ANKARA
HACI BAYRAM VELİ ÜNİVERSİTESİ
UZAKTAN EĞİTİM UYGULAMA VE
ARAŞTIRMA MERKEZİ

İŞLETMELERDE VERİ İLETİŞİMİ

Doç. Dr. Şaban Öztürk

saban.ozturk@hbv.edu.tr

Dersin Genel Hedefleri

- Öğrencilerin veri iletişiminin temel kavramlarını, protokollerini ve teknolojilerini anlamalarını sağlamak.
- İşletmelerde kullanılan çeşitli iletişim ağlarının yapısını, bileşenlerini ve işlevlerini incelemek.
- Günümüz veri iletişimi teknolojilerini (örneğin, Ethernet, Wi-Fi, Bluetooth) ve bunların işletme ortamındaki uygulama alanlarını öğrenmek.
- Veri iletişimi ağlarının güvenliğini sağlamak ve yönetmek için gerekli stratejileri ve araçları öğrenmek.
- İşletmelerde veri iletişimi ile ilgili ortaya çıkan sorunları tanımlamak ve bu sorunlara etkili çözümler geliştirmek.

Doç. Dr. Şaban Öztürk
saban.ozturk@hbv.edu.tr

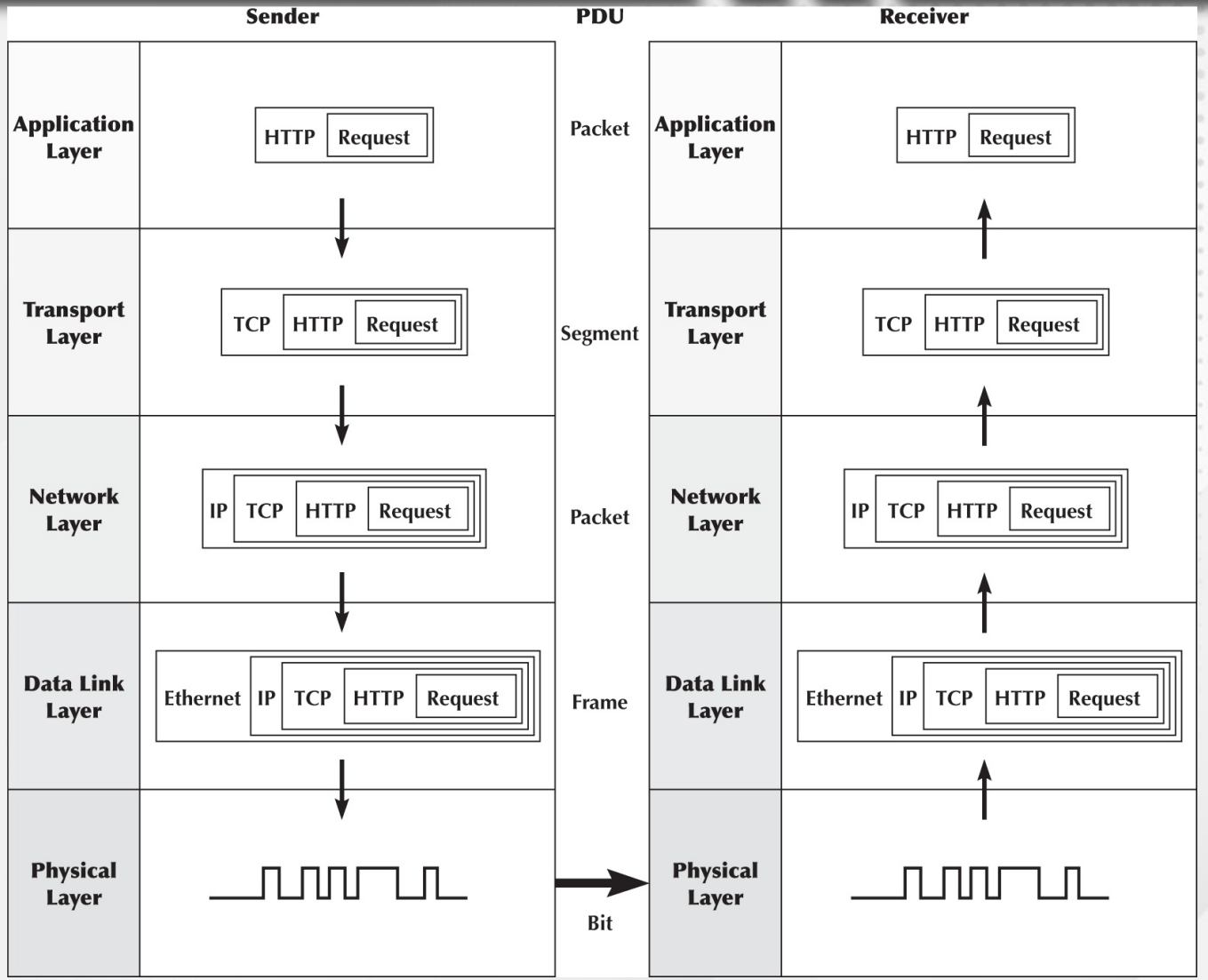
Hafta 4: Data Link Layer Çalışma Prensibi- Veri İletimi Prensipleri

Bölüm Hedefleri

- Data Link Layer Çalışma Prensibinin kavranması
- Veri İletimi Prensiplerinin kavranması

1. Data Link Layer

Veri bağlantı katmanı iletişimin göndericiden alıcıya genel yoldaki bir sonraki bilgisayara veya ağ cihazına taşınmasından sorumludur. Fiziksel ortamda iletilen mesajların nasıl hareket ettirileceğini kontrol eder. Hem gönderici hem de alıcı, birbirleriyle iletişim kuracakları kurallar veya protokoller konusunda anlaşmak zorundadır. Bir veri bağlantı protokolü, kimin ne zaman iletebileceğini, bir iletişimin nerede başladığını ve bittiğini ve bir alıcının bir iletim hatasını nasıl tanıyıp düzelteceğini belirler.



Kaynak: https://quizlet.com/20469384/itn-100-midterm-flash-cards/

1. Data Link Layer

Veri bağlantı kontrolünün gerekliliğini görmek için, doğrudan bağlı iki istasyon arasında etkili veri iletişimi için bazı gereksinimler ve amaçlar listelenir:

• Çerçeve senkronizasyonu: Veri blokları olan çerçeveler halinde gönderilir. Her çerçevenin başlangıcı ve sonu tanınabilir olmalıdır.

• Akış kontrolü: Gönderici istasyon, alıcı istasyonun bunları emebileceği hızdan daha hızlı bir hızda çerçeveler göndermemelidir.

• Hata kontrolü: Aktarım sistemi tarafından tanıtılan bit hataları düzeltilmelidir.

• Adresleme: Bir yerel ağ (LAN) gibi paylaşılan bir bağlantıda, bir iletimde yer alan iki istasyonun kimliği belirtilmelidir.

• Kontrol ve veri aynı bağlantıda: Genellikle kontrol bilgileri için fiziksel olarak ayrı bir iletişim yoluna sahip olmak istenmez. Bu nedenle, alıcı, gönderilen veriden kontrol bilgilerini ayırt edebilmelidir.

• Bağlantı yönetimi: Sürekli bir veri alışverişinin başlatılması, sürdürülmesi ve sonlandırılması, istasyonlar arasında bir miktar koordinasyon ve işbirliği gerektirir.

1. Data Link Layer

Veri bağlantı katmanı iki ana işlev gerçekleştirir ve bu nedenle genellikle iki alt katmana bölünür.

Mantıksal bağlantı kontrolü (logical link control [LLC] sublayer): veri bağlantı katmanının üstündeki ağ katmanına bağlantı sağlar. Gönderen bilgisayarda, LLC alt katman yazılımı, ağ katmanı yazılımıyla iletişim kurmakla sorumludur (örneğin, İnternet Protokolü [IP]) ve genellikle bir IP paketi olan ağ katmanı Protokol Veri Birimi'ni (PDU) bir veri bağlantı katmanı PDU'su ile (genellikle bir Ethernet çerçevesi) çevreler. Alıcı bilgisayarda, LLC alt katman yazılımı, veri bağlantı katmanı PDU'sunu kaldırır ve içerdiği iletiyi (genellikle bir IP paketi) ağ katmanı yazılımına iletilir.

Medya erişim kontrolü (media access control [MAC] sublayer): fiziksel donanımı kontrol eder. Gönderen bilgisayarda, MAC alt katman yazılımı, fiziksel katmanın bitleri fiziksel sembollere dönüştürme zamanını ve nasıl kontrol edeceğini belirler. Alıcı bilgisayarda, MAC alt katman yazılımı, LLC alt katmandan veri bağlantı katmanı PDU'sunu alır, bunu bir bit akışına dönüştürür ve fiziksel katmanın gerçekten bitleri devreye gönderdiği zamanı kontrol eder. Alıcı bilgisayarda, MAC alt katmanı fiziksel katmandan bir bit akışı alır ve bunu tutarlı bir PDU'ya dönüştürür, iletim sırasında herhangi bir hata olup olmadığını kontrol eder ve veri bağlantı katmanı PDU'sunu LLC alt katmanına iletilir.

1. Data Link Layer

Bir veri bağlantı protokolü üç işlev gerçekleştirir:

- Bilgisayarların ne zaman ileteceğini kontrol eder (medya erişim kontrolü)

- İletim hatalarını tespit eder ve düzeltir (hata kontrolü)

- Bir iletişimin başlangıcını ve sonunu bir PDU kullanarak tanımlar (ileti belirleme)

ANKARA
HACI BAYRAM VELİ ÜNİVERSİTESİ
UZAKTAN EĞİTİM UYGULAMA VE
ARAŞTIRMA MERKEZİ

2. Medya erişim kontrolü (media access control [MAC] sublayer)

Medya erişim kontrolü, bilgisayarların ne zaman iletim yapacağını kontrol etme ihtiyacını ifade eder. Noktadan-noktaya tam çift yönlü yapılandırmalarda, medya erişim kontrolü gereksizdir çünkü devrede sadece iki bilgisayar bulunur ve tam çift yönlü yapılandırma herhangi bir zamanda her iki bilgisayarın da iletim yapmasına izin verir.

Medya erişim kontrolü, birkaç bilgisayarın aynı iletişim devresini paylaştığı durumlarda önem kazanır, örneğin, yarı çift yönlü yapılandırmaya sahip noktadan-noktaya bir yapılandırma veya birkaç bilgisayarın aynı devreyi paylaştığı çok noktalı bir yapılandırmada. Burada, iki bilgisayarın aynı anda veri iletimi yapmamasını sağlamak kritiktir - ancak eğer yaparlarsa, bu sorundan kurtulmanın bir yolu olmalıdır. Medya erişim kontrolü için iki temel yaklaşım vardır: çekişme ve kontrol edilmiş erişim.

2. Medya erişim kontrolü (media access control [MAC] sublayer)

Çekişme durumunda, bilgisayarlar devrenin boş olduğu zamanı bekler (yani, başka bilgisayarlar iletim yapmıyorsa) ve sonra veri göndermek için herhangi bir zamanda iletim yaparlar. Çekişme, Ethernet - Yerel Ağlar (LAN'ler) gibi alanlarda yaygın olarak kullanılır.

Bir benzetme olarak, birkaç arkadaşla konuştuğunuzu varsayalım. İnsanlar dinler ve kimse konuşmuyorsa, konuşabilirler. Bir şey söylemek istiyorsanız, konuşmacı bitene kadar beklersiniz ve sonra konuşmaya çalışırsınız. Genellikle, insanlar önceki konuşmacı durduğunda tam olarak o anda ilk dalan kişiye yer verirler.

Bazen, iki kişi aynı anda konuşmaya çalışır, bu nedenle böyle bir sözlü çarpışma meydana geldikten sonra konuşmaya devam etmek için bir teknik olmalıdır.

2. Medya erişim kontrolü (media access control [MAC] sublayer)

Kontrollü erişimde, bir cihaz devreyi kontrol eder ve hangi istemcilerin ne zaman iletim yapabileceğini belirler. İki yaygın olarak kullanılan kontrollü erişim tekniği vardır: erişim istekleri ve anketleme.

Erişim istekleri tekniğiyle, iletim yapmak isteyen istemci bilgisayarlar, devreyi kontrol eden cihaza (örneğin, kablosuz erişim noktası) iletim için bir istek gönderir. Kontrol cihazı, bir seferde bir bilgisayara iletim izni verir. Bir bilgisayara iletim izni verildiğinde, diğer tüm bilgisayarlar, o bilgisayarın bitirmesini bekler ve sonra, iletim yapacak bir şeyleri varsa, bir erişim isteği göndermek için bir çekişme tekniği kullanırlar.

Erişim istekleri tekniği, öğrencilerin ellerini kaldırdığı bir sınıf durumuna benzer. Öğretmen, erişim noktasını kontrol eden kişi olarak hareket eder. Konuşmak istediklerinde, öğrenciler ellerini kaldırır ve öğretmen onları tanır, böylece katkıda bulunabilirler. Bitirdiklerinde, öğretmen tekrar devralır ve başka birinin konuşmasına izin verir. Ve tabii ki, bir sınıftaki gibi, kablosuz erişim noktası istediği zaman iletim yapmayı seçebilir.

2. Medya erişim kontrolü (media access control [MAC] sublayer)

Anketleme, bir istemci bilgisayara iletim yapmasına izin veren bir sinyal gönderme işlemidir. Periyodik olarak, kontrol cihazı (örneğin, bir kablosuz erişim noktası) istemciye veri gönderip göndermediğini sorgular. İstemci veri göndermek için hazırsa, bunu yapar. İstemci veri göndermek için hazır değilse, olumsuz yanıt verir ve kontrol cihazı başka bir istemciye veri gönderip göndermediğini sorar.

Anketlemenin birkaç türü vardır. Sıralı anketleme ile, denetleyici bir listedeki tüm istemcilere ardışık olarak çalışır, önce istemci 1'e anketleme yapar, sonra istemci 2'ye ve bu şekilde devam eder, tüm istemciler anketlenene kadar. Sıralı anketleme, bazılarının diğerlerinden daha sık anketlenmesini sağlamak için öncelikle istemcileri seçmek için değiştirilebilir. Örneğin, birinci istemcinin önceliğini artırabiliriz ve 1, 2, 3, 1, 4, 5, 1, 6, 7, 1, 8, 9 gibi bir anketleme sırası kullanarak önceliği artırabiliriz.

Genellikle, sıralı anketleme bir miktar beklemeyi gerektirir çünkü denetleyici bir istemciyi anketledikten sonra cevap beklemeli ve sonra gelen cevap, gönderilmeyi bekleyen bir giriş mesajı, gönderilecek hiçbir şey olmadığını belirten olumsuz bir yanıt veya geçici olarak hizmet dışı olan istemci (örneğin, arızalı veya kullanıcı tarafından kapatılmış) nedeniyle tam "zaman aşımı süresi" sona erer. Bir tür başarısız güvenli zaman aşımı kullanılmazsa, devre anketleme, hizmet dışı bir istemci üzerinde sonsuza dek kilitlenebilir.

Hub anketleme (genellikle token iletimi olarak adlandırılır) ile, bir cihaz anketi başlatır ve çok noktalı devre üzerindeki sonraki bilgisayara iletir, bilgisayar mesajını gönderir ve anketi sonraki bilgisayara iletir. Sonra, bilgisayar anketi bir sonraki bilgisayara iletir ve bu şekilde devam eder, ilk bilgisayara ulaşana kadar, ilk bilgisayar işlemi tekrar başlatır.

2. Medya erişim kontrolü (media access control [MAC] sublayer)

Hangi medya erişim kontrol yaklaşımı en iyisidir: kontrollü erişim mi yoksa mücadele mi?

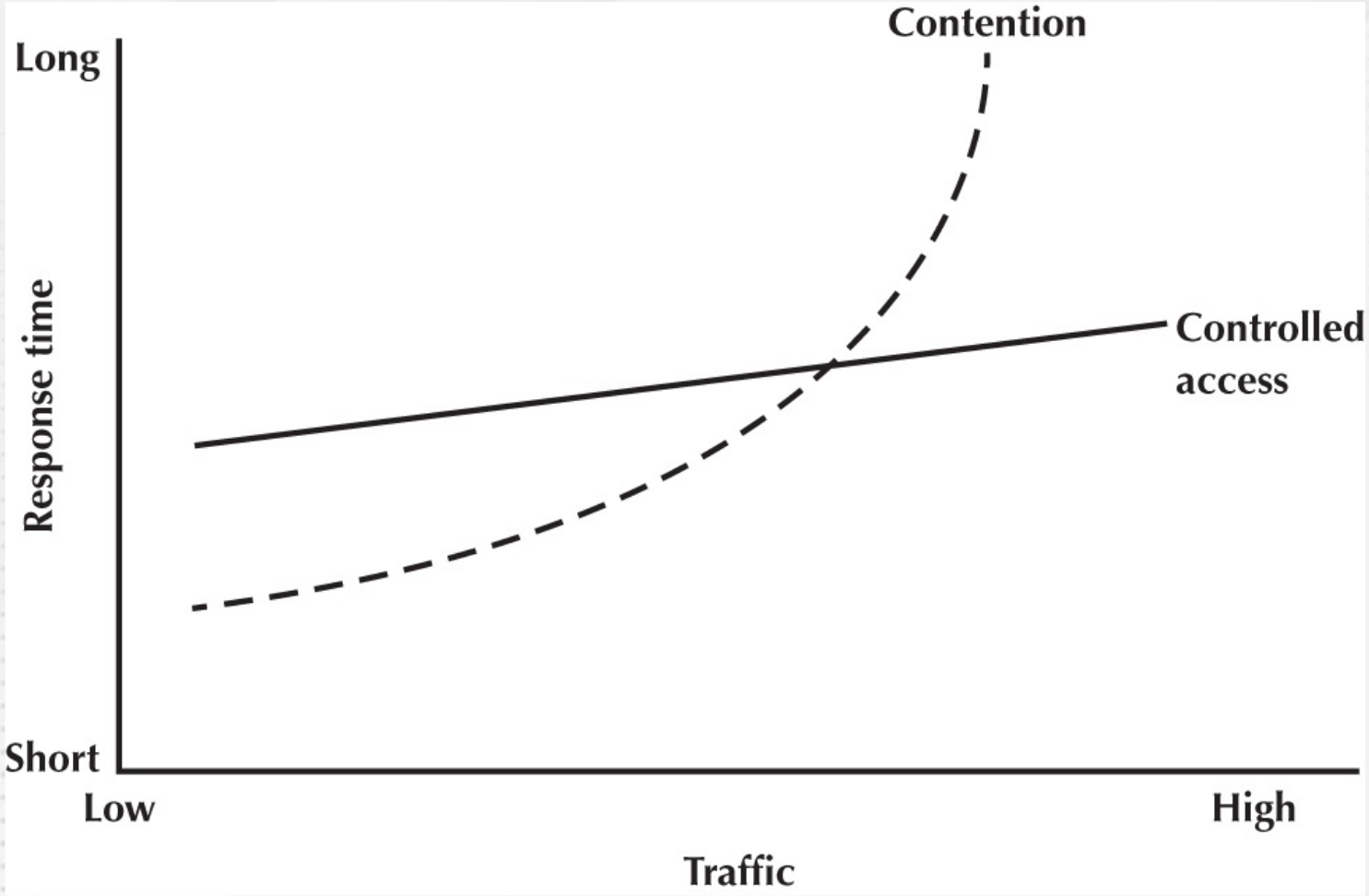
saban.ozturk@hbv.edu.tr

2. Medya erişim kontrolü (media access control [MAC] sublayer)

Basit bir cevap yoktur. Ana düşünce, hangi yaklaşımın ağ üzerinden kullanıcı verilerinin en fazla miktarını iletmeyi sağlayacağıdır. Genel olarak, düşük kullanıma sahip küçük ağlar için **çekişme** yaklaşımları, **kontrollü** yaklaşımlardan daha iyi çalışır. Bu durumda, her bilgisayar gerekli olduğunda iletim yapabilir, izin beklemeksizin. Kullanımın düşük olması nedeniyle, çarpışma olasılığı çok azdır. Buna karşılık, kontrollü erişim ortamlarında bulunan bilgisayarlar izin beklemek zorundadır, bu yüzden başka hiçbir bilgisayarın iletim yapması gerekmiyorsa bile, anketi beklemek zorundadırlar.

Büyük kullanıma sahip büyük ağlar için tam tersi geçerlidir: Kontrollü erişim daha iyi çalışır. Yüksek hacimli ağlarda birçok bilgisayar iletim yapmak istemektedir ve mücadele kullanarak çarpışma olasılığı yüksektir. Çarpışmalar, çarpışma sırasında devre kapasitesini boşa harcar ve daha sonra her iki bilgisayarın da yeniden iletim yapmasını gerektirir. Kontrollü erişim çarpışmaları engeller ve devreyi daha verimli kullanır ve yanıt süresi artmasına rağmen, bu daha aşamalı bir şekilde olur. En iyi erişim kontrolü tekniğini seçmenin anahtarı, kontrollü ve mücadele arasındaki geçiş noktasını bulmaktır. Doğru bir cevap olmasa da, çünkü ağdaki bilgisayarların ne kadar mesaj ilettiğine bağlıdır, çoğu uzman, geçiş noktasının genellikle yaklaşık 20 bilgisayar civarında olduğuna inanmaktadır (daha yoğun bilgisayarlar için daha düşük, daha az yoğun bilgisayarlar için daha yüksek). Bu nedenle, LAN'lerde veya kablosuz LAN'lerde sıklıkla kullanılan paylaşılan çoklu nokta devreleri gibi, her bir paylaşılan devreye en fazla 20 bilgisayar koymaya çalışırız.

2. Medya erişim kontrolü (media access control [MAC] sublayer)



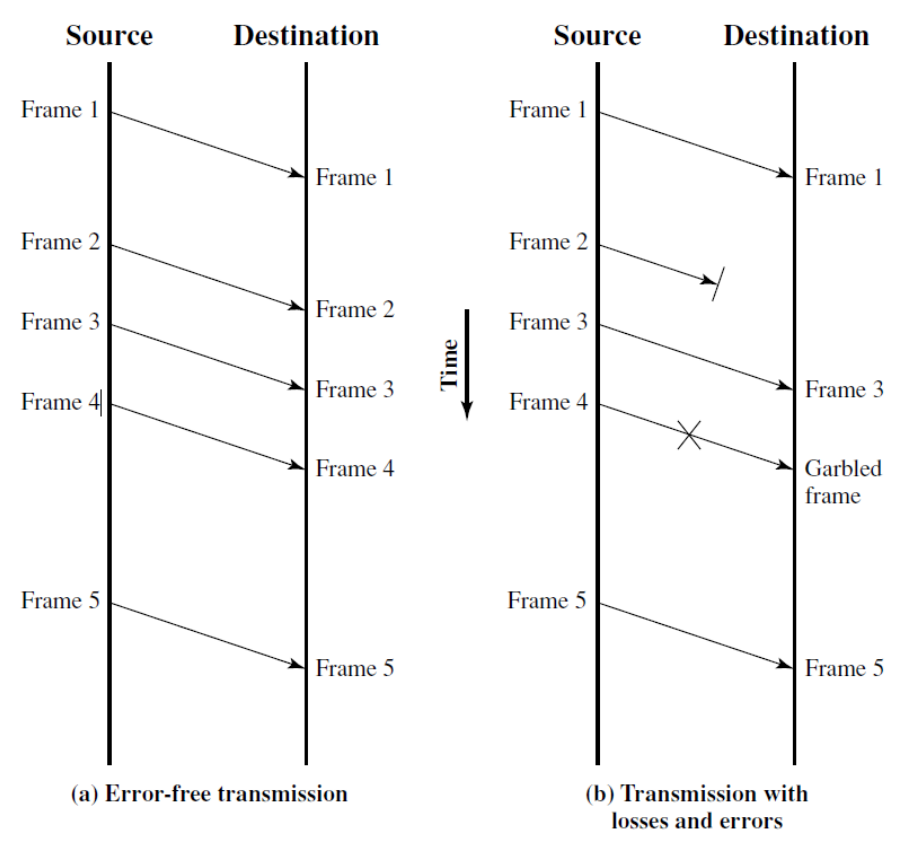
Kaynak: https://quizlet.com/190908365/isds-409-midterm-2-chapters-4-5-flash-cards/

saban.ozturk@hbv.edu.tr

3. Flow Control

Akış kontrolü, bir iletim biriminin bir alıcı birimini veri ile aşırı yüklememesini sağlayan bir tekniktir. Alıcı birim genellikle bir transfer için bir maksimum uzunlukta bir veri tamponu tahsis eder. Veriler alındığında, alıcı, verileri üst düzey yazılıma iletmek için belirli bir işlem yapmak zorundadır. Akış kontrolü olmadığında, alıcının tamponu eski verileri işlerken dolup taşabilir.

Başlamak için, hata olmadan akış kontrolü için mekanizmaları inceleyeceğiz. Kullanacağımız model, Zaman Sekans Diyagramı şeklinde tasvir edilmiştir. Her ok, iki istasyon arasında bir veri bağlantısında geçen tek bir çerçeveyi temsil eder. Veriler, her biri bir kısmı veriyi ve bazı kontrol bilgilerini içeren çerçeveler dizisi halinde gönderilir. Bir istasyonun tüm bitlerini bir çerçevenin ortam üzerine yayması için geçen süre, iletim süresidir; bu, çerçevenin uzunluğuna orantılıdır. Yayılma süresi, bir bitin kaynaktan hedefe bağlantı boyunca gitmesi için gereken süredir. Bu bölümde, iletilen tüm çerçevelerin başarıyla alındığını varsayarız; hiçbir çerçeve kaybolmaz ve hiçbiri hatalı gelmez. Ayrıca, iletilen çerçeveler, gönderildikleri sırayla gelir. Ancak, her iletilen çerçeve, alım öncesinde keyfi ve değişken miktarda gecikme yaşar.



Kaynak: https://courses.minia.edu.eg/Attach/9935Lect_9.pdf

saban.ozturk@hbv.edu.tr

3. Flow Control

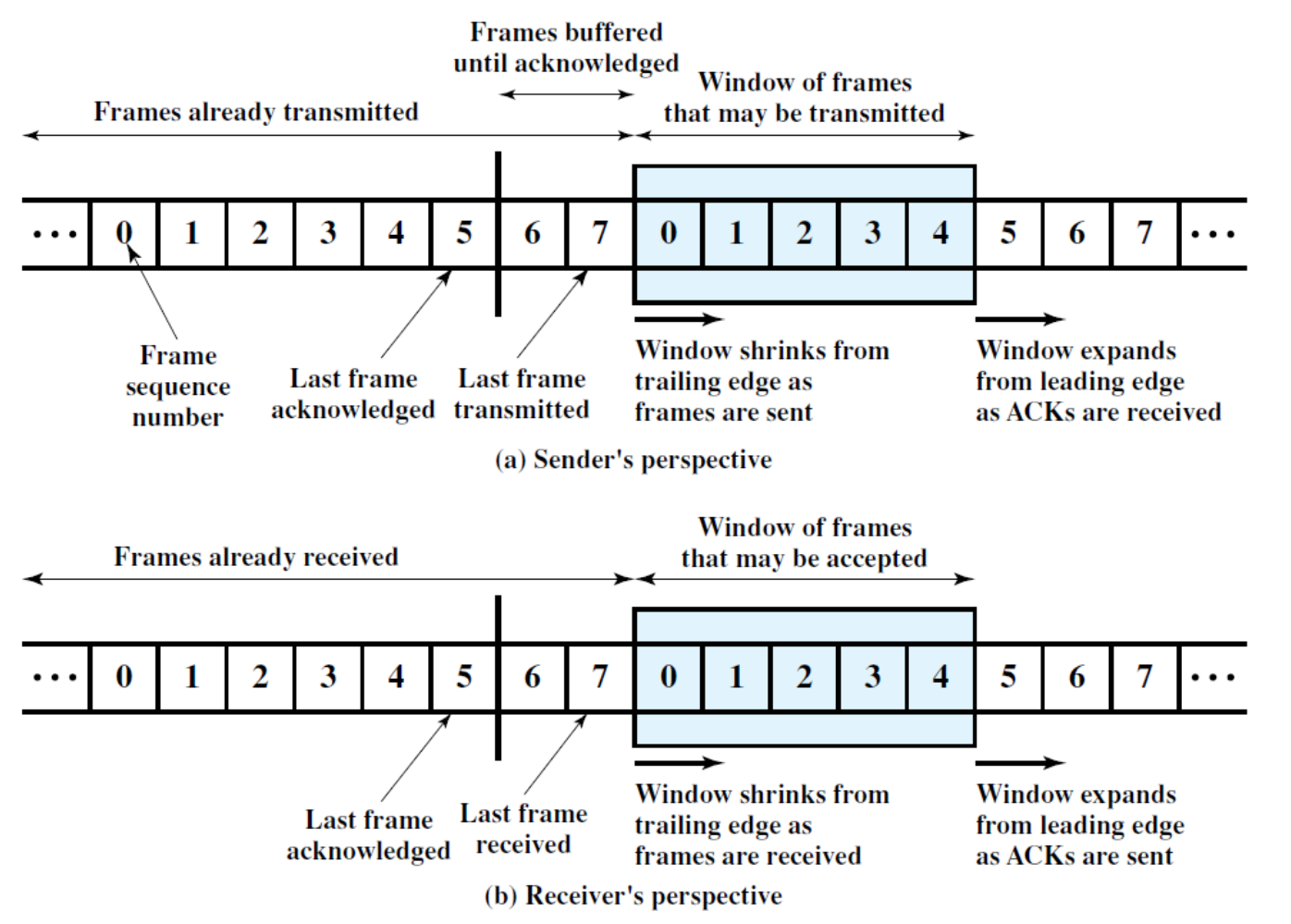
En basit akış kontrolü biçimi, *stop-and-wait akış kontrolü* olarak bilinir ve şu şekilde çalışır. Bir kaynak birimi bir çerçeve iletim eder. Hedef birim çerçeveyi aldıktan sonra, sadece alınan çerçeve için bir yanıtı geri göndererek bir sonraki çerçevenin kabul edilmesine istekli olduğunu belirtir. Kaynak, bir sonraki çerçeveyi göndermeden önce yanıtı alana kadar beklemek zorundadır. Hedef, basitçe onayı vermemek suretiyle veri akışını durdurabilir. Bu prosedür, bir mesajın birkaç büyük çerçeve içinde gönderildiği durumlarda iyi çalışır ve aslında iyileştirilemez. Ancak, sık sık bir kaynağın büyük bir veri bloğunu daha küçük bloklara böleceği ve veriyi birçok çerçevede ileteceği durumlar ortaya çıkar. Bu, şu nedenlerle yapılır:

Alıcının tampon boyutu sınırlı olabilir.
İletim ne kadar uzun olursa, tam bir çerçevenin yeniden iletimini gerektirecek hata olma olasılığı o kadar artar. Daha küçük çerçevelerle hatalar daha erken tespit edilir ve yeniden iletilmesi gereken veri miktarı daha az olur.
LAN gibi paylaşılan bir ortamda, bir istasyonun uzun süre ortamı işgal etmesine izin vermemek genellikle istenir, böylece diğer gönderici istasyonlarda uzun gecikmeler oluşmaz.

3. Flow Control

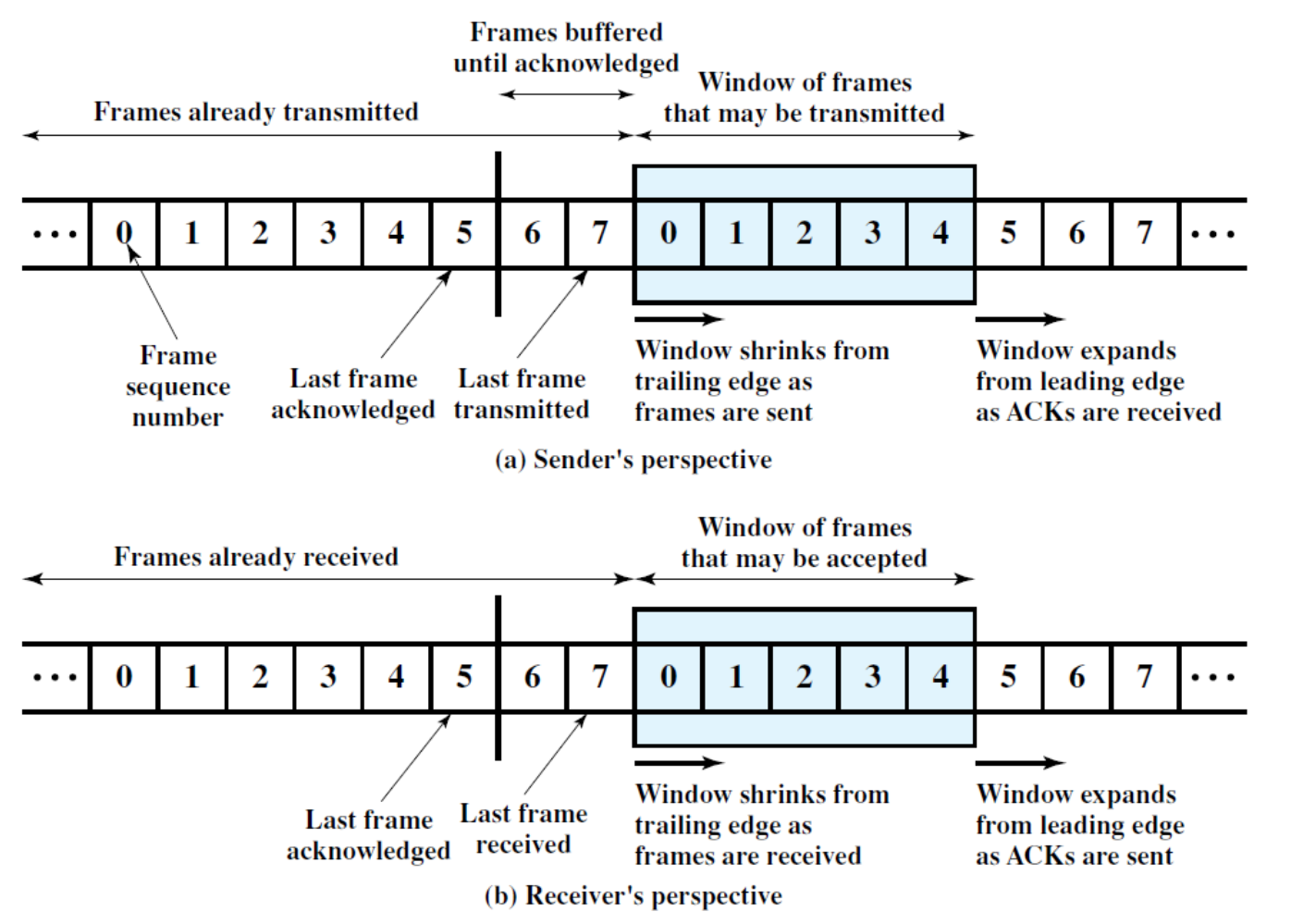
İki istasyon, A ve B, tam çift yönlü bir bağlantı aracılığıyla bağlandığında nasıl çalışabileceğini inceleyelim. İstasyon B, W çerçeveye kadar önbellek alanı ayırır. Bu sayede, B W çerçeve alabilir ve A, herhangi bir onay beklemeden W çerçeve gönderebilir. Hangi çerçevelerin onaylandığını takip etmek için her biri bir sıra numarası ile etiketlenir. B, bir çerçeveyi, beklenen bir sonraki çerçevenin sıra numarasını içeren bir onay ile tanır. Bu onay, aynı zamanda B'nin, belirtilen numarayla başlayarak bir sonraki W çerçeveyi almaya hazır olduğunu da ima eder. Bu şema ayrıca birden çok çerçeveyi tanımak için de kullanılabilir. Örneğin, B, çerçeveleri 2, 3 ve 4 alabilir ancak çerçeve 4 gelene kadar onayı tutabilir. Ardından, sıra numarası 5 ile bir onay göndererek, B, çerçeveleri 2, 3 ve 4'ü aynı anda tanır. A, göndermeye izin verilen sıra numaralarının bir listesini ve B, almaya hazır olduğu sıra numaralarının bir listesini tutar. Bu listelerin her biri, çerçeveler için bir pencere olarak düşünülebilir. İşlem, kaydıran pencere akış kontrolü olarak adlandırılır.

3. Flow Control



Kaynak: https://www.slideshare.net/slideshow/dc-chapter-07/36618933

3. Flow Control



Kaynak: https://www.slideshare.net/slideshow/dc-chapter-07/36618933

4. DATA LINK PROTOCOLS

Burada, mesajı belirten mesaj ayırma üzerinde duruyoruz, bu da bir mesajın nerede başladığını ve nerede durduğunu gösterir ve mesajın içindeki çeşitli parçaları veya alanları belirtir. Örneğin, bir mesajın veya veri paketinin hangi kısmının hata kontrol bölümü olduğunu net bir şekilde belirtmelisiniz; aksi takdirde, alıcı, bir hatanın olup olmadığını doğru bir şekilde belirlemek için bunu kullanamaz. Veri bağlantı katmanı, ağ katmanından aldığı pakete bir PDU ekleyerek bu işlevi gerçekleştirir. Bu PDU'ya çerçeve (frame) denir.

saban.ozturk@hbv.edu.tr

4. DATA LINK PROTOCOLS (Asenkron İletim)

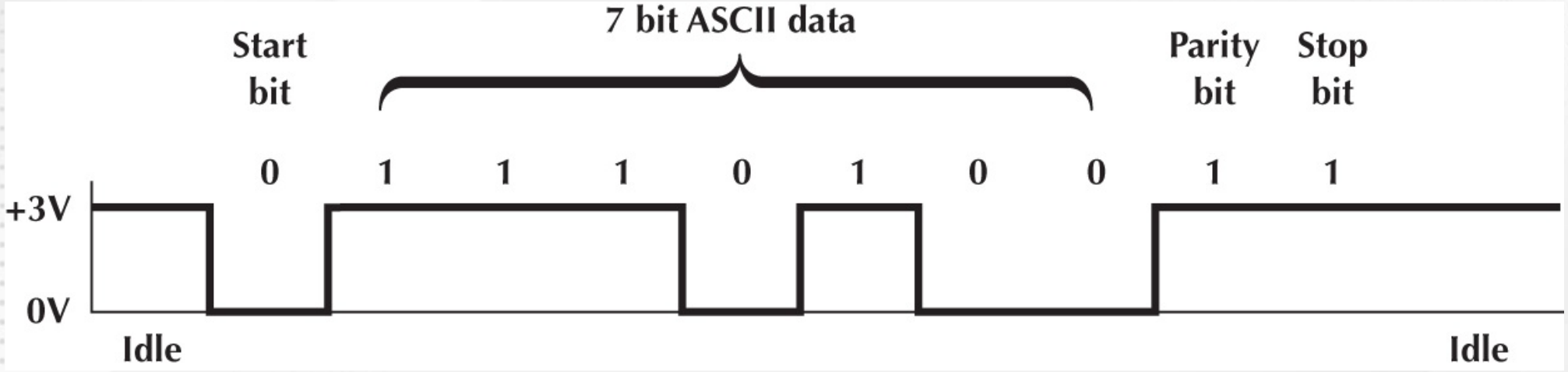
Asenkron iletim genellikle başlangıç-durdurma iletimi olarak adlandırılır çünkü ileten bilgisayar bir karakteri herhangi bir zamanda iletebilir ve alıcı bilgisayar o karakteri kabul eder. Bu tip iletişim genellikle yalnızca iki bilgisayarın olduğu noktadan noktaya tam çift yönlü devrelerde kullanılır. UNIX veya Linux bilgisayarına Telnet kullanarak bağlanıyorsanız, büyük ihtimalle asenkron iletim kullanıyorsunuzdur.

Asenkron iletimde, her karakter bağımsız olarak iletilir. İletimi ayırmak ve senkronizasyonu sağlamak için her bir karakterin önüne bir başlangıç biti ve arkasına bir durdurma biti konur. Örneğin, çiftli bit bit kontrolü ile 7-bitlik ASCII kullanıyorsak, her karakter için toplam iletim 10 bittir (1 başlangıç biti, harf için 7 bit, bir parity biti ve bir durdurma biti). Başlangıç biti ve durdurma biti birbirinin tersidir. Genellikle başlangıç biti 0 ve durdurma biti 1'dir. Karakterler arasında sabit bir mesafe yoktur çünkü terminal, karakter yazıldığı anda karakteri iletir ve bu, yazan kişinin hızına bağlı olarak değişir. Her bir karakterin başlangıç ve bitişinin (senkronizasyon) tanınması, başlangıç bitinin gelen bitlerin örnekleme işlemini başlatmasıyla karakterlerin veri bitlerinin doğru karakter yapısına yorumlanabilmesini sağlar. Bir durdurma biti, alıcının karakterin alındığını ve bir sonraki başlangıç bitinin tanınması için sıfırladığını bildirir.

Kaynak: https://www.slideshare.net/slideshow/dc-chapter-07/36618933

4. DATA LINK PROTOCOLS (Asenkron İletim)

Gönderici bir sonraki karakteri kullanıcıdan girmesini beklerken hiçbir veri göndermez; iletişim devresi boşta olur. Bu boşta zaman aslında yapaydır - her zaman devreye bir sinyal gönderilmelidir. Örneğin, +5 voltun bir 1'i ve 0 voltun bir 0'ı temsil ettiği tek kutuplu dijital işaretleme tekniğini kullanıyorsak 0 volt göndersek bile, bir sinyal gönderiyoruz, bu durumda bir 0 gönderiyoruz. Gönderici bir harfi ilettiğinde ve gönderecek daha fazla veri beklediğinde, sürekli bir durdurma biti dizisi gönderir. Bazı eski protokoller geleneksel tek durdurma bitinin yerine iki durdurma biti kullanır. Hem bir başlangıç biti hem de bir durdurma biti kullanımı değişiyor; bazı protokoller durdurma bitini tamamen kaldırmıştır.



Kaynak: https://iraqembededsystems.blogspot.com/2015/01/blog-post.html

4. DATA LINK PROTOCOLS (Senkron İletim)

Senkron iletimde, veriler bir veri grubunun tamamı aynı anda bir veri bloğu olarak iletilir. Bu veri bloğuna bir çerçeve adı verilir. Örneğin, bir terminal veya kişisel bilgisayar, kullanıcı tarafından yazılan tüm tuş vuruşlarını saklar ve bunları kullanıcı özel bir "ilet" tuşuna bastığında yalnızca iletilir. Bu durumda, tüm çerçevenin başlangıcı ve sonu işaretlenmelidir, her bir harfin değil. Senkron iletim sıklıkla noktadan-noktaya ve çok noktalı devrelerde kullanılır. Çok noktalı devreler için, her paket bir hedef adres ve bir kaynak adres içermeli ve medya erişim kontrolü önemlidir. Her çerçevenin başlangıcı ve sonu (senkronizasyon) bazen çerçevenin başına senkronizasyon karakterleri (SYN) eklenerek belirlenir. Protokole bağlı olarak, bir ila sekiz SYN karakteri olabilir. SYN karakterlerinden sonra, ileten bilgisayar binlerce bit içerebilecek uzun bir veri akışı gönderir. Alıcı bilgisayar, hangi kodun kullanıldığını bildiğinden, ilk karakter için uygun sayıda bit sayar, bunun ilk karakter olduğunu varsayar ve bilgisayara iletir. Daha sonra ikinci karakter için bitleri sayar ve böyle devam eder.

Özetlemek gerekirse, asenkron veri iletimi her karakterin tamamen bağımsız bir varlık olarak iletilmesi demektir ve her bir karakterin başlangıcını ve sonunu alıcı bilgisayara bildirmek için kendi başlangıç ve durdurma bitlerine sahiptir. Senkron iletim ise gönderici ve alıcı senkronize olduktan sonra tüm veri blokları çerçeve olarak iletilir.

Senkron iletim için birçok protokol vardır. Dört yaygın olarak kullanılan senkron veri bağlantı protokolünü tartışıyoruz.

4. DATA LINK PROTOCOLS (Senkron İletim)

Ethernet, 1973 yılında Bob Metcalfe tarafından tasarlanmış ve 1970'lerde Digital, Intel ve Xerox tarafından birlikte geliştirilmiş çok popüler bir LAN protokolüdür. O zamandan beri, Ethernet daha da geliştirilmiş ve IEEE 802.3ac adında resmi bir standart haline getirilmiştir.

Günümüzde kullanılan birkaç Ethernet versiyonu bulunmaktadır. Ethernet, bir içerikleme ortam erişim protokolü kullanır. Şekilde, bir Ethernet 803.3ac çerçevesini göstermektedir. Çerçeve, birbirini takip eden bir ve sıfırların (10101010) tekrarlayan bir deseni olan 7 baytlik bir başlangıç dizisiyle başlar. Ardından, çerçevenin başlangıcını işaretleyen bir çerçeve başlatıcı ayırıcı gelir. Hedef adres, alıcıyı belirtirken, kaynak adres göndereni belirtir. Uzunluk, çerçevenin mesaj kısmının 8 bitlik baytlar cinsinden uzunluğunu belirtir. VLAN etiket alanı, sanal LAN'ler (VLAN'ler) tarafından kullanılan isteğe bağlı 4 baytlık bir adres alanıdır. Ethernet çerçevesi, VLAN'ler kullanıldığında yalnızca bu alanda kullanılır; aksi takdirde, alan atlanır ve uzunluk alanı hemen kaynak adres alanını takip eder. VLAN etiket alanı kullanıldığında, ilk 2 bayt 24,832 sayısına (onaltılık 81-00) ayarlanır, ki bu açıkça imkansız bir paket uzunluğudur. Ethernet, bu uzunluğu gördüğünde, VLAN etiket alanının kullanımda olduğunu bilir. Uzunluk başka bir değere sahipse, VLAN etiketlerinin kullanılmadığını ve uzunluk alanının hemen kaynak adres alanını takip ettiğini varsayar.

Preamble	Start of Frame	Destination Address	Source Address	VLAN Tag	Length	DSAP	SSAP	Control	Data	Frame Check Sequence
7 bytes	1 byte	6 bytes	6 bytes	4 bytes	2 bytes	1 byte	1 byte	1–2 bytes	46–1,500 bytes	4 bytes

Kaynak: https://www.chegg.com/homework-help/questions-and-answers/transmission-efficiency-1000-byte-file-sent-response-web-request-http-tcp-ip-ethernet-hint-q15114153

5. Ethernet

DSAP ve SSAP, gönderici ile alıcı arasında kontrol bilgisi iletmek için kullanılır. Bunlar, paketin içerdiği ağ katmanı protokolünün türünü belirtmek için sıklıkla kullanılır (örneğin, TCP/IP veya IPX/SPX gibi). Kontrol alanı, hata kontrolü için çerçeve sıra numaralarını ve ACK ve NAK'ları tutmak için kullanılır, ayrıca iletişim halindeki bilgisayarların veri bağlantı katmanlarının diğer kontrol bilgilerini alışveriş etmesini sağlar. İlk baytın son 2 biti, iletilen kontrol bilgisi türünü ve kontrol alanının 1 veya 2 bayt olup olmadığını belirtmek için kullanılır (örneğin, kontrol alanının son 2 biti 11 ise, kontrol alanı 1 bayt uzunluğundadır). Çoğu durumda, kontrol alanı 1 bayt uzunluğundadır. Mesajın maksimum uzunluğu yaklaşık 1,500 bayttır. Çerçeve, hata tespiti için kullanılan bir CRC-32 çerçeve doğrulama dizisiyle sona erer.

Preamble	Start of Frame	Destination Address	Source Address	VLAN Tag	Length	DSAP	SSAP	Control	Data	Frame Check Sequence
7 bytes	1 byte	6 bytes	6 bytes	4 bytes	2 bytes	1 byte	1 byte	1–2 bytes	46–1,500 bytes	4 bytes

Kaynak: https://www.chegg.com/homework-help/questions-and-answers/transmission-efficiency-1000-byte-file-sent-response-web-request-http-tcp-ip-ethernet-hint-q15114153

5. Ethernet

Ethernet II, Ethernet'in başka bir yaygın kullanılan versiyonudur. Çerçevenin başlangıcını işaretlemek için bir ön ek kullanır. Ethernet 802.3ac ile aynı kaynak ve hedef adres formatına sahiptir. Tür alanı, bir ACK çerçevesini belirtmek veya çerçevenin içerdiği ağ katmanı paketinin türünü (örneğin, IP) belirtmek için kullanılır. Veri ve çerçeve kontrol dizisi alanları Ethernet 802.3ac ile aynıdır. Ethernet II'nin bir çerçevenin sonunu işaretlemek için alışılmadık bir yolu vardır. Bir sonraki çerçeve başladığında, gönderen bilgisayarın 96 bit boyunca hiçbir sinyal göndermemesi gerekmektedir (yani, ne 0 ne de 1). Bu 96 bitlik süre boyunca hiçbir sinyal gönderilmemesi, bir başka bilgisayarın devreye girip iletim başlatmasını önlemek amacıyla tasarlanmıştır. Yeni Ethernet türlerinin bazıları, mesaj alanında 9.000 bayte kadar kullanıcı verisini içeren devasa çerçevelere izin verir. Bazı satıcılar, 64.000 bayte kadar kullanıcı verisi tutabilen süper devasa çerçevelerle deneyler yapmaktadır. Devasa çerçeveler, gigabit Ethernet gibi bazı Ethernet türleri için yaygındır.

Preamble	Start of Frame	Destination Address	Source Address	Type	Data	Frame Check Sequence
7 bytes	1 byte	6 bytes	6 bytes	2 bytes	46–1,500 bytes	4 bytes

Kaynak: https://www.chegg.com/homework-help/questions-and-answers/transmission-efficiency-1000-byte-file-sent-response-web-request-http-tcp-ip-ethernet-hint-q15114153

6. İletişim Verimliliği

Bir veri iletişim ağı hedeflerinden biri, ağ üzerinden en yüksek miktarda doğru bilgiyi taşımaktır. Yüksek hacimli veri iletimi, sonuç olarak ağın verimliliğini artırır ve maliyeti düşürür. Ağ verimliliği, devrelerin hata oranları ve maksimum iletim hızı gibi özelliklerden etkilenir; ayrıca veri iletim ve alım ekipmanlarının hızı, hata tespiti ve kontrol yöntemi ve veri bağlantı katmanı tarafından kullanılan protokol de etkilidir.

Tartıştığımız her protokol, her bir iletişim mesajının başlangıcını ve sonunu belirlemek ve hata kontrolünü yönetmek için bazı bit veya baytlar kullanır. Bu bitler ve baytlar iletimin gerçekleşmesi için gereklidir, ancak mesajın bir parçası değildir. Kullanıcıya bir değer katmazlar, ancak iletilen toplam bit sayısına karşı sayılırlar.

Her iletişim protokolünün hem bilgi bitleri hem de başlık bitleri vardır. Bilgi bitleri, kullanıcının anlamını iletmek için kullanılır. Başlık bitleri, hata kontrolü gibi amaçlar için kullanılır ve karakterlerin ve paketlerin başlangıcını ve sonunu işaretlemek için kullanılır. Hata denetimi için kullanılan bir çiftlik biti bir başlık bitidir çünkü kullanıcı verilerini göndermek için kullanılmaz; hata ile ilgilenmiyorsanız, başlık hata denetimi biti atlanabilir ve kullanıcılar hala mesajı anlayabilirler.
İletim verimliliği, toplam bilgi biti sayısı (yani, kullanıcı tarafından gönderilen mesajdaki bitler) ile iletimdeki toplam bitlerin (yani, bilgi bitleri artı başlık bitleri) bölünmesiyle tanımlanır. Örneğin, asenkron iletimin iletim verimliliğini hesaplayalım. 7-bit ASCII kullandığımızı varsayalım. Bir doğruluk biti için 1 bitimiz, bir başlangıç biti ve bir duraklama bitimiz var. Bu nedenle, her harf için 7 bit bilgi vardır, ancak toplam bitler her harf için 10'dur (7 + 3). Asenkron iletim sisteminin verimliliği, 7 bit bilgiyi toplam 10 bit veya% 70'te bölerek hesaplanır.

Doç. Dr. Şaban Öztürk

saban.ozturk@hbv.edu.tr

6. İletişim Verimliliği

Başka bir deyişle, asenkron iletimde, veri oranının sadece %70'i kullanıcı için kullanılabilir; %30'u iletim protokolü tarafından kullanılır. 56 Kbps alım yapan bir modem kullanan bir iletişim devresine sahipsek, kullanıcı etkili bir veri oranı (veya verimlilik) olan 39,2 Kbps görür. Bu oldukça verimsizdir.

Her mesajdaki başlık bitlerinin sayısını azaltarak veya bilgi bitlerinin sayısını artırarak verimliliği artırabiliriz. Örneğin, asenkron iletimden duraklama bitlerini kaldırırsak, verimlilik %77.8'e yükselir. 56 Kbps hızında bir dial-up modem iletim hızı, 43.6 Kbps artar, bu çok büyük değil ama en azından biraz daha iyidir.

Aynı temel formül, senkron iletimin verimliliğini hesaplamak için de kullanılabilir. Bilgi bitlerinin sayısı, iletilen ileti içindeki bilgi karakterlerinin kaç tane olduğunu belirleyerek hesaplanır. Eğer bir çerçevenin ileti kısmında 100 bilgi karakteri bulunuyorsa ve 8-bitlik bir kod kullanıyorsak, o zaman 100 × 8 = 800 bilgi biti vardır. Toplam bit sayısı, 800 bilgi bitine ek olarak belirleme ve hata kontrolü için eklenen fazlalık bitlerdir. Başlangıç bayrağı (8 bit), bir adres (8 bit), bir kontrol alanı (8 bit), bir çerçeve kontrol dizisi (32 bitlik bir CRC-32 kullanıldığını varsayalım), ve bir bitiş bayrağı (8 bit) içerdiğini gösterir. Bu toplamda 64 fazlalık bit olduğuna göre, verimlilik 800/(800 + 64) = 92.6% olur. Eğer devre, kullanıcıya 56 Kbps veri hızı sağlıyorsa, o zaman kullanıcı için etkili veri hızı yaklaşık olarak 51.9 Kbps olur.

6. İletişim Verimliliği

Bu örnek, senkron ağların genellikle asenkron ağlardan daha verimli olduğunu ve bazı protokollerin diğerlerinden daha verimli olduğunu göstermektedir. Mesajın uzunluğu ne kadar uzun olursa (örneğin, 1000 karaktere karşılık olarak 100), protokol o kadar verimli olur. Örneğin, ileti 1000 byte içeriyorsa, verimlilik burada 99.2% veya 8000/(8000 + 64) olacak, böylece etkili veri hızı yaklaşık olarak 55.6 Kbps olacaktır.

Genel kural, mesaj alanı ne kadar büyük olursa, protokol o kadar verimli olur. Peki neden gerçekten verimliliği artırmak için 10,000 veya hatta 100,000 byte'lık paketler oluşturmuyoruz?

saban.ozturk@hbv.edu.tr

6. İletişim Verimliliği

Cevap, bir hatanın bulunduğu bir çerçeve her zaman yeniden gönderilmelidir. Dolayısıyla, bir bütün dosya bir büyük paket olarak (örneğin, 100 K) gönderilirse ve 1 bit hata alınırsa, tüm 100,000 byte tekrar gönderilmelidir. Açıkça, bu bir kapasite israfıdır. Dahası, bir çerçevenin bir hataya sahip olma olasılığı çerçevenin boyutuyla artar; daha büyük çerçeveler, daha küçük olanlardan daha fazla hata içerme eğilimindedir, sadece olasılık kuralları gereği.

saban.ozturk@hbv.edu.tr

6. İletişim Verimliliği

Throughput, hatalı çerçeveleri yeniden iletmeyi ve fazladan bitleri hesaba katarak alınan toplam bilgi bitlerinin saniyedeki miktarıdır. Genel olarak, daha fazla hataya sahip devreler için küçük çerçeveler daha iyi bir throughput sağlar, daha az hata içeren ağlar için ise daha büyük çerçeveler daha iyi bir throughput sağlar. Neyse ki, gerçek ağlarda çoğunlukla, Şekilde gösterilen eğri çok düzgündür, bu da neredeyse optimum performans sağlayan bir çerçeve boyutu aralığı bulunduğunu gösterir. Çerçeve boyutları farklı ağlar arasında büyük ölçüde değişir, ancak ideal çerçeve boyutu genellikle 2,000 ve 10,000 byte arasında olur.

7. Hata Kontrolü

Ağı hatalardan korumak için uygulanabilen kontrol mekanizmalarını öğrenmeden önce, insan hataları ve ağ hataları olduğunu fark etmelisiniz. Bir numarayı yanlış yazma gibi insan hataları genellikle uygulama programı aracılığıyla kontrol edilir. İletim sırasında meydana gelenler gibi ağ hataları ise ağ donanımı ve yazılımı tarafından kontrol edilir.

Ağ hatalarının iki kategorisi vardır: bozulmuş veri (değiştirilmiş veriler) ve kaybolmuş veriler. Ağlar, hem bozulmuş verileri hem de kaybolmuş verileri önlemek, tespit etmek ve düzeltmek için tasarlanmalıdır. Öncelikle hataların kaynaklarını ve bunları nasıl önleyebileceğimizi inceleyerek başlayacağız, ardından hata tespiti ve düzeltmeye geçeceğiz.

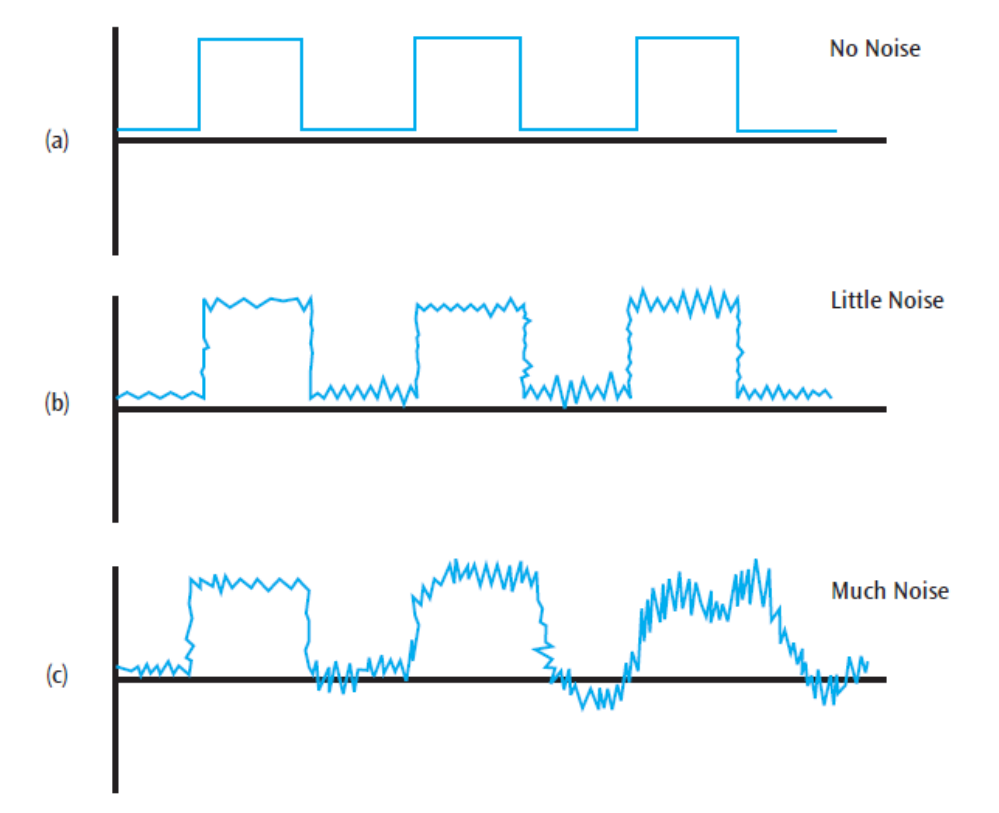
Veri iletişim ağlarında ağ hataları kaçınılmaz bir gerçektir. Devre tipine bağlı olarak, hatlardaki gürültü nedeniyle her birkaç saatte, dakikada veya saniyede bir oluşabilirler. Hiçbir ağ tüm hataları ortadan kaldıramaz, ancak çoğu hatayı uygun tasarım ile önlemek, tespit etmek ve düzeltmek mümkündür. Veri iletim devreleri sağlayan İnter-Exchange Taşıyıcıları (IXC'ler), kiraladıkları devrelerde beklenen tipik hata oranlarını ve hata desenlerini belirten istatistiksel ölçümler sağlarlar.

8.Hata Kaynakları

Hat gürültüsü ve bozulma, veri iletişimi hatalarına neden olabilir. Bu bölümde, gürültüye daha fazla maruz kalma olasılıkları nedeniyle fiber optik kablo gibi optik ortamlardan ziyade, özellikle elektriksel ortamlar olan çift sarmal tel ve koaksiyel kablo üzerinde durulmaktadır. Bu durumda, gürültü istenmeyen elektrik sinyalleridir (fiber optik kablolar için ise istenmeyen ışıktır). Gürültü, ekipman veya doğal bozulmalar tarafından eklenir ve iletişim devresinin performansını düşürür. Gürültü, fazladan bitler, eksik bitler veya bitlerin "ters çevrilmesi" şeklinde kendini gösterir (yani, 1'den 0'a veya tam tersine değiştirilmiş olanlar).

8.Hata Kaynakları

Beyaz gürültü veya Gauss gürültüsü (radyo ve telefonlardaki tanıdık arka plan hışırtısı veya statik) elektronların termal çalkantısı tarafından neden olur ve bu nedenle kaçınılmazdır. Ekipman mükemmel olsa ve teller her türlü harici müdahaleden mükemmel şekilde yalıtılmış olsa bile, hala bir miktar beyaz gürültü olacaktır. Beyaz gürültü genellikle bir sorun değildir, ta ki iletişimi silme derecesine kadar güçlü hale gelene kadar. Bu durumda, elektrik sinyalinin gücü arttırılır, böylece beyaz gürültüyü bastırır; teknik terimlerle ifade edildiği gibi, sinyal-gürültü oranı artırılır.

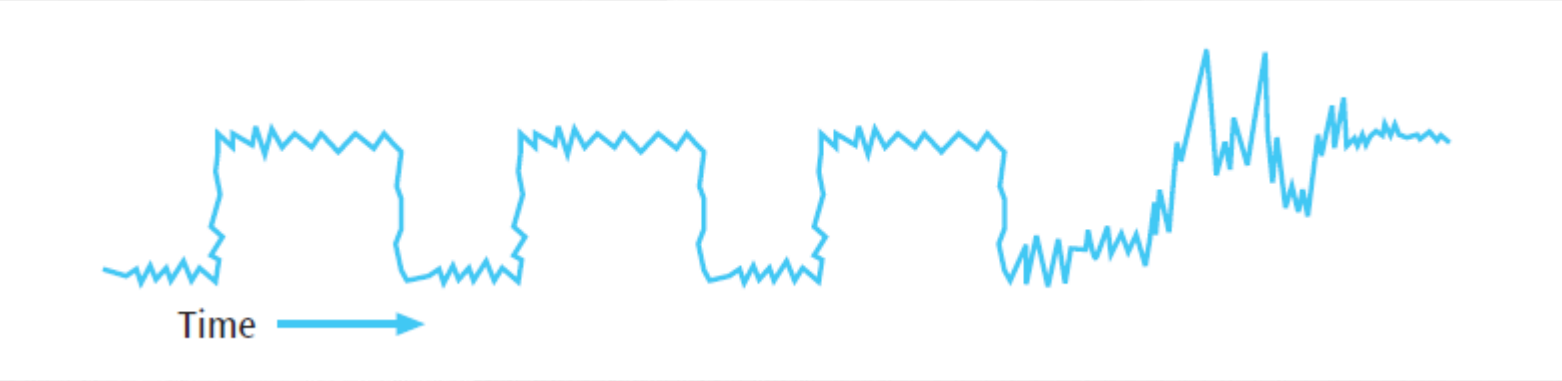


Kaynak: https://quizlet.com/375891977/exam-2-ds4250-ch-6-09-flash-cards/

saban.ozturk@hbv.edu.tr

8.Hata Kaynakları

Anlık gürültü (bazen sivri uçlar olarak adlandırılır), veri iletişimindeki hataların başlıca kaynağıdır. Bir tıklama veya çıtırtı sesi olarak duyulur ve 1/100 saniyeye kadar sürebilir. Bu tür bir tıklama sesi gerçekten sesli iletişimi etkilemez, ancak bir veri grubunu yok ederek bir grup hatası oluşturabilir. 1.5 Mbps hızında, 1/100 saniye süren bir sivri uç, 15.000 biti değiştirebilir. Anlık gürültünün bazı kaynakları, bitişik hatlardaki gerilim değişiklikleri, yıldırım çarpmaları sırasında şimşekler, floresan ışıklar ve devrelerdeki kötü bağlantılardır.

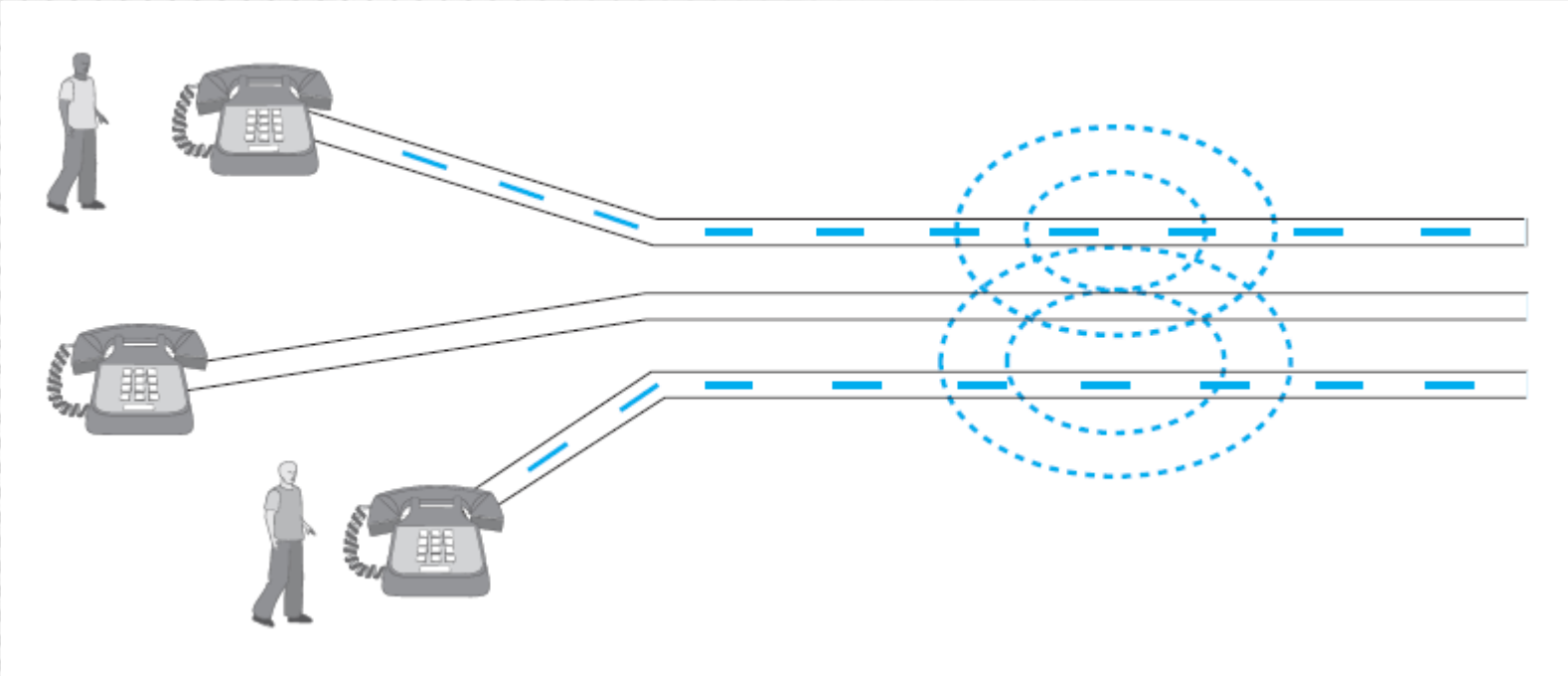


Kaynak:https://gnindia.dronacharya.info/ECE/7thSem/Downloads/DataCommunicationNetworks/UNIT-3/2.pdf

saban.ozturk@hbv.edu.tr

8.Hata Kaynakları

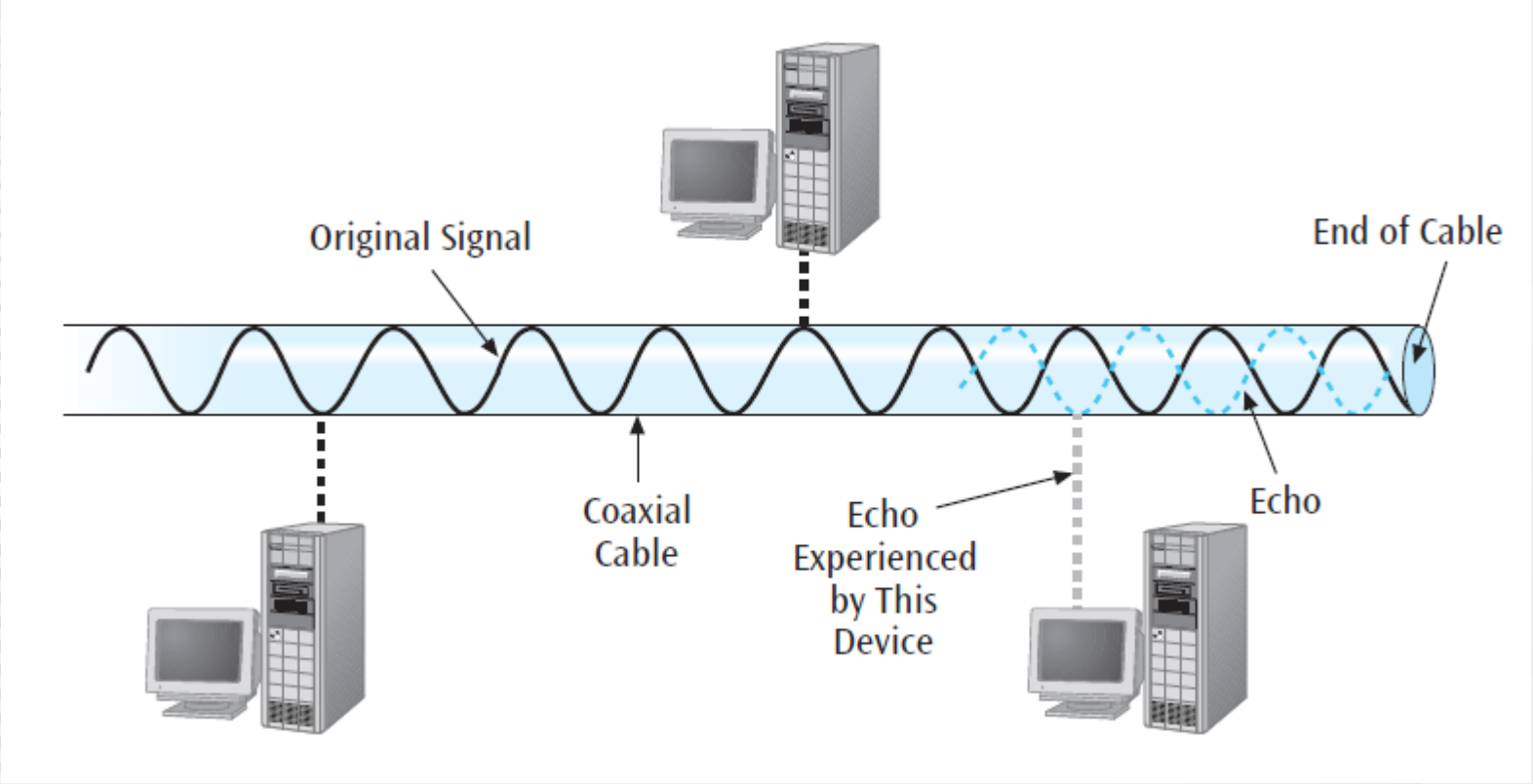
Çapraz veri, bir devrenin başka bir devredeki sinyalleri alması durumunda meydana gelir. Bir kişi, telefon görüşmeleri sırasında arka planda diğer konuşmaları duyduğunda çapraz konuşma yaşar. Ayrı sinyaller taşıyan kablolar arasında, birçok ayrı sinyal taşıyan çoklayıcı bağlantılarda veya bir antenin başka bir antenden minik bir yansımayı aldığı mikrodalga bağlantılarında meydana gelir. Hatlar arasındaki çapraz konuşma, iletişim mesafesinin artması, iki kablonun yakınlığının artması, sinyal gücünün artması ve daha yüksek frekanslı sinyallerle artar. Islak veya nemli hava da çapraz konuşmayı artırabilir. Beyaz gürültü gibi, çapraz konuşma normalde rahatsız edici olmayacak kadar düşük bir sinyal gücüne sahiptir.



Kaynak: https://slideplayer.com/slide/8391943/

8.Hata Kaynakları

Yankılar, sinyalin geri yansımasına neden olan kötü bağlantıların bir sonucudur. Yankının gücü tespit edilecek kadar güçlü ise hatalara neden olur. Çapraz konuşma ve beyaz gürültü gibi, yankılar da genellikle rahatsız edici olmayacak kadar düşük bir sinyal gücüne sahiptir. Yankılar, fiber optik kablolar arasındaki bağlantıların düzgün bir şekilde hizalanmaması durumunda da meydana gelebilir.



Kaynak: https://slideplayer.com/slide/8579571/

8.Hata Kaynakları

Zayıflama, bir sinyalin iletim bilgisayarından alıcı bilgisayara kadar seyahat ettiği süre boyunca kaybettiği güçtür. Bazı güç, ortam tarafından emilir veya alıcıya ulaşmadan önce kaybolur. Ortam güç emdikçe, sinyal zayıflar ve alıcı ekipmanı verileri doğru bir şekilde yorumlama şansına sahip olur. Bu güç kaybı, iletim yöntemi ve devre ortamının bir fonksiyonudur. Yüksek frekanslar, düşük frekanslardan daha hızlı güç kaybeder, bu nedenle alınan sinyal bileşen frekanslarının eşitsiz kaybıyla bozulabilir. Zayıflama, frekans arttıkça veya tel çapı azaldıkça artar.

Intermodülasyon gürültüsü, özel bir çapraz konuşma türüdür. İki devreden gelen sinyaller, başka bir sinyal için ayrılmış bir frekans bandına düşen yeni bir sinyal oluşturur. Bu tür gürültü, müzikteki harmoniklere benzer. Birleştirilmiş bir hat üzerinde, birçok farklı sinyal birlikte artırılır ve ekipmanın ayarlarında hafif değişiklikler, intermodülasyon gürültüsüne neden olabilir. Hatalı ayarlanmış bir modem, veri iletimi olmadığında güçlü bir frekans tonu iletebilir ve bu şekilde bu tür bir gürültü oluşturabilir.

8.Hata Kaynakları

Duruma bağlı olarak hataları önlemenin (veya en azından azaltmanın) birçok tekniği bulunmaktadır. **Ekranlama** (telleri yalıtkan bir kaplama ile koruma), ani gürültü, çapraz konuşma ve aramodülasyon gürültüsünü önlemenin en iyi yollarından biridir. Farklı miktarlarda ekranlama sağlayan birçok farklı türde tel ve kablo bulunmaktadır. Genel olarak, ekranlama ne kadar fazlaysa, kablo o kadar pahalı ve kurulumu o kadar zordur.

Kabloları gürültü kaynaklarından uzaklaştırmak (özellikle güç kaynaklarından) ani gürültü, çapraz konuşma ve aramodülasyon gürültüsünü azaltmanın bir yoludur. Ani gürültü için bu, ışıklardan ve ağır makinelerden kaçınmayı içerir. İletişim kablolarını güç kablolarından uzakta tutmak her zaman iyi bir fikirdir. Çapraz konuşma için, bu, kabloları diğer iletişim kablolarından fiziksel olarak ayırmayı gerektirir.

Çapraz konuşma ve ara modülasyon gürültüsü genellikle uygun olmayan çoklayıcı kullanımından kaynaklanır. **Çoklama tekniklerinin değiştirilmesi** (örneğin, FDM [Frekans Bölümleme Çoklaması] den TDM [Zaman Bölümleme Çoklaması] veya FDM'deki koruma bandlarının frekanslarının veya boyutlarının değiştirilmesi) yardımcı olabilir.

8.Hata Önleme

Birçok tür gürültü (örneğin, yankılar, beyaz gürültü), kötü bakımı yapılan ekipman veya kablolar arasındaki kötü bağlantılar ve birleştirmelerden kaynaklanabilir. Bu, özellikle fiber optik kabloların yankı durumu için geçerlidir, bu durum neredeyse her zaman kötü bağlantılar tarafından oluşturulur. Çözüm burada açıktır: **İletim ekipmanını ayarlayın ve bağlantıları yeniden yapın**.

Ses iletim hatlarında zayıflamayı önlemek için, **tekrarlayıcılar veya amplifikatörler hat boyunca düzenli aralıklarla yerleştirilir**. Onların arasındaki mesafe, iletim hattının birim uzunluğundaki güç kaybına bağlıdır. Bir amplifikatör, gelen sinyali alır, gücünü artırır ve devrenin bir sonraki bölümünde yeniden ileterek güçlendirir. Genellikle, telefon şirketinin ses devreleri gibi analog devrelerde kullanılırlar. Amplifikatörler arasındaki mesafe, zayıflamanın miktarına bağlıdır, ancak genellikle 1 ila 10 mil aralıklarında olur. Analog devrelerde, gürültü ve bozulmanın da sinyalle birlikte arttığını unutmamak önemlidir. Bu, önceki bir devreden gelen bazı gürültünün her seferinde sinyal güçlendirildiğinde yeniden üretildiği ve amplifiye edildiği anlamına gelir.

8.Hata Önleme

Table 6-1 Summary of errors and error-prevention techniques

Type of Error	Error-Prevention Technique
White noise	Install special filters for analog signals; implement digital signal regeneration for digital signals
Impulse noise	Install special filters for analog signals; implement digital signal processing for digital signals
Crosstalk	Install proper shielding on cables
Echo	Install proper termination of cables
Jitter	Use better-quality electronic circuitry, use fewer repeaters, slow the transmission speed
Attenuation*	Install device that amplifies analog signals; implement digital signal regeneration of digital signals

*Not a type of error, but indirectly affects error

Kaynak: https://www.slideserve.com/elina/chapter-6-errors-error-detection-and-error-control

saban.ozturk@hbv.edu.tr

8. Hata Tespiti

Çok yüksek hata tespiti performansı sunan veri iletim metodolojileri geliştirmek mümkündür. Hata tespiti yapmanın tek yolu, her mesajla ekstra veri göndermektir. Bu hata tespiti verileri, gönderenin veri bağlantı katmanı tarafından mesaj üzerinde yapılan bazı matematiksel hesaplamalara dayanarak her mesaja eklenir (bazı durumlarda, hata tespiti yöntemleri donanımın kendisine entegre edilmiştir). Alıcı, aldığı mesaj üzerinde aynı matematiksel hesaplamaları yapar ve sonuçlarını mesajla birlikte iletilen hata tespiti verileriyle eşleştirir. Eğer ikisi eşleşirse, mesajın doğru olduğu varsayılır. Eğer eşleşmezlerse, bir hata oluşmuştur.

Genel olarak, gönderilen hata tespiti verisinin miktarı ne kadar büyükse, hata tespit etme yeteneği o kadar fazladır. Ancak, hata tespiti verisi miktarı arttıkça, kullanışlı verinin veri kapasitesinden daha azının gerçek mesajı iletmek için kullanıldığından, kullanışlı verinin veri akışı azalır. Dolayısıyla, veri akışının verimliliği, istenen hata tespit miktarı arttıkça ters orantılı olarak değişir.

Üç iyi bilinen hata tespit yöntemi; parity checking (Parite Kontrolü), checksum (Kontrol Toplamı) ve cyclic redundancy checking (Döngüsel Redundans Kontrolü)'dir.

8. Hata Tespiti

Parity Checking (Parite Kontrolü): Parity checking, her veri iletiminde eklenen bir parite biti kullanır. Bu parite biti, verideki 1'lerin sayısını çift veya tek yaparak bir çiftlik kontrolü sağlar. Alıcı, veriyi aldığında parite bitini de kontrol eder ve eğer iletilen veride bir hata varsa, parite biti bunu belirler.

Parity, en eski ve en basit hata tespit yöntemlerinden biridir. Bu teknikle, iletilen her byte için bir ekstra bit eklenir. Bu ekstra parite bitinin değeri, iletilen her byte içindeki 1'lerin sayısına dayanır. Bu parite biti, byte içindeki toplam 1 sayısını (parite biti dahil) ya çift bir sayıya ya da tek bir sayıya yapmak üzere ayarlanır.

Pariteyle herhangi bir tek hata (bir 1'in 0'a veya tam tersine dönüşümü) algılanır, ancak hangi bitin hatalı olduğunu belirleyemez. Bir hata olduğunu bileceksiniz, ancak hatanın ne olduğunu bilemeyeceksiniz. Ancak iki bit değiştiğinde, parite kontrolü herhangi bir hatayı tespit etmeyecektir. Paritenin yalnızca tek sayıda bitin değiştirildiği durumlarda hataları tespit edebileceği açıktır; çünkü çift sayıda hata birbirini iptal eder. Bu nedenle, bir hata meydana geldiğinde bir hatayı tespit etme olasılığı yaklaşık olarak %50'dir. Bugün birçok ağ, düşük hata tespit oranından dolayı pariteyi kullanmamaktadır. Parite kullanıldığında, protokoller genellikle tek parite veya çift parite olarak tanımlanır.

8. Hata Tespiti

Parity Checking (Parite Kontrolü):

Table 6-2 Simple example of longitudinal parity

	Data							Parity
Row 1	1	1	0	1	0	1	1	1
Row 2	1	1	1	1	1	1	1	1
Row 3	0	1	0	1	0	1	0	1
Row 4	0	0	1	1	0	0	1	1
Parity Row	0	1	0	0	1	1	1	0

Table 6-3 The second and third bits in Rows 1 and 2 have errors, but longitudinal parity does not detect the errors

	Data							Parity
Row 1	1	+0	−1	1	0	1	1	1
Row 2	1	+0	+0	1	1	1	1	1
Row 3	0	1	0	1	0	1	0	1
Row 4	0	0	1	1	0	0	1	1
Parity Row	0	1	0	0	1	1	1	0

Kaynak: https://slideplayer.com/slide/5117722/

saban.ozturk@hbv.edu.tr

8. Hata Tespiti

Checksum (Kontrol Toplamı): Checksum, bir veri paketindeki tüm veri byte'larının toplamının belirli bir değere eşit olup olmadığını kontrol eder. Veri paketi gönderilirken, gönderen bu toplamı hesaplar ve alıcıya gönderir. Alıcı, veriyi aldığında toplamı yeniden hesaplar ve gönderilen toplamla karşılaştırır. Eğer toplamlar eşleşmezse, veride bir hata olduğu anlaşılır.

Checksum tekniğiyle, bir kontrol toplamı (genellikle 1 byte) mesajın sonuna eklenir. Kontrol toplamı, mesajdaki her karakterin ondalık değeri toplanarak, toplamın 255'e bölünmesi ve kalanın kontrol toplamı olarak kullanılmasıyla hesaplanır. Alıcı, kendi kontrol toplamını aynı şekilde hesaplar ve iletilen kontrol toplamı ile karşılaştırır. İki değer eşitse, mesajın hata içermediği varsayılır. Kontrol toplamı kullanımı, çoklu bit hızında hatalar için hataların yaklaşık %95'ini tespit eder.

8. Hata Tespiti

Checksum (Kontrol Toplamı): "This is cool." iletilmek istenirse, ASCII'de bu ileti ikili olarak şöyle görünecektir: 1010100 1101000 1101001 1110011 0100000 1101001 1110011 0100000 1100011 1101111 1101111 1101100 0101110.

İlk ikili değer - 1010100 - ondalıkta 84 değerine eşittir. 1101000 ise 104'e eşittir. Bir sonraki ikili değer 1101001, 105'e eşittir; 1110011, 115'e eşittir; 0100000, 32'ye eşittir; 1101001, 105'e eşittir; 1110011, 115'e eşittir; 0100000, 32'ye eşittir; 1100011, 99'a eşittir; 1101111, 111'e eşittir; tekrar 1101111, yine 111'e eşittir; 1101100, 108'e eşittir; ve 0101110, 46'ya eşittir.

Toplam 1167, ardından çıkış mesajına bir şekilde eklenir ve alıcıya gönderilir. Alıcı aynı karakterleri alır, ASCII değerlerini ekler ve iletişim sırasında hiçbir hata olmadıysa, aynı toplam olan 1167'yi elde etmelidir.

8. Hata Tespiti

Cyclic Redundancy Checking (Döngüsel Redundans Kontrolü): Cyclic redundancy checking (CRC), verideki bir dizi bit üzerinde belirli bir matematiksel işlem yaparak bir kontrol değeri üretir. Bu kontrol değeri, veri paketinin bir tür parmak izi gibidir. Gönderen, veri paketini göndermeden önce CRC değerini hesaplar ve alıcıya gönderir. Alıcı, veriyi aldığında aynı matematiksel işlemi tekrar uygular ve hesaplanan CRC değeri ile gönderilen CRC değerini karşılaştırır. Eğer bu değerler eşleşmezse, veride bir hata olduğu anlaşılır.

9. Hata Düzeltme

Bir hata tespit edildikten sonra düzeltilmelidir. Hatanın düzeltilmesi için en basit, en etkili, en az maliyetli ve en yaygın olarak kullanılan yöntem **yeniden iletim**dir.

İleri hata düzeltme, alıcı uçta hataları tespit edip düzeltmek için iletimin yeniden yapılmasına gerek kalmadan orijinal mesajı koruyan yeterli fazlalığı içeren kodları kullanır. Fazlalık veya gereken ek bitler farklı şemalara göre değişir. Eklenen ek bitlerin oranı, veri bitlerinin sayısına yaklaşık olarak eşit olan hata tespit etme bitlerinin sayısından küçük bir yüzdeye kadar değişir. Birçok hata düzeltme kodunun özelliklerinden biri, hataların arka arkaya gelmeden önce en az bir minimum sayıda hatasız bit olması gerektiğidir.

İleri hata düzeltme genellikle uydu iletiminde kullanılır. Dünya istasyonundan uyduya ve geri dönüşteki turda önemli bir gecikme bulunmaktadır. Hata oranları, ekipmanın durumuna, güneş lekelerine veya hava durumuna bağlı olarak değişebilir. Gerçekten de, bazı hava koşulları hatalar olmadan iletim yapmayı imkansız hale getirir ve bu nedenle ileri hata düzeltme hayati öneme sahiptir. Uydu ekipmanı maliyetleriyle karşılaştırıldığında, ileri hata düzeltmenin ek maliyeti önemsizdir.

9. Hata Düzeltme

Stop-and-wait hata kontrolü genellikle durdur ve bekle akış kontrol protokolüyle ilişkilendirilen bir tekniktir. Bir iş istasyonu (A İstasyonu), başka bir iş istasyonuna (B İstasyonu) bir veri paketi gönderir ve ardından durur ve B İstasyonundan bir yanıt bekler. Bu noktada dört şey olabilir.

İlk olarak, veri paketi hatasız bir şekilde ulaşırsa, B İstasyonu ACK gibi olumlu bir onay ile yanıt verir. A İstasyonu bir ACK aldığında, bir sonraki veri paketini iletmektedir.

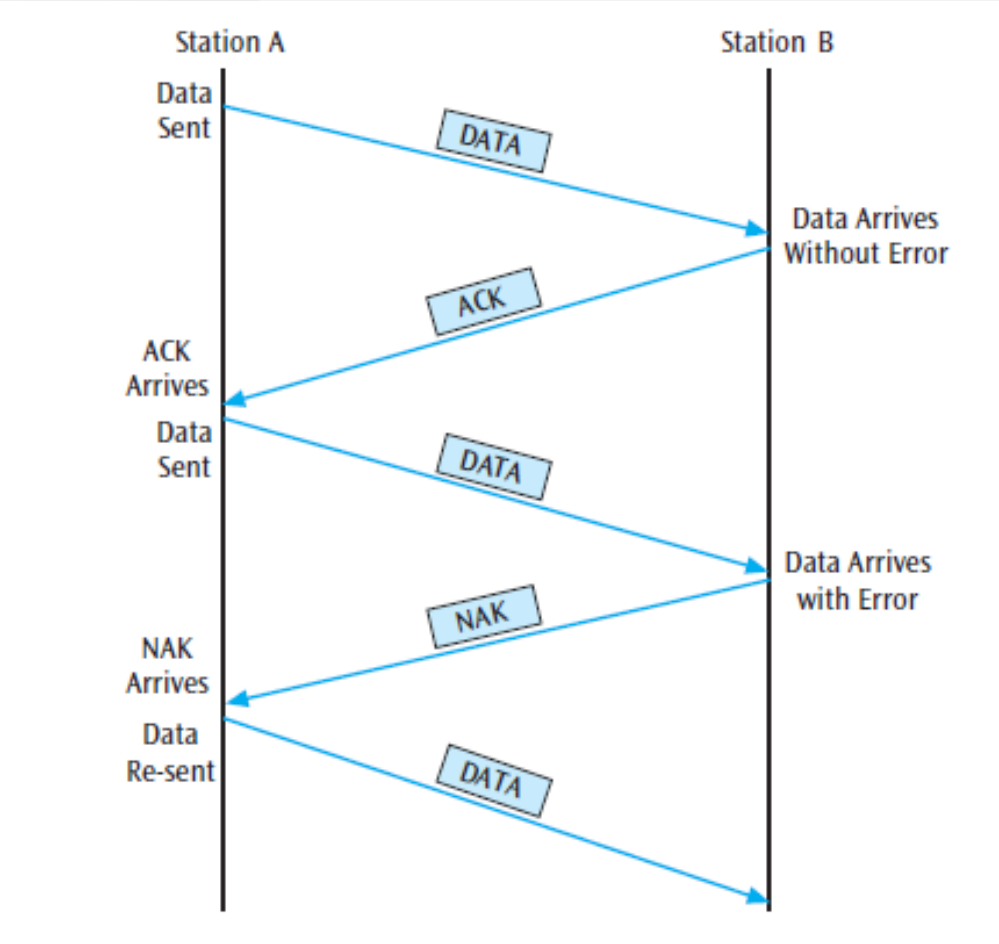
İkincisi, veri bir hatayla gelirse, B İstasyonu NAK veya REJ (reddetmek için) gibi bir negatif bir onay ile yanıt verir. A İstasyonu bir NAK aldığında, önceki veri paketini yeniden gönderir.

Üçüncüsü, bir paket B İstasyonuna bozulmadan ulaşır, B İstasyonu bir ACK ile yanıt verir, ancak ACK kaybolur veya bozulur. A İstasyonu bir onay formu beklemelidir, bu nedenle daha fazla paket iletemez. Belirli bir süre sonra (zaman aşımı denir), A İstasyonu son paketi yeniden gönderir. Ancak bu paket şimdi bozulmadan B İstasyonuna ulaşırsa, B İstasyonu bunun son aldığı paketle aynı paket olduğunu bilemez. Bu karışıklığı önlemek için, paketler sırasıyla 0, 1, 0, 1 ve benzeri şekilde numaralandırılır. A İstasyonu paket 0'ı gönderirse ve paket 0 için ACK kaybolursa, A İstasyonu paket 0'ı yeniden gönderir. B İstasyonu iki paket 0'ı arka arkaya (orijinal ve kopya) fark eder ve ilk paket 0'ın ACK'sinin kaybolduğunu çıkarır.

Dördüncüsü, A İstasyonu bir paket gönderir, ancak paket kaybolur. Paket B İstasyonuna ulaşmadığı için, B İstasyonu bir ACK döndürmez. A İstasyonu bir ACK almazsa, zaman aşımı yapar ve önceki paketi yeniden gönderir. Örneğin, A İstasyonu paket 1'i gönderir, zaman aşımına uğrar ve paket 1'i yeniden gönderir. Bu paket 1, B İstasyonuna ulaştığında, B İstasyonu bir ACK ile yanıt verir. A İstasyonu ACK'nin ilk paketi mi yoksa ikinciyi mi doğruladığını nasıl bilecek? Karışıklığı önlemek için, ACK'ler, paketler gibi numaralandırılır. Ancak, paketlerin (sırasıyla 0, 1, 0, 1 ve benzeri) aksine, ACK'ler 1, 0, 1, 0 ve benzeri şekilde numaralandırılır.

9. Hata Düzeltme

Basit durdur ve bekle hata kontrolünün en ciddi dezavantajlarından biri, yüksek derecede verimsiz olmasıdır. Durdur ve bekle hata kontrolü, yarı çift yönlü bir protokoldür, yani sadece bir istasyon aynı anda iletim yapabilir. İleten istasyonun bir onay beklerken harcadığı zaman, ek paketler göndermek için daha iyi harcanabilirdi. Durdur ve bekle yönteminden daha verimli teknikler mevcuttur. Bunlardan biri kaydırma penceresi tekniğidir.



Kaynak: https://www.chegg.com/homework-help/data-communications-and-computer-networks-7th-edition-chapter-6-problem-19rq-solution-9781285225869

9. Hata Düzeltme

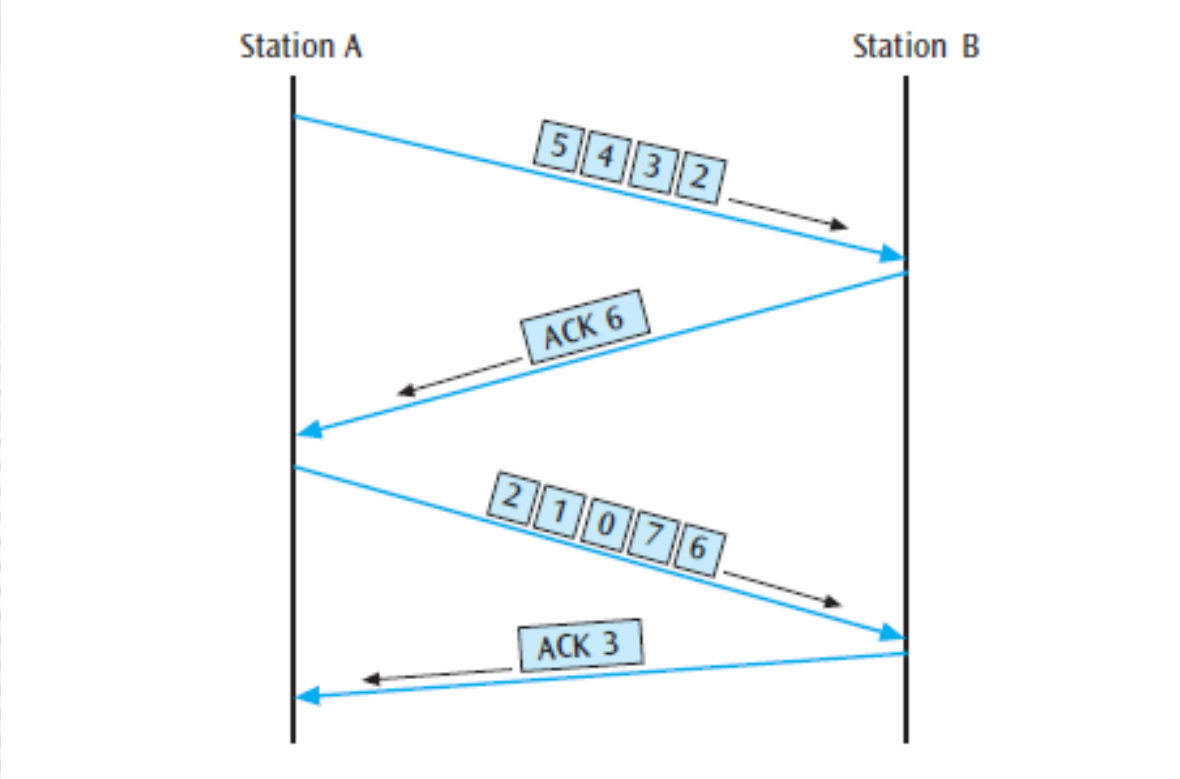
Kaydırma penceresi hata kontrolü, bir istasyonun herhangi bir onay almadan önce bir dizi veri paketini iletebilmesine izin veren akış kontrolü düzeni olan kaydırma penceresi protokolüne dayanmaktadır. Basitlik için, aşağıdaki örnekler maksimum 7 paketlik bir pencere boyutu olan standart protokolü dikkate alacaktır. 7'lik pencere boyutuyla bir kaydırma penceresi protokolündeki veri akışını takip etmek için paketler sırasıyla 0, 1, 2, 3, 4, 5, 6 ve 7 numaralandırılır. Paket numarası 7 iletim edildiğinde, numaralandırma sıfırdan tekrar başlar. Paketler 0'dan 7'ye numaralandırılmış olsa da, sekiz farklı pakete karşılık gelseler de, bir seferde yalnızca yedi veri paketi beklenir (onaylanmamış). Bir seferde en fazla yedi veri paketi beklenir olduğu için, aynı numaraya sahip iki veri paketi (örneğin, her ikisi de 4 numaralı) asla aynı anda iletilmez. Bir göndericinin maksimum 7 paketlik bir pencere boyutuna sahip olduğunu ve dört paket gönderdiğini varsayalım, hala bir onay beklemek zorunda kalmadan üç paket daha gönderebilir. Alıcı, göndericinin daha fazla veri göndermeden önce dört paketi de onaylarsa, göndericinin pencere boyutu tekrar 7'ye döner. Göndericinin beş paket gönderip durduğu bir senaryoyu düşünelim. Alıcı beş paketi alır ve onları onaylar. Onay alınmadan önce, pencere boyutu 7 olduğu için gönderici iki daha fazla paket gönderebilir. Onay alındığında, gönderici yeniden durmadan önce ek yedi paket gönderebilir. Alıcının göndericiye gönderdiği onay da numaralandırılır.

Doç. Dr. Şaban Öztürk

saban.ozturk@hbv.edu.tr

9. Hata Düzeltme

Kaydırma penceresi hata kontrolü,

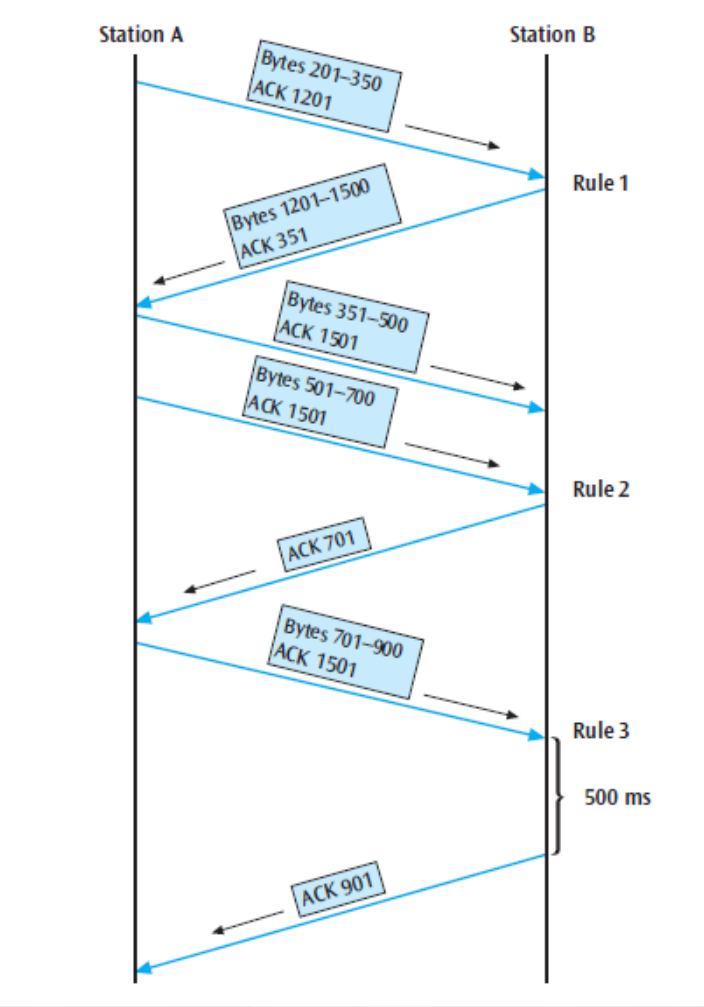


Kaynak: https://slideplayer.com/slide/13491092/

9. Hata Düzeltme

Kaydırma penceresi hata kontrolü,

Alıcı, bir şey alındığında her zaman veriyi onaylamalı mı? Yoksa bir alıcı, bir onay göndermeden önce bir süre daha bir şeyin gelip gelmediğini görmek için bekleyebilir mi?



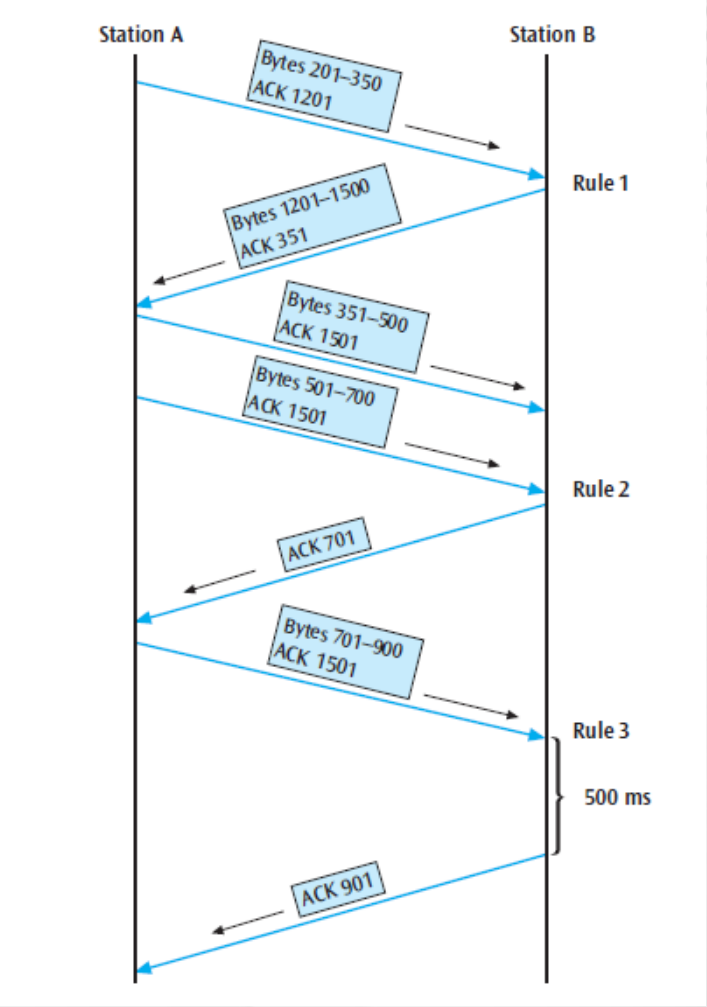
Kaynak: https://slideplayer.com/slide/13491092/

saban.ozturk@hbv.edu.tr

9. Hata Düzeltme

Kaydırma penceresi hata kontrolü,

TCP/IP dünyasında, alıcı istasyonlar bu soruyu çözmek için birkaç kuralı takip ederler. İlk kural (Şekil'de gösterildiği gibi) bir alıcının biraz önce veri aldığını ve veriyi geri göndermek istediğini varsayarsak, alıcının göndereceği veri ile bir ACK eklemesi gerektiğidir. Buna sırt çantası yapıştırma denir ve alıcının ayrı bir ACK mesajı göndermesinden kurtarır. İkinci kural ise alıcının göndericiye geri dönecek herhangi bir veriye sahip olmadığı ve alıcının önceki gönderilen bir veri paketinin alındığını yeni kabul ettiğinde, alıcının başka bir paketin gelip gelmediğini görmek için 500 milisaniye beklemesi gerektiğidir. Ancak, 500 milisaniye dolmadan önce ikinci bir paket gelirse, alıcı hemen bir ACK göndermelidir. Son olarak, üçüncü kural, alıcının ikinci bir paketin gelmesini beklediği ve 500 milisaniyenin dolması durumunda, alıcının ikinci bir paketi beklememesi ve bunun yerine hemen bir ACK göndermesi gerektiğini belirtir.



Kaynak: https://slideplayer.com/slide/13491092/

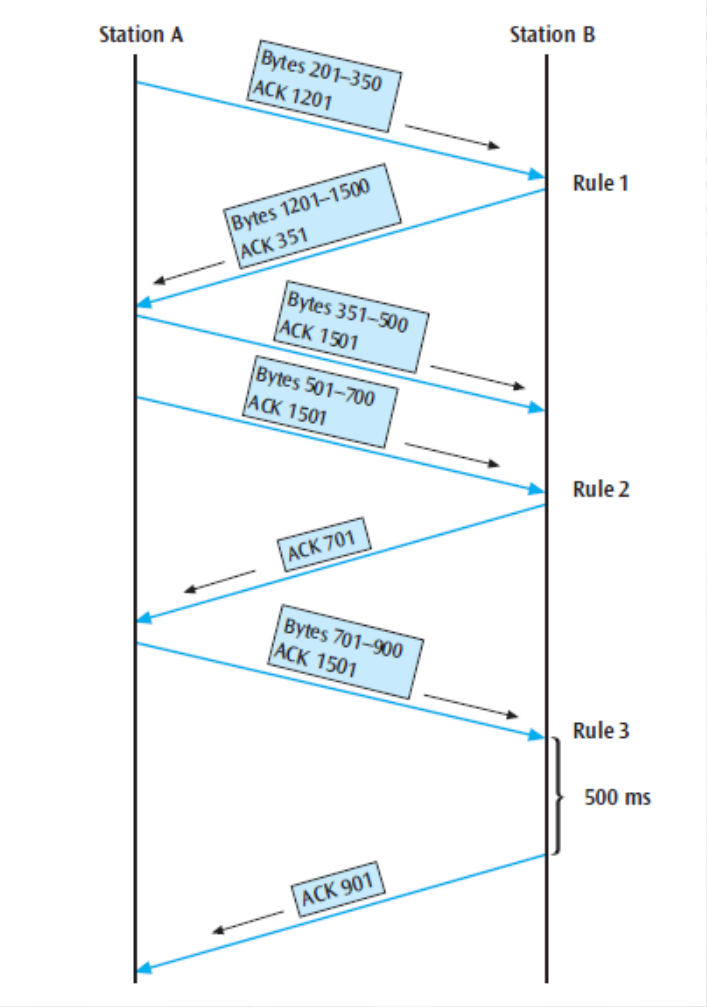
saban.ozturk@hbv.edu.tr

10. Pratikte Hata ile Mücadele

Çoğu ağ kablosu - özellikle LAN kabloları - çok güvenilirdir ve hatalar 1980'lerde olduğu kadar yaygın değildir.

Bu nedenle, LAN'lerde (yani Ethernet) kullanılan çoğu veri bağlantı katmanı yazılımı hataları tespit etmek için yapılandırılmış, ancak bunları düzeltmemektedir. Bir hata içeren bir paket her keşfedildiğinde, basitçe atılır. Kablosuz LAN'lar ve bazı Geniş Alan Ağları (WAN'lar), hataların daha olası olduğu yerlerde hala hem hata tespiti hem de hata düzeltme yaparlar.

Bunun sonucu olarak, hata düzeltmenin daha üst katmanlarda gerçekleştirilmesi gerekmektedir.



Kaynak: https://slideplayer.com/slide/13491092/

saban.ozturk@hbv.edu.tr

Hafta 4: Data Link Layer Çalışma Prensibi- Veri İletimi Prensipleri

Bölüm Özeti

- Data Link Layer Çalışma Prensibi
- Veri İletimi Prensipleri

Değerlendirme Soruları

1) Hangi medya erişim kontrol yaklaşımı en iyisidir: kontrollü erişim mi yoksa mücadele mi?

Kaynaklar

1) Stallings, W. (2007). Data and computer communications. Pearson Education India.

2) Forouzan, B. A. (2007). Data communications and networking. Huga Media.

3) Stallings, W., & Case, T. L. (2012). Business Data Communications-Infrastructure, Networking and Security.

4) Freer, J. (1996). Computer communications and networks. CRC Press.

5) Walrand, J., & Parekh, S. (2022). Communication networks: a concise introduction. Springer Nature.