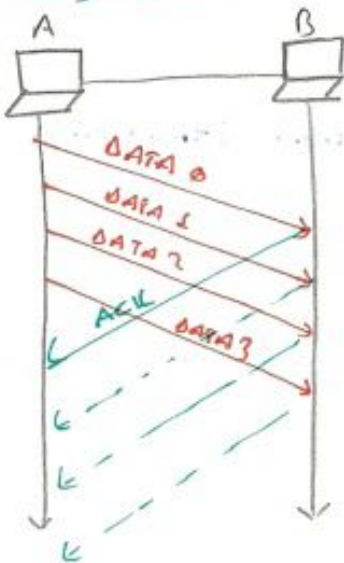
↓
Sabit

$$U = \frac{1}{1+2a}$$

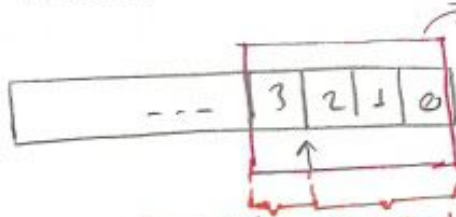
$$t_{\text{prop}} = \frac{\text{mesafe}}{\text{hız}} = \frac{d}{v}$$

$$t_{\text{frame}} = \frac{\text{frame size}}{\text{data rate}} = \frac{L(\text{length})}{R(\text{rate})} \quad \left\{ \begin{array}{l} \text{birimler} \\ \text{aynı} \\ \text{olmalı} \end{array} \right.$$

- Sliding window



→ Geri besleme gelene kadar bes durulmaz. Data yollanmaya devam edilir. Yollanan datalarda bir problem olma ihtimaline karşı dataların geteleşi (potholes) tutulur.



(potholes)
Çevre geldikçe
pencere sola
doğru kayar.

Window
gösterilme-
miş

Gönderilmiş
ama henüz
bir cevap yok
(Akıbeti bilinmiyor)

net %100 verimle
↑
kullanılır

$$U = \begin{cases} 1 & w \geq 2at+1 \\ \frac{w}{2at+1} & w < 2at+1 \end{cases}$$

$w=1$ olduğu zaman Stop & Wait olur