Sivas Cumhuriyet Üniversitesi Bilgisayar Ağları Dersi

Bölüm 3
Veri Bağı Katmanı

Halil ARSLAN

Bölümün hedefleri

- Veri bağı katmanının işlevlerini anlamak
- Çerçeveleme
  - Ethernet frame
- Akış denetimi
  - Sliding window protocols
- Hata bulma
  - Parity, CRC, Checksum
- Hata düzeltme
  - Hamming, ARQ
- Ortam erişim kontrolü (MAC)
  - CSMA/CD, Token-Ring, CSMA/CA
- Bu katmanda çalışan aygıtları tanımak
  - Switch, bridge

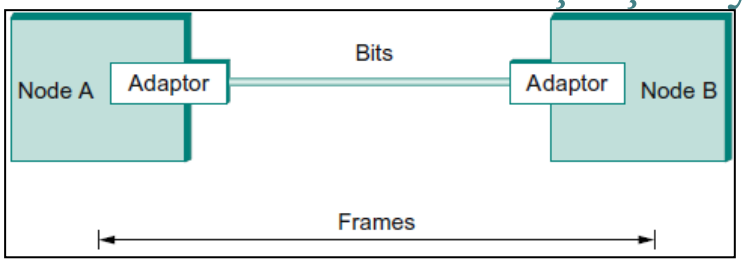
Veri Bağı Katmanı

OSI referans modelinin ikinci katmanı olan veri bağı katmanı;

- Verileri çerçevelemek ve adreslemek,
- Akış denetimi
- Hata denetimi,
- Hata düzeltme,
- Ortam erişim denetimi yapar.

Çerçeveleme(framing)

- Fiziksel katmanda veri bitler halinde kaynaktan hedefe gönderilirken, veri bağı katmanı bitleri "**frame**" lere böler.
- Çerçeveleme sayesinde bir kaynaktan hedefe giden ya da farklı kaynaklardan başka hedeflere giden mesajlar ayrıştırılır.
- Her mesaj tek bir çerçeve olabilir. Bu durumda;
  - Çok büyük bir çerçeve oluşur
  - Hata ve akış kontrolü zor olur
  - Oluşacak tek bir bit hatası, tüm mesajı etkiler
- Genelde, her mesaj birden çok çerçeveye bölünür (fragmentation).
  - Oluşacak tek bir bit hatası sadece bir çerçeveyi etkiler



Çerçeveleme(framing)

1- Sabit Çerçeve Boyutu

– Frame'ler için ayraçlar kullanmaya gerek yoktur. Örnek: ATM (48 byte)

2- Değişken Çerçeve Boyutu

– Frame'in başlangıç ve bitiş noktalarını belirlemek için ayraçlar kullanılmalıdır

– İki ana yaklaşım vardır. Karakter temelli yaklaşım ve bit temelli yaklaşım. Örnek: Ethernet2 (64-1518 byte)

(1500B payload + 14B Ethetnet Header + 4B FCS)

Çerçeveleme(framing)

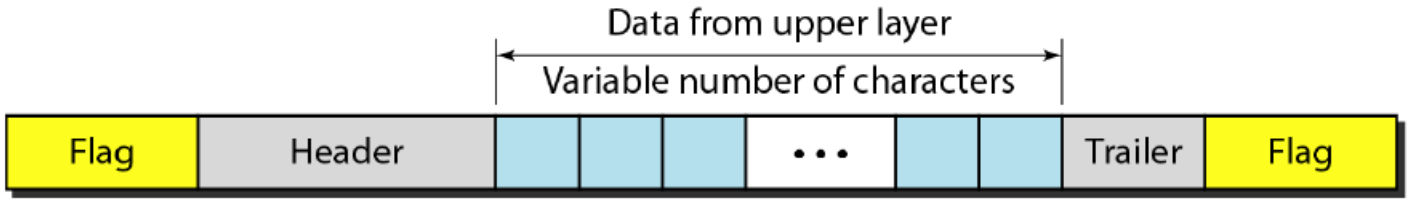
Karakter Temelli Çerçeveleme

• Veri karakterler olarak taşınır (ASCII 8-bit)

• Header (Başlık) kaynak ve hedef adreslerini içerir, trailer (kuyruk) hata kontrol bilgisini içerir. Hepsinin boyutu 8 bit ve katlarıdır.

• Frameleri birbirinden ayırmak için frame'in başına ve sonuna 8 bitlik "flag"ler eklenir

– Flaglerin içeriği protokole göre değişir



Çerçeveleme(framing)

Karakter Temelli Çerçeveleme

• Karakter temelli framing sadece metin türü verinin transfer edildiği dönemde popülerdi

• Flag text içinde kullanılmayan herhangi bir karakter olarak seçilirdi

• Günümüzde diğer veri türleri de gönderilmektedir

• Her karakter verinin içinde bulunabilir

• Alıcı frame'in sonuna geldiğini düşünebilir

• Önlemek için byte stuffing (byte doldurma) yaklaşımı karakter temelli framing ile birlikte kullanılır

Çerçeveleme(framing)

Bit Temelli Çerçeveleme

• Bir çok protokolde 8-bitlik 01111110 flag'i frame'in başlangıcını ve sonunu belirtmek için kullanılır

• Karakter temelli yaklaşımda karşılaşılan problemler görülebilir

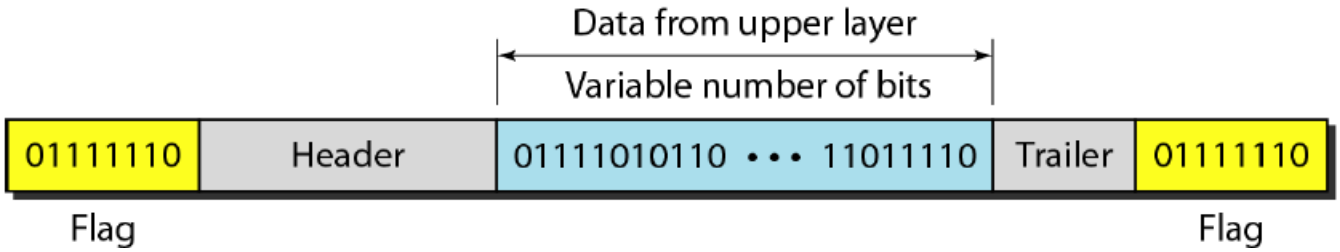
• Burada dolgu 1 karakter yerine sadece 1 bit ile yapılabilir

• Bu strateji bit dolgusu olarak bilinir

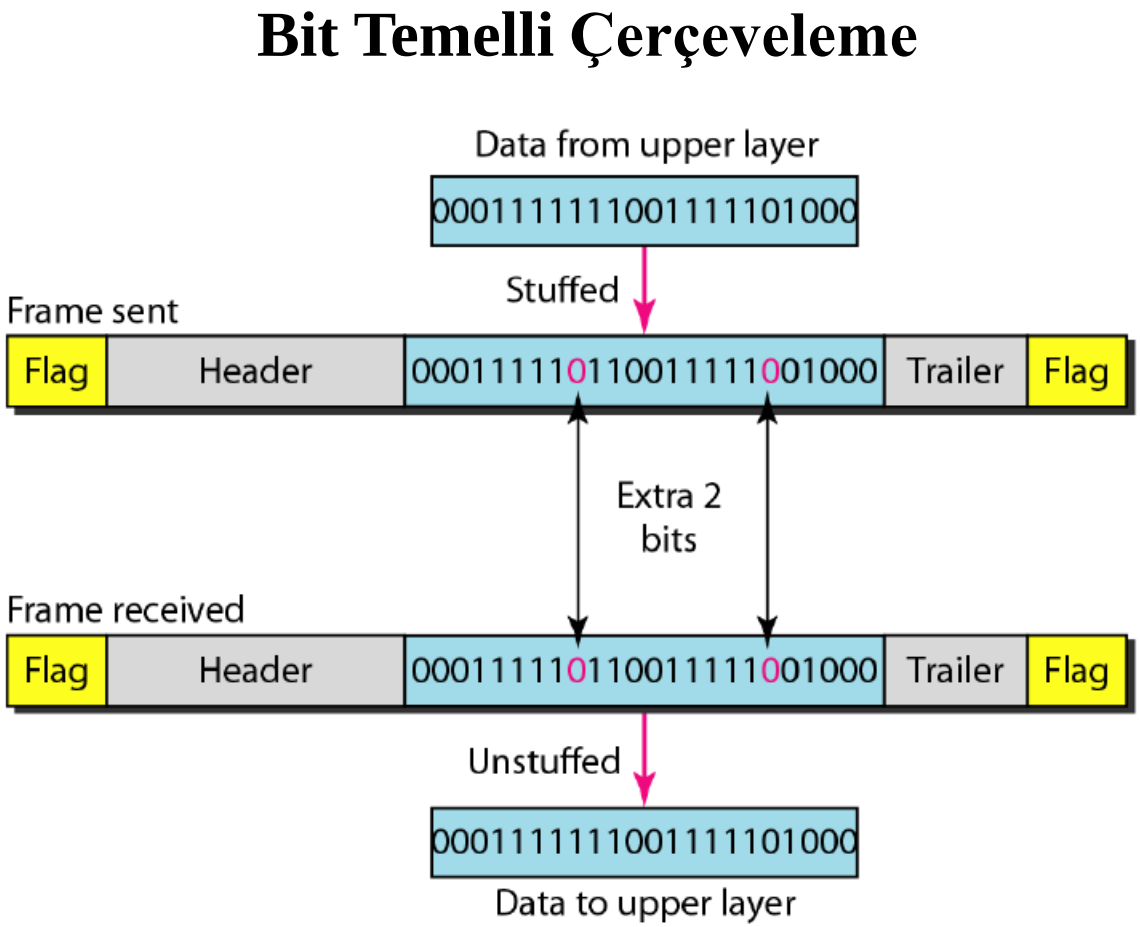
• Veri bloğunda 0'ı takip eden 5 tane 1 varsa (bir sonraki bitin değerine bakılmaksızın) mutlaka extra bir 0 eklenir

• Bu sayede flag 01111110 in veri içinde oluşmasına izin verilmez

• Alıcı extra 0 ları temizler

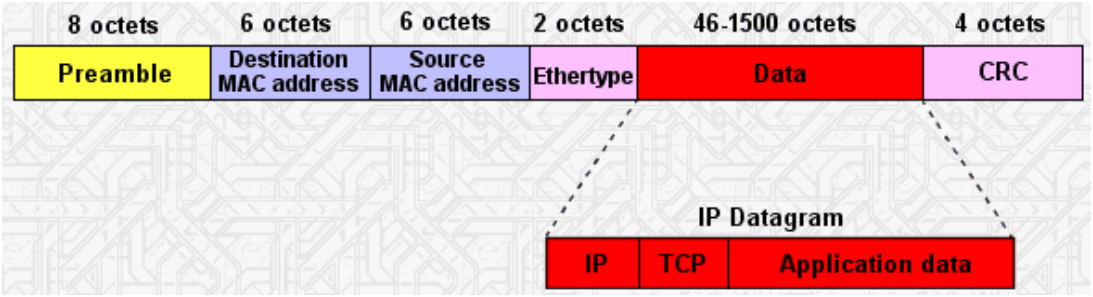


Çerçeveleme(framing)

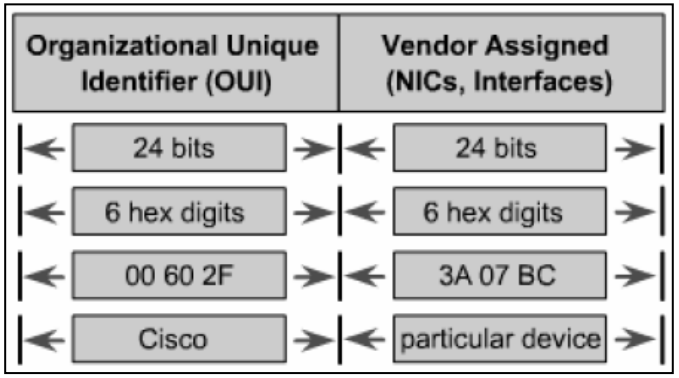


Çerçeveleme(framing)

Ethernet çerçeve Yapısı



Preamble: Başlangıç, 10101010 …. (7B) + 10101011 (1B-SDF) (Frame boyutundan sayılmaz)
Type: Üst katman protokol, IP:0800H, IPx:8137H, X.25:0805H
Data: En az 46 byte olmalı, daha azında çarpışma sezilemez.
CRC: preamble dışındaki tüm veri için hata denetimi yapılır.
Ethernet çerçevesi, host bilgisayar tarafından bakıldığında 14 byte'dir, preamble ve CRC alıcı ağ adaptöründe kaldırılır.

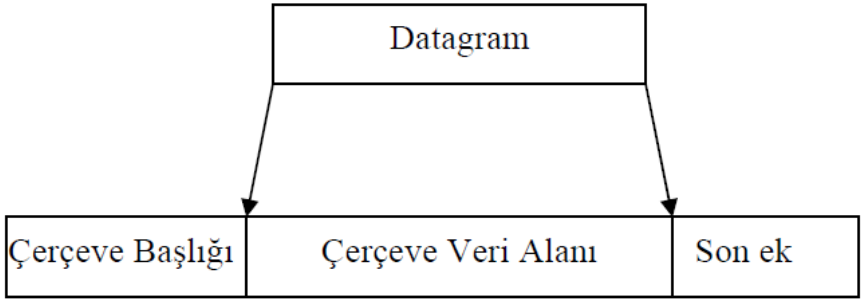


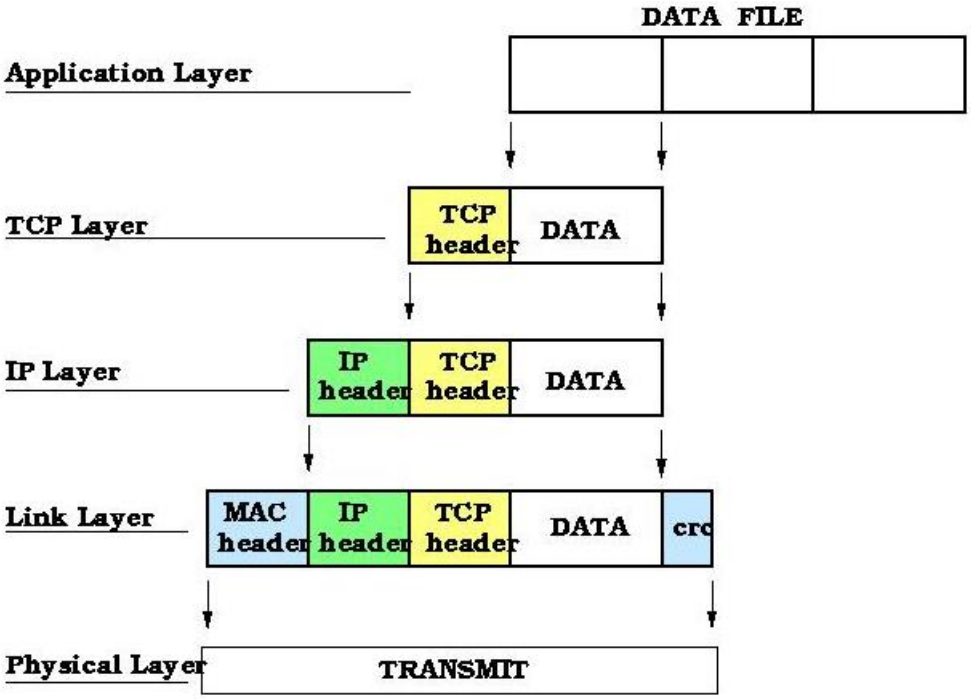
MAC (Fiziksel Adres) Formatı

Çerçeveleme(framing)

Kapsülleme (encapsulation)

IP datagramları geçtikleri fiziksel ağlarda o ağın çerçevesi içinde taşınırlar. Bu işleme kapsülleme denir. Kapsüllenen bir datagram ilgili çerçeve yapısında çerçeve verisi bölümüne yerleştirilir. Kapsülleme işlemi şekilde gösterildiği gibi varış düğümüne ulaşana kadar geçilen her fiziksel ağda tekrarlanır.





Akış Denetimi (Flow Control)

Göndericinin veriyi iletim hızı ile alıcının veriyi işleme hızının uyumlu hale getirilmesi sürecidir.

- Gelen veri kullanılmadan önce kontrol edilmeli ve işlenilmelidir,
- Bu işlemler genelde veri iletim hızından daha yavaştır,
- Gelen veri işlenene kadar bufferda tutulur,
- Düğümler arası çerçeve tamponlama kapasitesi sınırlıdır,
- Tampon dolmaya başladığında alıcı göndericiyi uyararak iletişimi durdurabilir

Aktarım (L4) katmanında uçtan-uca akış denetimi sağlansa da bazı protokollerde (ETHERNET – PAUSE, çok az desteği vardır) düğümler arası akış denetimi veri bağı katmanında gerçekleştirilir.

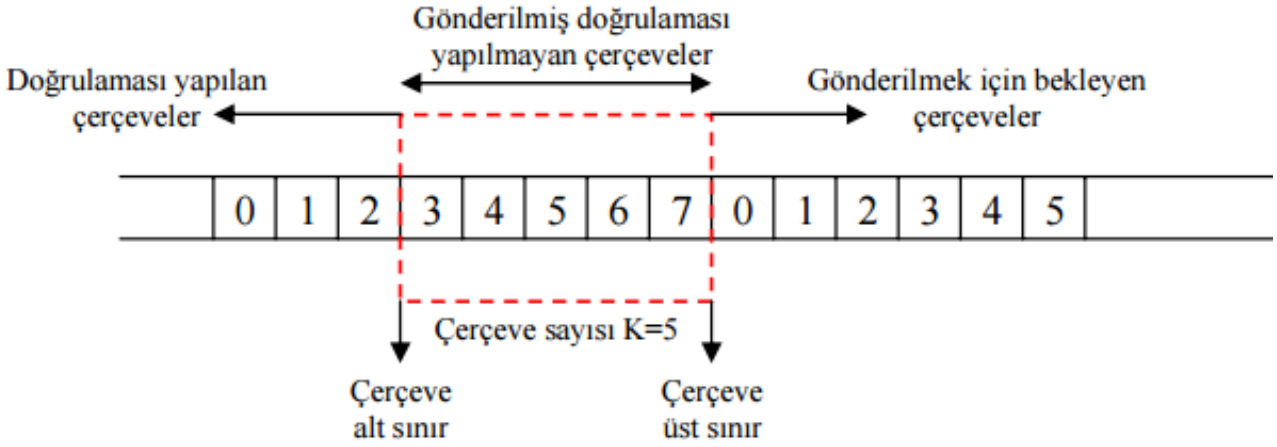
Veri bağlantı katmanında çerçeve akışını denetlemek için kullanılan protokole **Kayan Pencere Protokolü (sliding window protocols)** denir.

Akış Denetimi (Flow Control)

Kayan Pencere Protokolü

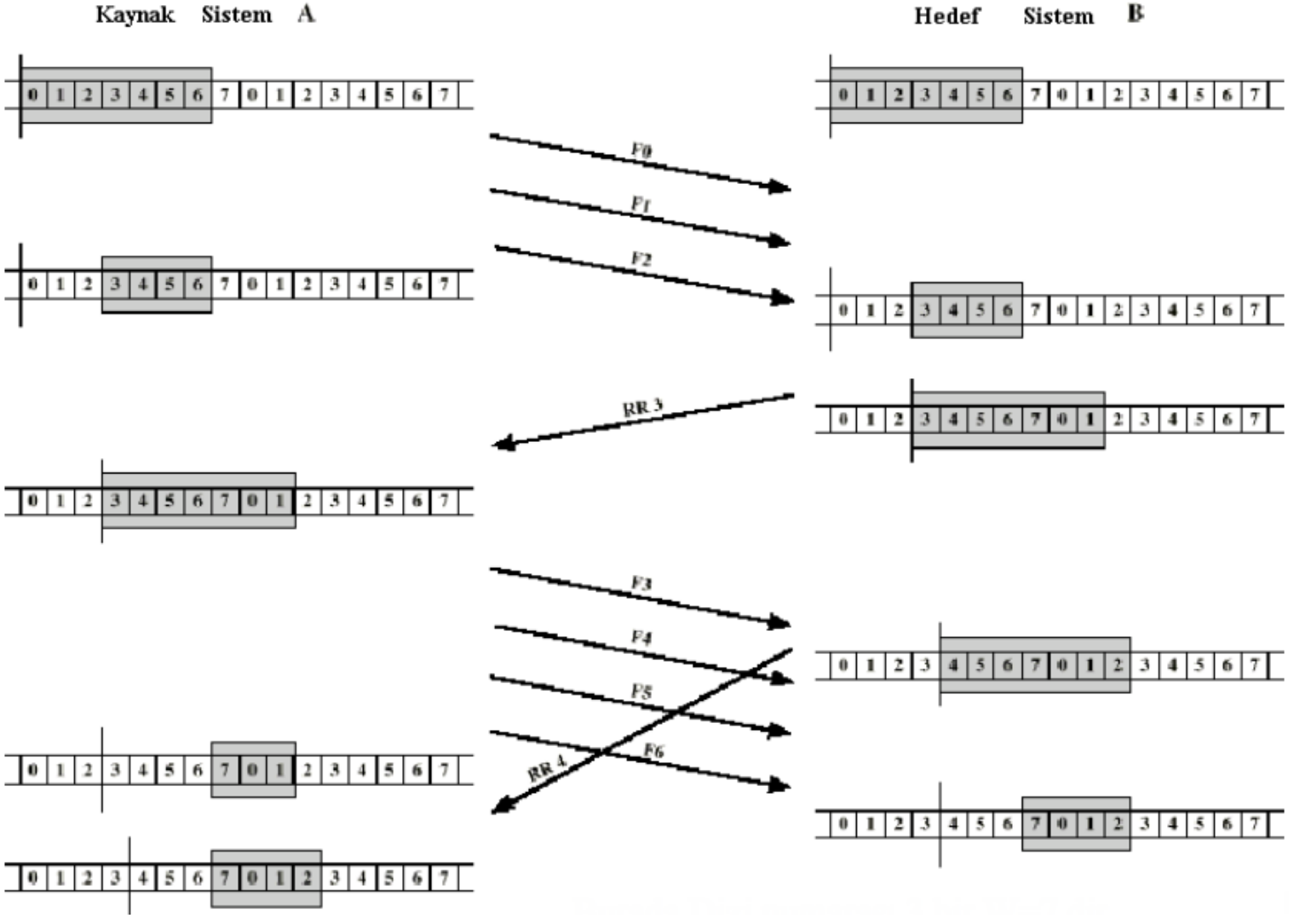
Birden fazla çerçevenin ardıl olarak iletimi sağlanabilir;

- Alıcı düğümün tamponu K uzunluğunda çerçeve kabul edebilir,
- Gönderici düğüm ACK (alındı onayı) gerekmeksizin K adet çerçeveyi iletebilir,
- Alıcı taraftan alınan ACK bir sonraki beklenen çerçeve numarasını ifade eder,
- Gönderici ve Alıcı tarafta gönderilen ve alınan çerçevelerin listesi tutulur,



Akış Denetimi (Flow Control)

Kayan Pencere Protokolü, Pencere boyutu K=7 örneği,



Hata Denetimi

Veri paketlerinin iletimi sırasında değişik nedenlere bağlı olarak bazı bitler bozulabilir. Bu bozulmalara **hata** denir.

Karşılıklı haberleşen ağ katmanları arasında veri iletiminde, iletim ortamındaki gürültü ve diğer olumsuz etkenler nedeniyle hatalar oluşabilir.

Gerekiyorsa hatalı verinin düzeltilmesi ya da yeniden gönderilmesi üzerine çalışılabilir. En basitinden, hatalı olduğu bilinen bir veri parçası atılarak, doğru olarak algılanması engellenebilir. Hata denetimi veri bağı katmanında yapılır.

Hata sezme işlemlerinin hepsi ek veri üretilmesini gerektirir. Üretilen ek veri, aktarılmak istenen veri ile birlikte taşınır. Bu da kullanıcıya sunulan veri kapasitesinin düşmesine yol açar.

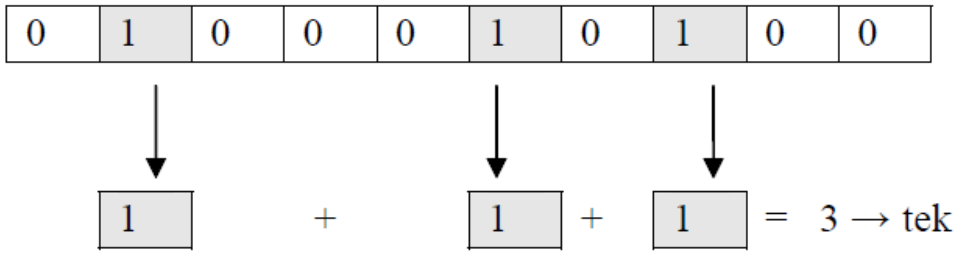
Hata Denetimi

Eşlik Sınaması (Parity Check):

Aktarılan veride oluşan tek sayıda hatayı sezmek için kullanılır. Bu amaçla veriye bir eşlik biti eklenir. Birlerin sayısının çift olmasına **çift eşlik**, tek olmasına da **tek eşlik** durumu denir.

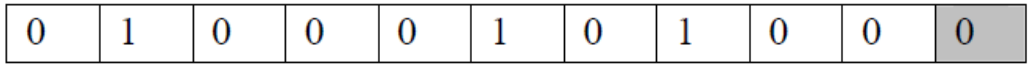
Tek eşlik

Veri:



Eşlik biti 0 olmalı

Veri + eşlik biti:

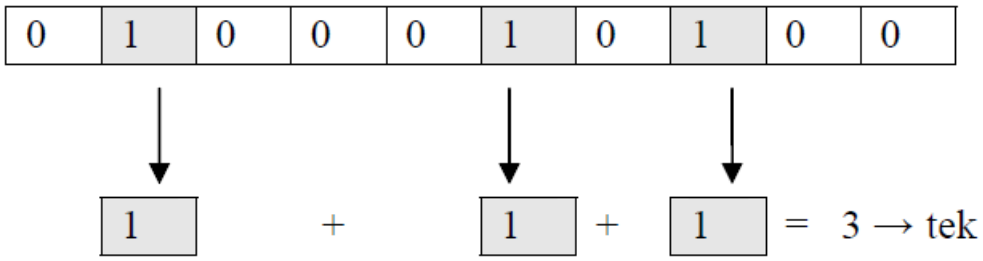


Hata Denetimi

Eşlik Sınaması (Parity Check):

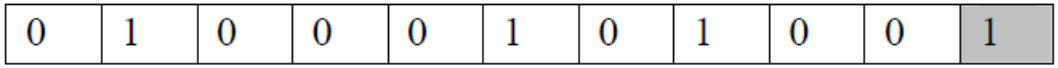
Çift eşlik

Veri:



Eşlik biti 1 olmalı

Veri + eşlik biti:



Bu teknik, 1,3,5 gibi tek sayıdaki bozulmaları tespit eder. Çift sayıdaki hatalarda başarısızdır. Dolayısıyla patlama hatalarında (burst errors) başarı oranı %50'dir. Genellikle düşük boyutlu veri transferlerinde kullanılır.

Hata Denetimi

Çevrimli Fazlalık Sınaması (Cyclic Redundancy Codes - CRC):

Cyclic redundancy check (CRC), LAN ve WAN ağlarda kullanılır.

Bu yöntemde taşınacak veriden hesaplanan bir sınama katarı verinin sonuna eklenir.

Sınama katarına tekniğin İngilizce adının (Cyclic Redundancy Check) kısaltılmışı olarak CRC katarı da denir.

Varış noktasında gelen verinin CRC katarı hesaplanır ve daha önce hesaplanarak veriye eklenmiş CRC katarı ile karşılaştırılır. CRC katarlarının farklı olması verinin bozulduğunu gösterir.

Hata Denetimi

Çevrimli Fazlalık Sınaması (Cyclic Redundancy Codes - CRC):

1- Gönderilecek n bitlik veri dizisi katsayıları veri dizisinin ilgili konumlarındaki bitlerin değerleri olan n. dereceden bir P(x) polinomu ile tanımlanır.

Örneğin; Gönderilecek bilgi bitleri dizisi “10110101” olarak verilsin,

P(x) polinomu	x^7	x^6	x^5	x^4	x^3	x^2	x^1	x^0
Verilen bit dizisi	1	0	1	1	0	1	0	1

$$P(x) = 1.x^7 + 0.x^6 + 1.x^5 + 1.x^4 + 0.x^3 + 1.x^2 + 0.x^1 + 1.x^0$$

$$P(x) = x^7 + x^5 + x^4 + x^2 + 1$$

Hata Denetimi

Çevrimli Fazlalık Sınaması (Cyclic Redundancy Codes - CRC):

2- P(x) polinomu, x^p ile çarpılır. p üreteç polinomunun derecesidir. G(x): üreteç polinomudur. Üreteç polinomları standart olarak tanımlanmış olup CRC kodunun uzunluğunu belirler.

Yukarıdaki örnek için $G(x) = x^3 + x + 1$ ise
G(x), 3. dereceden olup $x^P = x^3$ olur ve P(x) ile çarpılır.

$x^3.P(x) = x^3.(x^7 + x^5 + x^4 + x^2 + 1) = x^{10} + x^8 + x^7 + x^5 + x^3$ elde edilir.

Ad	Polinom	Uygulama
CRC-8	X^8+x^2+x+1	ATM başlık
CRC-10	$X^{10}+x^9+x^5+x^4+x^2+1$	ATM AAL
CRC-16	$X^{16}+x^{12}+x^5+1$	HDLC
CRC-32	$X^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$	LAN lar

Hata Denetimi

Çevrimli Fazlalık Sınaması (Cyclic Redundancy Codes - CRC):

3- $x^p.P(x)$ ifadesi, G(x) üreteç polinomuna bölünür. Kalan elde edilir.

$$
\begin{array}{l|l}
x^{10}+x^8+x^7+x^5+x^3 & x^3+x+1 \\
\cline{2-2}
x^{10}+x^8+x^7 & x^7+x^2 \\
\cline{1-1}
x^5+x^3 & \\
x^5+x^3+x^2 & \\
\cline{1-1}
x^2 & \text{Kalan: } R(x)
\end{array}
$$

Hata Denetimi

Çevrimli Fazlalık Sınaması (Cyclic Redundancy Codes - CRC):

4- Gönderilecek bit dizisi belirlenir.

$$T(x) = x^{P}.P(x) + R(x)$$

$$T(x) = \underbrace{1.x^{10} + 0.x^{9} + 1.x^{8} + 1.x^{7} + 0.x^{6} + 1.x^{5} + 0.x^{4} + 1.x^{3}}_{x^{3}.P(x)} + \underbrace{1.x^{2} + 0.x^{1} + 0.x^{0}}_{R(x)}$$

Gönderilecek bit dizisi: 10110101100 elde edilir. Dikkat edilirse CRC uzunluğu 3 bit olup, üreteç polinomunun derecesine eşittir.

Hata Denetimi

Çevrimli Fazlalık Sınaması (Cyclic Redundancy Codes - CRC):

5- Alıcı tarafta T(x) polinomu G(x)' e bölünür. Bölme işleminde kalan sıfır ise iletim hatası yoktur ve CRC kodu elimine edilir. Kalan sıfırdan farklı ise iletim hatası vardır.

$$
\begin{array}{l|l}
x^{10}+x^8+x^7+x^5+x^3+x^2 & x^3+x+1 \\
\cline{2-2}
x^{10}+x^8+x^7 & x^7+x^2 \\
\cline{1-1}
\quad x^5+x^3+x^2 & \\
\quad x^5+x^3+x^2 & \\
\cline{1-1}
\quad 0 &
\end{array}
$$

Hata Denetimi

Toplama Sağlaması (Checksum):

Checksum Internette yaygın kullanılmaktadır.

Gönderilen sayıların toplamı alınır ve birlikte gönderilir. (7,11,12,0,6) için (7,11,12,0,6,36).

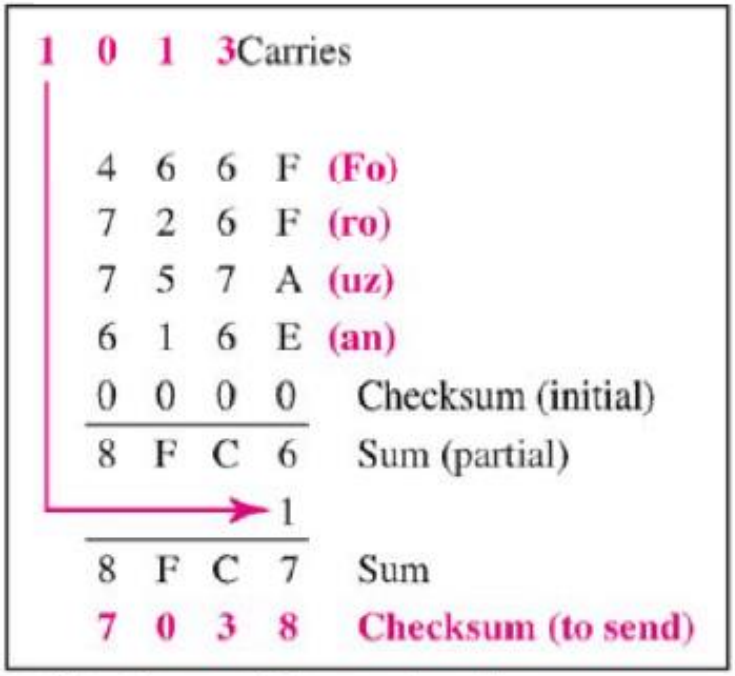
Alıcı gelen sayıları toplar ve gelen toplamla karşılaştırır. Aynı ise data alınır değilse atılır.

Checksum'da toplama işlemi one's complement (birin tümleyeni) aritmetiğiyle yapılır. Internette 16-bit checksum kullanılır.

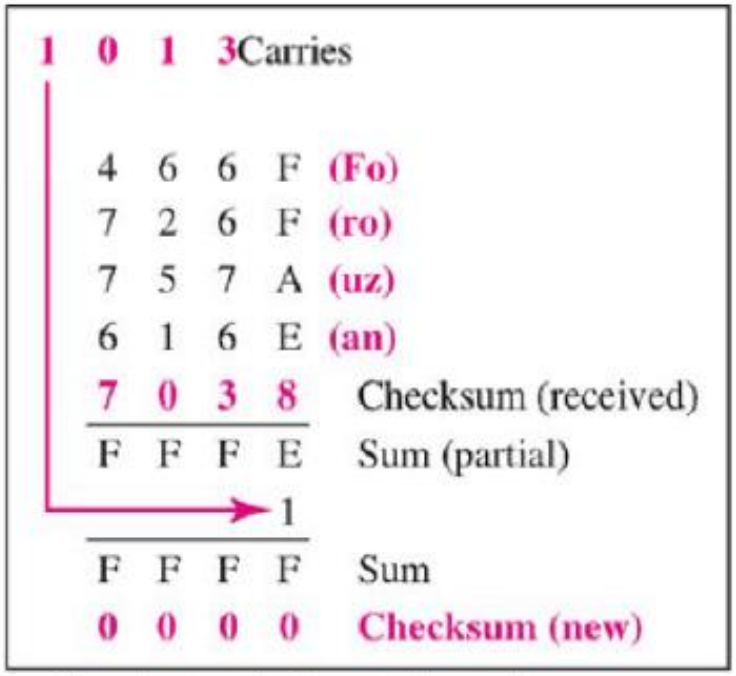
Hata Denetimi

Toplama Sağlaması (Checksum):

Gönderen mesajı 16-bit parçalara böler ve hepsini 1 tümleyene göre toplar. Toplamın tümleyeni alınır ve checksum elde edilir.
Alıcı aynı işlemleri tekrarlar. Checksum değeri 0 olursa hata yoktur.



a. Checksum at the sender site



b. Checksum at the receiver site

Hata Düzeltme

Hamming Kodları

Telekominkasyon sisteminde kullanılan hata düzeltme yöntemidir. Hamming kodları 2 bitlik hataların tespitini ve 1 bitlik hataların düzeltilmesini sağlar. 7 bitlik bir veri olası 7 tane farklı tek bitlik hata durumu vardır ve bu farklı durumlar toplam 3 bitlik hata kontrol bitleri kullanarak ifade edilebilir. Kontrol bitleri de bozulabileceğinden hata düzeltme için gereken bit sayısı 4'tür.

Hamming (11,7) algoritmasına göre;

1. İkinin katı olan bütün bit pozisyonları parity (kontrol) bitleri için
(ör: 1,2,4,8,16,32,64,128….vb.)

2. Diğer tüm pozisyonlar veriler (data) için
(ör: 3,5,6,7,9,10,11,12,13,14,15,17…..vb.)

3. Her bir parity biti verideki bazı diğer bitlerin kontrolünü yapar.

Hata Düzeltme

Hamming Kodları

1. pozisyonda (n = 1): 0 bit atla (0 = n-1), 1 bit kontrol et (n), 1 bit atla (n), 1 bit kontrol et (n), 1 bit atla (n)…. vb.

2. pozisyonda (n = 2): 1 bit atla (1 = n-1), 2 bit kontrol et (n), 2 bit atla (n), 2 bit kontrol et (n), 2 bit atla (n)…. vb.

4. pozisyonda (n=4): 3 bit atla (3 = n-1), 4 bit kontrol et (n), 4 bit atla (n), 4 bit kontrol et (n), 4 bit atla (n)…. vb.

8. pozisyonda (n=8): 7 bit atla (7 = n-1), 8 bit kontrol et (n), 8 bit atla (n), 8 bit kontrol et (n), 8 bit atla (n)… vb.

… şeklinde devam eder.

Hata Düzeltme

Hamming Kodları (Gönderen)

Daha önce açıklandığı gibi (11,7) lik gösterim 7'si data olan toplam 11 bitlik bir veri paketini ifade etmektedir. Örneğin bu 7 bitlik datamız "0110001" olsun. Çift Parity kullanıldığı düşünülürse;

	p1	p2	d1	p3	d2	d3	d4	p4	d5	d6	d7
Parity bitsiz data			0		1	1	0		0	0	1
p1	**0**		0		1		0		0		1
p2		**0**	0			1	0			0	1
p3				**0**	1	1	0				
p4								**1**	0	0	1
Parity bitli data	0	0	0	0	1	1	0	1	0	0	1

Eklenen parity bitleri ile birlikte iletilecek data "00001101001" olur.

Hata Düzeltme

Hamming Kodları (Alan)

Veri iletilirken baştan 3. bitin bozulduğunu ve karşı tarafa “0” değil de “1” olarak iletildiğini varsayalım. Bu durumda “00001101001” olan datamız alıcı düğüm tarafından “00101101001” olarak algılanacaktır.

	p1	p2	d1	p3	d2	d3	d4	p4	d5	d6	d7	Parity Kontrol	Parity bit
Alınan data	0	0	1	0	1	1	0	1	0	0	1		
p1	**0**		1		1		0		0		1	**Yanlış**	1
p2		**0**	1			1	0			0	1	**Yanlış**	1
p3				**0**	1	1	0					**Doğru**	0
p4								**1**	0	0	1	**Doğru**	0

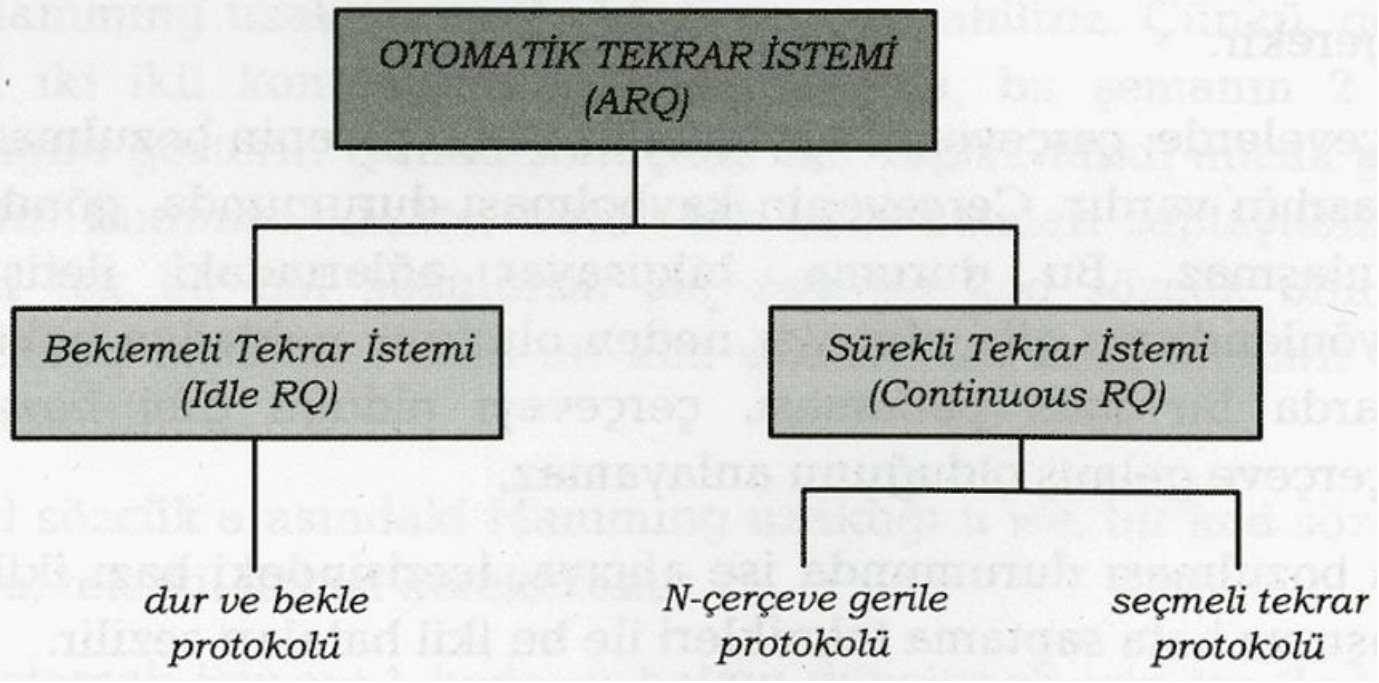
	p4	p3	p2	p1	
İkilik	0	0	1	1	
Ondalık			2	1	**Σ = 3**

Böylelikle hatalı bitin 3. sıradaki olduğunu tespit edilmiş olur.

Hata Düzeltme

Otomatik Tekrar İsteği (Automatic Repeat reQuest - ARQ)

Büyük boyutlu verilerde oluşan hatalar belirlenebilse de bunların alıcı tarafta düzeltilmesi, düzeltme kodunun veri boyutunu ciddi anlamda artıracağı için kullanılır değildir. Bu nedenle hatalı gelen çerçeveler gönderici düğümden tekrar istenir.



Hata Düzeltme

Dur ve Bekle ARQ (Stop and Wait - ARQ)

Frameler alıcıya işleyebileceğinden daha hızı gelmeye başlarsa, kullanılana kadar bir yerde bekletilmeleri/saklanmaları gerekir,

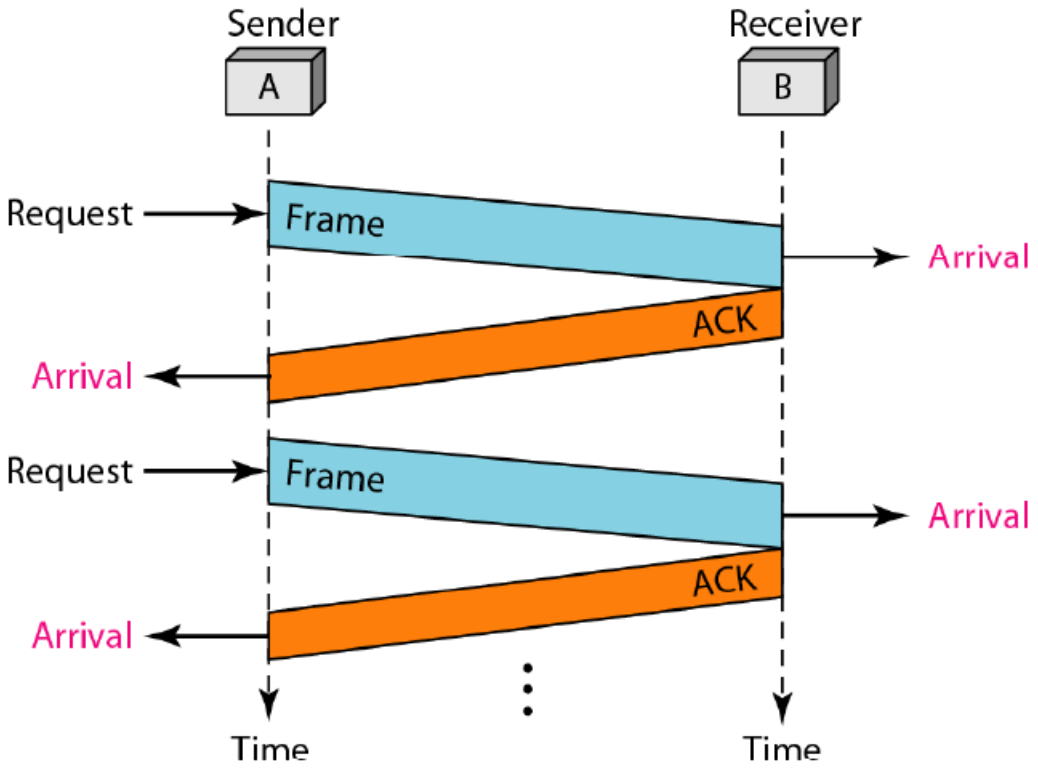
Alıcı genelde bir çok kaynaktan gelen frameleri saklayacak kadar büyük buffer'a sahip değildir. Bu durumda frameler red edilir,

Alıcı göndericiyi uyararak veri göndermesini yavaşlatmak / durdurmak zorundadır. Stop-And-Wait protokolünde gönderici bir frame gönderir ve alıcıdan onay (ACK) alana kadar başka frame göndermez,

Alıcı hatalı frame'i bulursa hiçbirşey yapmaz. Alıcının sessizliği hata olduğu anlamına gelir.

Hata Düzeltme

Dur ve Bekle ARQ (Stop and Wait - ARQ)



Hata Düzeltme

Dur ve Bekle ARQ (Stop and Wait - ARQ)

Stop-and-Wait Protokolü frame'in doğru sırasında olmayan ya da tekrar gönderilmiş olduğunu anlamaz. Çözüm frame'lere sıra numarası eklemektir.

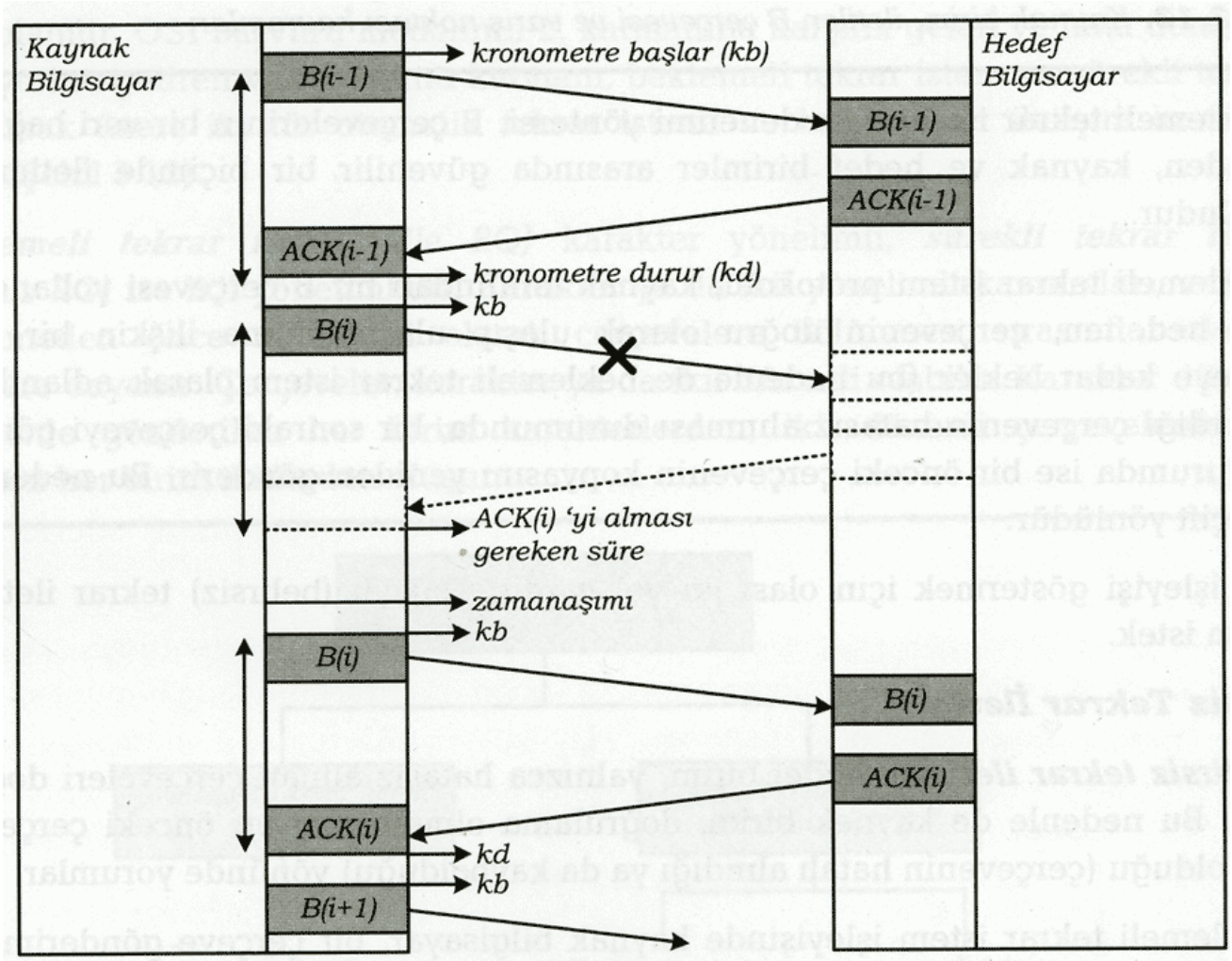
Alıcı sıra numarası doğru olmayan bir frame aldığında kayıp yada tekrarlı frameler olduğunu anlar.

Kayıp ve hatalı frame'ler tekrar gönderilmelidir. Gönderici alıcıya gönderdiği frame'in bir kopyasını saklar ve bir timer başlatır. Timer dolduğunda gönderilen frame için ACK alınmamışsa, frame yeniden gönderilir (kopya hala saklanır ve timer yeniden başlatılır).

ACK'lar da kaybolabileceği ve bozulabileceği için ACK'ların da extra bitlere ve sıra numaralarına ihtiyacı vardır. Bozuk veya sırasız gelen ACK lar dikkate alınmaz.

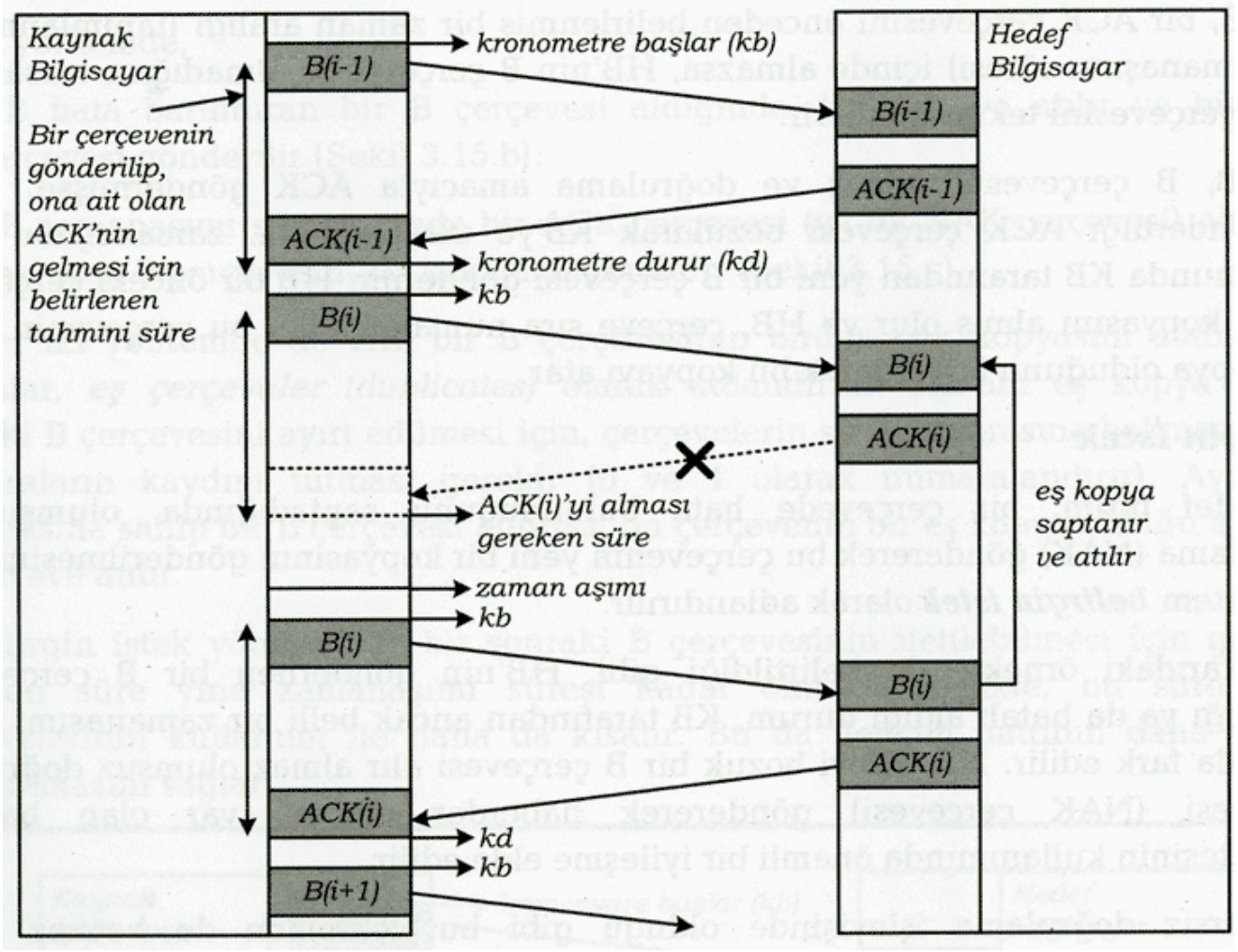
Hata Düzeltme

Dur ve Bekle ARQ (Stop and Wait - ARQ)



Hata Düzeltme

Dur ve Bekle ARQ (Stop and Wait - ARQ)



Hata Düzeltme

N-Çerçeve Gerile (Go-Back-N ARQ)

İletişimin/kanalın verimliliğini artırmak için, ACK beklenmeden birden fazla frame gönderilmelidir.

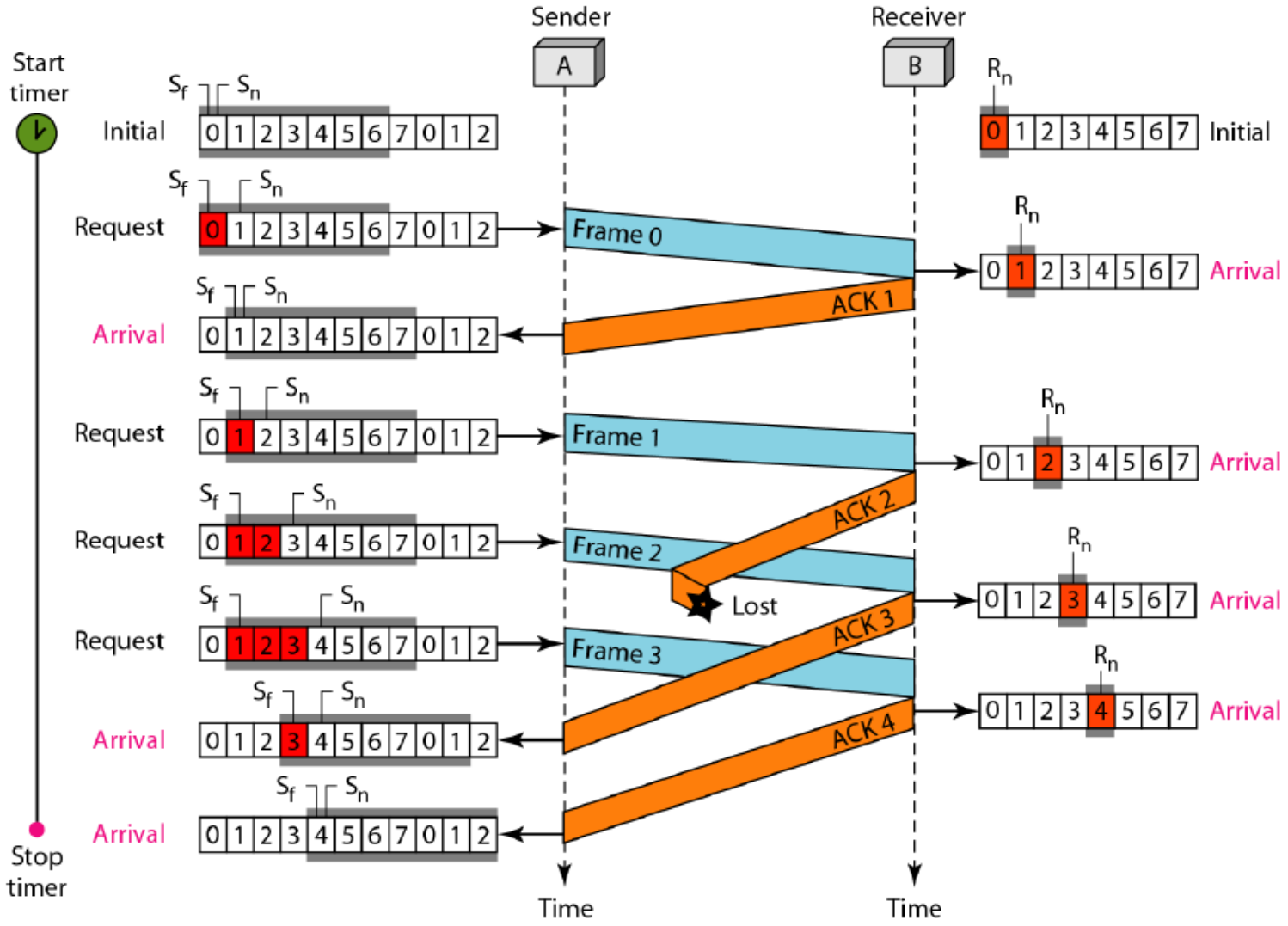
Go-Back-N ARQ protokolü kanal üzerinde birden fazla frame olmasını sağlar. Gönderilen frame'lerin bir kopyası ACK gelene kadar göndericide tutulur

Frameler sıra numaraları ile gönderilir. Başlıkta sıra numarası için m bit yer ayrıldığında 0 dan 2^m-1'e kadar sıra numarası kullanılır.

Sıra numaraları modulo-2^m aritmatiğine göre verilir. m=4 ise 0-15 arası sıra numaraları oluşur. 0, 1,2,3,4,5,6, 7,8,9, 10, 11, 12, 13, 14, 15,0,1,2,3,4,5,6,7,8,9,10..

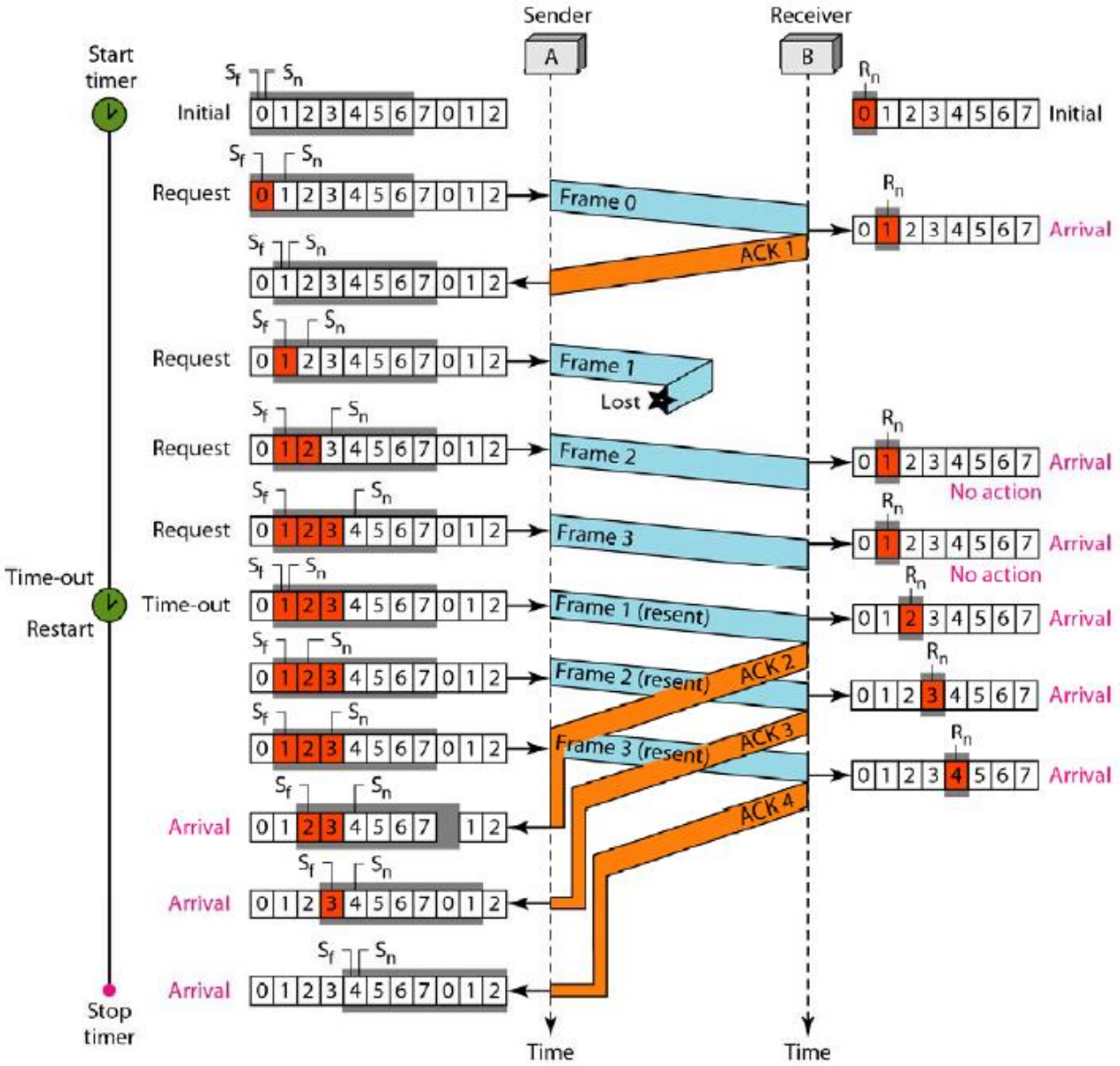
Hata Düzeltme

N-Çerçeve Gerile (Go-Back-N ARQ)



Hata Düzeltme

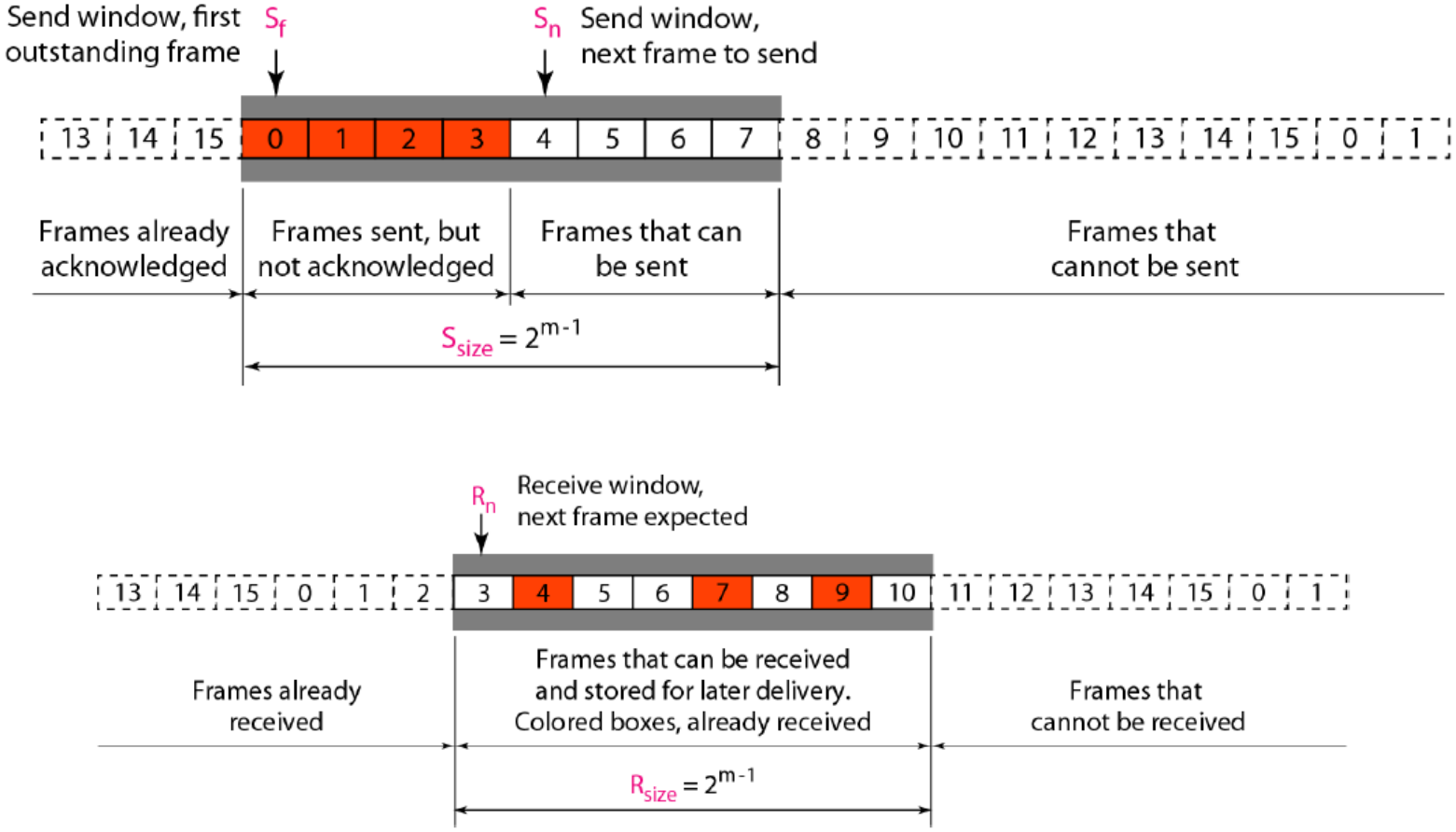
N-Çerçeve Gerile



Hata Düzeltme

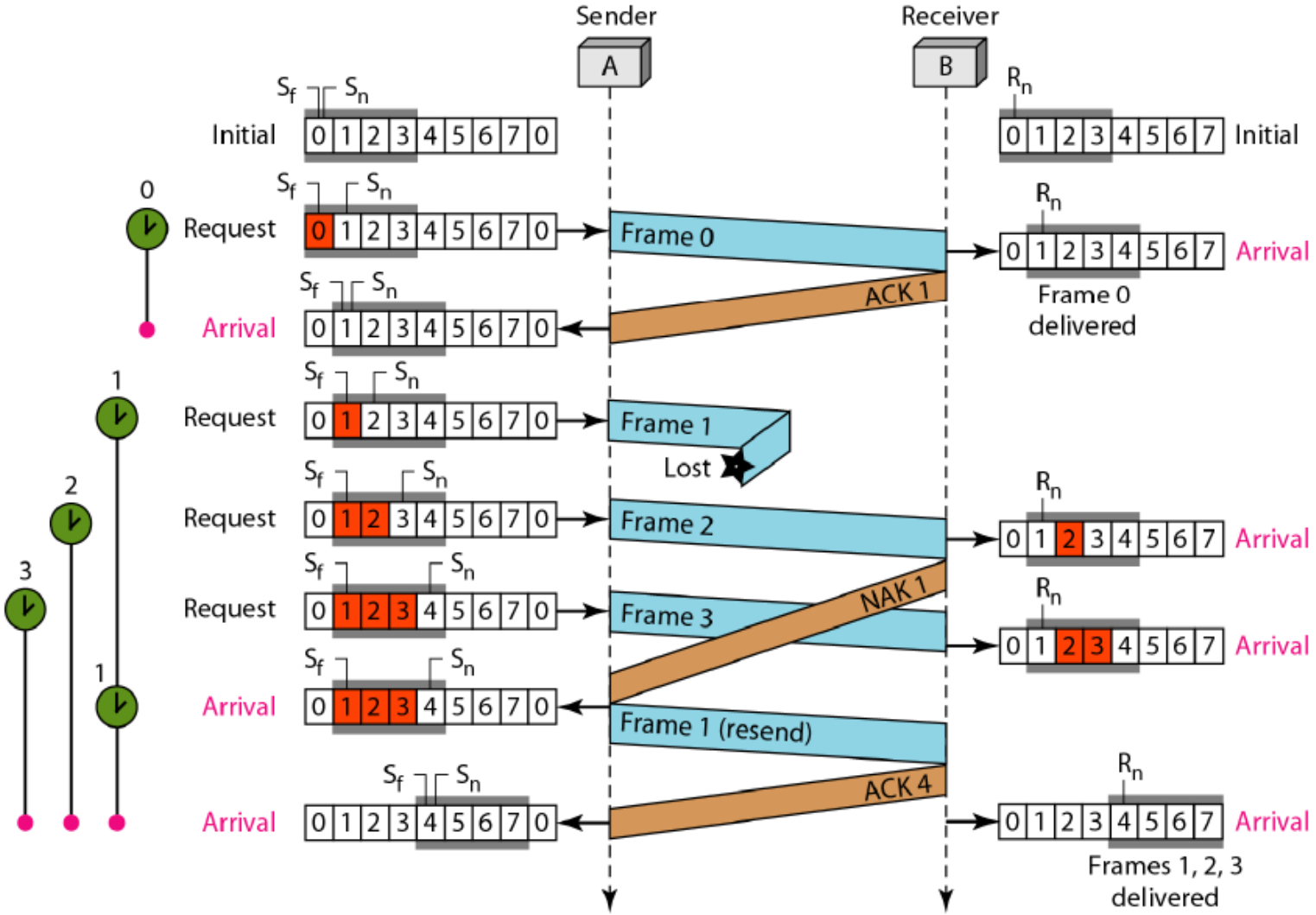
Seçici Tekrar İsteği (Selective Repeat ARQ)

Hatalı ve alınmayan frame'ler için NACK gönderilir.



Hata Düzeltme

Selective Repeat ARQ



Ortam Erişim Denetimi (MAC)

Medium Access Control (MAC)

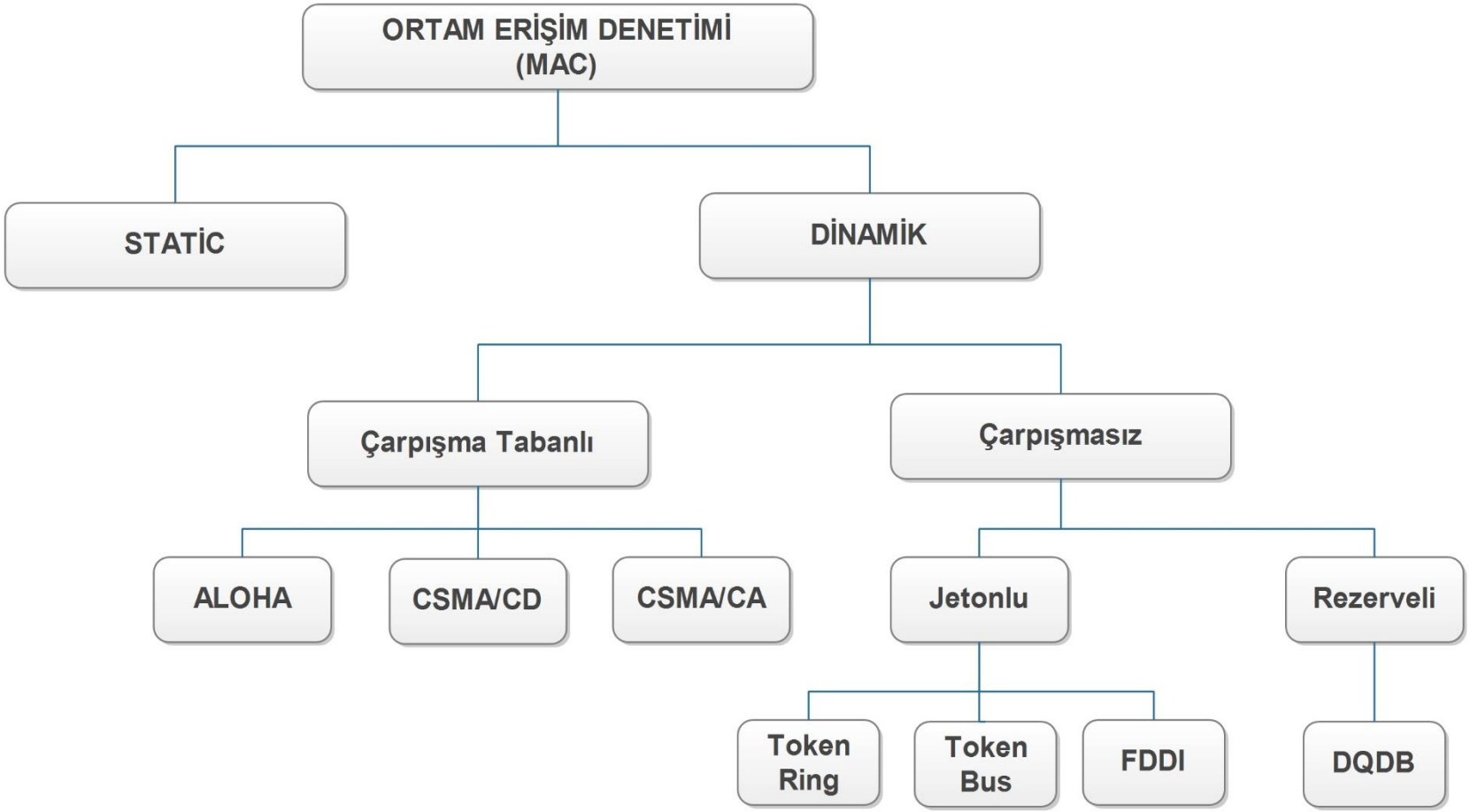
Veri bağı katmanı, mantıksal bağlantı denetimi (logic link control-LLC) ve ortam erişim denetimi (medium access control - MAC) adlı iki alt katmana sahiptir.

MAC alt katmanı, ağ üzerindeki düğümlerin iletişim ortamına nasıl erişeceklerini belirler. Ortam olarak bilgisayar ağları kablolu ve kablosuz iletim ortamı kullanırlar. İletim ortamının (PPP hariç) ortak kullanıldığı modeller için uygun erişim yöntemleri geliştirilmiştir.

Ortam erişim protokolleri kablolu ortamlar için Ethernet 802.3, 802.5 Token ring, FDDI, ATM vb. sayılabilirken, kablosuz ortamlar için WiFi 802.11 sayılabilir.

Ortam Erişim Denetimi (MAC)

Medium Access Control (MAC)



Ortam Erişim Denetimi (MAC)

Basit ALOHA

Rastgele erişim metotları ALOHA adlı protokolden türetilmişlerdir. ALOHA çoklu erişim sağlayan basit bir protokoldür.

ALOHA daha sonra iletişimden önce iletim ortamının dinlenmesi işlemi ile geliştirilmiştir. Buna “carrier sense multiple access”adı verilir.

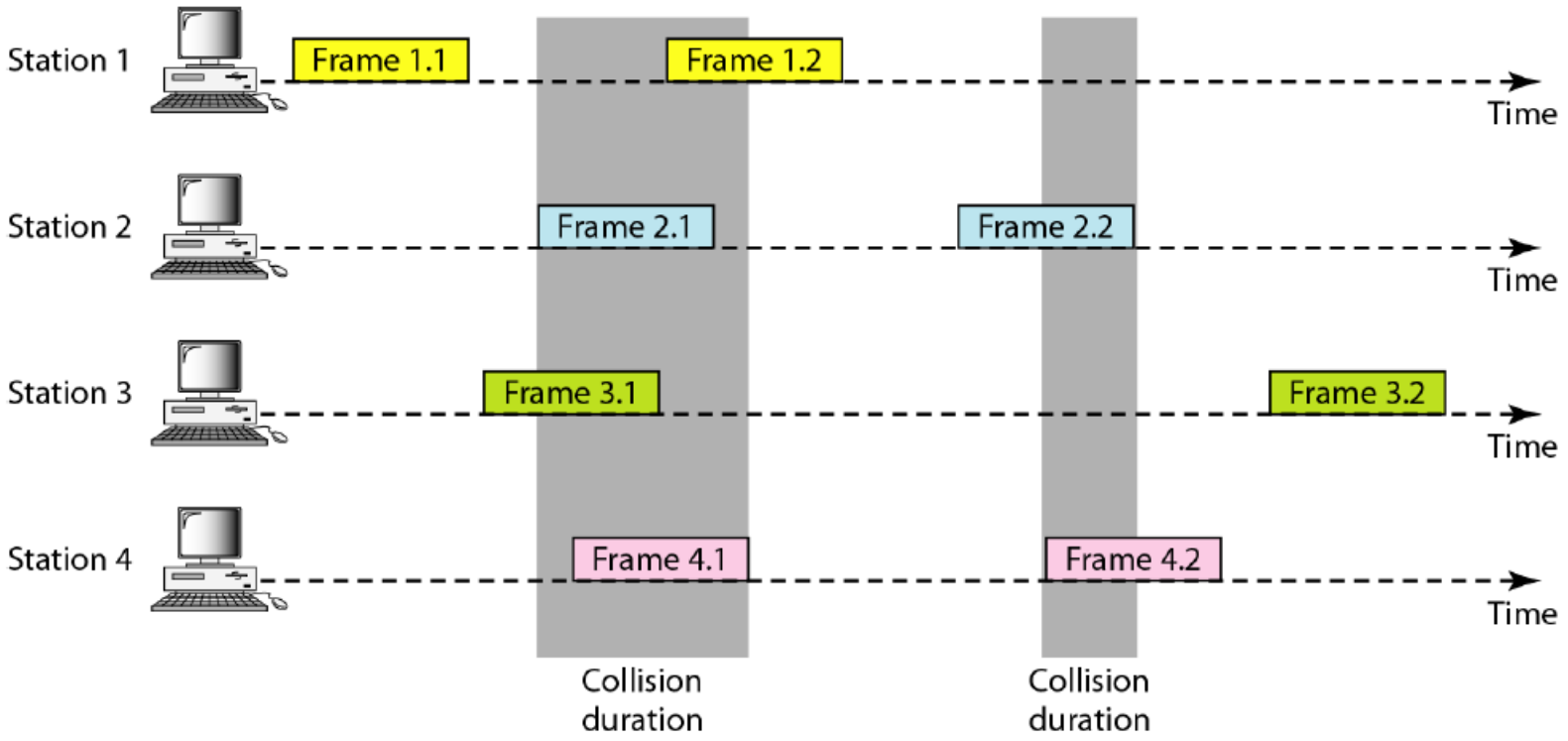
Daha sonra “carrier sense multiple access with collision detection (CSMA/CD)” ve “carrier sense multiple access with collision avoidance (CSMA/CA)” protokolleri ortaya çıkmıştır.

Ortam Erişim Denetimi (MAC)

Basit ALOHA

Orjinal ALOHA protokolü “Pure ALOHA” olarak bilinir.

Bir düğüm frame i olduğu zaman gönderir. Paylaşımlı kanal olduğu için çakışma olabilir.



Ortam Erişim Denetimi (MAC)

Basit ALOHA

Şekilde iletim ortamını kullanmak için "yarışan" 4 düğüm vardır. Her düğüm 2 frame gönderir, iletim ortamında 8 frame vardır. Çakışmalar olur ve sadece 2 frame hedefe ulaşabilir: Frame 1.1 ve Frame 3.2

Bozulan framelerin tekrar gönderilmesi gerekir. Bir frame gönderildiğinde alıcıdan ACK beklenir. Belli bir zaman aralığında alıcıdan ACK gelmezse, gönderici frame'in ya da ACK'nın çakıştığını kabul eder ve frame'i yeniden gönderir. Eğer tüm düğümler framelerini aynı anda tekrar gönderirse yeniden çakışma olur.

Pure ALOHA her düğümün yeniden gönderim yapmadan önce "rastgele" bir zaman dilimi beklemesini gerektirir. Rastgelelik tekrar çakışma olma olasılığını azaltır. Bu zaman dilimi "Back-off time (TB)" olarak adlandırılır.

Ortam Erişim Denetimi (MAC)

Carrier Sense Multiple Access (CSMA):Çakışma olasılığını azaltmak ve performansı artırmak için CSMA protokolü geliştirilmiştir. Hattı dinleyen düğüm hat boşsa çerçeveyi iletir. Aynı anda iki düğüm iletime geçtiği durumlarda çakışma olur.

Çarpışmayı bulmak (Collision Detect - CD)

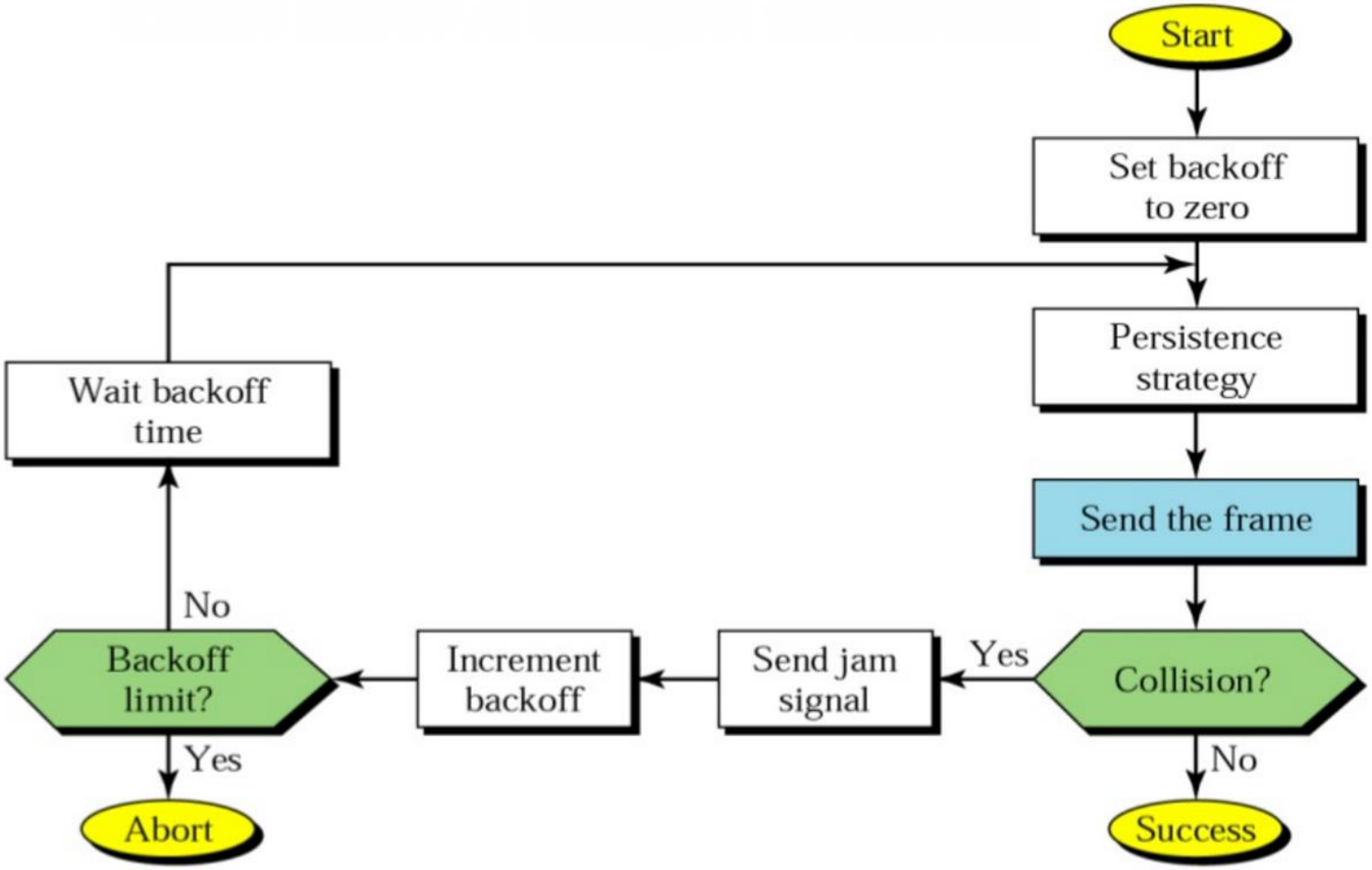
CSMA çakışma olma durumundan sonrası için bir prosedür tanımlamaz.

CSMA/CD çakışma sonrasını düşünür. Frame başarısız olmuşsa tekrar gönderilir. Çakışan ilk bitin durumuna göre hareket edilir. Çakışma anlaşıldığı anda gönderim durdurulur.

Çakışma enerji seviyesindeki değişim ile anlaşılır.

Ortam Erişim Denetimi (MAC)

CSMA/CD:JAM Sinyal, diğer düğümlerin çakışmayı algılaması için rastgele bit gönderimidir.



Ortam Erişim Denetimi (MAC)

Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA):CSMA/CD'nin ana fikri bir istasyonun gönderim yaparken aynı zamanda dinleyebilmesidir.

Kablosuz ortamda sinyal seviyesi ile çakışmanın tespiti zordur.

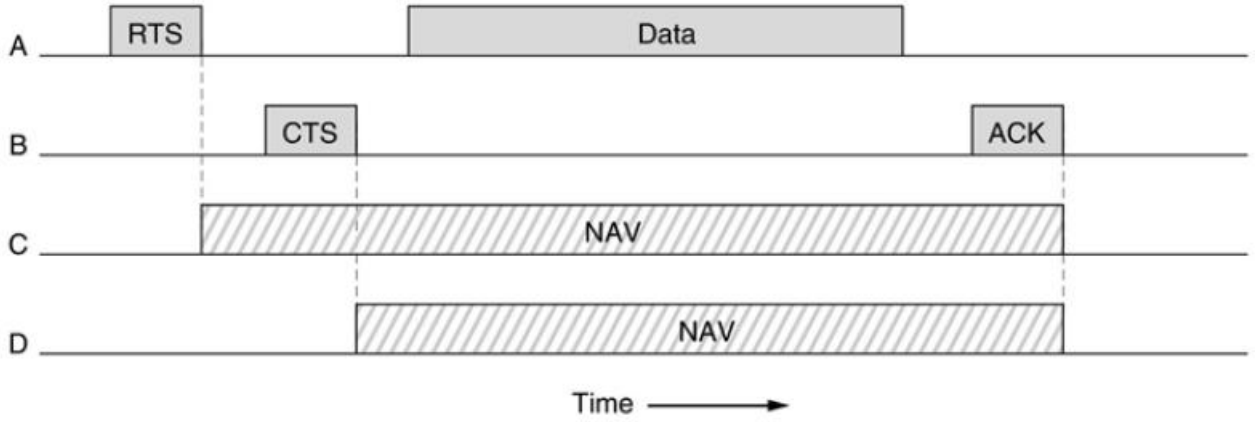
–Enerji gönderim sırasında kaybolur

Veri göndermek isteyen istasyon ortamı dinler, ortam boş ise veri gönderilir, ortam meşgul ise beklenir.

Ancak veri gönderilirken ortam dinlenemediği için oluşan çakışmalar o anda anlaşılmaz. Çakışma durumunda ikili geri-çekilme algoritması çalıştırılır.

Ortam Erişim Denetimi (MAC)

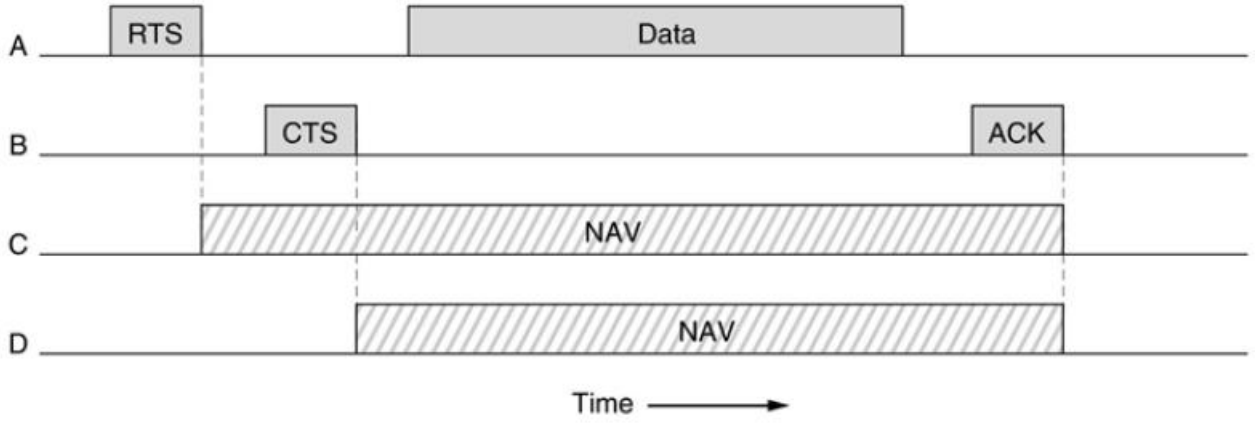
CSMA/CA: Bu modda gönderen (A) ve alıcı (B) arasında veri iletimi öncesinde bir haberleşme gerçeklenir. Bu haberleşmeyi duyan diğer istasyonlar da uygun şekilde davranırlar. Aşağıdaki örnekte C istasyonunun A'nın veri iletim alanında, D'nin ise B'nin alanında olduğunu düşünebiliriz.



CTS: Clear to Send, **RTS:** Request to Send, **NAV:** Network Tahsis Vektörü

Ortam Erişim Denetimi (MAC)

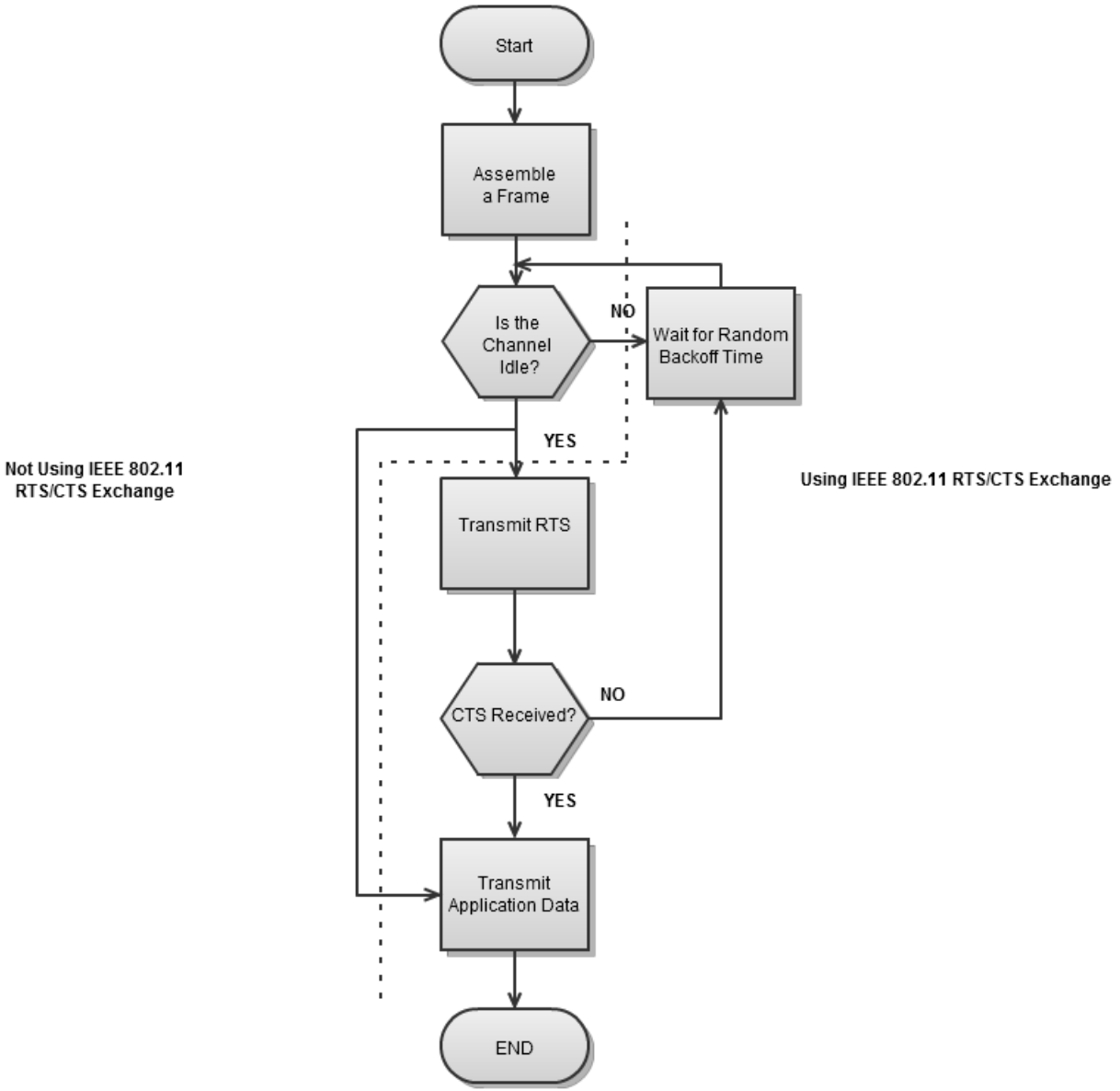
CSMA/CA:CTS mesajını alan A tüm veriyi gönderir ve bir ACK zamanlayıcısını (timer) çalıştırır. B veriyi doğru olarak alınca bir ACK çerçevesi gönderir. ACK çerçevesi A'daki zamanlayıcı dolmadan ulaşırsa işlem başarı ile tamamlanmış olur. Aksi halde herşey yeniden yapılır. Veri iletimi sırasında C ve D istasyonları bir sinyal yaymazlar sadece kanalın meşgul olduğunu kendilerine bildirirler.



CTS: Clear to Send, **RTS:** Request to Send, **NAV:** Network Tahsis Vektörü

Ortam Erişim Denetimi (MAC)

CSMA/CA:



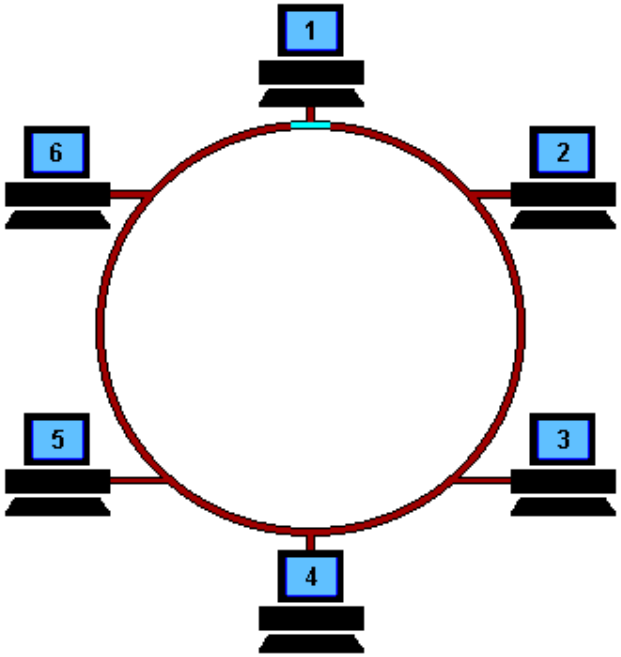
Ortam Erişim Denetimi (MAC)

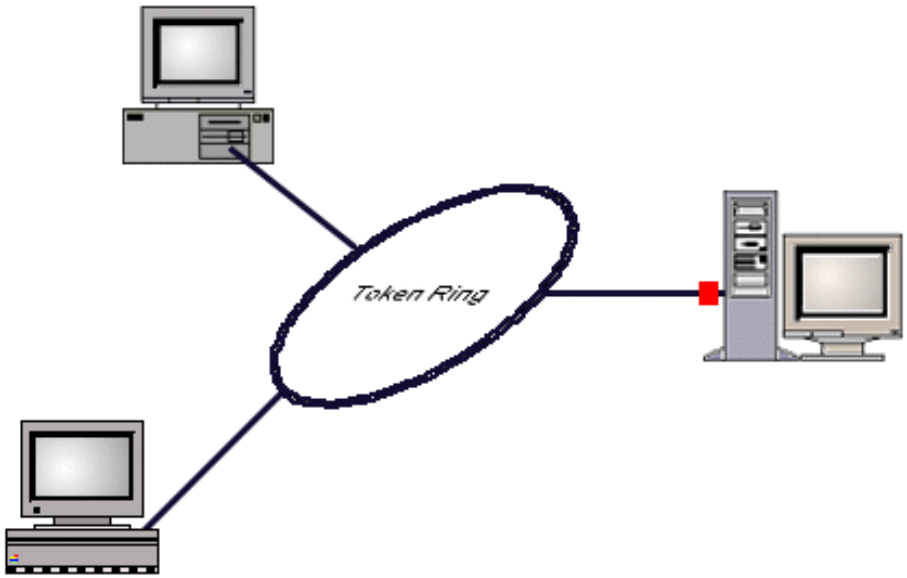
CSMA/CA:

IEEE 802.11					
Release date	Standard	Band (GHz)	Modulation	Advanced antenna technologies	Maximum data rate
1997	802.11	2.4	DSSS, FHSS	N/A	2 Mbits/s
1999	802.11b	2.4	DSSS	N/A	11 Mbits/s
1999	802.11a	5	OFDM	N/A	54 Mbits/s
2003	802.11g	2.4	DSSS, OFDM	N/A	54 Mbits/s
2009	802.11n	2.4, 5	OFDM	MIMO, up to four spatial streams	600 Mbits/s
2012 (expected)	802.11ad	60	SC, OFDM	Beamforming	6.76 Gbits/s
2013 (expected)	802.11ac	5	OFDM	MIMO, MU-MIMO, up to eight spatial streams	6.93 Gbits/s

Ortam Erişim Denetimi (MAC)

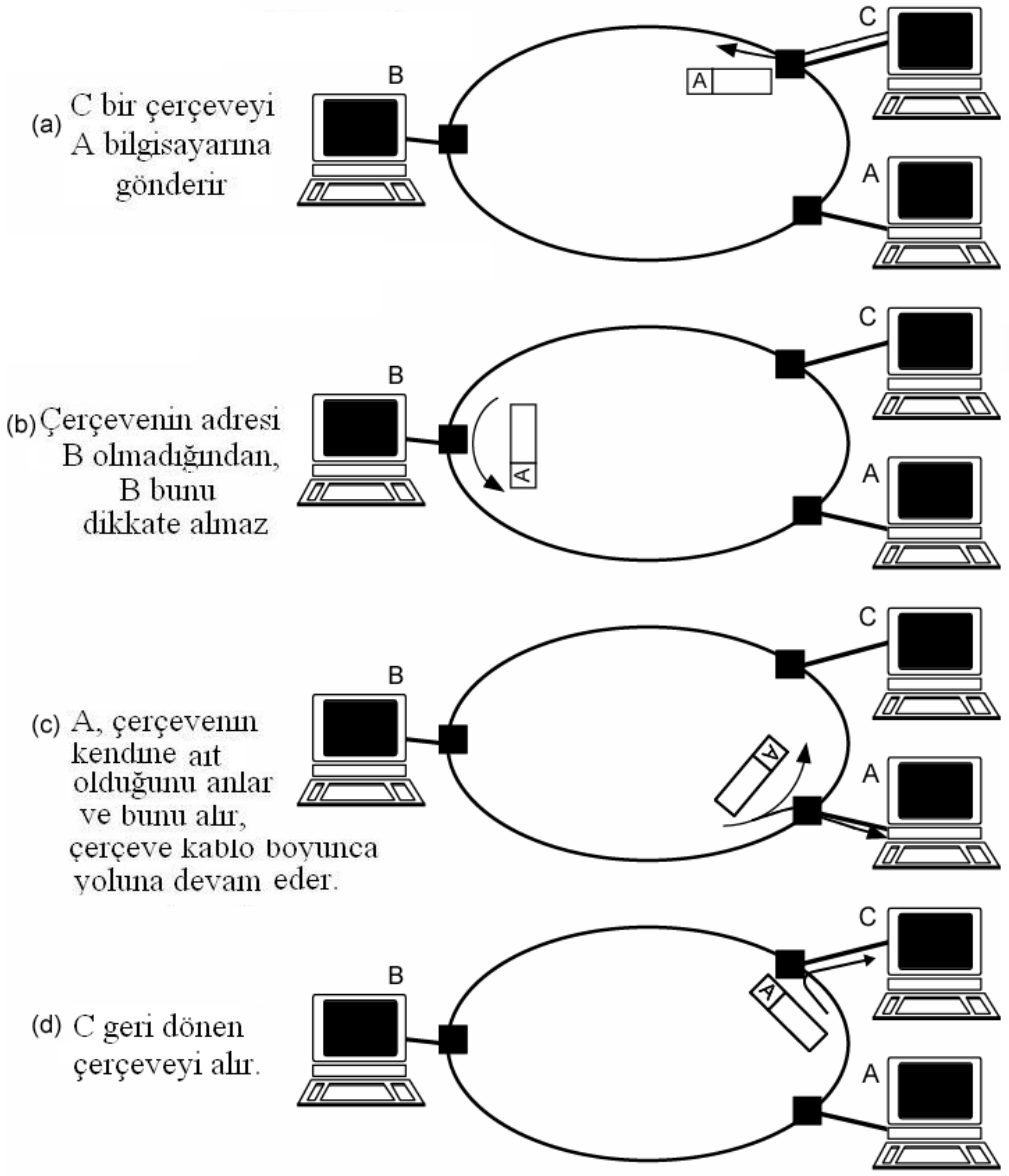
Token Ring: Halka topolojisini kullanan ortam erişim protokolüdür. Ağ erişimi için token kontrolü kullanır. Ağa erişmek isteyen düğüm, token adı verilen özel bir bit dizisine sahip olmak zorundadır. Token'ı elde eden düğüm ağa erişim hakkına sahip iken diğer düğümler sadece ağı dinleyebilir. Veri iletimini tamamlayan düğüm token'ı tekrar ağa bırakır. Token 3 byte'lık özel bir bit dizisidir.





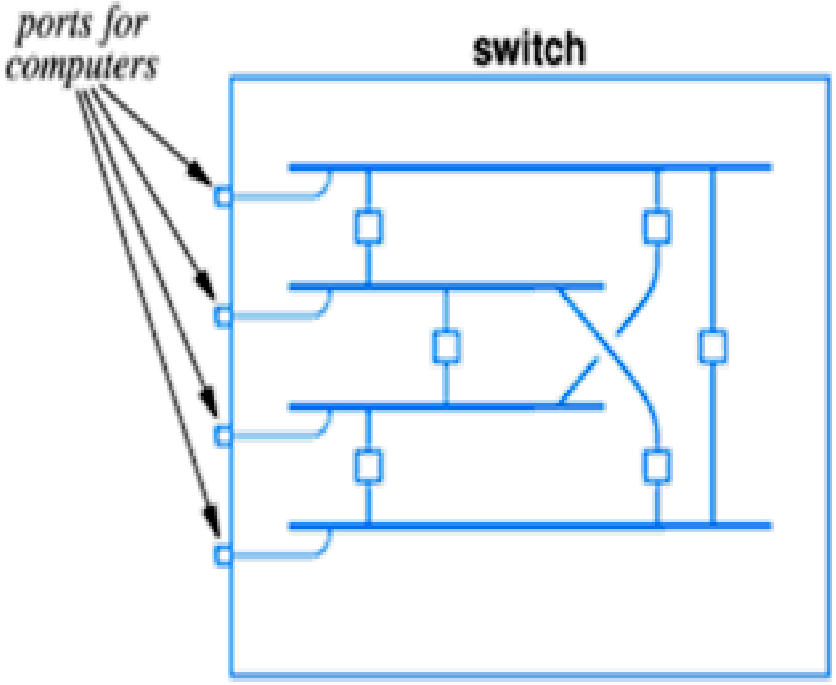
Ortam Erişim Denetimi (MAC)

Token Ring:



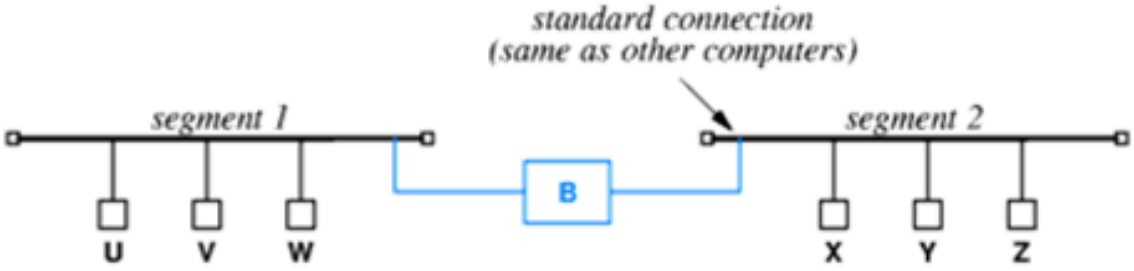
L2'de Çalışan Aygıtları

Anahtar (Switch): Bir anahtar, her bir ucuna farklı bilgisayarların bağlanabildiği bir aygıttır. Her ağda bir bilgisayarın bulunduğu köprülerle birbirine bağlı bir ağ grubuna benzer. Aynı anda n/2 adet bilgisayar birbiri ile haberleşebilir. (n: uç sayısı)



L2'de Çalışan Aygıtları

Köprü (Bridge):İki ağı birbirine bağlamak için kullanılırlar. Veri Bağı Katmanında çalışırlar. Kendilerine gelen çerçeveleri inceleyerek içlerinde bulunan adreslere göre hareket ederler. Üzerlerinde yazılım bulunmaktadır.



Bir ağda bulunan bilgisayarın gönderdiği bilginin, gerekmiyor ise, diğer ağa gitmesini engeller. Ağları birbirinden ayırır. Her ağ için bir liste oluşturur.