

YILDIZ TEKNİK ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

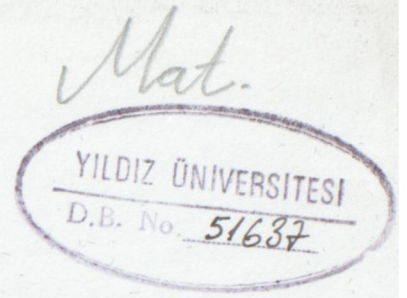
Veri İletişimi ve Appc Programlama

Yüksek Lisans Tezi

Hülya Şahintürk

1992

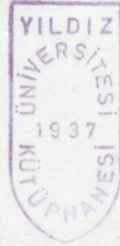
Ref
MTM
110
1992



YILDIZ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTUSU

85 R
209
114

VERİ İLETİŞİMİ VE APPC PROGRAMLAMA



YÜKSEK LİSANS TEZİ

MATEMATİK MÜHENDİSİ HOLYA ŞAHİNTÜRK

Y.T.O.
KÜTÜPHANE DOK. DAİ. BAŞKANLIĞI

İSTANBUL 1992

YILDIZ TEKNİK ÜNİVERSİTESİ
KÜTÜPHANE DOKÜMANTASYON
DAİRE BAŞKANLIĞI

Kot	209
Alındığı Yer	114
	F.B. Enstitüsü
Tarih	23.9.1996
Fatura	
Fiyatı	30 Bin
Ayniyat No	1/7
Kayıt No	52521
UDC	
Ek	

Y.T.Ü.
KÜTÜPHANE DOK. DAİ.

VERİ (BİLGİ) İLETİŞİMİ VE APPC PROGRAMLAMA

içindekiler

Sayfa

BÖLÜM 1 : Bilgi iletişim nedir?.....	1
Kısım 1.1 : Bilgi iletişim nedir?.....	1
1.1.1. Bilgi iletişimin Faydaları nelerdir?.....	2
1.1.2. Bilgi iletişim Hiyerarşisi.....	3
1.1.3. Bilgi iletişim Sistemlerinde Uygulama Türleri Nelerdir.....	3
1.1.3.1. Bilgi Giriş, Sorgulama ve Kayıt	6
1.1.3.2. Uzaktan Toplu Giriş.....	9
1.1.3.3. Uzak İş Girişi.....	10
1.1.3.4. Dağıtılmış Bilgi İşleme.....	13
Kısım 1.2 : Bilgi iletişim Sisteminin Esasları.....	14
1.2.1. Kod kümeleri.....	14
1.2.2. Protokoller.....	15
1.2.3. Analog ve Dijital.....	16
Kısım 1.3 : Bilgi iletişim Sistemleri Öğeleri.....	16
BÖLÜM 2 : iletişim Sistem Bileşenleri.....	19
Kısım 2.1. iletişim Sistemleri.....	19
2.1.1. Hat Bağlantı Çeşitleri.....	20
2.1.2. Hatlar.....	23
2.1.3. Modemler.....	24
2.1.4. Hata denetimi.....	27
2.1.5. iletişim Uyarlayıcısı.....	28
Kısım 2.2. Bilgi İletim Durumları.....	30

2.2.1. Sayısal Hatlar.....	32
2.2.2. Hatların Denetimi.....	32
Kısım 2.3. Bilgi Doğrulama Yöntemleri.....	36
2.4.1. Hat Denetiminde Kullanılan Yöntemler..	38
2.4.1.1. İletişim Uyarlayıcıları.....	43
2.4.1.2. İletişim Denetleyicileri.....	44
2.4.1.3. Terminaller.....	44

BÖLÜM 3 : AS/400 Gerçekleştirme ve Programlama

KISIM 3.1 : APPC' nin Genel Görünümü.....	46
KISIM 3.2 : Hat Desteği.....	48
3.2.1. Aktarmalı Noktadan Noktaya Bağlantı.....	48
3.2.2. Aktarmasız Noktadan Noktaya Bağlantı....	49
3.2.3. Aktarmasız Çok Noktalı Bağlantı.....	49
3.2.4. IBM Token-Ring Yerel İletişim Bölge AĞI.	50
KISIM 3.3 : OS/400 Desteği.....	52
KISIM 3.4 : İletişime Başlayabilmek için Gereken Hazırlıklar.....	54
3.4.1. Hat Tanımı.....	54
3.4.2. Denetim Birimi Tanımı.....	58
3.4.3. Aygıt Tanımı.....	62
3.4.4. Mod Tanımı.....	64
KISIM 3.5 : ICF Kütükleri.....	65
3.5.1. Genel Bilgiler.....	65
3.5.2. ICF Kütükleri için DDS Kodlamaları.....	67
3.5.3. ICF Kütüklerinin yaratılması.....	76
KISIM 3.6 : APPC Programlama.....	80

3.6.1. Genel Bilgiler.....	80
3.6.2. ICF Kütükleri için RPG/400 Deyimleri....	82
3.6.3. Özet.....	85
Örnek programlar.....	86
Kaynaklar.....	104
Özgeçmiş.....	105

Teşekkür,

Bu konu üzerinde çalışmamı sağlayan ve tezimin hazırlanmasında bana gösterdiği derin anlayış ve yardımları için Sayın hocam Prof. Tahir Şişman'a , ayrıca ; kısıtlı bir zaman çerçevesi içinde yardımlarını esirgemeyen, çalışmamda bana her türlü kaynağı temin eden ve AS/400 sisteminde bizzat çalışmamı sağlayan, değerli arkadaşım Sayın Süleyman Öneş'e ve eşim Mehmet Şahintürk' e bana verdiği bilgiler ve sağladığı kaynaklar için teşekkürü bir borç bilirim.

Hülya Şahintürk

ÖZET

Bu çalışma veri iletişiminin nasıl gerçekleştiği, bilgi iletişim kavramlarını , bilgi işlem sistemlerinde özellikle IBM AS/400 sisteminde kaç çeşit iletişim kullanıldığını ve bütün bu olayları öğrenmek isteyenler için hazırlanmıştır. Özellikle öğrenciler için önemli bir kaynaktır. Kitaptaki bütün metinler basit bir yolla sunulmuştur.

Bu çalışma bir sistem iletişiminin içsel olarak efektif ve çabuk çalışmasını anlatacak şekilde organize edilmiştir. Çalışmamdaki bütün konular IBM AS/400 sistemine göre ele alınmış , örneklerle ve şekillerle anlatılmıştır.

1. Bölümde bilgi iletişimi anlamak için gerekli olan temel tanımlamalar ve basit kavramları açıklamaktadır.
2. Bölümde kaç çeşit bilgi iletişim olduğunu ve bunların bileşimini göstermektedir. İlk bakışta bu kısım belki biraz daha teknik görünebilir, fakat ilk önce 1. bölümde sunulan bütün kavramlardan sonra anlayabilinir. Aynı zamanda bu kısımda ana giriş/çıkış aygıtları veya çevre birimleri ile bağlantısı için gerekli teknikler verilmiştir.
3. Bölümde her çeşit AS/400 iletişimini ele almakta ve iletişime başlamak için nelere ihtiyaç duyduğu anlatılmaktadır. İletişim kodlaması için OS/400 işletim sisteminde gerekli desteği ele almakta ve son olarak 3. bölüm APPC kodlama ve programlama geçiş ile son bulmaktadır.

Top Coding: Finally into part and with APPC coding or programming.

ABSTRACT

***** END OF SOURCE *****

This book is prepared for those who want to understand what a data communication is and how it operates and what its concepts or what sort of communications use in data processing system especially with IBM AS/400. In particular, it is an important material for students. Whole text in this book is presented in a simple way.

This study is organized to introduce you, quickly and effectively to the inner working of a system communication. In this study all subjects are explained according to AS/400. This study consists of three parts and each part is illustrated with examples.

Part 1 ; provides basic definition and necessary explanation to understand data communication and also introduces many basic concepts.

Part 2 ; shows what kind of communications and its compounds. At first glance, this chapter may appear very technical, but once you understand all the definitions presented in part 1. You will find this internal journey both easy and enjoyable. And in this part also presents the techniques necessary for connecting the main input/output devices and peripherals.

Part 3 ; is the last part in this study that deals with all sorts of AS/400 communications. And what else is needed for starting communications on the system. What kind of support is used for coding. Finally this part ends with APPC coding or programming.

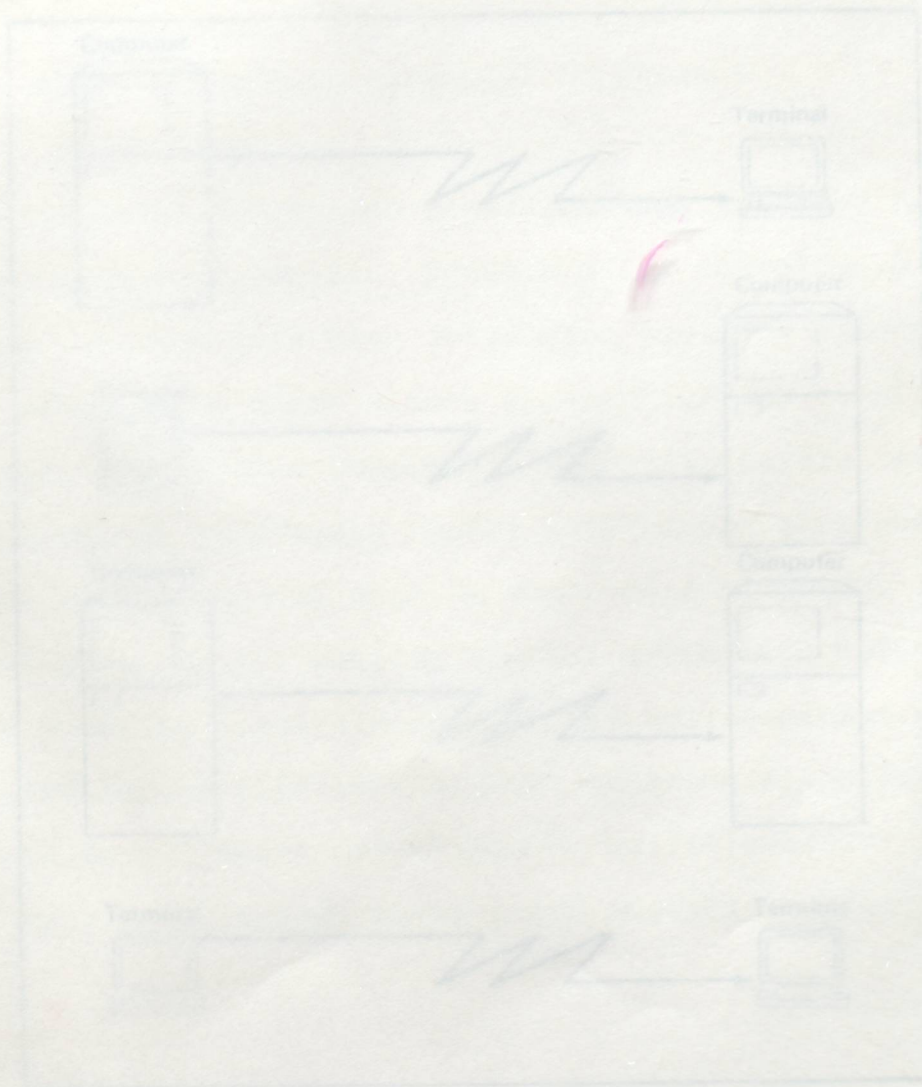
E FILE HULYA/KAYNAK
R APPC1S

R*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
FAPPC1ICFO E WORKSTN
C WRITEFMT1
C WRITEFMT2
C SETON LR

* * * * * E N D O F S O U R C E * * * * *

BÖLÜM 1.1.7. ÖRNEK İKİTANIM KODI ?

Bölüm 1.1.7. ÖRNEK İKİTANIM KODI. Bu bölümde uzakta
bulunan ve terminal ile çalışan bilgisayarların bir
birbirleriyle iletişim kurması için gerekli olan kodları
göstermektedir. Bu kodlar, bilgisayarların birbiriyle
iletişim kurması için gerekli olan kodları göstermektedir.
Bölüm 1.1.7. ÖRNEK İKİTANIM KODI. Bu bölümde uzakta
bulunan ve terminal ile çalışan bilgisayarların bir
birbirleriyle iletişim kurması için gerekli olan kodları
göstermektedir. Bu kodlar, bilgisayarların birbiriyle
iletişim kurması için gerekli olan kodları göstermektedir.

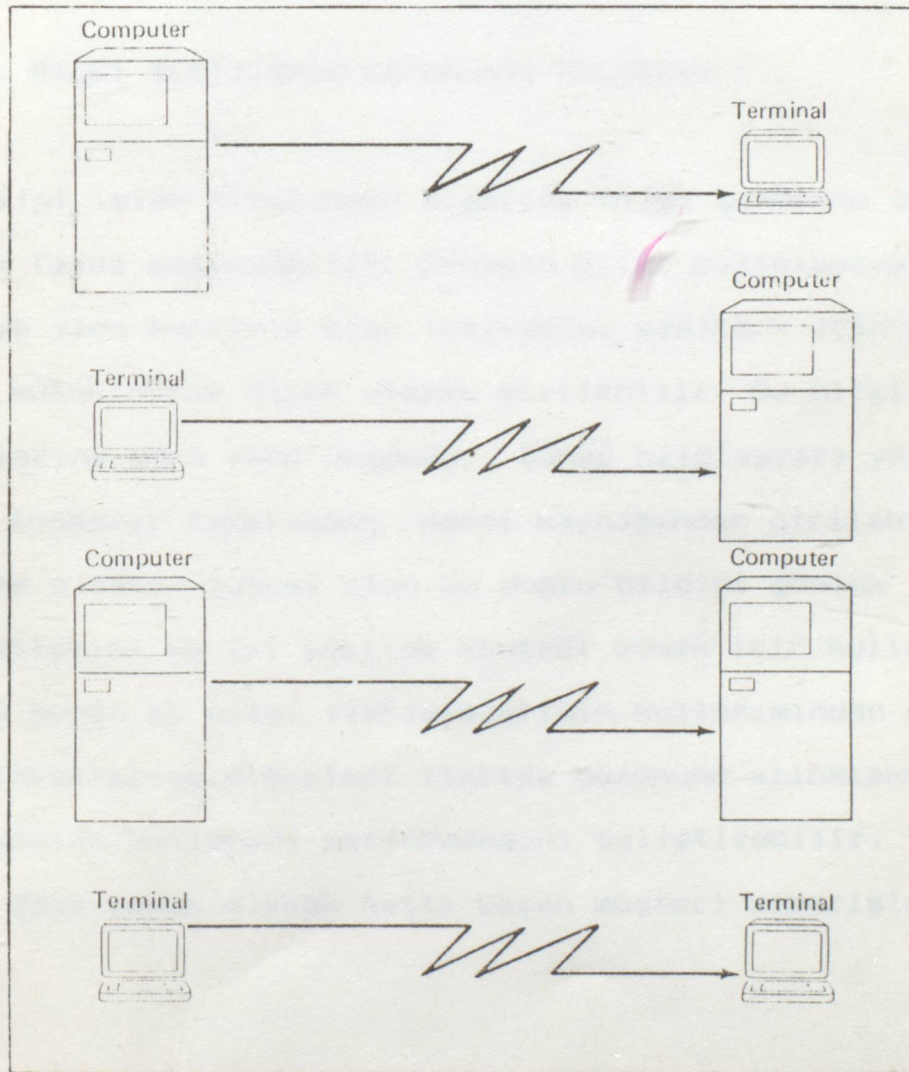


Bölüm 1.1.7. ÖRNEK İKİTANIM KODI

BÖLÜM 1. BİLGİ İLETİŞİM

KISIM 1.1 BİLGİ İLETİŞİM NEDİR ?

Bilgi iletişim (data communication), genellikle uzakta bulunan ve terminal adı verilen giriş/çıkış aygıtlarıyla bir bilgisayar arasındaki,veya birden fazla bilgisayarın kendi aralarındaki bilgi alışverişi ile programlar arası iletişimidir. Kısaca iletişim bilgilerin bir yerden diğerine aktarımıdır. Bilgi iletişimi aşağıdaki bir yada daha fazla



Veri iletişimi ve APPC Programlama

bilgi işlem aygıtları arasındaki bilgi gönderilmesi ile gerçekleşir.

- . Bir bilgisayardan bir diğer bilgisayara
- . Bir bilgisayardan bir terminale
- . Bir terminalden bir bilgisayara
- . Bir terminalden bir diğer terminale

Bir bilgi iletişim sistemi bir grup terminallerden, bilgisayarlardan, programlardan, ve gönderim ekipmanlarından oluşur.

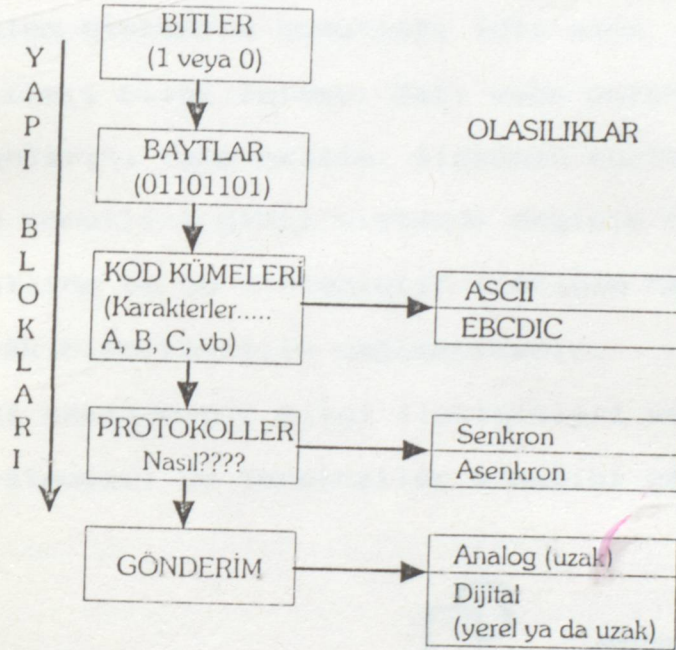
1.1.1. BİLGİ İLETİŞİMİN FAYDALARI NELERDİR ?

Bir bilgi işlem cihazından diğerine bilgi gönderme özelliği birçok fayda sağlayabilir. Örneğin bilgi postalamaya, kayıta, delgiye yada kontrole olan ihtiyacını azaltmak için uzak bilgi kütüklerine direk olarak girilebilir. Bu bilgi diğerlerine göre daha doğrudur. Çünkü bilgisayara yetkili bilgi insanlar tarafından kendi kaynağından girilebilir. İşletme o zaman güncel olan bu doğru bilgiyi günlük iş hareketlerini en iyi şekilde kontrol etmek için kullanabilir. Ortaya konan bu bilgi iletişimlerinin kullanımından dolayı bu gibi faydalar ve diğerleri ileride gözönüne alınacaktır. Bu faydalar kullanıcı performansını geliştirebilir. Çünkü işler daha çabuk olacak hatta bazen müşteri siparişleri yada

taleplerinin cevapları daha doğru olacaktır.

1.1.2. İLETİŞİM HİYERARŞİSİ

Bunu aşağıdaki tablodan anlamak daha faydalı olacaktır.



1.1.3. BİLGİ İLETİŞİM SİSTEMLERİNDE UYGULAMA TÜRLERİ NELERDİR?

Bugünün bilgi işlem sistemlerinin çoğu aşağıdakilerin biri yada daha fazla işlevleri yerine getirmesi için kullanılır.

- . Bilgi girmek: Bilgi kütüklerinin içine direk olarak bilgi girişi.
- . Bilgi sorgulama: Belirlenmiş bir bilgi için bu bilgiye ait

dosyada yapılan araştırma

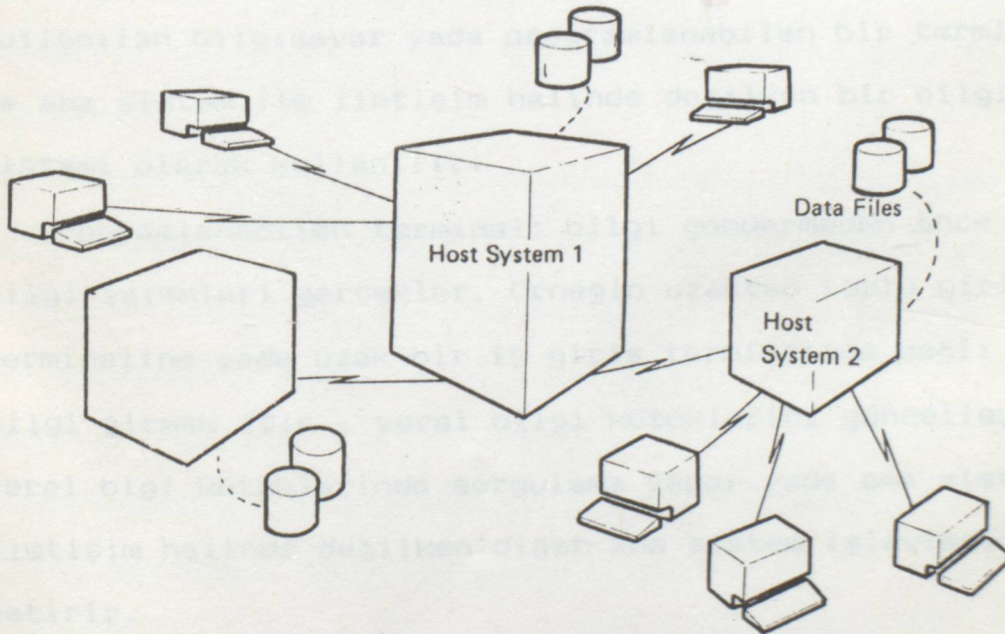
. Kayıt güncelleme: Dosyalara yüklenmiş bilgileri değiştirme ekleme veya silme yapma

. Uzaktan toplu giriş: Uzak bir bölgeden bilgi kütüklerine toplu bilgi girme

. Uzaktan iş girişi: Veriyi gönderme ve işleme için uzak bir bilgi işlem sistemine komutları icra etme

. Dağıtılmış bilgi işleme: Bazı yada bütün bilgi işlem fonksiyonlarını (ana bellek, sistemin kontrolü, işlem, giriş ve çıkış komutları gibi) sistemde değişik bölgelerde lokalleştirme ve bu sistemlerin iletişim hatları ve cihazları kullanarak birbirleriyle bağlantısıdır.

Aşağıdaki şekilde bir bilgi iletişimleri konfigürasyonunda bilgi sistemleri ve terminaller örneğini gösterir.



Bu şekil bilgi iletişimlerinin uygulamaları açısından ve aynı zamanda aşağıdaki açıklamalar içinde faydalı olabilir.

. Ana sistem 1: Denetleyen bilgisayar yada bilgi iletişim konfigürasyonundaki en yüksek seviyeli sistemdir. Ana sistem 1 bağlantılı diğer ana sistemlere yada sistemler içinde iletişim programlarını kontrol eden ve başlatan programları içerir. Bazı konfigürasyonlarda bütün sistemler aynı denetim işlerine, görevlerine sahip olabilirler bundan bunlara eşdeş sistemler denir. Bir eşdeş konfigürasyonunda ana sistem yoktur.

. Ana sistem 2: Kendisine bağlı terminaller ve ana sistem 1 ile iletişim halindedir.

. Uzaktan toplu giriş yada uzaktan iş giriş terminali: bir ana sistem ile iletişim halindeyken terminal olarak kullanılan bilgisayar yada programlanabilen bir terminaldir. Ve ana sistem ile iletişim halinde değilken bir bilgi işlem sistemi olarak kullanılır.

. Programlanabilen terminal: bilgi göndermeden önce bazı bilgi işlemleri gerçekler. Örneğin uzaktan toplu giriş terminaline yada uzak bir iş giriş terminaline bağlı yerel bilgi girmek için , yerel bilgi kütüklerini güncelleştirir, yerel bilgi kütüklerinde sorgulama yapar yada ana sistem ile iletişim halinde değilken diğer ana sistem işlevlerini yerine getirir.

. Programlanamaz bir terminal veri iletişimleri sistem konfigürasyonunda giriş/Çıkış istasyonudur (Bir yazıcı yada klavye/görüntü terminali gibi.)

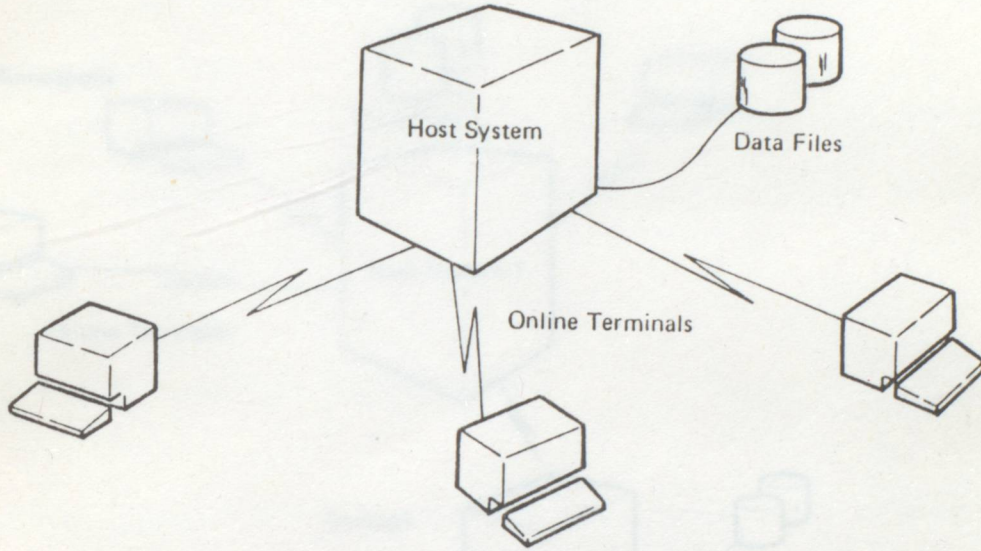
1.1.3.1. BİLGİ GİRİŞ, SORGULAMA VE KAYIT GÜNCELLEME

Bilgi girişinde ana sistemdeki kütüklere yerel ve uzak terminaller arayıcılığıyla doğrudan bilgi girilmesidir. Genellikle tek yönlü bir uygulama tipidir. Bilgi terminalden ana bilgisayara iletilir. Bilginin doğruluğu uygulama programı tarafından sınanır.

Kayıt güncelleştirmede yerel ve uzak terminallerden girilen bilgilerle sisteme bağlı çevrim içi kütüklere yeni kayıt eklenebilir, kayıtlar iptal edilebilir veya güncelleştirme yapılabilir. Bu tür uygulamalar, bilginin bütünlüğünün (data integrity) korunması için, sistem tasarımının daha dikkatle yapılmasını gerektirir.

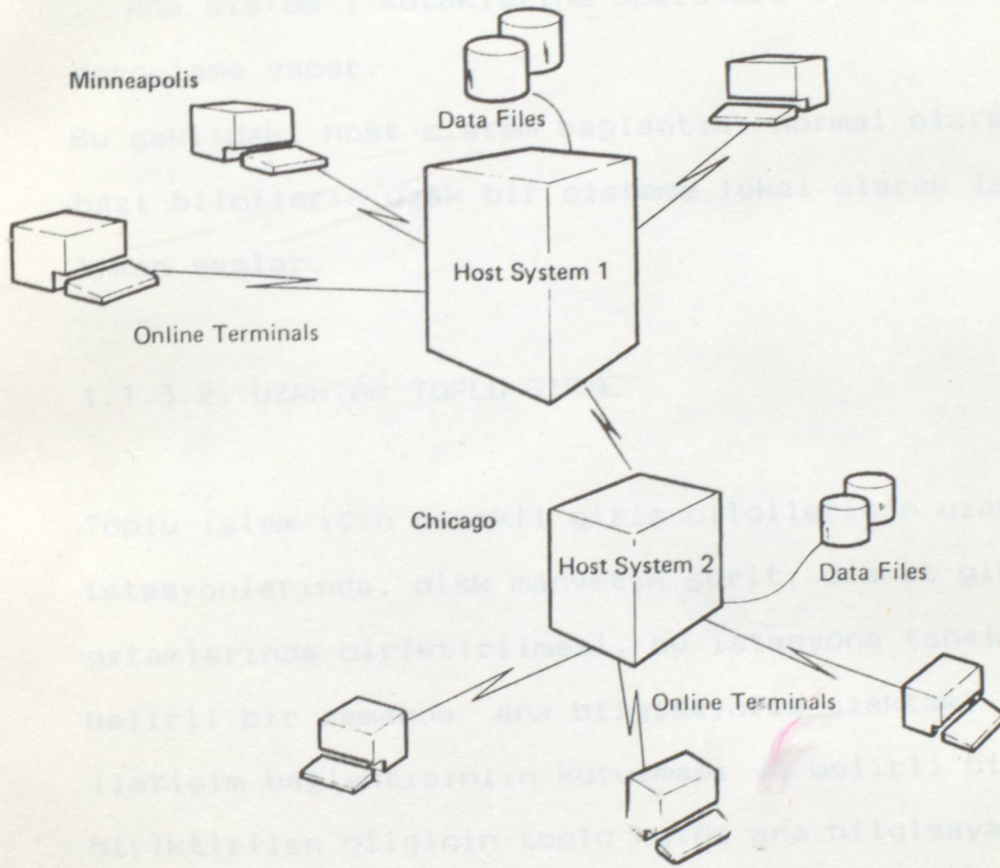
Soru sorma (sorgulama) da terminallerden ana bilgisayara bağlı bilgi kütüklerine erişilerek, belirli bir kayda ait bilgiler terminallerden görüntülenir. Kayıt üzerinde herhangi bir değişiklik yapılması söz konusu değildir.

Aşağıdaki örnekte ana sisteme bağlı terminalleri göstermektedir. Bu sistem bilgi giriş, sorgulama ve kayıt güncelleştirmesi için kullanılabilir.



Bu sistemde müşteri siparişleri uzak bir terminalden ana sistem kütüklerine direk olarak girilebilir. Aynı zamanda, diğer terminaller ana sistem kütüklerine güncelleme veya sorgulama yapabilir. Bu sistemde bilgi, alınır ve işlem görür ve ana sistem yada terminaller tarafından anında, hemen kullanıma hazırdır. Buna etkileşimli iletişim yada gerçek zamanlı bilgi işlem denir.

Aşağıdaki şekilde görüldüğü gibi bir ana sistem terminallere ek olarak bir diğer ana sistem ile bağlantılı olabilir.



Bu konfigürasyonda ana sistem 2 kendi terminalleri için bir ana sistemdir ve ana sistem 1 için bir terminaldir. Host sistem 2 aşağıdakileri gerçekleştirir.

- . Ana sistem 1' e özet bilgi gönderir.
- . Ana sistem 1' den bilgi alır ve işler.

. Ana sistem 1 operatörü için bilgi kütüklerine sorgulama yapar.

. Ana sistem 1 kütüklerine operatörü için veri kütüklerine sorgulama yapar.

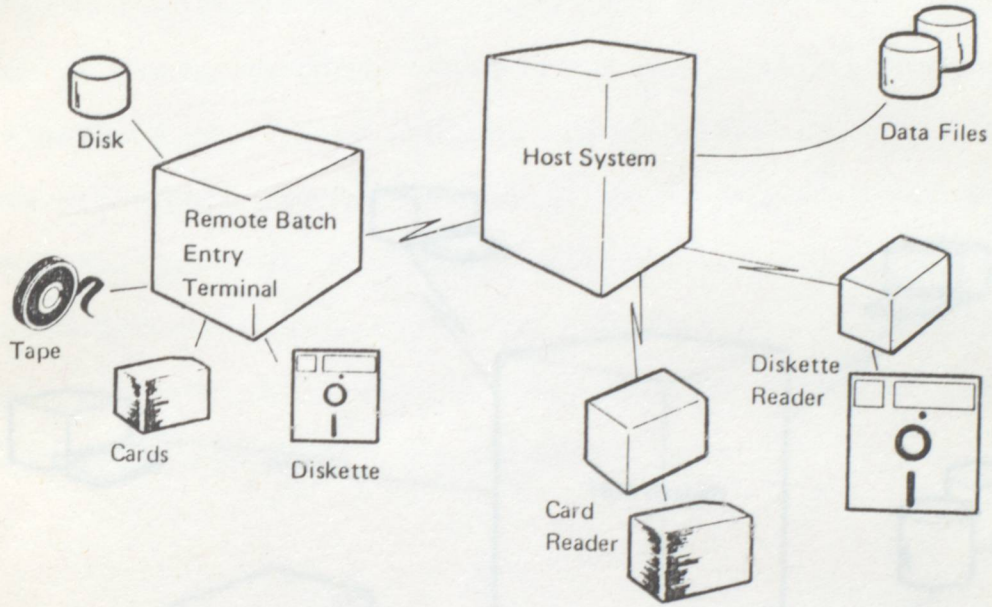
Bu şekildeki host sistem bağlantısı normal olarak gönderilmiş bazı bilgilerin uzak bir sisteme lokal olarak işlenebilmesine imkan sağlar.

1.1.3.2. UZAKTAN TOPLU GİRME

Toplu işlem için gerekli giriş bilgilerinin uzaktaki terminal istasyonlarında, disk manyetik şerit, disket gibi kayıt ortamlarında biriktirilmesi, bu istasyona tahsis edilen belirli bir zamanda ana bilgisayarla uzaktaki terminalin iletişim bağlantısının kurulması ve belirli bir süre içinde biriktirilen bilginin toplu halde ana bilgisayara iletilmesidir. İletilen bilgiler daha sonra ana bilgisayarda işlenir.

Aşağıdaki örnekte ana sistem için uzaktan toplu giriş sistemi olarak gösterilmiştir.



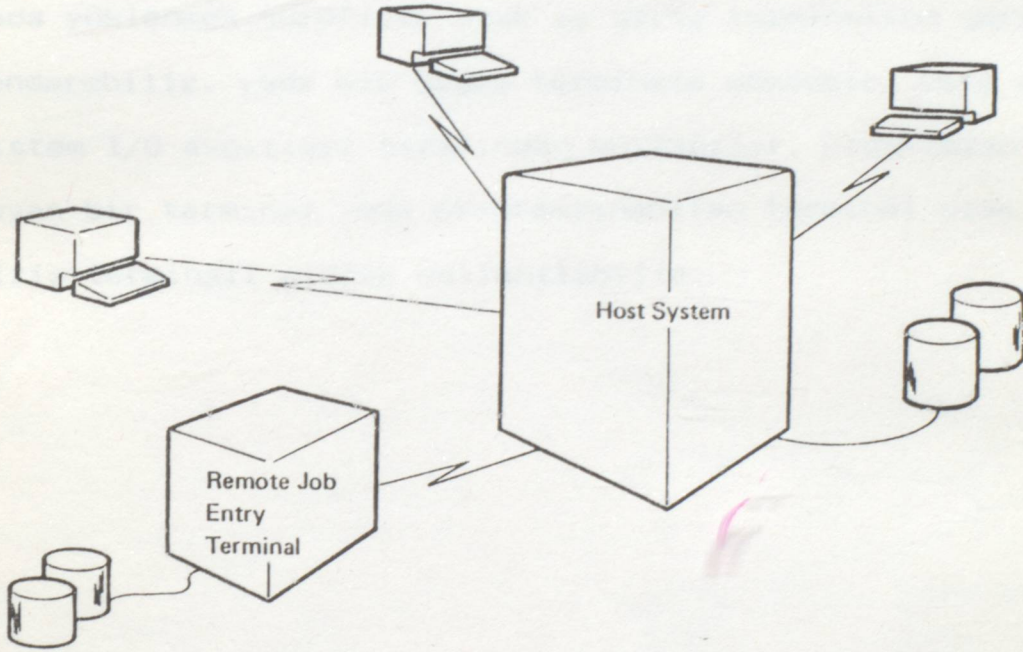


1.1.3.3. UZAK İŞ GİRİŞİ

Uzaktaki terminalden bir işin yürütülmesi için gerekli iş denetim deyimlerinin ve giriş bilgilerinin ana bilgisayara iletilmesidir. İş yürütüldükten hemen sonra sonuçlar uzaktaki terminalden alınabilir, başka bir terminale iletilebilir veya

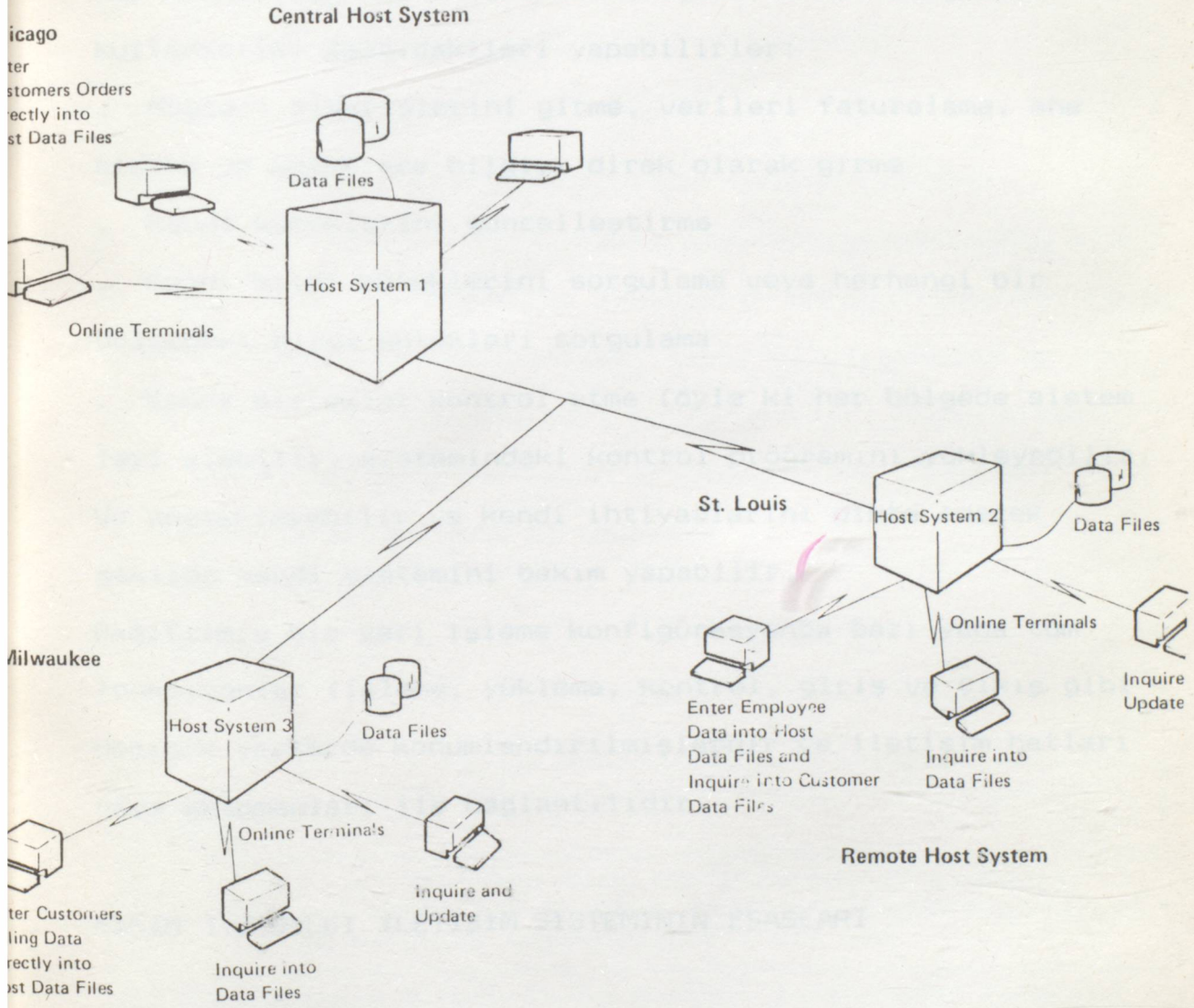
ana sisteme giriş/cıkış bazı aygıtlarıyla yazdırılabilir.

Aşağıdaki şekilde bir uzak iş giriş terminali gösterilmiştir.



Uzak iş giriş terminali ana sistem ile bağlantılı olduğunda, uzak iş giriş terminal kullanıcısı ana bilgisayara ya veri yada işlem komutları gönderebilir. Ana sistem o zaman uzak iş giriş kullanıcısından komutlara göre bilgiyi işler. Böylece uzak iş giriş terminal kullanıcısı işlem kapasitesinden faydalanabilir. Sonuç sonraki bir kullanım için yazdırılmak yada yüklenmek suretiyle uzak iş giriş terminaline geri gönderebilir. yada bir başka terminale gönderir veya ana sistem I/O aygıtları tarafından kullanılır. Programlanmayan bir terminal yada programlanabilen terminal uzak iş giriş terminali olarak kullanılabilir.

1.1.3.4. DAĞITILMIŞ BİLGİ İŞLEME



Uzak Ana Sistem

Bu şekil dağıtılmış bilgi işlem konfigürasyonunu gösterir. Bu konfigürasyonda, uzak sistemler ve terminaller bir merkezi işlem birimine bağlanmışlardır (ana sistem 1 gibi).

Her ana sistem kendi verilerini işler ve her bölgedeki kullanıcılar aşağıdakileri yapabilirler:

- . Müşteri siparişlerini girme, verileri faturalama, ana sistem ve kütüklere bilgiyi direk olarak girme
- . Kendi kütüklerini güncelleştirme
- . Kendi bilgi kütüklerini sorgulama veya herhangi bir bölgedeki bilgi kütükleri sorgulama
- . Kendi sistemini kontrol etme (öyle ki her bölgede sistemleri açabilir), sistemindeki kontrol programını yükleyebilir. Ve destekleyebilir ve kendi ihtiyaçlarını dikte edecek şekilde kendi sistemini bakım yapabilir.

Dağıtılmış bir veri işleme konfigürasyonda bazı yada tüm fonksiyonlar (işleme, yükleme, kontrol, giriş ve çıkış gibi değişik yerlerde konumlandırılmışlardır ve iletişim hatları veya ekipmanları ile bağlantılıdır.

KISIM 1.2 BİLGİ İLETİŞİM SİSTEMİNİN ESASLARI

1.2.1. Kod kümeleri

Bir kod kümesi, 256 değişik bit grubunun neleri gösterdiğini anlamak için kullanılan standarttır (8 bitlik baytlar

kullanarak). Bilgisayarlar arasında iletişimin kurulabilmesi için aynı kod kümelerini kullanmaları gerekmektedir. Kod kümelerine iki örnek:

- . ASCII
- . EBCDIC

1.2.2. Protokoller

İletişim dilleri protokoller olarak adlandırılırlar.

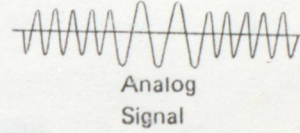
Protokoller, bağlı bilgisayarlar arasında aktarılmakta olan bilginin biçimini tanımlar. İnsanların anlaşabilmek için aynı dili kullanmak zorunda olmaları gibi, bilgisayarlar da veri aktarımında aynı protokolleri kullanmalıdır. Protokoller, veri gönderimi için baytların gruplandırılma yöntemini ve aşağıdaki alanları kontrol ederler:

- . Hata denetimi
- . Gönderim hızı

KOD KÜMESİ	Ö Z E L L İ K L E R İ
ASCII	American Standard code for Information Exchange (Bilgi değiş tokuşu için Amerikan Standart Kodu) . 128 karakter gösterebilen 7 bit kod
EBCDIC	Extended Binary-Coded Decimal Interchange Code Genişletilmiş İkili-Kodlanmış Onluk Değiş Tokuş kodu . 256 karakter gösterebilen 8 bit kod . IBM ve IBM uyumlu anabilgisayar ekipmanı üreten firmalar tarafından kullanılan standart

1.2.3. Analog ve dijital

Bahsettiğimiz sinyal tipleri (bit, bayt ve kod kümeleri) dijital sinyaller olarak bilinir. Diğer tip sinyaller, frekans düzeyinin yoğunluğundan oluşan analog sinyaldir. Telefon hattındaki ses analog sinyale örnektir.



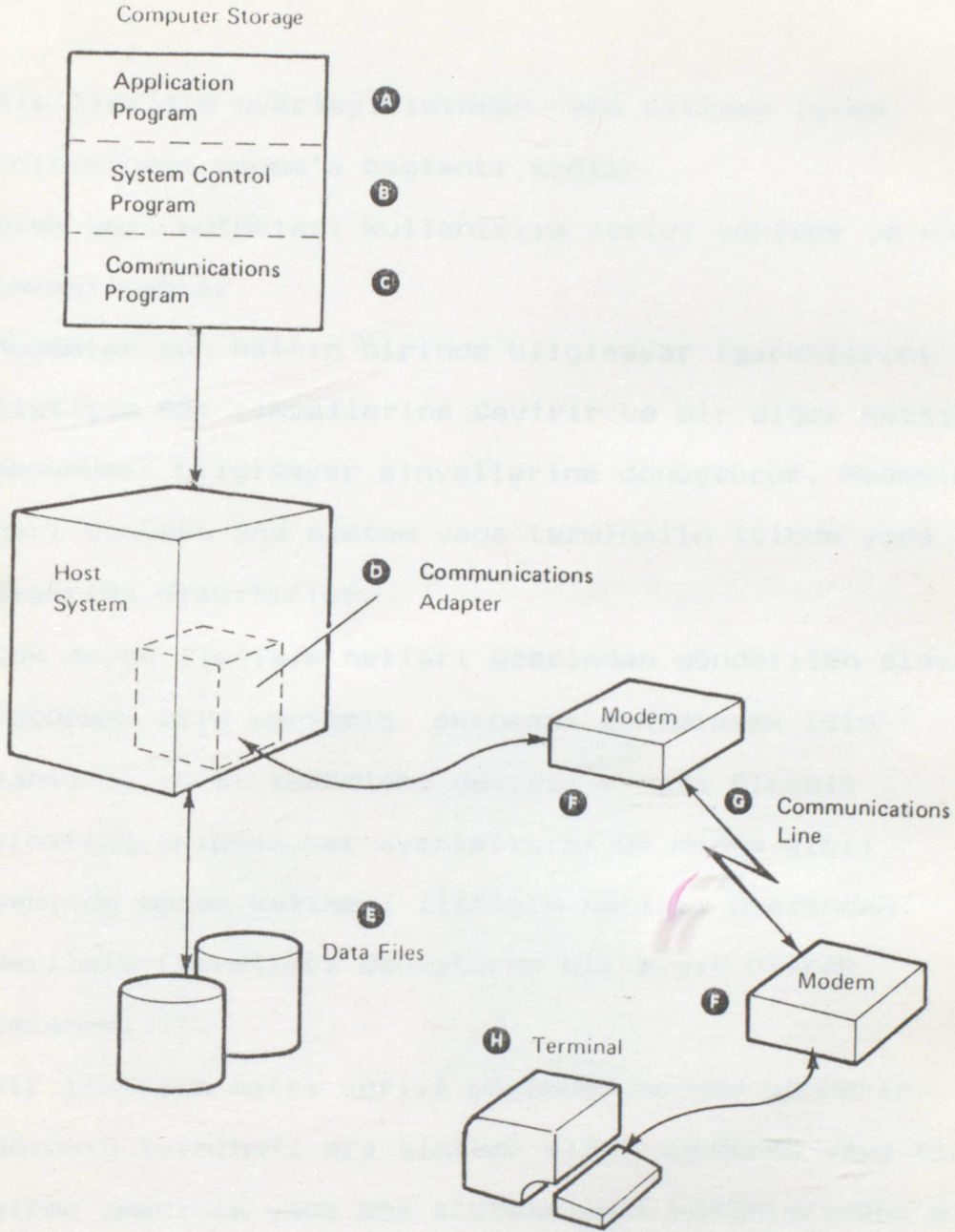
KISIM 1.3 BİLGİ İLETİŞİM SİSTEMLERİNİN ÖGELERİ

Bir bilgi iletişim sistemlerin ögeleri, sistemin nasıl tasarlanacağına bağlı olarak farklılık gösterebilir. Bunlarda veri iletişimleri için temel ögelere gereksinme vardır.

Şimdiki örnekte bütün ögeleri gerektiren bir konfigürasyondaki ana sistem ve bir terminal gösterir.

A- Uygulama programları kullanıcı işlerini yerine getiren programlardır. IBM sistemlerinin çoğunda uygulama programları aynı programlama dilleri ile yazılmışlardır. (RPG II, RPG III, RPG 400, FORTRAN, COBOL gibi). Müşteri, bilgi iletişim programları için uygun olmayan bu programları kullanırlar.

B- Sistem kontrol programı (kullanıldığında) iletişim programı vasıtasıyla uygulama programından kullanılması için veri hazırlar.



C- Bu konfigürasyonda ana sistem iletişim programı ana sistem ve terminal arasındaki iletişimi kontrol eder. Bazı sistemler veri gönderimini kontrol etmeye yardım etmesi için aynı zamanda bir mikro işlemci(ler) kullanabilir.

D- Bir iletişim uyarlayıcısından ana sisteme işlem ünitesinden modem'e bağlantı sağlar.

E- Disk veri kütükleri kullanıcıya veriyi yükleme ve kullanma imkanı sağlar.

F- Modemler son hattın birinde bilgisayar işaretlerini iletişim hat sinyallerine çevirir ve bir diğer hattın sonundaki bilgisayar sinyallerine dönüştürür. Modemleri geri dönüşte ana sistem yada terminalin içinde yada dışarıda olabilirler.

Birçok devre iletişim hatları üzerinden gönderilen sinyalleri dönüştürmek için yapılmış ekipmanı tanımlamak için kullanılır. (Veri tamamlama devresi = data circuit terminating, ekipman, hat uyarlatıcısı ve modem gibi) . Bu şekilde modem kelimesi iletişim hatları üzerinden gönderilmiş işaretleri dönüştüren bir aygıt olarak tanımlanabilir.

G- Bir iletişim hattı veriyi modemden modeme gönderir.

H- Görüntü terminali ara sisteme bilgi gönderme veya bilgi girme amacıyla yada ana sisteme veri kütüklerinden bilgiyi geri çağırma amacıyla kullanılır.

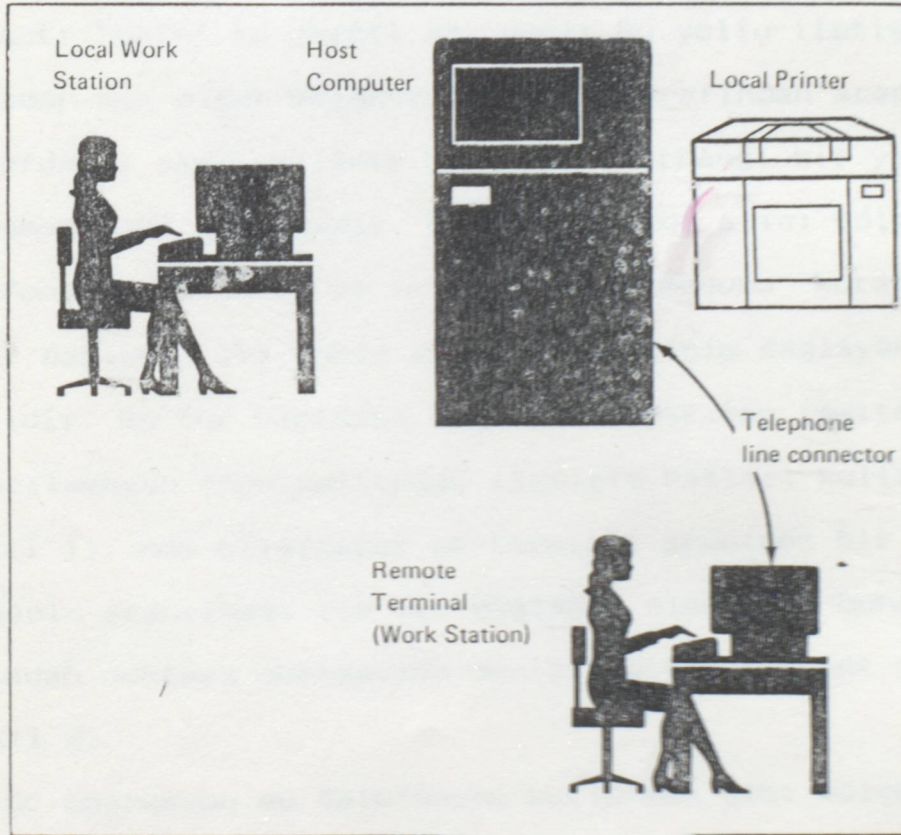
ÜNİTE 2: İLETİŞİM SİSTEM BİLEŞENLERİ

KISIM 2.1 İLETİŞİM SİSTEMLERİ

İletişim sistemlerini açıklamak için önce birbirine bağlamayı uzak giriş/çıkış aygıtı ve bir sistem düşünelim. (Şekil 1) .

Uzak giriş/çıkış aygıtı bir görüntü /klavye çalışma istasyonu olabilir. Bunu da sık sık terminal olarak adlandıracağız.

Merkezi olarak lokalleştirilmiş bilgisayarı çoğu kez



- . Anasistem
- . Ana bilgisayar
- . Ana merkezi işlem Ünitesi
- . Ana işlemci

diye adlandırabiliriz.

2.1.1. Hat Bağlantı Çeşitleri

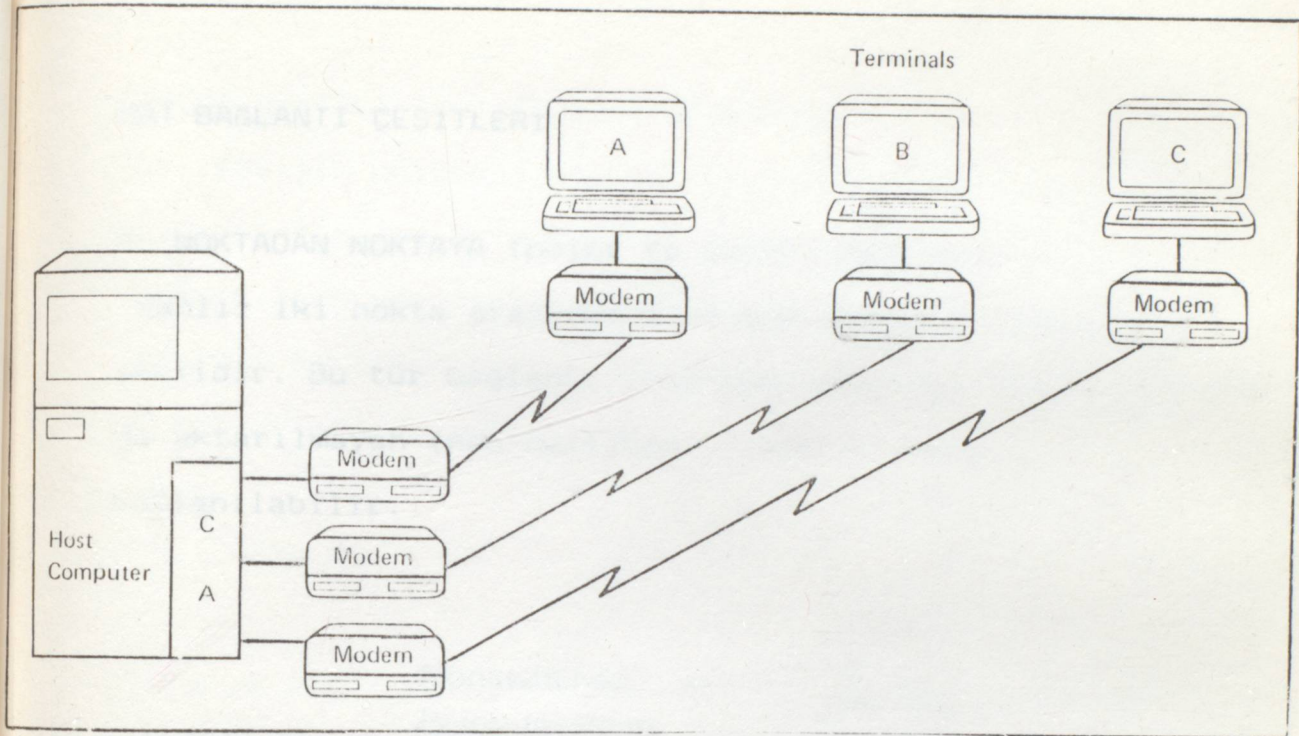
Uzak terminal telefon hatları vasıtasıyla ana bilgisayarla bağlantı kurulabilir. Bu çeşit bağlantıya noktadan noktaya bağlantı (point to point) ağı denir. Bu yollu iletişim bir bölgeden bir diğer bölgeye operator tarafından aranarak (telefonla) yapılabilir. Bu tıpkı herhangi bir yeri arayıp konuşmak gibi bir şeydir. Diğer operator alıcı bölgedeki telefona cevap verir ve ikiside bilgi modunu kurarlar. Bu çeşit bağlantı iki nokta arasında iletişim sağlayan bağlantı şeklidir. Bu tür bağlantı için hem aktarılan (switched) hemde aktarılamayan (non-switched) iletişim hatları kullanılabilir. (şekil 1). Ana bilgisayar ve terminal arasında bir diğer bölgenin aranılması ile bir bağlantı olduğunda buna aktarılan noktadan noktaya operasyonu denir. (switched point to point). (şekil 2)

Bu tür operasyon ev telefonunu kullanmak gibi düzenli telefon servisini kullanır. Terminallerden ana bilgisayarı bağlantı kurmak için bu metodu kullanılırsa telefon konuşmalarında

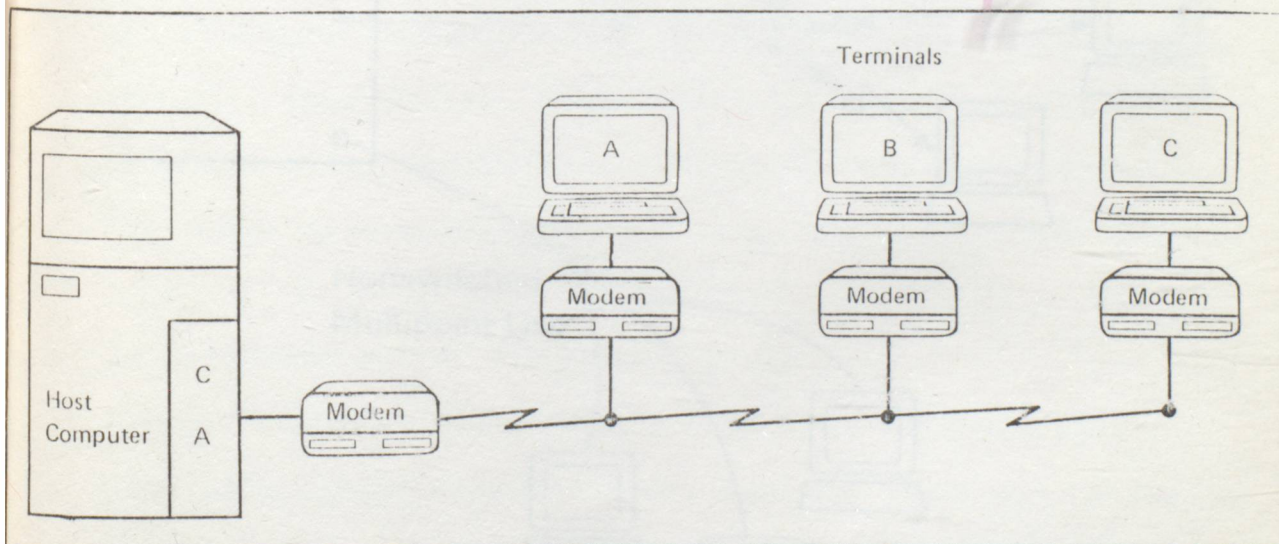
olduđu gibi aynı esneklik derecesine sahiptir. Herhangi bir yerde herhangi bir zamandaki herhangi bir sayıdaki terminalleri arıyabilir. Aynı zamanda herhangi sayıdaki terminallerde ana bilgisayarını arıyabilir. Eğer numara meşğul ise arayıcı bekler ve sonra tekrar dener. Bu tür hatları kullanmak için uzak modem ilk önce yerel modemler tarafından çevrilmelidir. Ondan sonra ancak uzak modem çağrını cevaplar ve iletişim başlar.

Bu aramalarda telefon aramaları için normal bir konuşma gibi ücret alınır. Peki başka alternatifimiz var mı?. Elbette vardır. Ayrıca aktarılamayan tipte bir operasyon vardır. Buda kiralanmış hatlarla gerçekleşir.

Aktarılan noktadan noktaya operasyona döndüğümüzde veri gönderimi için düzenli telefon kullanamayız. Telefon ve telefon aygıtları temelde ses taşımaları düzeninde yapılmıştır. Veri taşımak için uygun değildir. Ses trafiğı analog sinyallerden (sürekli değişken) yapılmıştır. Bilgi trafiğı dijital (sayısal) (açık yada kapalı) işaretlerden oluşmuştur. Mühendisler ses telefon hatları üzerinden veri geçmesine imkan verecek şekilde modem denilen bir cihaz üretmişlerdir.



A point-to-point network

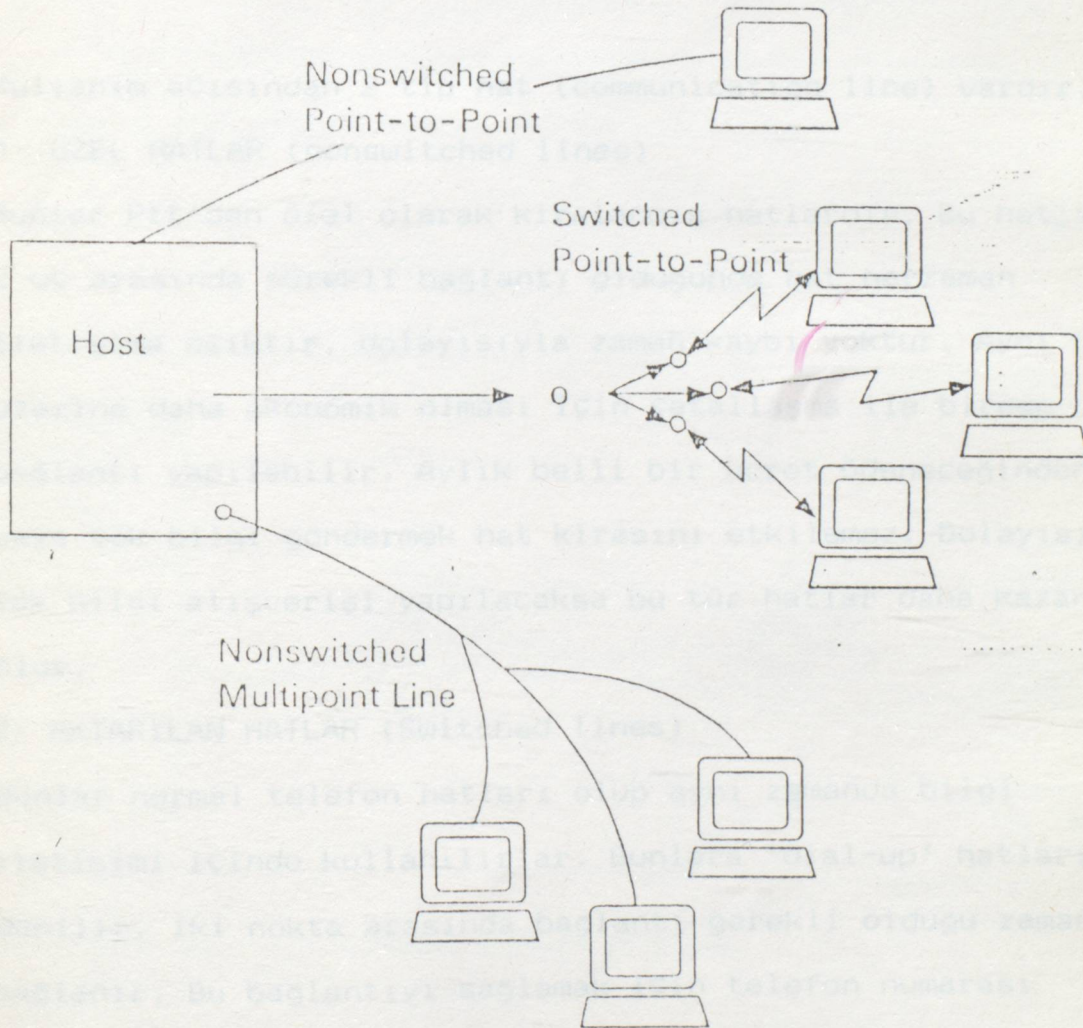


A multipoint network

HAT BAĞLANTI ÇEŞİTLERİ

1- NOKTADAN NOKTAYA (point to point) BAĞLANTI

Yalnız iki nokta arasında iletişim sağlayan bağlantı şeklidir. Bu tür bağlantı için hem aktarılan (switched) hem de aktarılmayan (non-switched) iletişim hatları kullanılabilir.



2- ÇOK NOKTALI (multipoint) BAĞLANTI

Bir iletişim hattında bir takım çatallaşmalar yapılarak birçok istasyona hizmet vermesi sağlanabilir. Bu çeşit bağlantılara çok noktalı bağlantı adı verilir. Bu durumda iletişim hattının özel olarak kiralanmış bir hat (non-switched) olması gerekir.

2.1.2. HATLAR

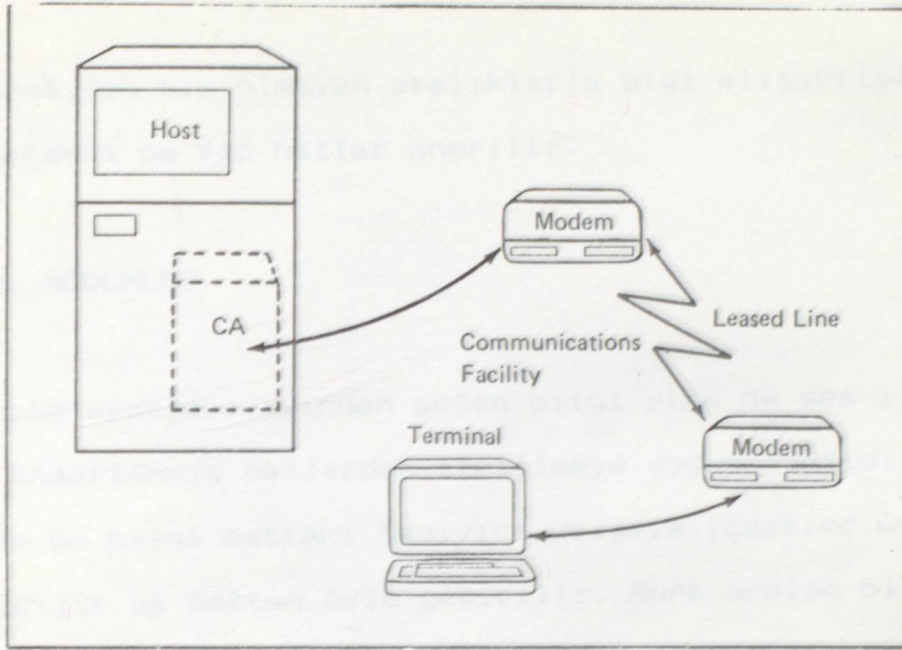
Kullanım açısından 2 tip hat (communication line) vardır.

1- ÖZEL HATLAR (nonswitched lines)

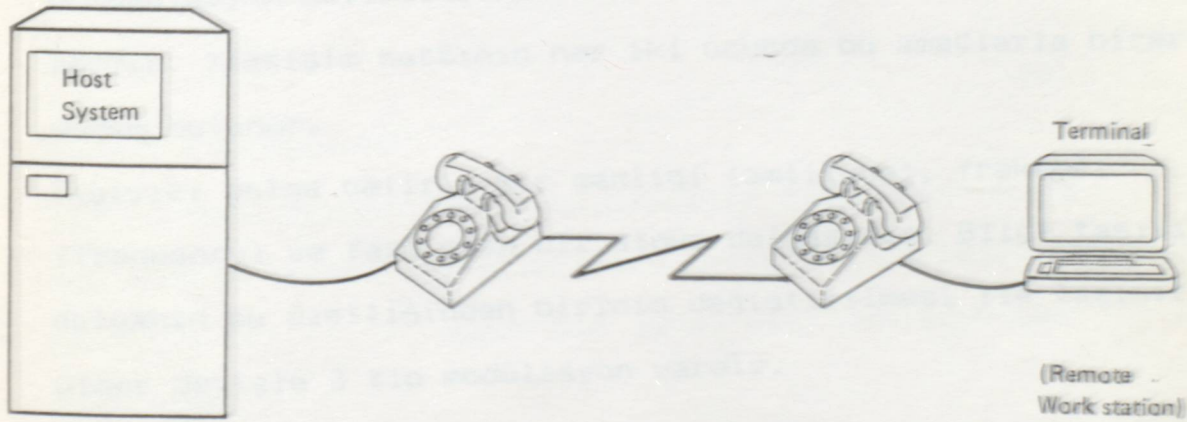
Bunlar Ptt'den özel olarak kiralanmış hatlardır. Bu hatlarda 2 uç arasında sürekli bağlantı olduğunda hat her zaman iletişime açıktır, dolayısıyla zaman kaybı yoktur. Aynı hat üzerine daha ekonomik olması için çatallaşma ile birden çok bağlantı yapılabilir. Aylık belli bir ücret ödeneceğinden, az veya çok bilgi göndermek hat kirasını etkilemez. Dolayısıyla çok bilgi alışverişi yapılacaksa bu tür hatlar daha kazançlı olur.

2- AKTARILAN HATLAR (Switched lines)

Bunlar normal telefon hatları olup aynı zamanda bilgi iletişimi içinde kullanılırlar. Bunlara 'dial-up' hatlarında denilir. İki nokta arasında bağlantı gerekli olduğu zaman sağlanır. Bu bağlantıyı sağlamak için telefon numarası çevirir gibi karşı tarafın numarası çevrilir. İki taraf



Şekil-1 Aktarılmayan noktadan noktaya iletişim.



Şekil-2 Aktarılan noktadan noktaya iletişim.

arasında çok sık olmayan aralıklarla bilgi alışverişi yapılacaksa bu tip hatlar önerilir.

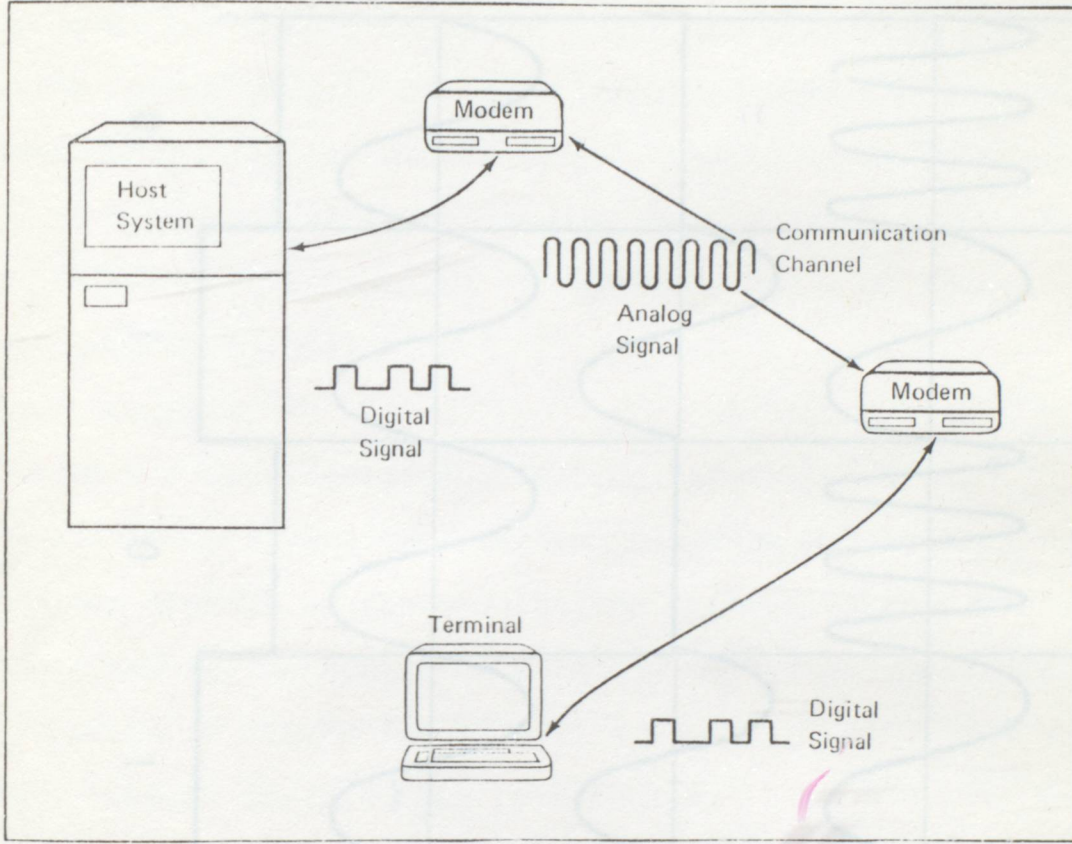
2.1.3. MODEMLER

İletişim uyarlayıcısından geçen bilgi yine de ses iletimi için tasarlanmış hatlardan iletmeye uygun değildir. Bu yüzden bu bilgi hattaki taşıyıcı dalgaya (carrier wave) bindirilir ve hattan öyle geçirilir. Buna analog bilgi adı da verilir. Dijital sinyalin analog sinyale dönüşüm işlemine MODÜLASYON denir. Bu bilginin karşı tarafta tekrar bilgisayarın anlayacağı şekilde taşıyıcıdan kurtarılacak şekilde çözülmesinde DEMODÜLASYON denir. Bu iki işlevi gerçekleştiren iletişim araçları ise MODÜLasyon ve DEModülasyon kelimelerinin ilk harflerinden oluşan MODEMLerdir. İletişim hattının her iki ucunda bu amaçlarla birer modem bulunur.

Taşıyıcı dalga belirli bir genliği (amplitude), frekansı (frequency) ve fazı olan bir sinüs dalgasıdır. Bilgi taşıyıcı dalganın bu özelliğinden birinin değiştirilmesi ile taşınır. Diğer deyişle 3 tip modülasyon vardır.

- . Genlik modülasyonu
- . Frekans modülasyonu
- . Faz modülasyonu

Şimdi bunlara sırayla kısaca değinecek olursak:



Genlik Modülasyonu

Genlik modülasyonu, taşıyıcı sinyalin genişliğinin veri gösterecek tarzda değiştirilmesi sürecidir. Örneğin büyük bir genişlik bir '1' i, düşük bir (yada hiç) genişlik ise '0' ı gösterebilir.

Frekans Modülasyonu

Frekans modülasyonunda, taşıyıcının frekansı veri gösterecek şekilde değiştirilir. Yüksek frekans periyodu '1' i , düşük frekans periyodunda '0' ı gösterebilir.

MODÜLASYON TEKNİKLERİ

İKİLİ KODLANMIŞ BİLGİ:

SİNYAL VAR

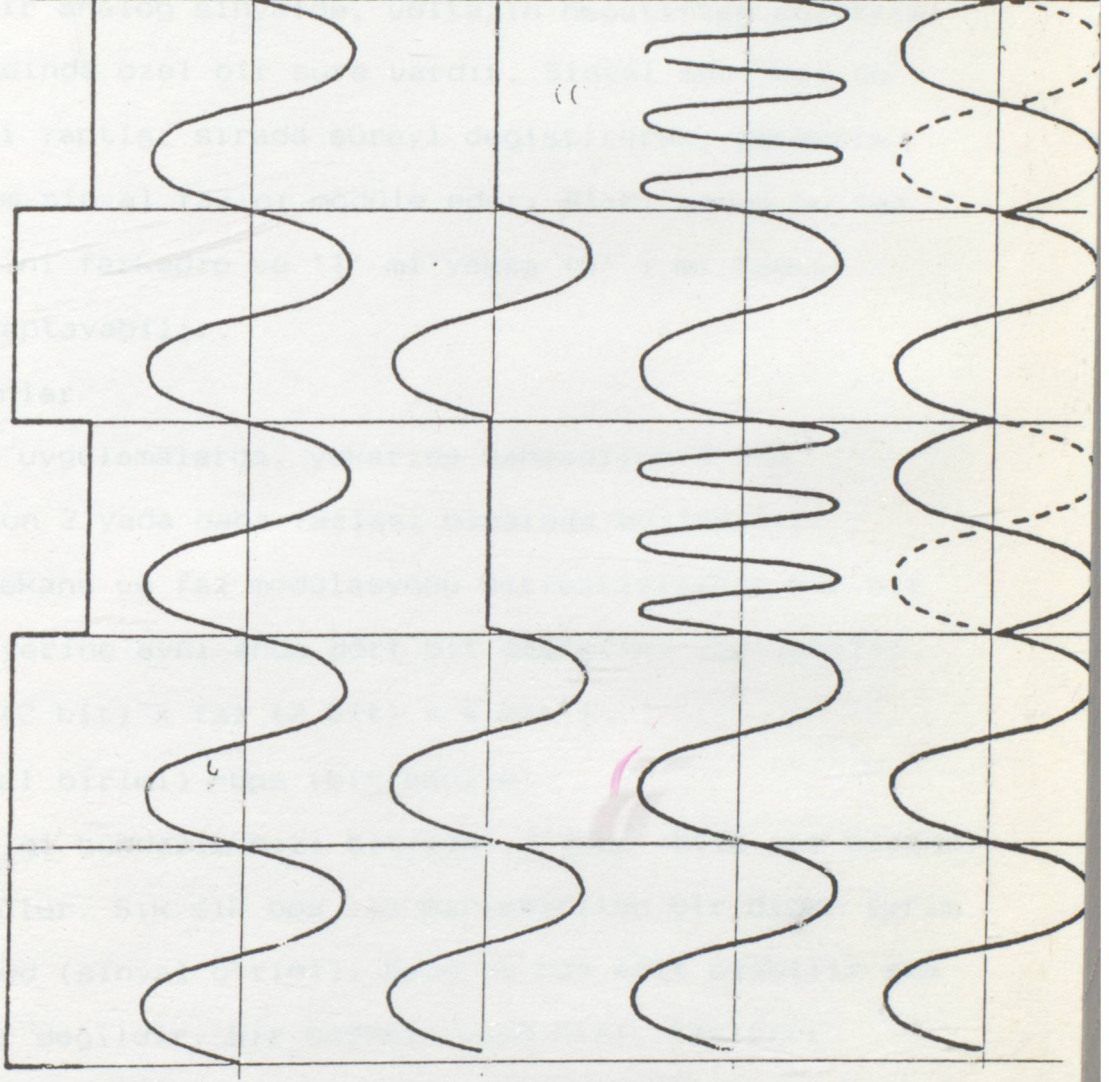
SİNYAL YOK

TAŞIYICI DALGA

GENLİK MODÜLASYONU

FREKANS MODÜLASYONU

FAZ MODÜLASYONU



Faz Modülasyonu

Herhangi bir analog sinyalde, voltajın negatiften pozitifte geçişi arasında özel bir süre vardır. Sinyal seviyesi bu değişikliği yaptığı sırada süreyi değiştirerek, gönderim yapan modem sinyal fazını modüle eder. Alıcı modem bu faz değişikliğini farkedip ve '1' mi yoksa '0' ı mı temsil ettiğini saptayabilir.

Kombinasyonlar

Yüksek hız uygulamalarda, yukarıda bahsedilen 3 tip modülasyonun 2 yada daha fazlası birarada kullanılır.

Örneğin frekans ve faz modülasyonu birleştirilirse bir bit gösterimi yerine aynı anda dört bit gösterimi yapılabilir.

[frekans (2 bit) x faz (2 bit) = 4 bit]

Baud (sinyal birimi) -bps (bit/saniye)

Modemin bilgi gönderim hızı bit/saniye (bps =bits per second) olarak ölçülür. Sık sık bps ile karıştırılan bir diğer terim vardır: Baud (sinyal birimi). Baud ve bps eşit olabilir ama aynı şeyler değildir. Bir modemin baud hızı, taşıyıcı sinyalin hız ölçüsüdür. Bazı modülasyon teknikleri kullanarak taşıyıcının bir devrinde bir çok bit yerleştirebileceği için baud hızı bps hızından daha az olabilir. Örneğin, bir bilginin 4 bitini 600 baundluk modemin taşıyıcısında modüle ettiğinizde, modem işlemini 2400 bps hızıyla tamamlar.

2.1.4. HATA DENETİMİ

Bir modem gönderilen veriyi eski haline getirmek için demodüle etmenin yanında, bazı hata denetimleride yapar. Aşağıdaki değişik tipte hata denetimleri yer almaktadır. Bunlar temel hata denetim metodlarıdır.

- . Eşlik Sağlaması (Parity ckecking)
- . Sağlama Toplamı (checksum)
- . Çevrimsel Hata Denetimi (CRC, Cylic Redundancy Check)

Bu metodların herbiri veri gönderim hatalarını saptamak için (kimi zaman düzeltmek) için kullanılır.

Eşlik Sağlaması

Bir eşlik biti, hata denetiminde yararlanmak için karektere eklenen bittir. Tek eşlik kullanıldığında, bir bayttaki '1' lerin sayısı toplanır. '1' lerin sayısı çiftse, bitlerin toplam sayısı eşlik bitine '1' koyularak tek hale getirilir. Bitlerin sayısı tekse, eşlik bitine '0' yerleştirilir. Alıcı aygıt, alınan bayttaki '1' bitlerinin toplam sayısının tek olup olmadığını kontrol eder. Değilse, gönderimde bir hata olduğu ortaya çıkar.

Aynı süreç (tersi) 'Çift eşlik ' kontrolünde de kullanılır.

Sağlama Toplamı

Bir sağlama toplamı biti, eşlik bitine benzer ancak, bir sağlama toplamı bitinin değeri, bir bayt grubunda bulunan her kolondaki '1' lerin sayısı tarafından saptanır. Örneğin, gönderilmekte olan blokta 12 bayt varsa, her baytın sekiz

pozisyonunun herbiri için sekiz sağlama toplamı biti olacaktır. Sağlama toplamı, yatay denetimi (longitudinal redundancy check) olarak da bilinir.

Çevrimsel Hata denetimi

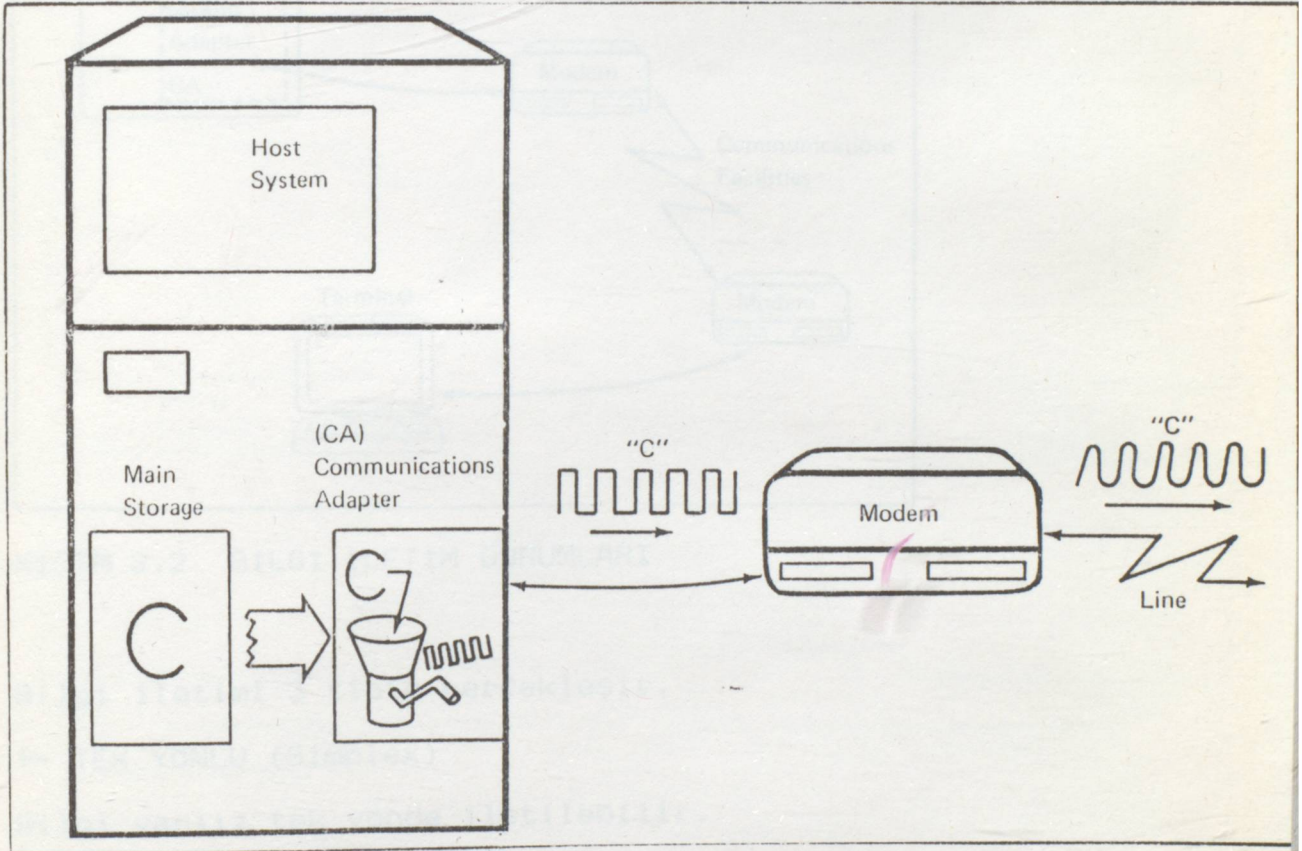
Çevrimsel hata denetim baytı, gönderilen bloğun ikili değerinde gerçekleştirilen bir fonksiyon tarafından saptanır. Daha sonra bu bayt gönderilen bloğun sonuna eklenir. Alım işlemi bittiğinde, alıcı aynı işlemi gerçekleştirir. Sonra da elde ettiği sonuçları gönderilen çevrimsel hata denetimi ile karşılaştırır. Aynı değilse gönderimde hata olduğu ortaya çıkar.

2.1.5. İLETİŞİM UYARLAYICISI (CA= COMMUNICATION ADAPTER)

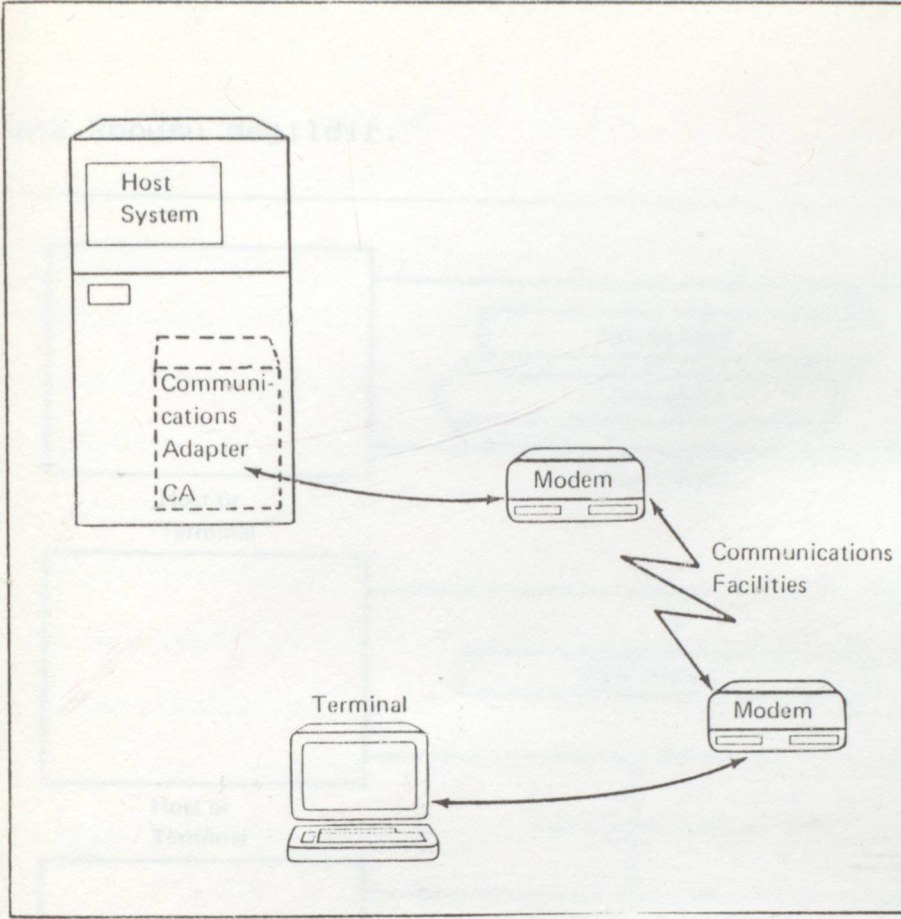
Bellekten paralel bitler biçiminde gelen bilgileri seri bitlere dönüştürerek hat üzerine vermeye veya hat üzerinden tan seri biçimde gelen bilgiyi paralel şekle dönüştürerek bilgisayar belleğine iletmeye yarayan bir parçadır. Bir başka deyişle, bilgisayara uygun biçimdeki bilgiyi hat üzerinden iletişime daha uygun şekle getirir veya tersini yapar.

İletişim uyarlayıcısından hatta çıkılırken bilgi hala sayısal biçimdedir ve iletme tam uygun değildir. Bu yüzden ayrıca modeme gereksinme duyulur. İletişim uyarlayıcısı karakterleri bilgisayar belleğinden paralel olarak alır, seri olarak modeme göndermek için bit bit gönderir. Alış bilgisi

hat ve modemden bit-bit alınır, iletişim uyarlayıcısı karakterleri bilgisayar belleğine paralel olarak göndermek için tekrar düzenler.



İletişim Uyarlayıcısı Operasyonu.



KISIM 2.2 BİLGİ İLETİM DURUMLARI

Bilgi iletimi 3 tipte gerçekleşir.

1- TEK YÖNLÜ (Simplex)

Bilgi yalnız tek yönde iletilebilir.

2- YARI ÇİFT YÖNLÜ (Half Duplex=HDX)

Bilginin bir yönde iletimi tamamlanınca hattın yönü değişir.

İletimin yön değiştirmesi için geçen zamana dönüş zamanı

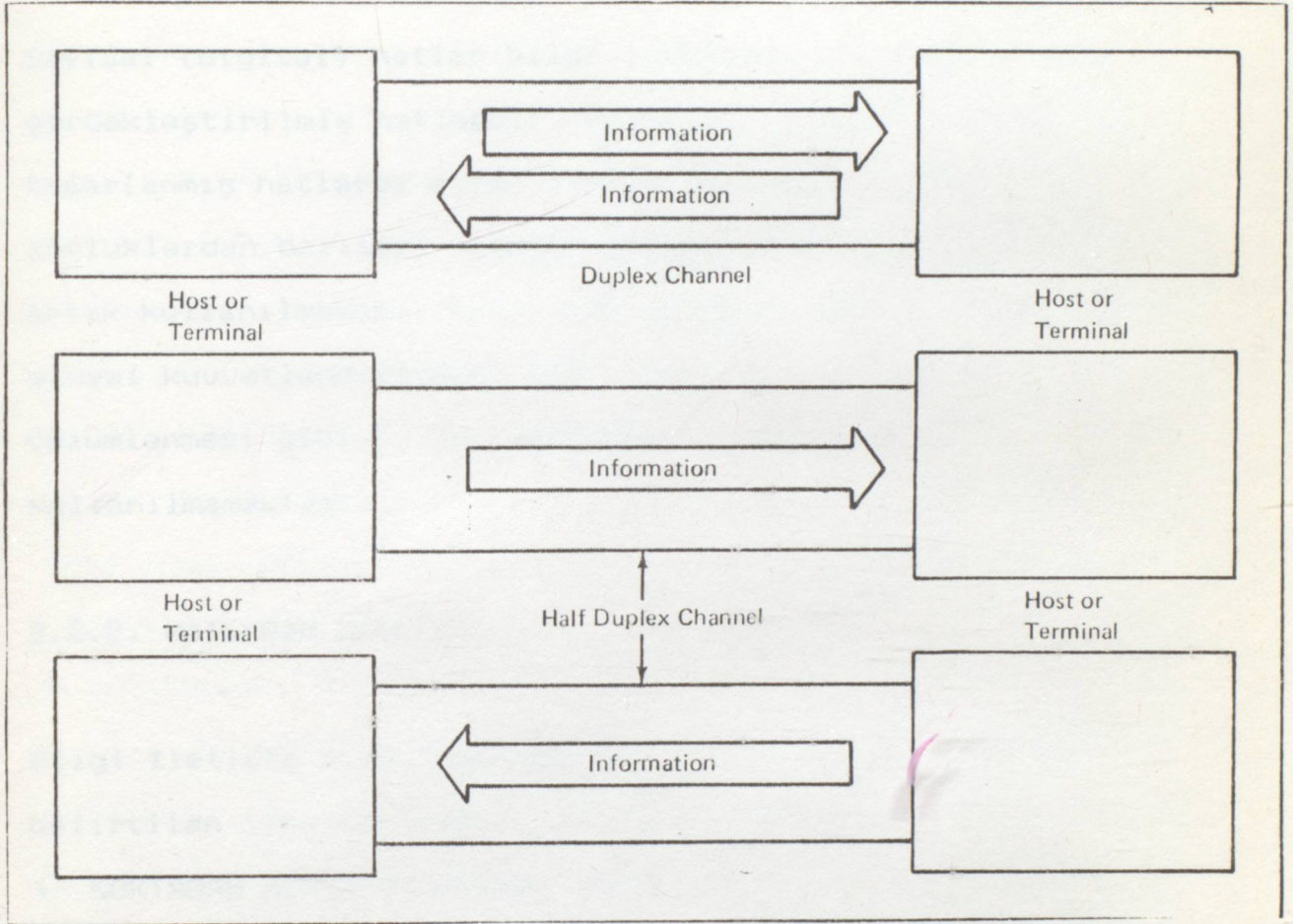
(turn-around time) adı verilir.

3- ÇİFT YÖNLÜ (Full Duplex=FDX)

Bilgi aynı anda her 2 yönde iletilebilir. Bu nedenle yarı

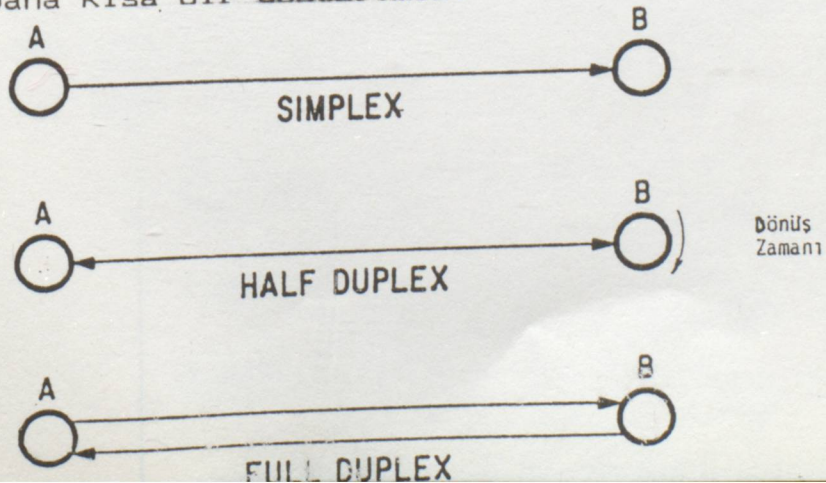
çift yönlü iletimde olduğu gibi bir gecikme (dönüş zamanı)

söz konusu değildir.



. Duplex vs Half Duplex

Veya daha kısa bir österimle:



2.2.1. SAYISAL HATLAR

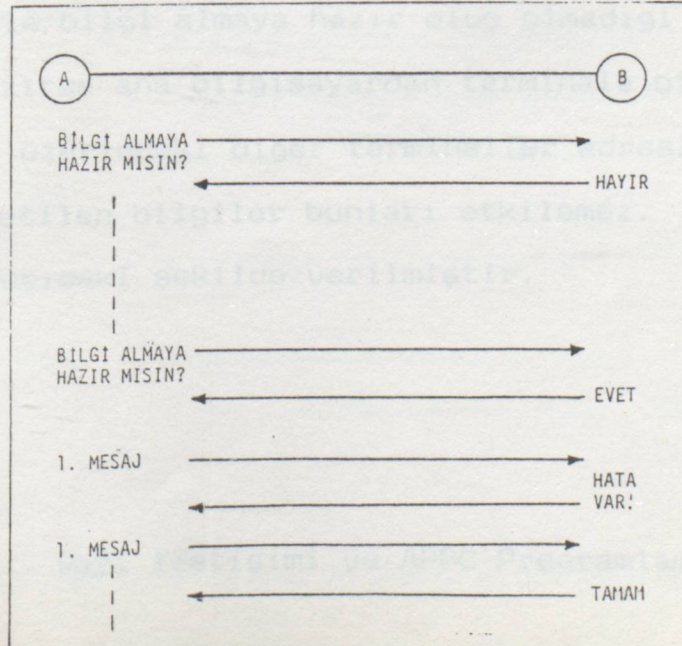
Sayısal (digital) hatlar bilgi iletişimi için özel olarak gerçekleştirilmiş hatlardır. Dolayısıyla ses iletişimi için tasarlanmış hatlarda bilgi iletimi sırasında görülen zorluklardan bazıları ortadan kalkar. Örneğin, modemlerin artık kullanılmaması, hızın çok artması, parazit ayıklama ve sinyal kuvvetlendirilmesi gibi konuların kolaylıkla çözümlenmesi gibi... Bu tip hatlar henüz bizde kullanılmamaktadır.

2.2.2. HATLARIN DENETİMİ

Bilgi iletişim sistemlerinde hatların denetimi aşağıda belirtilen ilkelere uygun şekilde gerçekleşir.

1- NOKTADAN NOKTAYA BAĞLANTI DENETİMİ

Noktadan noktaya bağlanmış bir bilgi iletişim sisteminde hat denetimi ilke olarak aşağıdaki şekilde gerçekleşir.



Bazen her iki istasyon aynı anda karşı tarafa hazır olup olmadığını sorabilir. Buna Çekişme (contention) adı verilir. Bu durumda karışıklığı önlemek için taraflardan birisi diğerlerinden daha öncelikli seçilir. Öncelikli seçilen tarafa birincil istasyon (primary station), diğer tarafta bağımlı istasyon (tributary station) adı verilir. Birincil istasyondan iletilen bilgi sona ermeden diğerleri hattı kullanamazlar.

2- ÇOK NOKTALI BAĞLANTI DENETİMİ

Bu bağlantıda denetim, adresleme (addressing) ve tarama (polling) yöntemleriyle sağlanır. Taraflardan birisi yine birincil seçilir.

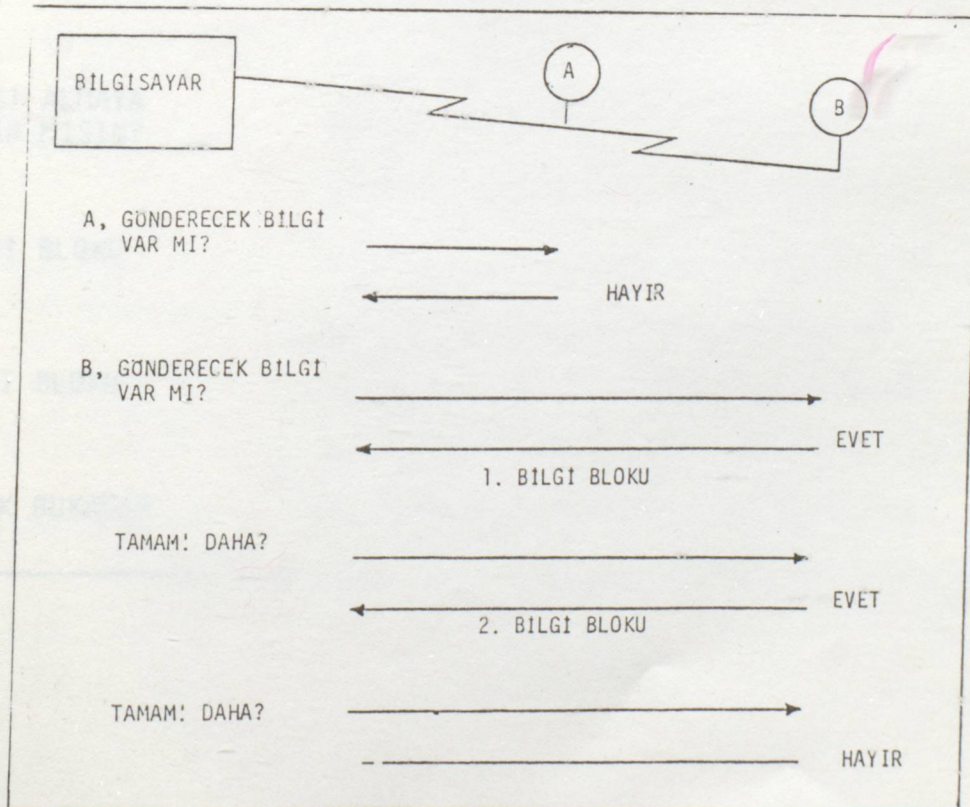
3- ADRESLEME VE TARAMA

Adresleme, bir terminale bilgi göndermek için kullanılan yöntemdir. Hat üzerinde bulunan bütün terminallerin tanıtıcı birer adresi vardır. Ana bilgisayar veya birincil istasyon, belirli bir terminale bilgi göndermeden önce bu terminalin adresini iletir. Böylece terminal adreslenmiş olur. Adresleme ile terminale bilgi almaya hazır olup olmadığı sorulur. Terminal hazırsa ana bilgisayardan terminale bilgi iletimi başlar. Hat üzerindeki diğer terminaller adreslenmediklerinden iletilen bilgiler bunları etkilemez. Adreslemenin ilkeleri aşağıdaki şekilde verilmiştir.

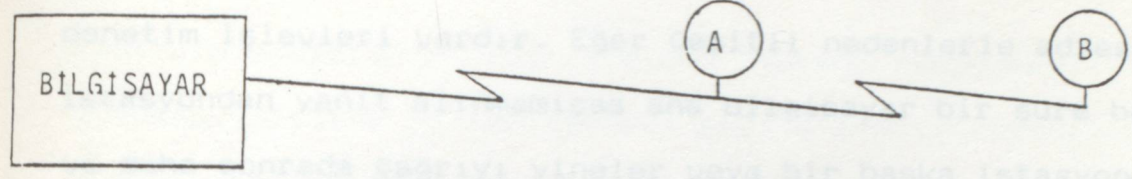
Tarama

Tarama terminallerden bilgi gönderilmesine olanak sağlayan yöntemdir.

Terminallerin veya birincil olmayan tarafın, birincil seçilen bilgisayara bilgi göndermeleri durumunda, bilginin hangi taraftan geldiğini anlayabilmek için yine hat denetiminin ana bilgisayar önceden tanımlanmış bir tarama listesine göre (polling list) terminalleri tarayarak gönderecek bilgileri olup olmadığını sorar. Olumlu yanıt gönderen ilk terminal bilgi iletmeye başlar. Taramanın ilkeleri aşağıda verilmiştir.



Aktarılan terminal belirli bir süre içinde cevap vermezse...



A, BİLGİ ALMAYA
HAZIR MISIN?
← EVET

1. BİLGİ BLOKU
← TAMAM

2. BİLGİ BLOKU
← TAMAM

ŞİMDİLİK BUKADAR
→

B, BİLGİ ALMAYA
HAZIR MISIN?
← EVET

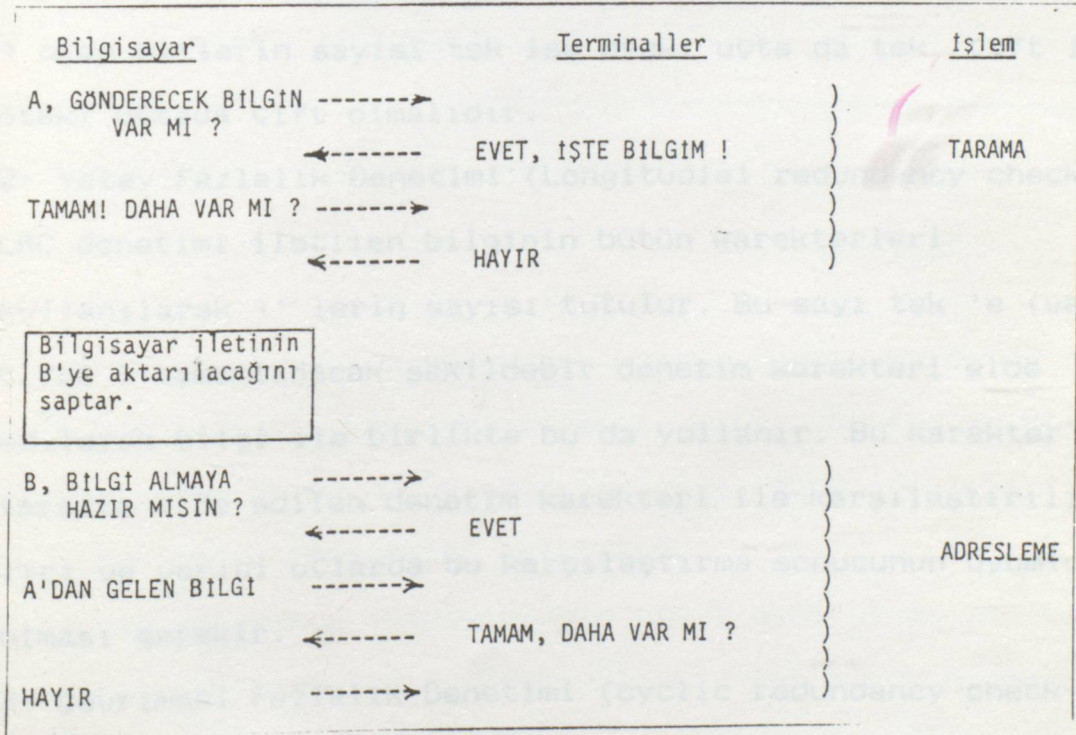
3. BİLGİ BLOKU
← HATA VAR !

3. BİLGİ BLOKU
← TAMAM

ŞİMDİLİK BUKADAR
→

Adreslenen terminal belirli bir süre içinde çağrıya yanıt vermediğinde, ana bilgisayarın gerçekleştirmesi gereken bazı denetim işlevleri vardır. Eğer çeşitli nedenlerle adreslenen istasyondan yanıt alınmamışsa ana bilgisayar bir süre bekler ve daha sonrada çağrıyı yineler veya bir başka istasyonu adresler. İşte çağrıyı yinelemeden veya bir başka istasyonu adreslemeden evvel beklenilen bu süreye zaman aşımı (time out) adı verilir.

Eğer iki terminalin birbirine gönderecek mesajları varsa (message switching), bu iletişim, A' dan B' ye gidecek bilgi olduğu varsayımıyla, şöyle gerçekleşir.



KISIM 2.3 BİLGİ DOĞRULAMA YÖNTEMLERİ

İletişim hatlarında bilgi iletimi olumsuz yönde etkileyen gürültü, girişim (interference) gibi etmenler nedeniyle normal giriş/Çıkış işlemlerine oranla bilgi iletişiminde hata olasılığı daha fazladır. Örneğin her 5000 baytın 1 inde hata olabilir veya hattaki 0,01 saniyelik bir kesinti 10-20 bitin kaybolmasına neden olabilir. Bu yüzden iletilen bilgi üç yöntemle doğrulanır.

1- Düşey Fazlalık Denetimi (vertical redundancy check-VRC)
VRC denetimi, iletilen bütün karakterlerdeki 1 olan bitlerin sayısının tek veya çift olduğunu belirtir. Bilgi iletilirken 1 olan bitlerin sayısı tek ise öteki uçta da tek, çift ise öteki uçta da çift olmalıdır.

2- Yatay Fazlalık Denetimi (Longitudinal redundancy check-LRC)
LRC denetimi iletilen bilginin bütün karakterleri kullanılarak 1' lerin sayısı tutulur. Bu sayı tek 'e (veya Çifte) tamamlanacak şekilde bir denetim karakteri elde edilerek bilgi ile birlikte bu da yollanır. Bu karakter karşı tarafta elde edilen denetim karakteri ile karşılaştırılır. Alıcı ve verici uçlarda bu karşılaştırma sonucunun uyumlu olması gerekir.

3- Çevrimsel Fazlalık Denetimi (cyclic redundancy check-CRC)
İletilen bilginin ikili değeri bir değişmezle bölünür. Bölüm atılır ve blok denetim karakteri denilen ve bölmeden kalanla

VRC →	○	○	●	○	●	●	○	●	●	●				●
B	○	○	○	○	○	○	○	○	○	○				●
A	○	○	○	○	○	○	○	○	○	○				○
8	○	○	○	○	○	○	○	○	○	○				●
4	○	○	○	○	○	○	○	○	○	○				○
2	○	○	○	○	○	○	○	○	○	○				○
1	○	○	○	○	○	○	○	○	○	○				○
KARAKTERLER	1	2	3	4	5	6	7	X	Y	Z				L
														R
														C

● = 1 OLAN BİTLER
 ○ = 0 OLAN BİTLER

Blok
 Sonu
 Karakteri

VRC ve LRC Bilgi Doğrulama Yöntemleri

oluşturulan 16 bitlik bir değer ortaya çıkar. Bu değer bilgiyle birlikte karşıya yollanır. Karşı tarafta da bilginin alınmasından sonra bu karakterin aynısı elde edilirse bilginin doğru iletilildiği anlaşılır.

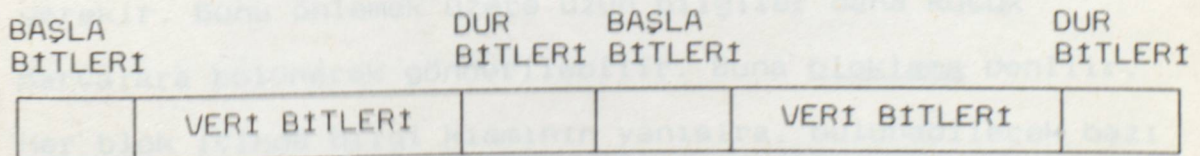
iletişim sırasında istasyonların hepsi aynı bilgi doğrulama yöntemini kullanmalıdır. Herhangi bir iletişimde normal olarak ya VRC, LRC, CRC veya VRC ile LRC' nin bileşimi kullanılır.

KISIM 2.4.1. HAT DENETİMİNDE KULLANILAN YÖNTEMLER

Bilgi iletişim hatlarının denetlenmesinde kullanılan yöntemlere protokol adı verilir. Hat denetiminde 3 protokol vardır.

1- Başlama / Bitme (start / stop)

Daha önceki yıllarda kullanılan ve günümüzde geçerliliğini hemen hemen yitirmiş olan bu yönteme eş zamanlı olmayan iletim (asynchronous) adı da verilir. Yada zaman uyumsuz protokoller denilebilir. Bunlar uziletişim hattındaki veri bitlerinin birer birer gönderildiği protokollerdir. Alıcının, bir karakterin (baytın) başını ve sonunu bilmesi için her baytın başına ve sonuna ek bitler eklenir. Bunlar başla (start) ve dur (stop) bitleri olarak bilinir. Alıcı modem start bitini aldığı anda (stop bitini saptayıncaya kadar) bu bitten sonra gelen kısmın istenilen karakter olduğunu anlar. Start ve stop bitlerinin sırasını ve sayısını belirlemek için kullanılan biçim ve kurallar protokoller tarafından saptanır. İlan tahtası (bulletin board) ve On-line servislerde iletişim kurmak için kullanılan kişisel bilgisayarlar zamanuyumsuz protokoller kullanılır.



Bu tüm bayt grubu (veri ve protokol ek baytları) bir iletim birimi (frame) olarak bilinir. Bu yöntemle bilgi karakter karakter iletildiğinden iletim hızı düşüktür. iletim sırasında bir kaç bitin kaybolması sonucu bütün mesaj karışabilir.

1		1	1	0	0	0	1	0	0		0	D
1		1	1	0	0	0	0	0	1		0	A
1		1	1	1	0	0	0	1	1		0	T
1		1	1	0	0	0	0	0	1		0	A
											SP Bit	ST Bit

Start bitleri

Stop bitleri

Start/Stop Yönetiminde Denetim Bitleri

2- İkili Eş Zamanlı İletişim (Binary Synchronous Communication-BSC)

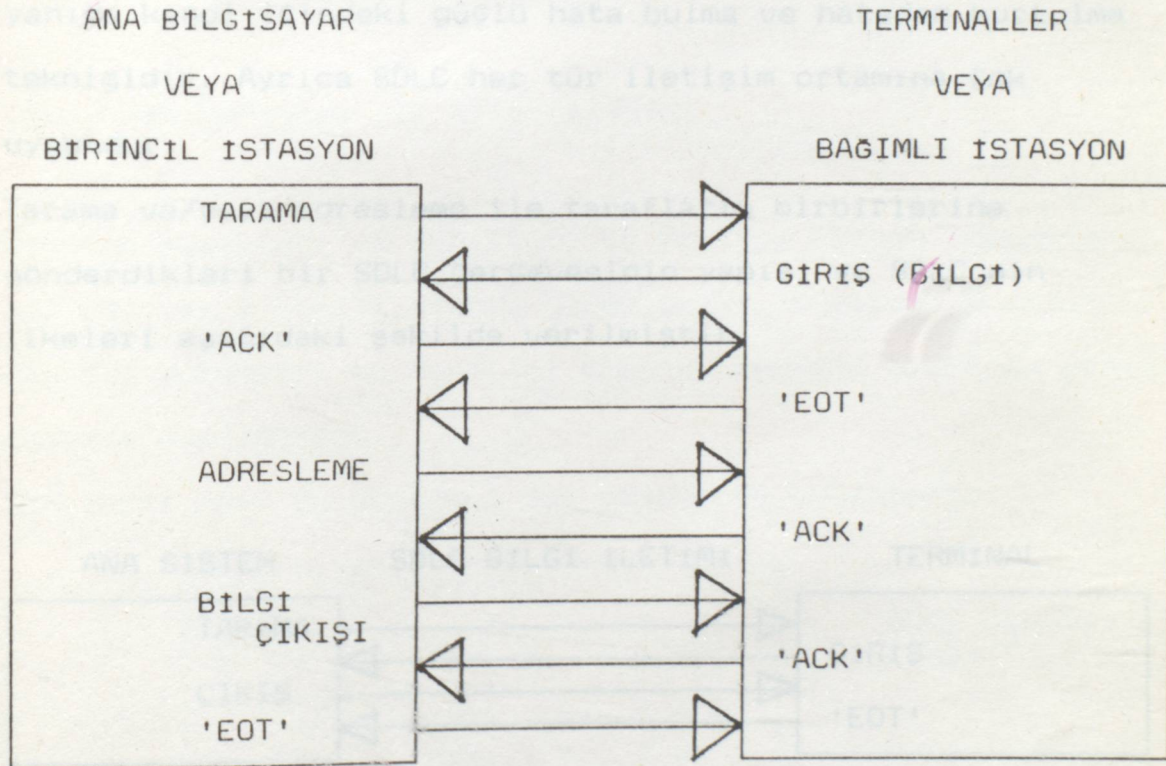
BSC hat denetiminde ana bilgisayara veya terminallere iletilen bilgi BSC denetim karakterleri denilen karakterlerle birlikte sürekli bir bit dizisi görünümünde iletilir. Bu bir tür bilgi bloğudur. Bilginin çok uzun olması durumunda parazit kapma, bilgi kaybı vs. gibi hata durumları ortaya çıkarsa bilginin tümünü tekrar iletmek gerekir. Bunu önlemek üzere uzun bilgiler daha küçük parçalara bölünerek gönderilebilir. Buna bloklama denilir. Her blok içinde bilgi kısmının yanısıra, bulunabilecek bazı

BSC denetim karakterleri şunlardır.

3- Eş Zamanlı SYN: Eş Zamanlama Denetimi (Synchronous Data Link Control)
STX: Bilgi (metin) Başlangıcı
ETX: Bilgi sonu
EOT: İletim Sonu
ACK: Olumlu Yanıt
NAK: Olumsuz Yanıt

Bu protokol IBM'in orijinal zaman uyumlu protokolüdür.

Aşağıdaki şekilde BSC iletimine bir örnek görülmektedir.

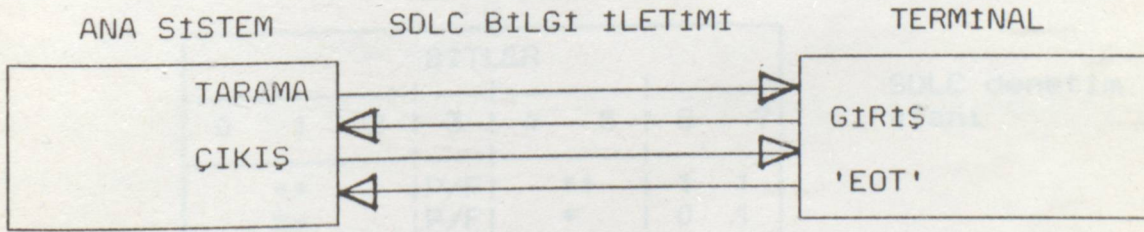


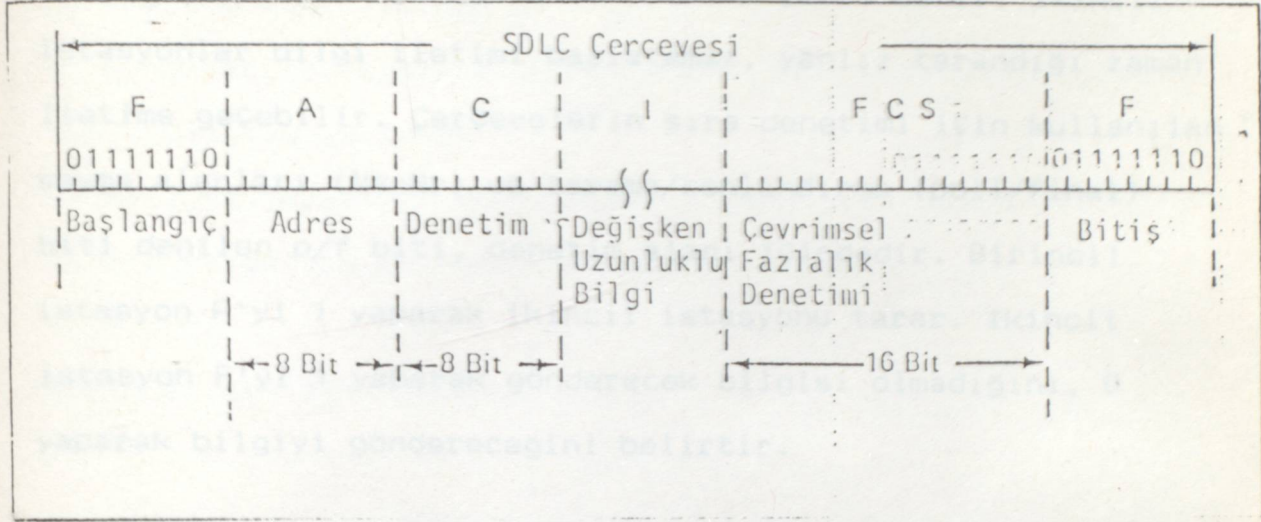
BSC Hat Denetimi ile Bilgi İletimi

3- EşZamanlı Bilgi Bağlantısı Denetimi (Synchronous Data Link Control- SDLC)

En yeni yöntem olan SDLC hat denetiminde ana bilgisayar ile terminaller arasında veya birincil ve bağımlı (primary and tributary) istasyonlar arasında iletilen bütün bilgiler, denetim komutları ve yanıtları çerçeve (frame) denilen gruplar şeklinde yollar. Bu tür iletimde daha az sayıda dönüş (turnaround) gerektiğinden hem iletim hızı hemde hat verimi artmaktadır. Sdlc'nin diğer yöntemlere oranla en üstün yanı sıra kendi içindeki güçlü hata bulma ve hatadan kurtulma tekniğidir. Ayrıca SDLC her tür iletişim ortamına çok uygundur.

Tarama ve/veya adresleme ile tarafların birbirlerine gönderdikleri bir SDLC çerçevesinin yapısı ve SDLC nin ilkeleri aşağıdaki şekilde verilmiştir.





SDLC Hat Denetimi ile Bilgi İletimi

Denetim komutları ve yanıtları bu çerçeve içinde 'denetim' adı verilen 8 bitlik alanda gider gelir. Buradaki 8 bitin çeşitli durumları bu denetimi sağlar. Denetim alanının yapısı aşağıda verilmiştir.

IBM tarafından bu protokol SNA ortamında kullanılır.

BITLER							
0	1	2	3	4	5	6	7
**			P/F	**		1	1
Nr			P/F	*		0	1
Nr			P/F	Ns		0	

SDLC denetim alanı

TARAMA/SONLANMA BİTİ

Birincil istasyonun gönderdiği çerçevelere istek (veya komut)

ikincil istasyonların gönderdiklerine yanıt denir. İkincil istasyonlar bilgi iletimi başlatamaz, yalnız tarandığı zaman ilettime geçebilir. Çerçevelerin sıra denetimi için kullanılan sayma alanları (Ns-Nr) ve tarama/sonlandırma (poll/final) biti denilen p/f biti, denetim alanı içindedir. Birincil istasyon P'yi 1 yaparak ikincil istasyonu tarar. İkincil istasyon F'yi 1 yaparak gönderecek bilgisi olmadığını, 0 yaparak bilgiyi göndereceğini belirtir.

2.4.1.1. İLETİŞİM UYARLAYICILARI

İlk başlarda bilgi iletişim uyarlayıcılarının ana işlevlerine değinerek, paralel bilgiyi seri bitlere dönüştürdüğünü ve tersini de yaptığını belirtmiştik. Şimdi bu noktada bilgi iletişim uyarlayıcılarının diğer işlevlerinede değinerek görevlerini toplu bir şekilde görelim.

- . Bilgiyi almak ve göndermek,
- . Bilgiye denetim karakterlerini eklemek veya bilgiyi bu karakterlerden temizlemek ,
- . Alınan ve gönderilen bilginin doğruluğunu denetlemek ve onaylamak ,
- . Zaman aşımı denetlemek ,
- . Aktarılan hatlar için otomatik arama ve/veya cevaplama işlemlerini yapmak ,
- . Çok noktalı hatlarda tarama yapmak veya başka yerden gelen

taramalara yanıt vermek ,

. Karakterleri bitlere dönüştürmek (Yani paralel bilgiyi seri bilgiye çevirmek ,

. Bitleri karaktere dönüştürmek.(yani seri gelen bitleri tekrar karaktere dönüştürüp belleğe vermek.) .

2.4.1.2. İLETİŞİM DENETLEYİCİLERİ

Programlanabilir denetim birimleri olan iletişim denetleyiciler (CCU), iletişim ağının gerektirdiği programlama işlevlerinin hemen hepsini gerçekleştirmek üzere tasarlanmışlardır. Böylece, ana bilgisayarın yükü hafifletilmiş, daha esnek bir iletişim sistemi elde edilmiş ve bütün olarak sistem verimi artırılmıştır.

İletişim denetim işlevleri denetleyicilere yüklenen ve burada işlem yapan bir denetim programı tarafından gerçekleştirilir. Bu tür bir iletişim ortamında bütün hatların fiziksel ve mantıksal denetimi iletişim denetleyici tarafından yapılır. Hatlar üzerinde her tür terminal bulunabileceği gibi, belirli bir terminalden çeşitli uygulamalara erişme olanağı da vardır.

2.4.1.3. TERMINALLER

'Terminal', bir denetim birimi ve bir veya daha fazla giriş/çıkış aygıtından oluşan bir grup bilgi iletişim

ekipmanına verilen isimdir. Ancak terminal sözcüğü genellikle görüntü birimi veya yazıcıları ifade etmek üzere kullanılmaktadır.

Uzak terminallerin bağlantısında denetim biriminden yararlanmak zorunludur. Bu birimin işlevleri şöyle özetlenebilir.

- . Terminal ile iletişim hattı arasındaki bağlantıyı sağlar.
- . Terminal giriş/Çıkış trafiğini denetler.
- . Terminalin ara belleğini (buffer) ve bazı seçimlik özelliklerini gerçekleştirecek devreleri barındırır.
- . Terminalden iletilen bilgiye hat denetim karakterlerini katar; terminale yollanan Çıkış bilgilerindeki hat denetim karakterlerini tanır, denetim karakterlerinin gerektirdiği işlemleri yerine getirir ve bunları bilgi akımından çıkarır.
- . İletişim sırasında hata bulma ve düzeltme işlemlerini gerçekleştirir.

BÖLÜM 3

AS/400 APPC GERÇEKLEŞTİRME VE PROGRAMLAMA

KISIM 3.1 : APPC NİN GENEL GÖRÜNÜMÜ

OS/400 işletim sisteminin sağladığı olanaklar kısaca şunlardır

1- AS/400 her türlü iletişim sistemini kullanabilir.

Desteklenen iletişim yöntemleri şunlardır.

- . BSC
- . SNA
- . Zaman Uyumsuz (asynchronous)

2- Uygulama programları arasındaki iletişimi, iletişim sisteminin ICF (intersystem communication function) işlevi sağlar.

3- APPC (advanced program to program communication) de ICF'in desteklediği bir iletişim türüdür. APPC ile yapılan iletişimde AS/400, uzak sistemlerde programlar (etkileşimli ve toplu) başlatılabilir ve/veya uzak programlar yerel (local) sistemde programlar başlatabilir.

4- APPC ile programlar birbirleriyle konuşabilir.

5- APPC için gerekli bütün yazılımlar OS/400 içinde vardır.

6- APPC, SNA ister.

Hiyerarşik yapısı şöyledir; kim kimin içindedir

Kişiler 3.2 : NAT DESTEĞİ

AS/400 her tür hat bağlantısını destekler. Bu türler şunlardır.

AS/400

OS/400

3.2.1. Aktarmalı: Nokta-Nokta Bağlantı (Switched Point

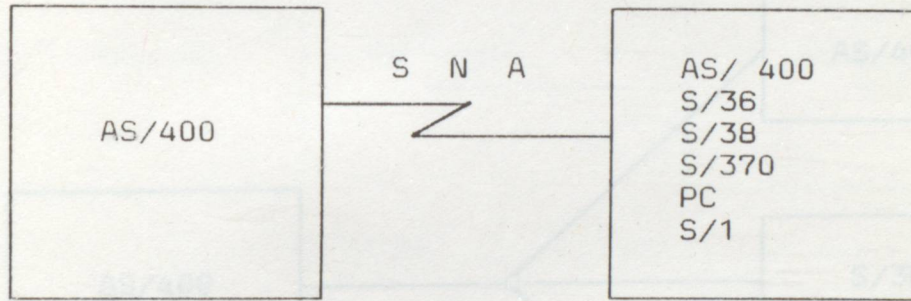
to point) Bu tür bağlantı tipine dial-up da denilir. Bağlantı

görsel kadar para

ICF

APPC

A P P C



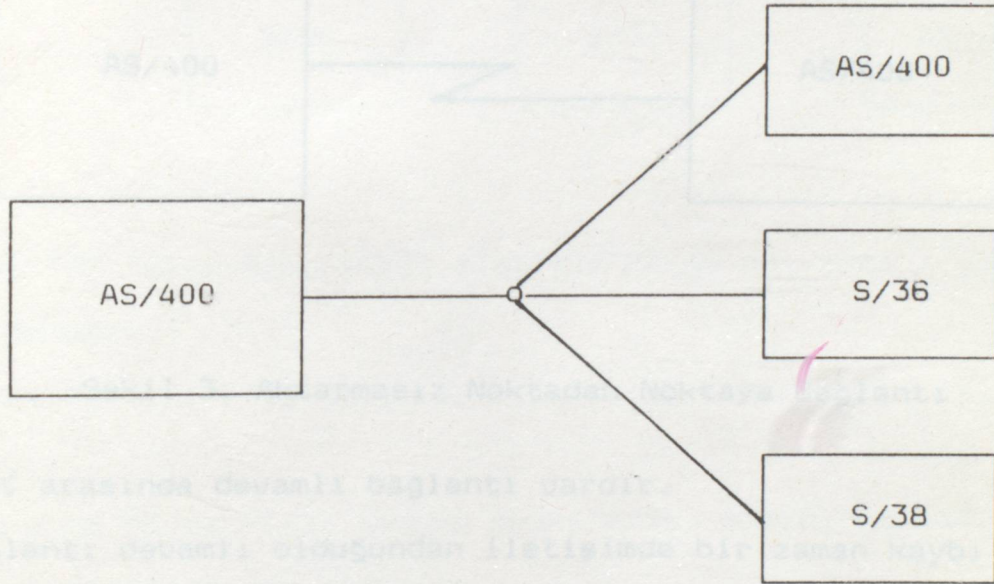
APPC, AS/400 sisteminin diğer IBM sistemleriyle SNA mantıksal birim 6.2 (LU 6.2) yapısını kullanarak heberleşmesini sağlar.

Yukarıdaki şekilde S/370 olarak geçen sistem, gerçekten bir sistem\370 veya 30XX, 43XX, 9370 olabilir. Fakat bunlarında benzer düzeyde bir APPC desteğine, örneğin CICS/VS' e sahip olmaları gerekmektedir. APPC türü iletişimde 2 mantıksal birim (LU) arasında en çok 512 çalışma devresi (session) başlatmak mümkündür.

KISIM 3.2 : HAT DESTEĞİ

AS/400 her tür hat bağlantısını destekler. Bu türler şunlardır.

3.2.1. Aktarmalı Noktadan Noktaya Bağlantı (Switched Point to point) Bu tür bağlantı tipine dial-up ta denilir. Bağlantı süresi kadar para ödenir.



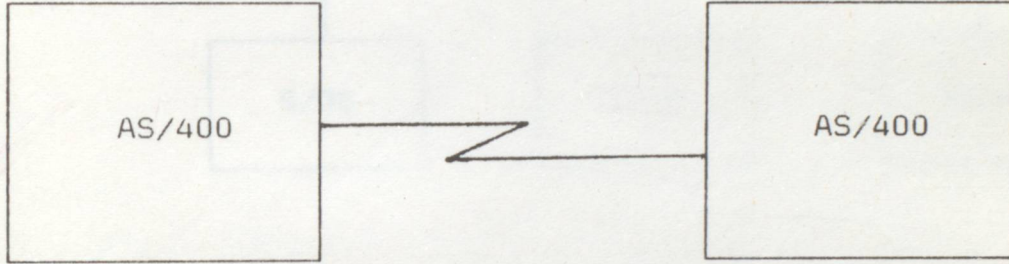
Şekil 2. Aktarmalı Noktadan Noktaya Bağlantı

- . Kendiliğinden (automatic) veya elle (manuel) aramalı veya cevap vermeli olabilir.
- . Bilgi iletişimine başlamadan evvel 2 taraf arasında

bağlantı sağlanmış olmalıdır.

. Bu amaçla normal telefon hatları kullanılabilir.

3.2.2. Aktarmasız Noktadan Noktaya Bağlantı (Nonswitched Point to point)



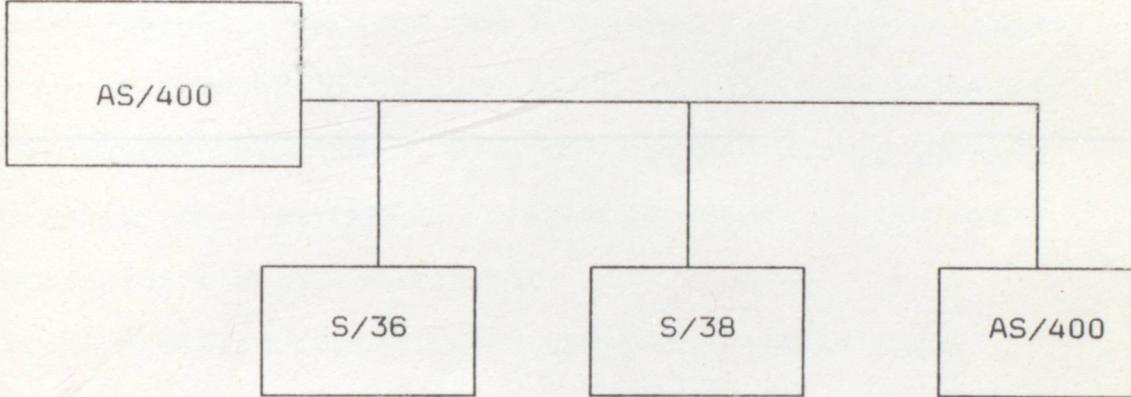
Şekil 3. Aktarmasız Noktadan Noktaya Bağlantı

- . 2 uç arasında devamlı bağlantı vardır.
- . Bağlantı devamlı olduğundan iletişimde bir zaman kaybı olmaz.

3.2.3. Aktarmasız Çok Noktalı Bağlantı (Nonswitched Multipoint)

- . Hattın bazı bölümleri ortak kullanıldığından masraflar azalır.
- . Yoğun bir bilgi trafiği olacaksa hattın ortak kullanılacak

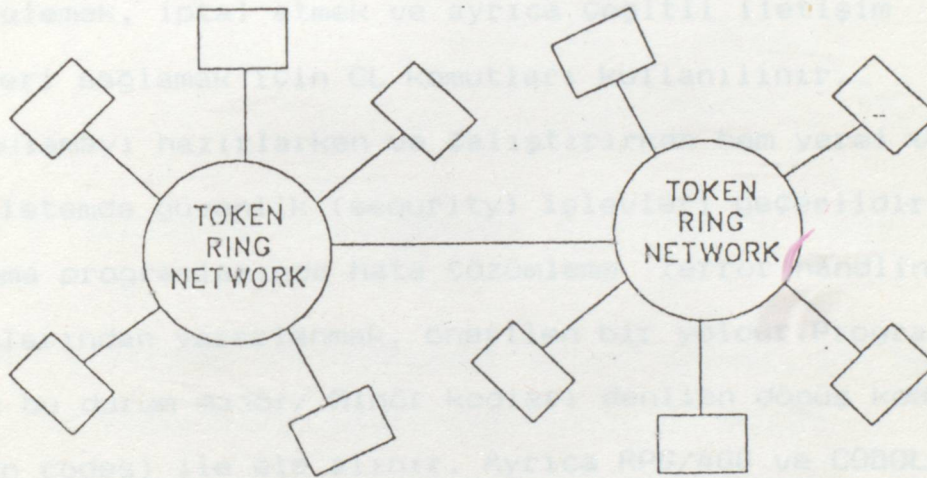
bölümlerinin tasarımı iyi yapılmalıdır.



3.2.4. IBM Token-Ring Yerel İletişim Bölge Ağı (LAN - Local Area Network)

- . Önce PC tipi bilgisayarlar için geliştirilmiş bir iletişim ağıdır.
- . Şu anda PC ve PS/2' lerden başka S/36, S/370, AS/400' ler de bu bağlantıda yer alabilir.
- . LAN birden fazla aynı lokasyonda bağlanması ve aralarında bilgi alışverişi yapılmasıdır. Aynı lokasyon denilince 10 km'lik bir yayılmadan söz edilir. Hız çok iyidir, 1 Mbps den fazla dır.
- . LAN'a uygun bir bağlantı şekli token-ring dir. Diğer i ise PC network tür.
- . PrensiP RiNG içinde bir token dolaşıyor. Token '1

yakalayan , adres verip bilgiyi gönderiyor. Karşı taraf bilgiyi alıp regenerate edip tokene tekrar yükleyip ilk yere gönderiyor. Alınan ve gönderilen bilgi aynıysa TOKEN serbest oluyor.



KISIM 3.3 OS/400 DESTEĞİ (APPC için)

İki uygulama programı arasında bilgi alışverişi

1- ICF kütükleri ile düzenlenir. Bu amaçla DDS kodlaması yapmak gerekir. Yapılan DDS kodlaması ile bilginin formatı tanımlanır ve bilgi alışverişinde kullanılacak işlemlerin özellikleri belirtilir. S/36 dan bilinen sistem destekli formatlar da (\$\$FORMATS), eğer istenirse , ufak tefek değişikliklerde kullanılabilir.

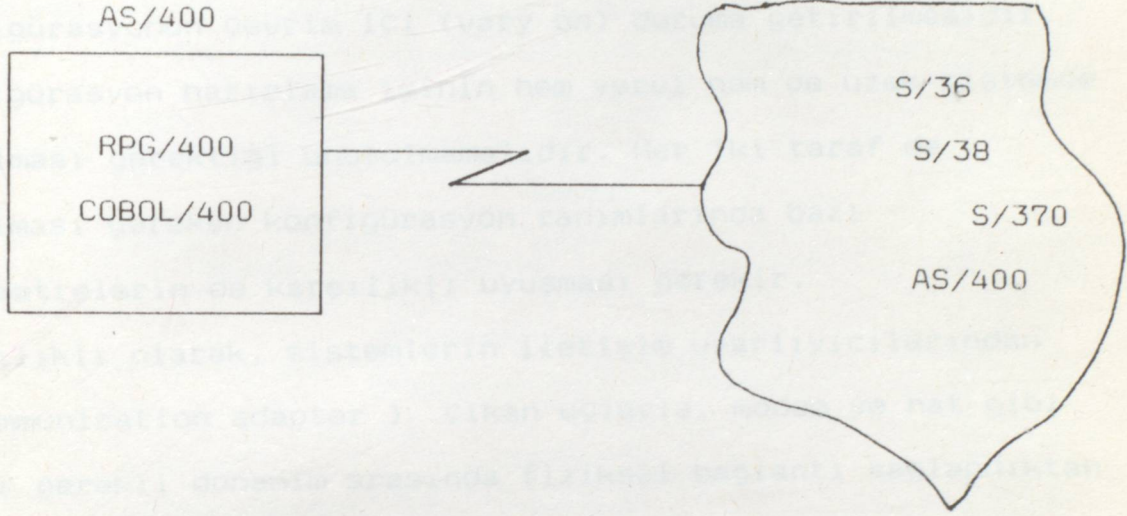
İletişim konfigürasyonlarını yaratmak, değiştirmek, görüntülemek, iptal etmek ve ayrıca çeşitli iletişim işlevleri sağlamak için CL komutları kullanılır.

3- Uygulamayı hazırlarken ve çalıştırırken hem yerel ve hemde uzak sistemde güvenlik (security) işlevleri geçerlidir.

Uygulama programlarında hata çözümüleme (error handling) olanaklarından yararlanmak, önerilen bir yoldur. Programlar içinde bu durum majör/ Minör kodları denilen dönüş kodları (return codes) ile ele alınır. Ayrıca RPG/400 ve COBOL/400' dede çeşitli durum kodları (status codes), benzer amaçla kullanılabilir. ICF' in desteklediği programlama dilleride RPG/400 ve COBOL/400, C/400 ve PL/I dir.

Yerel ve uzak sistemlerde aynı programlama dilinde yazılmış programlar birbiriyle konuşabileceği gibi, ayrı dillerdeki iki programda (örneğin biri RPG diğeri COBOL) birbiriyle konuşabilir. Bir programcının, çok ufak farklılıklar dışında,

karşı taraftaki sistemin ne olduğu ve orada hangi protokol kullanıldığı konusuyla ilgilenilmesi ve program mantığında bir değişiklik yapması gerekmez.



KISIM 3.4 İLETİŞİME BAŞLAYABİLMEK İÇİN GEREKEN HAZIRLIKLAR

Bir APPC iletişime başlayabilmek için 2 aşamalı bir hazırlık gerekir. Bu aşamalardan birincisi konfigürasyonu hazırlamaktır. İkincisi de programları hazırlamaktır. Bu amaçla şu tanımlar yapılmalıdır.

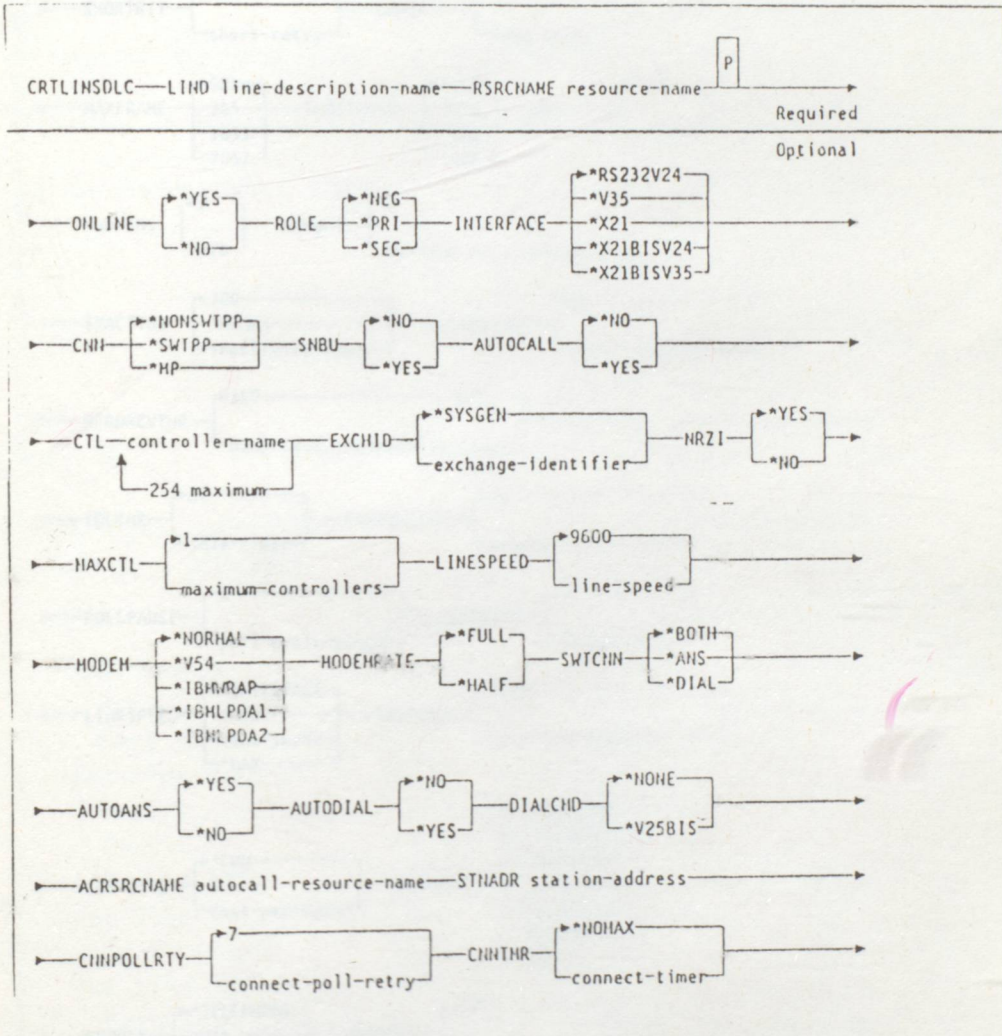
- . Hat tanımı
- . Denetim birimi tanımı
- . Aygıt tanımı
- . Mod tanımı

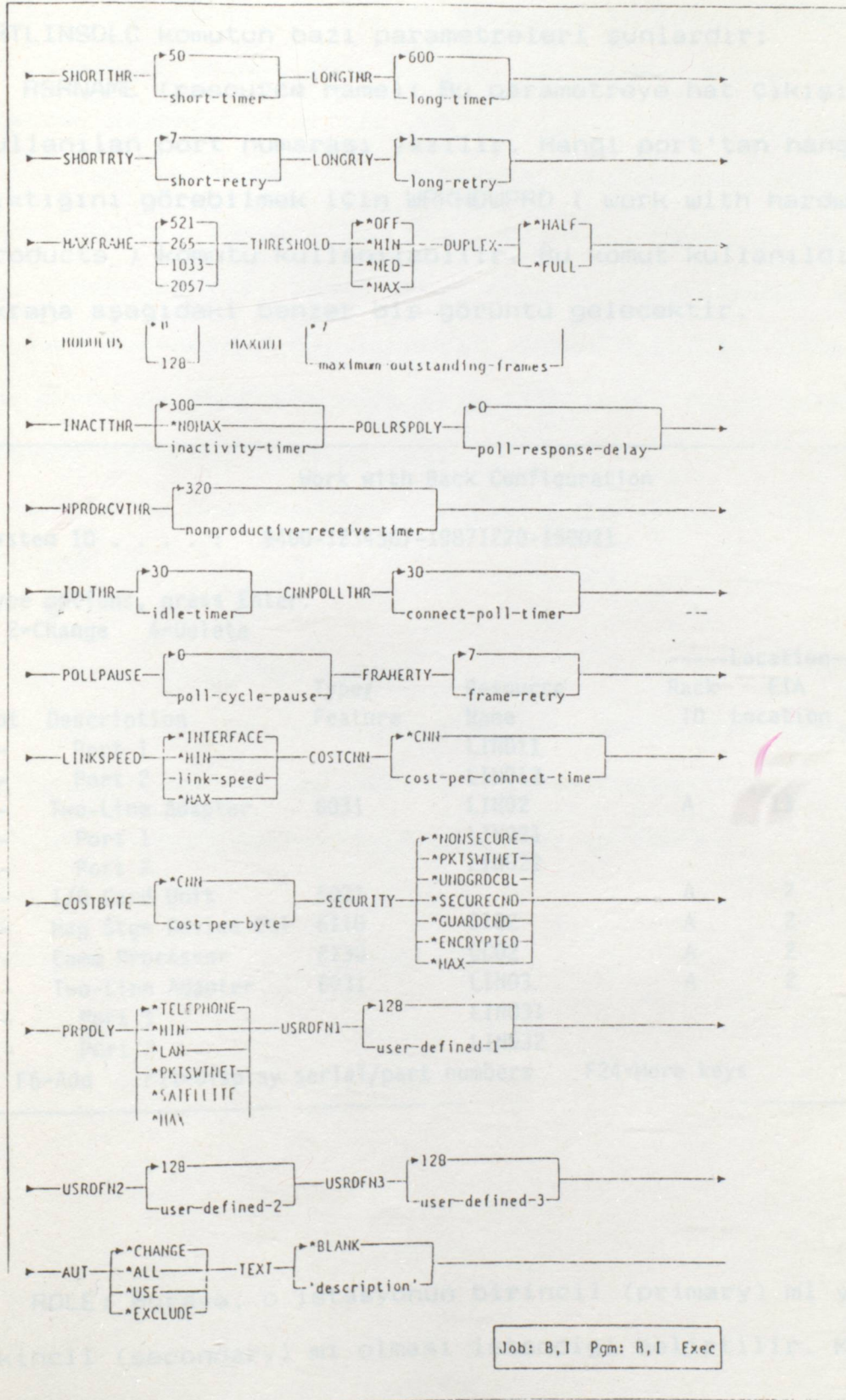
Bu tanımların her seferinde yapılması gerekmez. Her konfigürasyon için bir kere yapılması yeterlidir. Daha sonra istenirse tanımlar üzerinde değişiklikler yapılabilir. Üçüncüsü APPC iletişiminin ikinci aşaması ise hazırlanan bu konfigürasyonun çevrim içi (vary on) duruma getirilmesidir. Konfigürasyon hazırlama işinin hem yerel hem de uzak sistemde yapılması gerektiği unutulmamalıdır. Her iki taraf da yapılması gereken konfigürasyon tanımlarında bazı parametrelerin de karşılıklı uyuşması gerekir. Karşılıklı olarak, sistemlerin iletişim uyarlıyıcılarından (communication adapter) çıkan uçlarla, modem ve hat gibi diğer gerekli donanım arasında fiziksel bağlantı sağlandıktan sonra tanımlara geçilir.

3.4.1. HAT TANIMI

Hat tanımı yapabilmek için CRTLINS DLC (create line description SDLC) komutu kullanılır.

Bu komutun parametreleriyle birlikte görünümü şöyledir.





CRTLINS DLC komutun bazı parametreleri şunlardır:

. RSRNAME (resource name): Bu parametreye hat çıkışı için kullanılan port numarası yazılır. Hangi port'tan hangi hattın çıktığını görebilmek için WRKHDWPRD (work with hardware products) komutu kullanılabilir. Bu komut kullanıldığında ekrana aşağıdaki benzer bir görüntü gelecektir.

Work with Rack Configuration						
System ID : 9400-1234567-19871220-150021						
Type options, press Enter.						
2=Change 4=Delete						
		Type/ Feature	Resource Name	-----Location-----		
Opt	Description			Rack ID	EIA Location	Card Slot
-	Port 1		LIN011			
-	Port 2		LIN012			
-	Two-Line Adapter	6031	LIN02	A	13	11
-	Port 1		LIN021			
-	Port 2		LIN022			
-	I/O Card Unit	5021		A	2	
-	Mag Stge Device Ctl	6110	SI02	A	2	1
-	Comm Processor	6130	CC02	A	2	2
-	Two-Line Adapter	6031	LIN03	A	2	3
-	Port 1		LIN031			
-	Port 2		LIN032			
F6=Add F11=Display serial/part numbers F24=More keys						More...

. ROLE: Buraya, o istasyonun birincil (primary) mi yoksa ikincil (secondary) mi olması istendiği belirtilir. Karşı

tarafın tanımları yapılırken buradakinin tersi alınmalıdır.

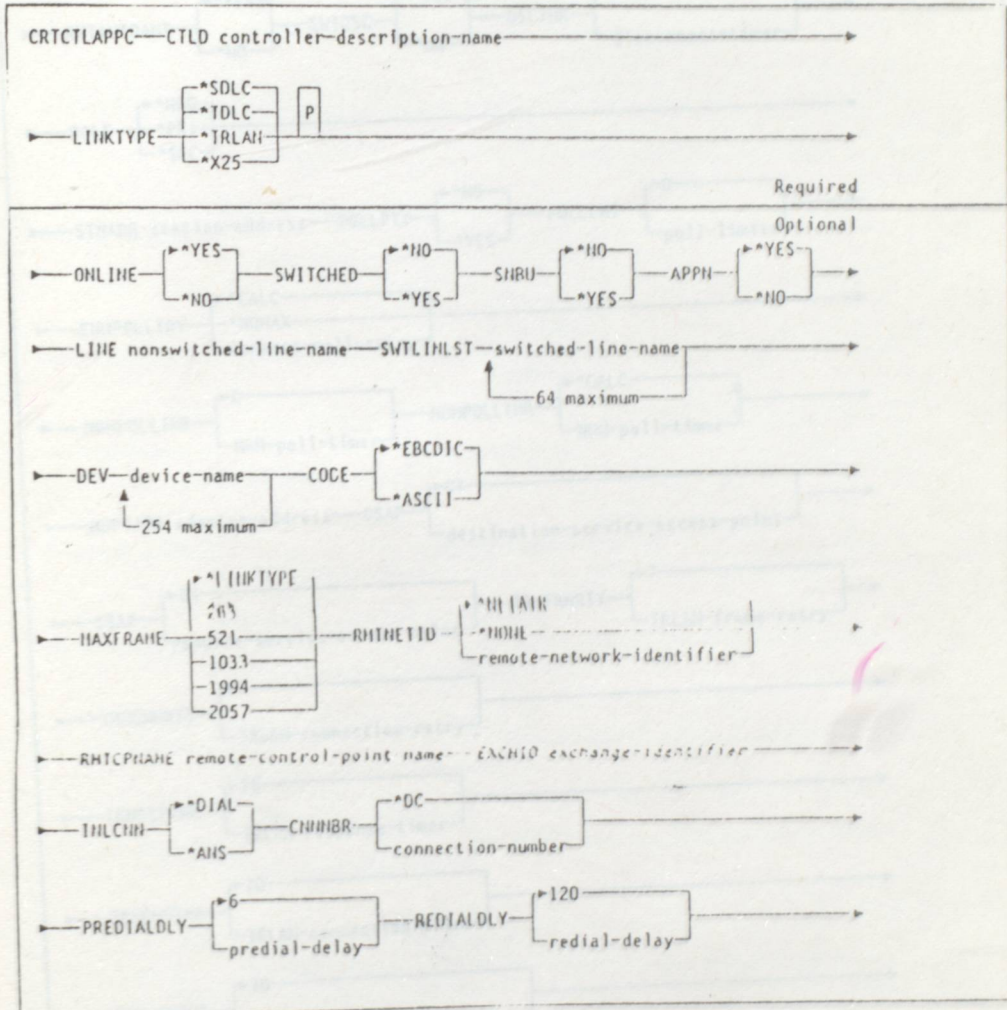
. EXCHID (Exchange ID): Buraya, tanımlamakta olan hat için sisteme atanacak olan bit değişim tanıtım kodu yazılmalıdır. Bunun formatı AS/400 için 056XXXXX şeklindedir. XXXXX yerine arzulanan bir rakam grubu yazılabilir. Fakat bu rakamları unutmamak gerekir, çünkü karşı tarafın tanımlarında da bunlara gereksinim duyulacaktır. Karşı taraf bir başka AS/400 ise orada da 056XXXXX formatı kullanılmalı fakat XXXXX'e değişik rakamlar verilmelidir.

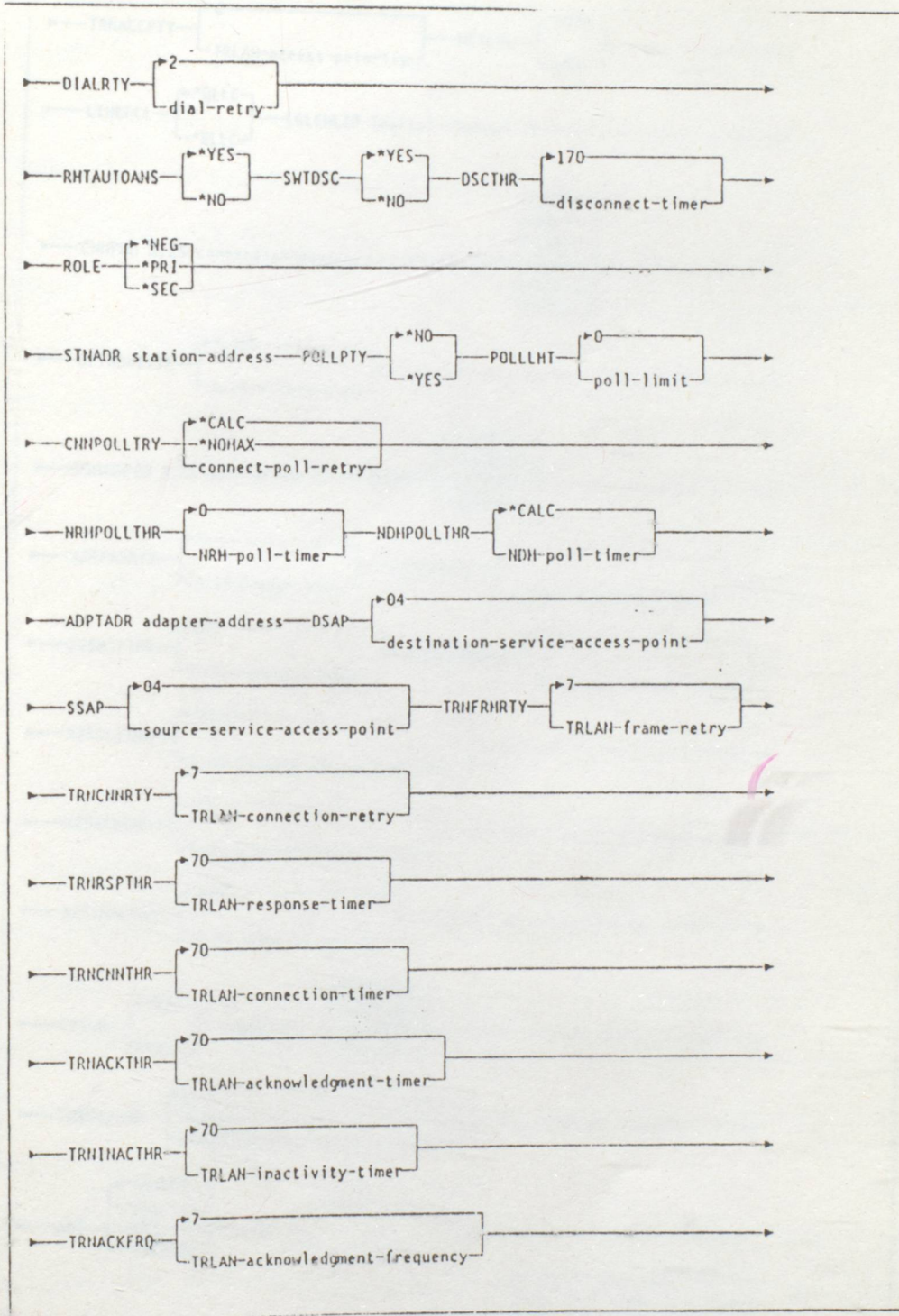
S/38 için bu değişim kodu 022 ile, S/36 için ise 03E ile başlamalıdır.

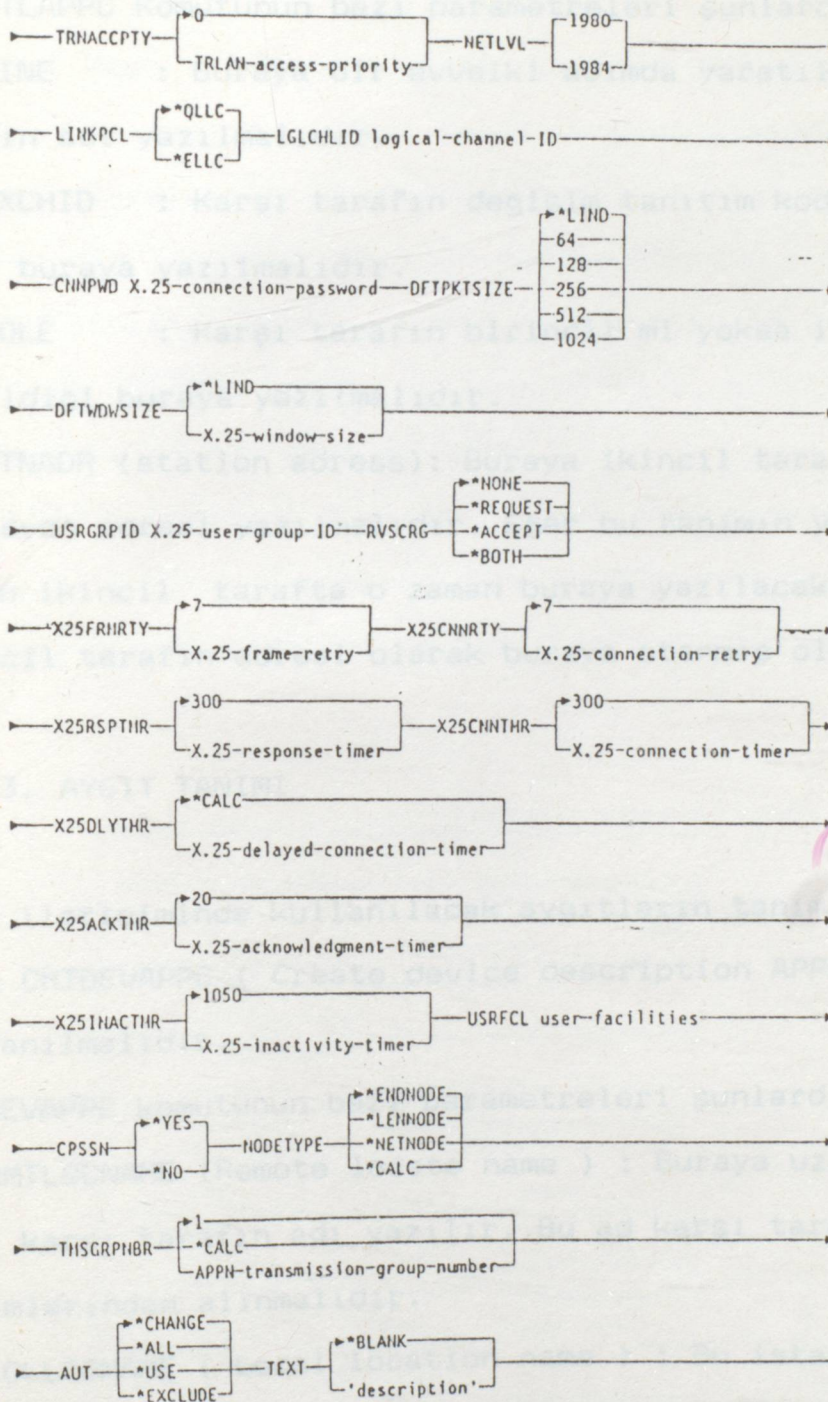
3.4.2 DENETİM BİRİMİ TANIMI

Denetim birimi tanımı yapabilmek için CRTCTLAPPC (create controller description APPC) komutu kullanılır.

Bu komutun parametreleriyle görünümü şöyledir.







Job: B,I Pgm: B,I Exec

CRTCTLAPPC Komutunun bazı parametreleri şunlardır.

- . LINE : Buraya bir evvelki adımda yaratılmış olan hattın adı yazılmalıdır.
- . EXCHID : Karşı tarafın değişim tanıtım kodu (exchange ID) buraya yazılmalıdır.
- . ROLE : Karşı tarafın birincil mi yoksa ikincil mi seçildiği buraya yazılmalıdır.
- . STNADR (station adress): Buraya ikincil tarafa atanan istasyon adresi yazılmalıdır. Eğer bu tanımın yapıldığı yer zaten ikincil tarafta o zaman buraya yazılacak rakam o ikincil tarafın adresi olarak buraya atanmış olur.

3.4.3. AYGIT TANIMI

APPC iletişimde kullanılacak aygıtların tanımını yapabilmek için CRTDEVAPPC (Create device description APPC) komutu kullanılmalıdır.

CRTDEVAPPC komutunun bazı parametreleri şunlardır.

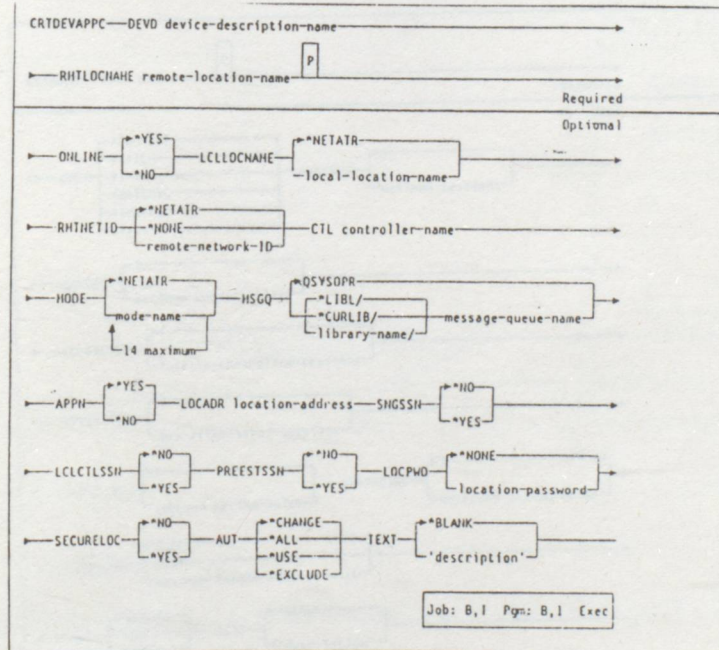
- . RMTLOCNAME (Remote locate name) : Buraya uzak istasyonun yani karşı tarafın adı yazılır. Bu ad karşı tarafın tanımlarından alınmalıdır.
- . LCLLOCNAME (Local location name) : Bu istasyona atanacak olan isim, bu parametrede verilir. Karşı tarafın RMTLOCNAME parametresine burada atanan isim aynen yazılmalıdır.
- . CTL (Controller) : Buraya, bir evvelki adımda tanımlanan

denetim biriminin adı yazılmalıdır.

. MOD (Mode) : Buraya, karşı tarafla da uyumlu olan bir mod adı yazılmalıdır. Mod, yerel ve uzak sistemler arasındaki çalışma devrelerinin (sessions) özelliklerini belirtmeye yarayan bir tanımdır. Varolan modlar DSPMODD (Display mode) ile görülebilir. Varolanlardan farklı bir mod yaratılmak istenirse CRTMODD komutu kullanılmalıdır. Bu komutun görünümü bir sonraki adımda verilmektedir.

. LOCADR (Location Address) : Bir APPC denetim birimi ile çalışacak aygıtın tanımında buraya 00 yazmak gerekir.

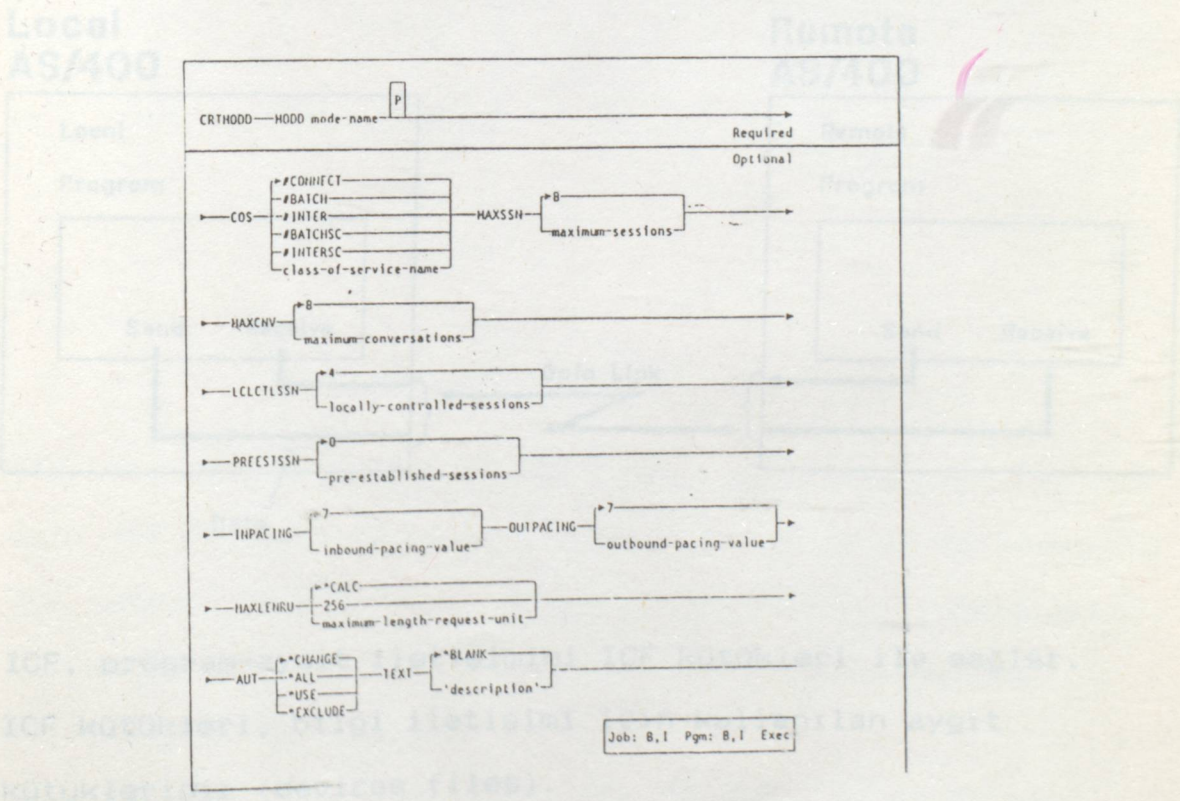
Bu komutun parametreleriyle birlikte görünümü şöyledir.



3.4.4. MOD TANIMI

Aynı isimli bir mod her iki sistemde de bulunmalıdır. Eğer varolan modlar yetersizse veya aynı isimli birer mod tanımı karşılıklı olarak yoksa, CRTMODD (Create mode description) komutu kullanılarak yeni modlar yaratabilir. Bir APPC iletişimi yeni modlar yaratmak zorunlu değildir. Daha evvel yaratılmış olan veya IBM'in hazır gönderdiği mod tanımları APPC iletişimi için kullanılabilir.

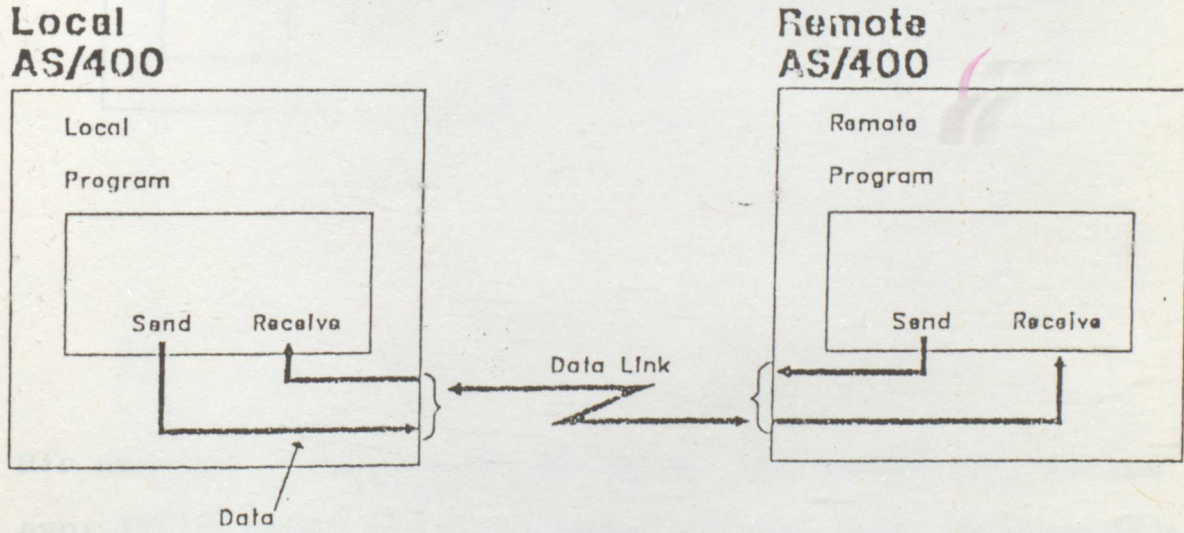
CRTMODD komutunun parametreleriyle birlikte görünümü şöyledir.



KISIM 3.5. ICF KÜTÜKLERİ

3.5.1. GENEL BİLGİLER

ICF bilgi iletişimi için hazırlanan aygıt kütükleridir. AS/400 ICF ile gerçekleşir. OS/400 ICF 'in gerçekleştirdiği bir başka işlevde AS/400 ile diğer donanım aygıtları arasında iletişimi sağlamaktadır. Bir uygulama programında programlama dili işlemleri ICF iletişim işlevleri birlikte ele alınır.

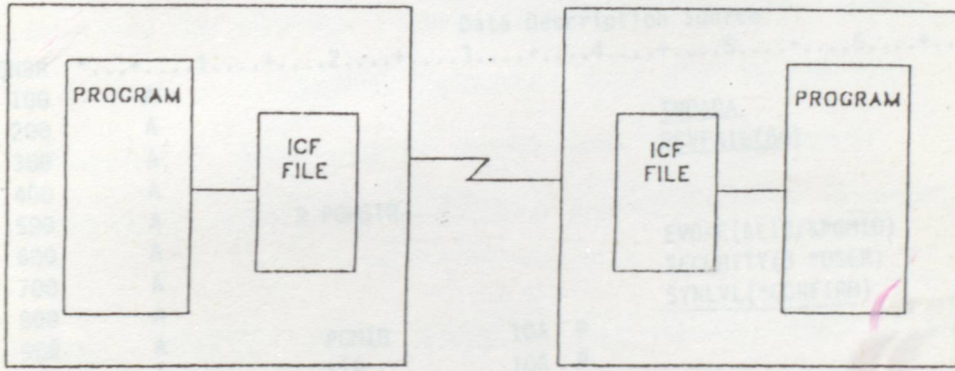


ICF, program-aygıt iletişimini ICF kütükleri ile sağlar. ICF kütükleri, bilgi iletişimi için kullanılan aygıt kütükleridir (devices files).

Bir ICF kütüğü, uygulama programı tarafından kullanılacak olan kayıt formatlarını içerir. Bu kayıt formatları daha sonra RPG ve/veya C/400 ve/veya COBOL tarafından kullanılır.

LOCAL AS/400

REMOTE AS/400



Bir program içinde birkaç ICF kütüğü kullanılabilir. Ayrıca aynı ICF kütüğünü birkaç programın kullanılması da mümkündür. ICF kütüğü, program dışı tanımlanmış (externally described) bir kütük olup, kaynak deyimleri SEU kullanılarak DDS belirtimlerinde kodlanır.

SEU ile girilirken file tipi ICCF verilir. CL de prompt

istenirken BC yani IPBC gibi yazılır.

3.5.2. ICF Kütükleri için DDS Kodlamaları:

ICF kütükleri için diğer aygıt kütüklerine benzer bir DDS kodlaması yapmak gerekir. Bu amaçla, ICF DDS anahtar sözcüklerinden (keywords) yararlanılır. Aşağıdaki şekilde tipik bir ICF kütüğü için DDS kodlaması görülmektedir.

SEQNBR	*...+....1....+....2....+....3....+....4....+....5....+....6....+....7.	Data Description Source
100	A	
200	A	INDARA
300	A	RCVFAIL(66)
400	A	
500	A	R PGMSTR
600	A	EVOKE(&LIB/&PGMID)
700	A	SECURITY(3 *USER)
800	A	SYNLVL(*CONFIRM) ?
900	A	PGMID 10A P
1000	A	LIB 10A P
1100	A	
1200	A	R ITEMRQ
1300	A	RCVDETACH(82)
1400	A	CONFIRM
1500	A	INVITE
1600	A	PARTNM 5A
1700	A	
1800	A	R ITEMDS
1900	A	PARTDS 25A
2000	A	
2100	A	R ERRDES
2200	A	ERRORD 40A
2300	A	
2400	A	R PGMEND
2500	A	DETACH
2600	A	
2700	A	R PGMERR
2800	A	FAIL
2900	A	
3000	A	R ITEMOK
3100	A	RSPCONFIRM
		***** END OF SOURCE *****

Bir ICF kütüğü hazırlamak için kullanılan ICF DDS anahtar sözcükleri şunlardır.

. EVOKE

Kütük veya kayıt düzeyi olarak kullanılabilir.

Karşı sistemde bir program başlatmaya yarar. Karşı sistemde başlatılan programa hedef (target) program adı verilir. Programlar arası iletişimi ve akışı denetleyen ve hedef programı ne zaman başlatacağına karar veren esas programa da kaynak (source) program denir. Hedef programı başlatmak için EVOKE işlemini bir kez kullanmak yeterlidir. Eğer karşı programın başlatılması başarısızlıkla sonuçlanmışsa EVOKE işlemi tekrarlanabilir.

EVOKE, INVITE' li ve INVITE' siz olmak üzere 2 şekilde kullanılır. INVITE ile birlikte kullanıldığında kaynak program okuma (read) durumunda kalır, karşı programın ise önce bir okuma ve daha sonrada bir yazma (write) işlemi yapması beklenir. INVITE' siz kullanıldığında kaynak programın bir başka yazma işlemi yapması gerekir. Hedef program ise yine önce bir okuma ve sonra da bir yazma yapabilir.

Göstergelerle koşullandırılarak da kullanılabilir.

Sequence Number	Type	Conditioning					Name	Length	Location	Function
		Condition Name	Condition	Condition	Condition	Condition				
00010	A						RCD			
00020	A	01								EVOKE(LIBRARY1/PROGRAM1)
00030	A	02								EVOKE(LIBRARY2/PROGRAM2)
00090	A						RCD2			EVOKE(&FIELD2/&FIELD1 'ABC' 10.1 * FIELD3)
00100	A									
00110	A						FIELD1	10A	P	
00120	A						FIELD2	10A	P	
00130	A						FIELD3	5B	P	

EVOKE

INVITE

Kütük veya kayıt düzeyi olarak kullanılabilir.

Karşı programı bilgi göndermeye davet etmek amacıyla kullanılır. Kullanılması program performansını artırıcı etki yapar.

Göstergelerle kuşullandırılarak ta kullanılabilir.

Sequence Number	Type	Conditioning					Name	Length	Location	Function
		Condition Name	Condition	Condition	Condition	Condition				
00010	A	01					RCD1			INVITE
00020	A									
00030	A						FID1	10		
00040	A						FID2	5		

INVITE

SECURITY

Kütük veya kayıt düzeyi olarak kullanılabilir.

Karşı tarafta bir program başlatırken oraya güvenlik bilgileri göndermeye yarar. Bir ICF kütüğü içinde EVOKE sözcüğü ile birlikte kullanılmalıdır. Aksi takdirde hata verir. Bu sözcük ile belirtilebilecek güvenlik bilgileri şunlardır.

1. Kimlik Tanıtımı- Profile ID
2. Parola- Password
3. Kullanıcı Tanıtımı- User ID

Bu bilgilerle birlikte bazı ayrılmış sözcükler (reserved words) de kullanılabilir:

Security Number	Conditioning	Name	Length	Location	Function
00010	01				SECURITY(2 'JONES' 3 'WHITE')
00020					
00030					
00040					
00050					
00060		R RCD1			
00070	03				SECURITY(2 'JONES' 3 'USER')
00080					EVOKE(LIB2/PGM2)
00090					
00100		R RCD2			
00110					EVOKE(LIB3/PGM3)
00120					
00130		FIELD1	5A		
00140					
00150		R RCD3			
00160	50				SECURITY(2 &CLVAR1 3 &CLVAR2)
00170					EVOKE(LIB4/PGM4)
00180		CLVAR1	10A		
00190		CLVAR2	10A		

SECURITY

* USER : AS/400' deki kullanıcı kimliği kullanılacaktır.
* NONE : Boş (null) bir güvenlik değeri kullanılacaktır
SECURITY sözcüğü göstergelerle koşullandırılarak da
kullanılabilir.

. DETACH

Kütük veya kayıt düzeyi olarak kullanılabilir.

Karşı programa buradaki programın bilgi gönderme işinin
bittiğini ve iletişimi sonlandırmak istediğini belirtmeye
yarar. Herhangi parametresi yoktur. Genellikle kaynak ve
hedef program, DETACH ve RCVDETACH sözcüklerini karşılıklı
olarak kullanırlar. Bir taraf DETACH gönderirse, karşı taraf
da bunu RCVDETACH olarak algılar.

DETACH sözcüğü göstergelerle koşullandırılarak
kullanılabilir.

Conditioning				Name	Length	Location	Function
Requester Number	Condition Name	Condition Name	Condition Name				
1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64
65	66	67	68	69	70	71	72
73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88
89	90	91	92	93	94	95	96
97	98	99	100	101	102	103	104
105	106	107	108	109	110	111	112
113	114	115	116	117	118	119	120
121	122	123	124	125	126	127	128
129	130	131	132	133	134	135	136
137	138	139	140	141	142	143	144
145	146	147	148	149	150	151	152
153	154	155	156	157	158	159	160
161	162	163	164	165	166	167	168
169	170	171	172	173	174	175	176
177	178	179	180	181	182	183	184
185	186	187	188	189	190	191	192
193	194	195	196	197	198	199	200
201	202	203	204	205	206	207	208
209	210	211	212	213	214	215	216
217	218	219	220	221	222	223	224
225	226	227	228	229	230	231	232
233	234	235	236	237	238	239	240
241	242	243	244	245	246	247	248
249	250	251	252	253	254	255	256
257	258	259	260	261	262	263	264
265	266	267	268	269	270	271	272
273	274	275	276	277	278	279	280
281	282	283	284	285	286	287	288
289	290	291	292	293	294	295	296
297	298	299	300	301	302	303	304
305	306	307	308	309	310	311	312
313	314	315	316	317	318	319	320
321	322	323	324	325	326	327	328
329	330	331	332	333	334	335	336
337	338	339	340	341	342	343	344
345	346	347	348	349	350	351	352
353	354	355	356	357	358	359	360
361	362	363	364	365	366	367	368
369	370	371	372	373	374	375	376
377	378	379	380	381	382	383	384
385	386	387	388	389	390	391	392
393	394	395	396	397	398	399	400
401	402	403	404	405	406	407	408
409	410	411	412	413	414	415	416
417	418	419	420	421	422	423	424
425	426	427	428	429	430	431	432
433	434	435	436	437	438	439	440
441	442	443	444	445	446	447	448
449	450	451	452	453	454	455	456
457	458	459	460	461	462	463	464
465	466	467	468	469	470	471	472
473	474	475	476	477	478	479	480
481	482	483	484	485	486	487	488
489	490	491	492	493	494	495	496
497	498	499	500	501	502	503	504
505	506	507	508	509	510	511	512
513	514	515	516	517	518	519	520
521	522	523	524	525	526	527	528
529	530	531	532	533	534	535	536
537	538	539	540	541	542	543	544
545	546	547	548	549	550	551	552
553	554	555	556	557	558	559	560
561	562	563	564	565	566	567	568
569	570	571	572	573	574	575	576
577	578	579	580	581	582	583	584
585	586	587	588	589	590	591	592
593	594	595	596	597	598	599	600
601	602	603	604	605	606	607	608
609	610	611	612	613	614	615	616
617	618	619	620	621	622	623	624
625	626	627	628	629	630	631	632
633	634	635	636	637	638	639	640
641	642	643	644	645	646	647	648
649	650	651	652	653	654	655	656
657	658	659	660	661	662	663	664
665	666	667	668	669	670	671	672
673	674	675	676	677	678	679	680
681	682	683	684	685	686	687	688
689	690	691	692	693	694	695	696
697	698	699	700	701	702	703	704
705	706	707	708	709	710	711	712
713	714	715	716	717	718	719	720
721	722	723	724	725	726	727	728
729	730	731	732	733	734	735	736
737	738	739	740	741	742	743	744
745	746	747	748	749	750	751	752
753	754	755	756	757	758	759	760
761	762	763	764	765	766	767	768
769	770	771	772	773	774	775	776
777	778	779	780	781	782	783	784
785	786	787	788	789	790	791	792
793	794	795	796	797	798	799	800
801	802	803	804	805	806	807	808
809	810	811	812	813	814	815	816
817	818	819	820	821	822	823	824
825	826	827	828	829	830	831	832
833	834	835	836	837	838	839	840
841	842	843	844	845	846	847	848
849	850	851	852	853	854	855	856
857	858	859	860	861	862	863	864
865	866	867	868	869	870	871	872
873	874	875	876	877	878	879	880
881	882	883	884	885	886	887	888
889	890	891	892	893	894	895	896
897	898	899	900	901	902	903	904
905	906	907	908	909	910	911	912
913	914	915	916	917	918	919	920
921	922	923	924	925	926	927	928
929	930	931	932	933	934	935	936
937	938	939	940	941	942	943	944
945	946	947	948	949	950	951	952
953	954	955	956	957	958	959	960
961	962	963	964	965	966	967	968
969	970	971	972	973	974	975	976
977	978	979	980	981	982	983	984
985	986	987	988	989	990	991	992
993	994	995	996	997	998	999	1000

. RCVDETACH

Kütük veya kayıt düzeyi olarak kullanılabilir.

Karşı programdan DETACH işleminin alındığını belirtmek amacıyla kullanılır. Açılacak göstergenin kodlanmasından sonra seçimlik bir metin kısmı yazılabilir. Eğer yazılırsa, bu metin kısmının herhangi bir işlevi yoktur; yalnızca açıklayıcı bilgiler içerir. Metin kısmı 50 karakteri geçmemelidir. Gerekirse gerisi kesilip atılır. Göstergelerle koşullandırılmaz.

Sequence Number	Line Type	Line Number	Condition Name	Name	Length	Location	Function
00010	A						RCVDETACH(44 'Transaction is
00020	A						finished')
00030	A			R RCD			

RCVDETACH

FAIL

Kütük veya kayıt düzeyi olarak kullanılabilir.

Karşı programa, alınan veya gönderilen bilginin geçersiz olduğunu belirtmek amacıyla kullanılır. Parametresi yoktur.

Genellikle kaynak ve hedef program , FAIL ve RCVFAIL sözcüklerini karşılıklı olarak kullanırlar. Bir taraf FAIL gönderirse, karşı taraf da bunu RCVFAIL ile algılar. Göstergelerle koşullandırılarak kullanılabilir.

Sequence Number	Line Type	Line Number	Conditioning					Name	Length	Reference	Location	Functions
			Condition Name	Indicator	Indicator	Indicator	Indicator					
00010	A	1										
00020	A	2										
00030	A	3						RCD				

RCVFAIL

SYNLVL

Kütük veya kayıt düzeyi olarak kullanılabilir.

Bu sözcük, programın eşzamanlama düzeyini belirtmek amacıyla kullanılabilir. SYNLVL sözcüğü yalnızca EVOKE ile birlikte kullanılabilir. Belirtebilecek düzeyler şunlardır:

- * CONFIRM : Yerel veya uzak programda CONFIRM sözcüğü kullanılacaktır.
- * NONE : CONFIRM sözcüğü kullanılmayacaktır.

Bu sözcük göstergelerle koşullandırılarak kullanılabilir.

Sequence Number	Line Type	Line Number	Conditioning					Name	Length	Reference	Location	Functions
			Condition Name	Indicator	Indicator	Indicator	Indicator					
00010	A	1						RCD				
00020	A	2										EVOKE(LIBRARY1/PROGRAM1)
00030	A	3										SYNLVL(*CONFIRM)

SYNLVL

. CONFIRM

Kütük veya kayıt düzeyi olarak kullanılabilir.

Gönderilen bilginin, karşı program tarafından doğru alındığı -
nın onaylanması amacıyla kullanılır. Bu sözcük yalnızca
SYNLVL (* CONFIRM) ile birlikte kullanılabilir. Bu durumdada
genellikle RSPCONFIRM sözcüğü ile karşılıklı olarak
kullanılır. Bir taraf onay (CONFIRM) istediğinde, diğer
taraf bilgiyi doğru almışsa, bunu RSPCONFIRM göndererek
belirtir.

Göstergelerle koşullandırılarak kullanılabilir.

Sequence Number				Conditioning				Name	Length	Location	Function
1	2	3	4	5	6	7	8				
Type				Condition Name				Reference ID	Data Type	Line	Pos
1	2	3	4	5	6	7	8				
00010	A							RCD			
00020	A	01								CONFIRM	
	A										
	A										
	A										

CONFIRM Keyword

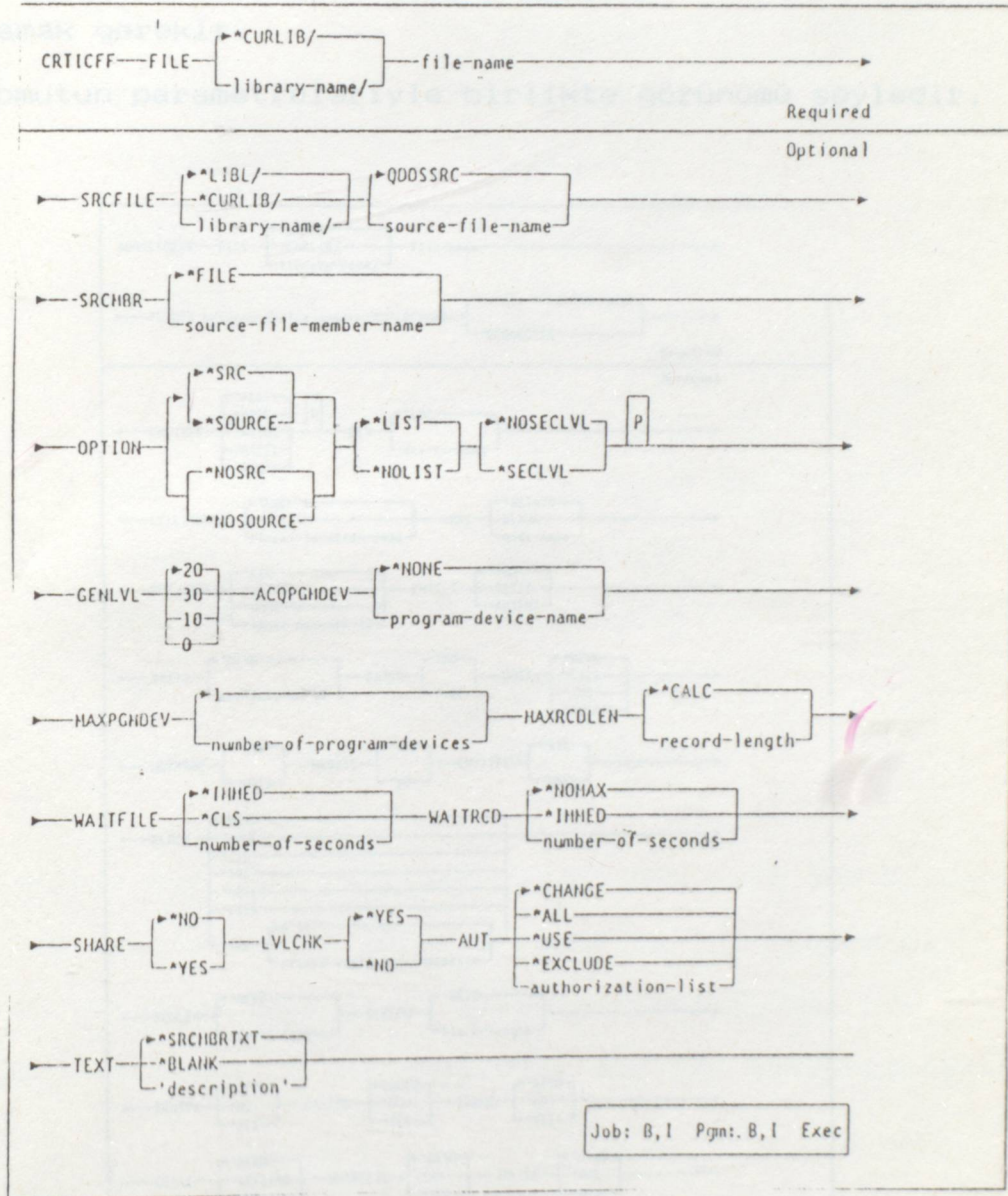
. RSPCONFIRM

Kütük veya kayıt düzeyi olarak kullanılabilir.

Karşı taraftan alınan bir CONFIRM isteğine olumlu yanıt
vermek amacıyla kullanılır. Parametresi yoktur.
Göstergelerle koşullandırılarak kullanılabilir.

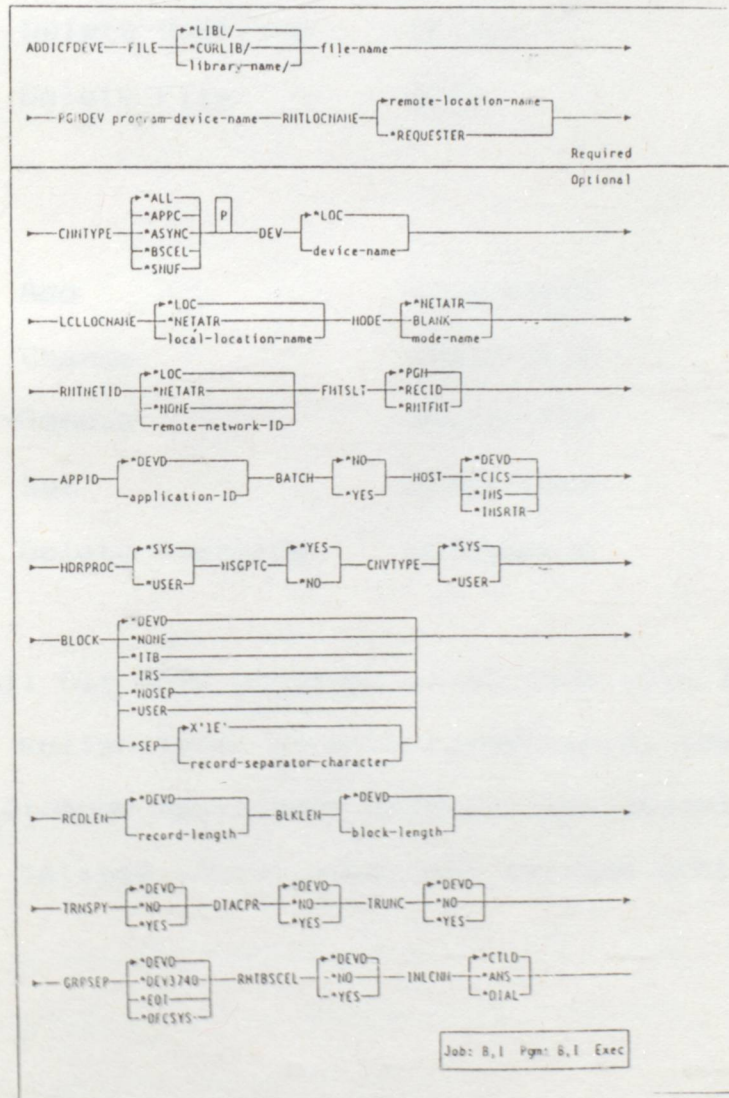
Sequence Number	Form Type	Conditioning					Name	Length	Location	Functions				
		Condition Name												
		Initial	Initial	Initial	Initial	Initial								
1	2	3	4	5	6	7	8	9	10	11				
12	13	14	15	16	17	18	19	20	21	22				
23	24	25	26	27	28	29	30	31	32	33				
34	35	36	37	38	39	40	41	42	43	44				
45	46	47	48	49	50	51	52	53	54	55				
56	57	58	59	60	61	62	63	64	65	66				
67	68	69	70	71	72	73	74	75	76	77				
78	79	80	81	82	83	84	85	86	87	88				
89	90	91	92	93	94	95	96	97	98	99				
100	101	102	103	104	105	106	107	108	109	110				
111	112	113	114	115	116	117	118	119	120	121				
122	123	124	125	126	127	128	129	130	131	132				
133	134	135	136	137	138	139	140	141	142	143				
144	145	146	147	148	149	150	151	152	153	154				
155	156	157	158	159	160	161	162	163	164	165				
166	167	168	169	170	171	172	173	174	175	176				
177	178	179	180	181	182	183	184	185	186	187				
188	189	190	191	192	193	194	195	196	197	198				
199	200	201	202	203	204	205	206	207	208	209				
210	211	212	213	214	215	216	217	218	219	220				
221	222	223	224	225	226	227	228	229	230	231				
232	233	234	235	236	237	238	239	240	241	242				
243	244	245	246	247	248	249	250	251	252	253				
254	255	256	257	258	259	260	261	262	263	264				
265	266	267	268	269	270	271	272	273	274	275				
276	277	278	279	280	281	282	283	284	285	286				
287	288	289	290	291	292	293	294	295	296	297				
298	299	300	301	302	303	304	305	306	307	308				
309	310	311	312	313	314	315	316	317	318	319				
320	321	322	323	324	325	326	327	328	329	330				
331	332	333	334	335	336	337	338	339	340	341				
342	343	344	345	346	347	348	349	350	351	352				
353	354	355	356	357	358	359	360	361	362	363				
364	365	366	367	368	369	370	371	372	373	374				
375	376	377	378	379	380	381	382	383	384	385				
386	387	388	389	390	391	392	393	394	395	396				
397	398	399	400	401	402	403	404	405	406	407				
408	409	410	411	412	413	414	415	416	417	418				
419	420	421	422	423	424	425	426	427	428	429				
430	431	432	433	434	435	436	437	438	439	440				
441	442	443	444	445	446	447	448	449	450	451				
452	453	454	455	456	457	458	459	460	461	462				
463	464	465	466	467	468	469	470	471	472	473				
474	475	476	477	478	479	480	481	482	483	484				
485	486	487	488	489	490	491	492	493	494	495				
496	497	498	499	500	501	502	503	504	505	506				
507	508	509	510	511	512	513	514	515	516	517				
518	519	520	521	522	523	524	525	526	527	528				
529	530	531	532	533	534	535	536	537	538	539				
540	541	542	543	544	545	546	547	548	549	550				
551	552	553	554	555	556	557	558	559	560	561				
562	563	564	565	566	567	568	569	570	571	572				
573	574	575	576	577	578	579	580	581	582	583				
584	585	586	587	588	589	590	591	592	593	594				
595	596	597	598	599	600	601	602	603	604	605				
606	607	608	609	610	611	612	613	614	615	616				
617	618	619	620	621	622	623	624	625	626	627				
628	629	630	631	632	633	634	635	636	637	638				
639	640	641	642	643	644	645	646	647	648	649				
650	651	652	653	654	655	656	657	658	659	660				
661	662	663	664	665	666	667	668	669	670	671				
672	673	674	675	676	677	678	679	680	681	682				
683	684	685	686	687	688	689	690	691	692	693				
694	695	696	697	698	699	700	701	702	703	704				
705	706	707	708	709	710	711	712	713	714	715				
716	717	718	719	720	721	722	723	724	725	726				
727	728	729	730	731	732	733	734	735	736	737				
738	739	740	741	742	743	744	745	746	747	748				
749	750	751	752	753	754	755	756	757	758	759				
760	761	762	763	764	765	766	767	768	769	770				
771	772	773	774	775	776	777	778	779	780	781				
782	783	784	785	786	787	788	789	790	791	792				
793	794	795	796	797	798	799	800	801	802	803				
804	805	806	807	808	809	810	811	812	813	814				
815	816	817	818	819	820	821	822	823	824	825				
826	827	828	829	830	831	832	833	834	835	836				
837	838	839	840	841	842	843	844	845	846	847				
848	849	850	851	852	853	854	855	856	857	858				
859	860	861	862	863	864	865	866	867	868	869				
870	871	872	873	874	875	876	877	878	879	880				
881	882	883	884	885	886	887	888	889	890	891				
892	893	894	895	896	897	898	899	900	901	902				
903	904	905	906	907	908	909	910	911	912	913				
914	915	916	917	918	919	920	921	922	923	924				
925	926	927	928	929	930	931	932	933	934	935				
936	937	938	939	940	941	942	943	944	945	946				
947	948	949	950	951	952	953	954	955	956	957				
958	959	960	961	962	963	964	965	966	967	968				
969	970	971	972	973	974	975	976	977	978	979				
980	981	982	983	984	985	986	987	988	989	990				
991	992	993	994	995	996	997	998	999	1000	1001				
1002	1003	1004	1005	1006	1007	1008	1009	1010	1011	1012				
1013	1014	1015	1016	1017	1018	1019	1020	1021	1022	1023				
1024	1025	1026	1027	1028	1029	1030	1031	1032	1033	1034				
1035	1036	1037	1038	1039	1040	1041	1042	1043	1044	1045				
1046	1047	1048	1049	1050	1051	1052	1053	1054	1055	1056				
1057	1058	1059	1060	1061	1062	1063	1064	1065	1066	1067				
1068	1069	1070	1071	1072	1073	1074	1075	1076	1077	1078				
1079	1080	1081	1082	1083	1084	1085	1086	1087	1088	1089				
1090	1091	1092	1093	1094	1095	1096	1097	1098	1099	1100				
1101	1102	1103	1104	1105	1106	1107	1108	1109	1110	1111				
1112	1113	1114	1115	1116	1117	1118	1119	1120	1121	1122				
1123	1124	1125	1126	1127	1128	1129	1130	1131	1132	1133				
1134	1135	1136	1137	1138	1139	1140	1141	1142	1143	1144				
1145	1146	1147	1148	1149	1150	1151	1152	1153	1154	1155				
1156	1157	1158	1159	1160	1161	1162	1163	1164	1165	1166				
1167	1168	1169	1170	1171	1172	1173	1174	1175	1176	1177				
1178	1179	1180	1181	1182	1183	1184	1185	1186	1187	1188				
1189	1190	1191	1192	1193	1194	1195	1196	1197	1198	1199				
1200	1201	1202	1203	1204	1205	1206	1207	1208	1209	1210				
1211	1212	1213	1214	1215	1216	1217	1218	1219	1220	1221				
1222	1223	1224	1225	1226	1227	1228	1229	1230	1231	1232				
1233	1234	1235	1236	1237	1238	1239	1240	1241	1242	1243				
1244	1245	1246	1247	1248	1249	1250	1251	1252	1253	1254				
1255	1256	1257	1258	1259	1260	1261	1262	1263	1264	1265				
1266	1267	1268	1269	1270	1271	1272	1273	1274	1275	1276				
1277	1278	1279	1280	1281	1282	1283	1284	1285	1286	1287				
1288	1289	1290	1291	1292	1293	1294	1295	1296	1297	1298				
1299	1300	1301	1302	1303	1304	1305	1306	1307	1308	1309				
1310	1311	1312	1313	1314	1315	1316	1317	1318	1319	1320				
1321	1322	1323	1324	1325	1326	1327	1328	1329	1330	1331				
1332	1333	1334	1335	1336	1337	1338	1339	1340	1341	1342				
1343	1344	1345	1346	1347	1348	1349	1350	1351	1352	1353				
1354	1355	1356	1357	1358	1359	1360	1361	1362	1363	1364				
1365	1366	1367	1368	1369	1370	1371	1372	1373	1374	1375				
1376	1377	1378	1379	1380	1381	1382	1383	1384	1385	1386				
1387	1388	1389	1390	1391	1392	1393	1394	1395	1396	1397				
1398														

Bir ICF kutuğunun hangi aygıtlarla iletişime geçileceği
ADDICFDEVE (Add ICF device entry) komutu ile belirlenir.
Eğer ICF kutuğu kullanılarak birden fazla aygıtla iletişim
yapılacaksa bu durumda aygıt sayısı kadar ADDICFDEVE komutu



Bir ICF kütüğünün hangi aygıtlarla iletişim yapabileceği ADDICFDEVE (Add ICF device entry) komutu ile belirtilir. Eğer ICF kütüğü kullanılarak birden fazla aygıtla iletişim yapılacaksa bu durumda aygıt sayısı kadar ADDICFDEE komutu kullanmak gerekir.

Bu komutun parametreleriyle birlikte görünümü şöyledir.



Diğer ICF kütüğü komutları da şunlardır.

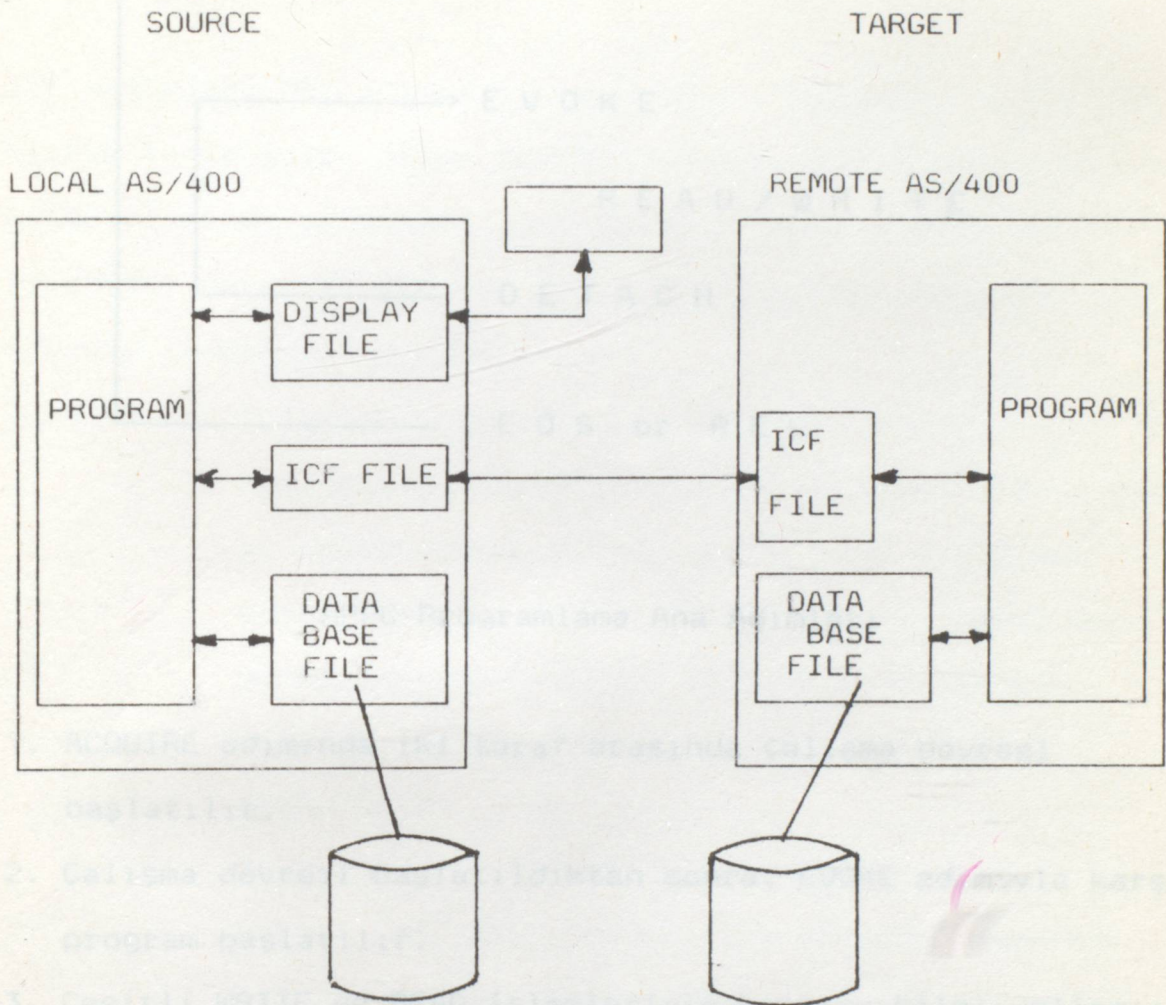
&

. Create	CRTICFF
. Change	CHGICFF
. Override	OVRICFF
. Delete Override	DLTOVR
. Delete File	DLTF

&

. Add	ADDICFDEVE
. Change	CHGICFDEVE
. Remove	RMVICFDEVE
. Set	OVRICFDEVE
. Delete Override	DLTOVRDEVE

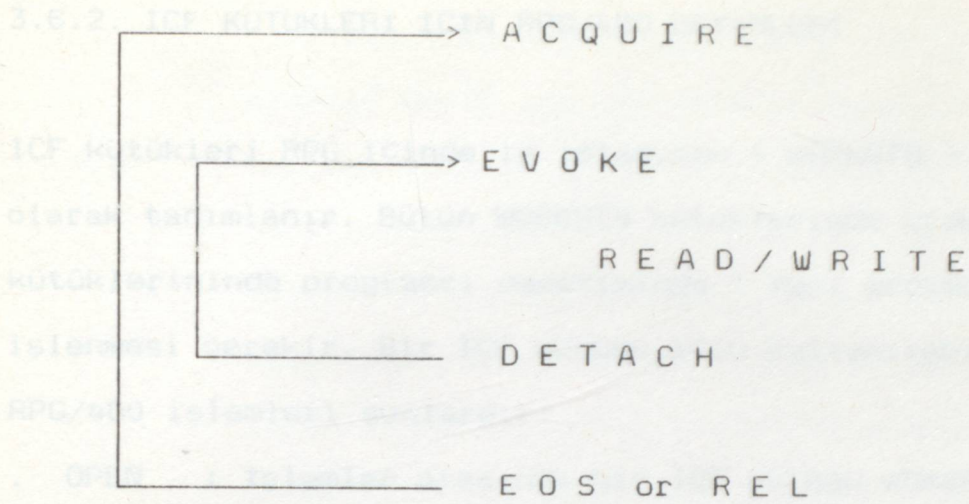
Etkileşimli bir APPC programı yazabilmek için ICF kütüğünün yanısıra, kullanılacak görüntü formatlarını içeren bir görüntü kütüğünde hazırlamak gerekir. Etkileşimli bir programın çalışma ilkesi aşağıdaki şekilde gösterilmiştir.



KISIM 3.6 APPC PROGRAMLAMA

3.6.1. GENEL YAPISI

Bir APPC programı hazırlayabilmek için 5 ana adımdan oluşan bir APPC program yapısı gerçekleştirilmelidir. Bu 5 ana adım aşağıdaki şekilde gösterilmiştir.



APPC Programlama Ana Adımları

1. ACQUIRE adımımda iki taraf arasında çalışma devresi başlatılır.
2. Çalışma devresi başlatıldıktan sonra, EVOKE adımıyla karşı program başlatılır.
3. Çeşitli WRITE ve READ işlemleriyle karşıya bilgi yollanır veya gelen bilgi alınır.
4. Bilgi alışverişi tamamlanınca karşı programa DETACH işlemi yollanarak programın bitirileceği belirtilir.
5. Son adımda ise çalışma devresi serbest bırakılır.

3.6.2. ICF KÜTÜKLERİ İÇİN RPG/400 DEYİMLERİ

ICF kütükleri RPG içinde iş istasyonu (WORKSTN) kütükleri olarak tanımlanır. Bütün WORKSTN kütüklerinde olduğu gibi ICF kütüklerinin de programcı denetiminde (full procedural) işlenmesi gerekir. Bir ICF kütüğü için kullanılabilecek RPG/400 işlemleri şunlardır:

- . OPEN : İşlemler arasında bir ICF kütüğü açmak için kullanılır. Normalde bir ICF kütüğünü açmak için bu işleme gerek yoktur. Ve program başlarken bütün kütükler kendiliğinden açılır. Fakat bu normal sürecin dışında programcının, kütüğü istediği zaman açmak üzere OPEN kullanması yeterlidir.
- . CLOSE : Açık olan bir ICF kütüğünü kapatmak için kullanılır. Normalde bir ICF kütüğünü kapatmak için bu işleme gerek yoktur. Ve program sona erdiğinde bütün kütükler kendiliğinden kapanır. Fakat bu normal sürecin dışında programcının, kütüğü istediği zaman kapatmak üzere CLOSE kullanması gerekebilir.
- . WRITE : Program aygıtına bilgi göndermeye yarar. Bir başka deyişle, karşı tarafa gönderilecek bilginin program aygıtı yoluyla hat üzerine verilmesini sağlar. Bu amaçla ICF kütüğü içinde hazırlanmış

olan kayıt formatlarından bilgi göndermeye uygun olanlar WRITE ile birlikte kullanılırlar.

- . READ : Program aygıtından bilgi okunması ve programa getirilmesi amacıyla kullanılır. Bir tarafın WRITE ile yolladığı bilgiyi almak için karşı taraftaki programda READ kodlanmalıdır. READ işlemide tıpkı WRITE gibi, ICF kütüğündeki kayıt formatlarından bilgi almaya uygun olanlarla birlikte kullanılır.
- . EXFMT : Bu işlem kodu WRITE ve READ' ün ardarda kullanımına eşdeğerdir anlamdadır. EXFMT kullanıldığında, önce bir WRITE ve hemen ardından da bir READ işlemi kodlanmış gibi sonuç alınır. Görüntü kütüklerindeki formatların kullanımı için çok uygun olan bu işlem kodu, ICF kütüklerinde pek kullanılmaz ve WRITE ile READ'in ayrı ayrı kullanılması tercih edilir. Bunun nedeni, ayrı kullanıldığında , her write işleminden sonra dönüş kodu (return code) adı verilen kodların program içinde denetlenebilmesidir. Halbuki EXFMT ile bu durum ortadan kalkar.
- . ACQ : Bir aygıtı program arayıcılığıyla ele geçirmek üzere kullanılır. Bir ICF kütüğünün, program başlayınca ilk önce hangi aygıtı ele geçireceği

Program 47 önceden belirtilmişse, program içinde ACQ işlemi
Program 48 ile belirtilir. Bir ICF kütüğü birkaç aygıt
Program 49 kullanarak bilgi iletişimi yapacaksa, daha sonra
Derlenen 50 kullanılacak aygıtlar birbirini izleyen bir REL
Deneme 51 ve bir ACQ işlemi ile ele geçirilirler.

- . REL : Daha önce ele geçirilmiş bir program aygıtının serbest bırakılması amacıyla kullanılır. Bir ICF kütüğü ele geçirdiği bir aygıtı serbest bıraktıktan sonra bir başka aygıtı ele geçirebilir.

Yukarıdaki işlemlerden başka RPG/400 'ün bilinen diğer işlem kodları d, eğer biliniyorsa , APPC programlarında kullanılabilir.

Yerel ve uzak programlardaki ICF kütüklerinde, birbirlerine karşılıklı gelen format ve alan isimleri aynı veya farklı olabilir.

3.6.3. ÖZET

Bir APPC programı hazırlayabilmek için gerekli işlemler şöyle özetlenebilir.

- . İletişim konfigürasyonu yaratılır ve çevrim içi duruma getirilir.
- . ICF kütüğü için DDS hazırlanır.
- . ICF kütüğü CRTICFF komutu ile yaratılır.

- . Program aygıtı ADDICFDEVE ile ICF kütüğüne eklenir.
- . Program hazırlanır.
- . Program derlenir.
- . Derlenen program gerçek bilgilerle çalıştırılmadan önce deneme bilgileriyle test edilir.



ICE FILE HULYA/KAYNAK
ER APPC1ICE

ER*	...	1	...	2	...	3	...	4	...	5	...	6	...	7
	A			R FMT1						EVOKE(HULYA/APPC1T)				
	A									SECURITY(2 'SERAP' 3 'HULYA				
	A			R FMT2						DETACH				

* * * * E N D O F S O U R C E * * * *

CE FILE HULYA/KAYNAK2
ER APPC1T

BR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
PGM
DSPLIBL OUTPUT(*PRINT)
ENDPGM

* * * * E N D O F S O U R C E * * * *

CE FILE HULYA/KAYNAK
ER APPC2S

```
BR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
0      FAPPC2ICFO      E      WORKSTN
0      FMALZEMESIF     E      K      DISK
0      C
0      C              *IN99    WRITEFMT1
0      C              DOWEQ 'O'
0      C              READ MALZEMES          99
0      C              *IN99    IFEQ 'O'
0      C              WRITEFMT2
0      C              END
0      C              END
0      C              WRITEFMT3
0      C              SETON          LR
```

* * * * * E N D O F S O U R C E * * * * *

RCE FILE HULYA/KAYNAK
BER MALZEMES

NBR*	1	2	3	4	5	6	7
00	A	R MALFMT					
00	A	MALNO	6S 0				
00	A	DEPO1	6S 0				
00	A	DEPO2	6S 0				
00	A	DEPO3	6S 0				
00	A	K MALNO	6S 0				
00	A	DEPO3	6S 0				
00	A	R FMT3					

* * * * END OF SOURCE * * * *

* * * * END OF SOURCE * * * *

E FILE HULYA/KAYNAK
R APPC2ICE

R*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
A APPC2ICE R FMT1 EVOKE(HULYA/APPC2T)
A MALZETA SECURITY(2 'SERAP' 3 'HULYA'
A R FMT2
A MALNO 6S 0
A DEPO1 6S 0
A DEPO2 6S 0
A DEPO3 6S 0
A R FMT3 DETACH

* * * * E N D O F S O U R C E * * * *

SOURCE FILE HULYA/KAYNAK2
NUMBER APPC2T

NUMBER*	...	1	...	2	...	3	...	4	...	5	...	6	...	7
100		FAPPC2ICTIF		E				WORKSTN						
200		FMALZETARO		E				DISK						
300		C		*IN11		DOWEQ'O'								
400		C				READ TFMT2						77		
500		C				WRITEMALEMTT								
600		C				END								
700		C				SETON						LR		

*** END OF SOURCE ***

RCE FILE HULYA/KAYNAK2
BER MALZETAR

NBR*	...	1	...	2	...	3	...	4	...	5	...	6	...	7
00	A			R MALEMTT										
00	A			MALNO			6S	0						
00	A			DEPO1			6S	0						
00	A			DEPO2			6S	0						
00	A			DEPO3			6S	0						
00	A			DEPO3			6S	0						

* * * * * E N D O F S O U R C E * * * * *

SOURCE FILE HULYA/KAYNAK2
NUMBER APPC2ICT

NUMBER*	...	1	...	2	...	3	...	4	...	5	...	6	...	7
100	A													
200	A			R	TFMT2									
300	A				MALNO			6S 0						
400	A				DEPO1			6S 0						
500	A				DEPO2			6S 0						
600	A				DEPO3			6S 0						

RCVDETACH(11)

* * * * * END OF SOURCE * * * * *

SOURCE FILE HULYA/KAYNAK
 NUMBER APPC4S

NUMBER*	1	2	3	4	5	6	7
100	FMALZEMESIF	E	K	DISK			
200	FAPPC4DF CF	E		WORKSTN			
300	FAPPC4ICFCF	E		WORKSTN			
400	F				KNUM	1	
500	F				KID	DEVICE	
600	F				KINFDS	ERRDS	
700	IERRDS	DS					
800	I			401 402	MAJOR		
900	I			403 404	MINOR		
100	C	'IST400'	ACQ APPC4ICF				
200	C		MOVEL'IST400	'DEVICE			
300	C	BAS	TAG				
400	C		EXFMTGIRMALNO				
500	C	03	GOTO SON				
600	C	MALNO	CHAINMALZEMES		15		
700	C	*IN15	IFEQ '1'				
800	C		EXFMTGITUZAK				
900	C	N90	WRITEFOREVOK				
100	C		SETON		90		
200	C		WRITEFORSEND				
300	C		READ RCVDATA		20		
400	C	R	IFEQ '*'				
500	C		EXFMTYOKUZAK				
600	C		GOTO BAS				
700	C		END				
800	C		EXFMTUDEPOLAR				
900	C		GOTO BAS				
100	C		ELSE				
200	C		EXFMTYDEPOLAR				
300	C	*IN03	IFEQ '1'				
400	C		GOTO SON				
500	C		ELSE				
600	C		GOTO BAS				
700	C		END				
800	C		END				
900	C	SON	TAG				
100	C	90	WRITEFORDET				
200	C	'IST400'	REL APPC4ICF				
300	C		SETON		LR		

* * * * * E N D O F S O U R C E * * * * *

RCE FILE HULYA/KAYNAK
BER MALZEMES

NBR*	...	1	...	2	...	3	...	4	...	5	...	6	...	7
00	A			R MALFMT										
00	A			MALNO		6S 0		5	10	MALZEME NO.				
00	A			DEPO1		6S 0		5	22					
00	A			DEPO2		6S 0		10	10	191 BITIRMEK İÇİN F3 E BAS				
00	A			DEPO3		6S 0		15	10					
00	A			K MALNO				15	2	MAJOR				
	A			MINOR		2		16	10					
	A							16	2	MINOR				
	A			R YDEPOLAR										
	A							5	10	MALZEME NO.				
	A			MALNO		6 0		5	22	EDTCDE(2)				
	A							10	10	DEPO1				
	A			DEPO1		6 0		10	22	EDTCDE(2)				
	A							11	10	DEPO2				
	A			DEPO2		6 0		11	22	EDTCDE(2)				
	A							12	10	DEPO3				
	A			DEPO3		6 0		12	22	EDTCDE(2)				
	A			MAJOR		2		15	10					
	A							15	2	MAJOR				
	A			MINOR		2		16	10					
	A							16	2	MINOR				
	A							20	10	191 BITIRMEK İÇİN F3 E BAS				
	A			R UDEPOLAR										
	A							5	10	MALZEME NO.				
	A			MALNO		6 0		5	22					
	A							10	10	DEPO4				
	A			DEPO4		6 0		10	22	EDTCDE(2)				
	A							11	10	DEPO5				
	A			DEPO5		6 0		11	22	EDTCDE(2)				
	A			MAJOR		2		15	10					
	A							15	2	MAJOR				
	A			MINOR		2		16	10					
	A							16	2	MINOR				
	A							20	10	191 BITIRMEK İÇİN F3 E BAS				
	A			R GITUZAK										
	A													
	A							10	2	ISTEDİĞİNİZ MALZEME YEREL				
	A									YOK. UZAK SİSTEM ARANACAK				
	A							12	12	LOTTEM ENTER A BASARAK BEN				
	A									DEPATH(H1)				
	A			R YOKUZAK										
	A							15	2	İSTEDİĞİNİZ MİMARIA UZAK SİS				
	A									YOK. YENİ MİMARIA GİBİNİZ.				

*** END OF SOURCE ***

RCE FILE HULYA/KAYNAK
BER APPC4DF

```
NBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
00      A      R GIRMALNO      CAO3(03)
00      A      5 10'MALZEME NO:'
00      A      MALNO      6 0I 5 22
00      A      10 10'İŞİ BİTİRMEK İÇİN F3 E BAS
00      A      MAJOR      2 15 10
00      A      15 2'MAJOR'
00      A      MINOR      2 16 10
00      A      16 2'MINOR'
00      A      R YDEPOLAR      CAO3(03)
00      A      5 10'MALZEME NO:'
00      A      MALNO      6 0 5 22EDTCDE(Z)
00      A      10 10'DEPO1:'
00      A      DEPO1      6 0 10 22EDTCDE(Z)
00      A      11 10'DEPO2:'
00      A      DEPO2      6 0 11 22EDTCDE(Z)
00      A      12 10'DEPO3:'
00      A      DEPO3      6 0 12 22EDTCDE(Z)
00      A      MAJOR      2 15 10
00      A      15 2'MAJOR:'
00      A      MINOR      2 16 10
00      A      16 2'MINOR:'
00      A      20 10'İŞİ BİTİRMEK İÇİN F3 E BAS
00      A      R UDEPOLAR      CAO3(03)
00      A      5 10'MALZEME NO:'
00      A      MALNO      6 0 5 22
00      A      10 10'DEPO4:'
00      A      DEPO4      6 0 10 22EDTCDE(Z)
00      A      11 10'DEPO5:'
00      A      DEPO5      6 0 11 22EDTCDE(Z)
00      A      MAJOR      2 15 10
00      A      15 2'MAJOR:'
00      A      MINOR      2 16 10
00      A      16 2'MINOR:'
00      A      20 10'İŞİ BİTİRMEK İÇİN F3 E BAS
00      A      R GITUZAK      LOCK
00      A      10 2'İSTEDİĞİNİZ MALZEME YEREL
00      A      YOK. UZAK SİSTEM ARANACAK
00      A      12 12'LÜTFEN ENTER A BASARAK BEK
00      A      DSPATR(HI)
00      A      R YOKUZAK
00      A      15 2'GİRDİĞİNİZ NUMARA UZAK SİS
00      A      YOK. YENİ NUMARA GİRİNİZ.'
```

* * * * E N D O F S O U R C E * * * *

RCE FILE HULYA/KAYNAK
BER APPC4ICE

NBR*	...	1	...	2	...	3	...	4	...	5	...	6	...	7
00	A			R FOREVOK						EVOKE(HULYA/APPC4T)				
00	A									SECURITY(2 'SERAP' 3 'HULYA				
00	A			R FORSEND						INVITE				
00	A			MALNO			6S 0							
00	A			R RCVDATA										
00	A			R			1							
00	A			DEPO4			6S 0							
00	A			DEPO5			6S 0							
00	A			R FORDET						DETACH				

* * * * * E N D O F S O U R C E * * * * *

* * * * * E N D O F S O U R C E * * * * *

RCE FILE HULYA/KAYNAK2
 BER APPC4T

NBR*	...	1	...	2	...	3	...	4	...	5	...	6	...	7
00		FAPPC4ICTCF		E				WORKSTN						
00		FMALKUTT IF		E		K		DISK						
00		C		*IN11		DOWEQ '0'								
00		C				READ RECEIVE						90		
00		C		*IN11		IFEQ '0'								
00		C		MALNO		CHAINMALKUTT						91		
00		C		*IN91		IFEQ '1'								
00		C				MOVE '*'		R						
00		C				MOVE *ZEROS		DEPO4						
00		C				MOVE *ZEROS		DEPO5						
00		C*				CALL 'APPC4TBE'								
00		C				WRITETARFOR								
00		C				ELSE								
00		C				MOVE 'A'		R						
00		C*				CALL 'APPC4TBE'								
00		C				WRITETARFOR								
00		C				END								
00		C				END								
00		C				END								
00		C				SETON						LR		

* * * * * E N D O F S O U R C E * * * * *

RCE FILE HULYA/KAYNAK2
BER MALKUTT

NBR*	...	1	...	2	...	3	...	4	...	5	...	6	...	7
00	A			R		MALKUTTF								
00	A					MALNO		6S		0				
00	A					DEPO4		6S		0				
00	A					DEPO5		6S		0				
00	A			K		MALNO								

* * * * * E N D O F S O U R C E * * * * *

* * * * * E N D O F S O U R C E * * * * *

RCE FILE HULYA/KAYNAK2
BER APPC4ICT

NBR*	...	1	...	2	...	3	...	4	...	5	...	6	...	7
00	A													
00	A			R RECEIVE										
00	A			MALNO			6S	0						
00	A			R TARFOR										
00	A			R			1							
00	A			DEPO4			6S	0						
00	A			DEPO5			6S	0						

RCVDETACH(11)

* * * * * E N D O F S O U R C E * * * * *

KAYNAKLAR

IBM, Data Communications Concepts and Planning Module 1:
General Concepts, 1981

IBM, Data Communications Concepts and Planning Module 2:
Communications planning, 1981

IBM, Data Communications Concepts and Planning Module 3:
Applications, 1981

IBM, Data Communications Concepts, 1983

IBM, AS/400 Communications: User Guide, 1988

IBM, AS/400 Communications: Programmers Guide, 1988

ÖZGEÇMİŞ

1966 yılında istanbul'da doğdu. ilk, orta, lise ve yüksek öğrenimini sırasıyla Beşiktaş Sinanpaşa ilkokulu (1977) ,istanbul Etiler lisesi (1983) ve Yıldız Üniversitesi MÜhendislik Fakültesi matematik mühendisliği (1987) bölümünde tamamlayarak iş hayatına atılmıştır. Sırasıyla Aarsal Cam Sanayii ve Tic. A.Ş IBM SİSTEM/34 de, Goodyear lastikleri A.Ş. IBM SİSTEM/ 34 VE SİSTEM/ 38 de uygulama programcısı olarak çalıştı. Son olarak Abaküs Bilgi İşlem ve Tic. A.Ş. de analist olarak çalıştı. Şu anda matematik mühendisliği bölümünde yüksek lisans programına devam etmektedir. Evlidir.

