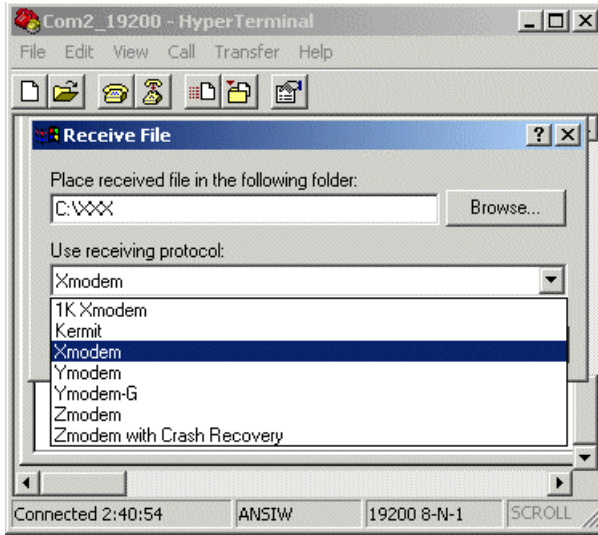


Dosya Aktarımları - XMODEM (Ver.1.0a)

Sahadaki mikro denetleyicilerle bilgisayarlar arasındaki iletişimde yararlanabilecek bir çok teknik vardır. Sürekli iletişim için kullanılan yöntem ModBus gibi endüstri standardı bir protokol kullanılmaktadır. Az sıklıkla kullanılan, büyük hacimli veri iletişimlerinde ise "Dosya Aktarımı" uygulamalarından yararlanılabilir. Önerimiz "standart" bir dosya aktarım tekniği seçmeniz olacaktır.



Dosya aktarım standartlarından Ethernet tabanlı olanlar için ilk akla gelenler Internet'in meşhur ettiği FTP (File Transfer Protocol) ve kuzeni TFTP (Trivial File Transfer Protocol) olacaktır. Seri iletişim tabanlı olanlar içinse XModem, YModem, ZModem ve Kermit sayılabilir.

Bütün bunların içinde en basiti olan XModem'le işe başlamak akıllıca olacaktır. Bütün Windows™ işletim sistemlerinde yer alan HyperTerminal™, XModem'in çeşitli uyarlamalarını desteklemektedir.

1977'de Ward Christensen tarafından belirlenen bu protokol zamanla bütün modemlerin desteklediği bir standarda dönüşmüş durumdadır. Güncel bilgisayar teknolojisi açısından protokol eski ve yavaş bulunmaktadır. Ancak gömülü sistemler ve mikro denetleyiciler için yeterince iyi durumdadır.

XModem

Bütün iletişim protokolleri gibi XModem de paket tabanlıdır. Paketin içindeki veri bloğu 128 bayttan oluşur. Her paket için onay gerektirir:

<SOH> 1 bayt	<PaketNo> 1 bayt	<255 - PaketNo> 1 bayt	<Veri> 128 bayt	<Sağlama Top.> 1 bayt
-----------------	---------------------	---------------------------	--------------------	--------------------------

Bu arada bazı ASCII kısaltmaları hatırlamakta fayda olacaktır:

ASCII Kısaltma	Bayt (Desimal)	Kontrol Kodu
<NULL>	0	CTRL-@
<SOH>	1	CTRL-A
<STX>	2	CTRL-B
<ETX>	3	CTRL-C
<EOT>	4	CTRL-D
<ACK>	6	CTRL-F
<NAK>	21	CTRL-U
<CAN>	24	CTRL-X
<SUB>	26	CTRL-Z

Paket numarası 1'den başlayarak birer birer artar. 256 ile modulosu alındıktan sonra kullanılır. Üçüncü bayt ise paket numarası 255 ten çıkartıldıktan sonra kalan değerdir. Sağlama toplamı ise verileri ASCII değerlerinin toplandıktan sonra yine 256 ile modulosu alınarak hesaplanır.

Tipik bir akış aşağıda verilmektedir.

Alıcı	<NAK>
Gönderen	<SOH><1><254><1.Paket><Sağ. Toplamı>
Alıcı	<ACK>
Gönderen	<SOH><2><253><2. Paket><Sağ. Toplamı>
Alıcı	<ACK>
...
Gönderen	<SOH><X><255 - X><Sonuncu Paket><Sağ. Toplamı>
Alıcı	<ACK>
Gönderen	<EOT>
Alıcı	<ACK>

Aşağıdaki kurallara da dikkat edilmesi halinde başarılı bir iletişim sağlanır:

- Alıcı; hatalı paket numarası, farklı sağlama toplamı gibi iletişim hatalarında <ACK> yerine <NAK> göndererek paketin tekrar gönderilmesini ister. Bu tür denemeler on kez tekrarlandıktan sonra hatanın sürmesi durumunda iletişim kesilir.
- Sonuncu paket için kalan verinin uzunluğu 128 baytan az kalırsa eksik karakterlerin hepsi <SUB> ile doldurulur.
- Her iki taraf istediği zaman üç adet <CAN> göndererek iletişimi kesebilir.

Ancak uygulamalarda sağlama toplamıyla (%99.29 güvenilirlik) yapılan kontrol yetersiz gelmeye başlayınca toplam yerine 16 bitlik, CRC (Cyclic Redundancy Check = dairesel artıklık denetimi) değerinin kullanımına başlandı.

XModem-CRC

Artık XModem denildiğinde XModem-CRC anlaşılmaktadır. HyperTerminal™'deki XModem seçeneği de bu protokolü kullanmaktadır.

<SOH> 1 bayt	<PaketNo> 1 bayt	<255 - PaketNo> 1 bayt	<Veri> 128 bayt	<CRC.> 2 bayt
-----------------	---------------------	---------------------------	--------------------	------------------

XModem'den olan farklılara gelince:

- Toplam paket uzunluğu bir bayt artarak 133 olmuştur.
- Alıcı, iletişimi başlatırken <NAK> değil "C" gönderir.

Uygulamada güçlük çıkaran kısım CRC'nin hesabıdır. Bütün CRC hesaplarında temelde

$$x^{16} + x^{12} + x^5 + 1$$

tarzında bir polinom kullanılır. Yukarıdaki polinomla hesaplanan CRC; CRC-CCITT olarak anılmakta, bilgisayarlarımızın disklerinde, disket sürücülerinde ve XModem'de kullanılmaktadır. <Veri> üzerindeki her baytın bu polinomda üretilen değere bölünmesinden sonra kalan artıklar yardımıyla CRC değeri oluşturulmaktadır.

Bu işlemin kodlama aşamasında 2 tabanında, XOR'larla nasıl yapıldığını Terry Ritter'ın yazısı açıklamaktadır. Bizim için önemli olan bu işlemin oldukça basit bir algoritmayla yapılabilmesidir.

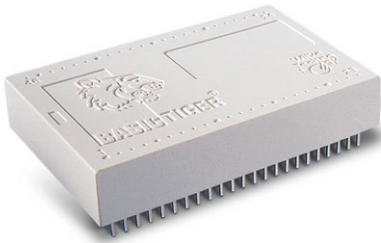
Algoritma, aşağıdaki Tiger Basic™ ile yazılmış uygulamanın CRC_BIT yordamı içinde yer almaktadır.

```

SUB CRC_BIT(String Buf$; WORD BufStart, BufLen; VAR WORD wCRC)
`
' Buf$      : CRC hesabı yapılacak paketin yer aldığı değişken
' BufStart  : Kaçınıcı karakterden başlanılacağı
' Buflen   : Kaç karakter için yapılacağı
' wCRC     : Hesaplanan CRC Değeri
`
LONG ITmp, ICRC
BYTE J
WORD wCnt
    ICRC = 0
    For wCnt = 0 To BufLen - 1
        ITmp = Asc(MID$(Buf$, BufStart + wCnt, 1)) * 0100h
        ICRC = ICRC BITXOR ITmp
        For J = 0 To 7
            If (ICRC BITAND 08000H) <> 0 Then
                ICRC = ((ICRC * 2) BITXOR 01021H) BITAND 0FFFFh
            Else
                ICRC = (ICRC * 2) BITAND 0FFFFh
            EndIf
        Next
    Next
    wCRC = ICRC BITAND 0FFFFh
END

SUB CRC_TAB(String Buf$; WORD BufStart, BufLen; VAR WORD wCRC)
    WORD wCnt, wCRCTmp
    BYTE B
    wCRCTmp = 0
    FOR wCnt = 0 TO BufLen - 1
        B = ASC(MID$(Buf$, BufStart + wCnt, 1))
        wCRCTmp = CRCTab((wCRCTmp SHR 8) BITXOR B) BITXOR (wCRCTmp SHL 8)
    NEXT
    wCRC = wCRCTmp
END

```



CRC_BIT görüldüğü gibi her bit üzerinde işlem yapar ve dolayısıyla oldukça fazla işlemci zamanı gerektirir. Ancak denetleyicinizin bellek miktarı yeterliyse önceden hazırlayacağınız tablolarla bu işlemi hızlandırmak mümkün olur. 256 word (512 bayt) kullanan bir tablo yardımıyla algoritma CRC_TAB ile son derece hızlı bir şekilde yapılabilmektedir.

XModem-1K

Bu protokolle, CRC ile güvenilirliđi artan XModem her paket sonrasında verilen onay nedeniyle biraz yavař kalınca veri uzunluđu 128 bayttan 1024'e ıkartılır.

<ETX> 1 bayt	<PaketNo> 1 bayt	<255 - PaketNo> 1 bayt	<Veri> 1024 bayt	<CRC.> 2 bayt
-----------------	---------------------	---------------------------	---------------------	------------------

XModem-CRC'den olan farklara gelince:

- Toplam paket uzunluđu 1029 olmuřtur.
- İlk Bayt <ETX> olmuřtur

XModem-1KG

Seri iletiřimde donanım tokalařması (CTS/RTS) var ise bu protokolden yararlanılabilir. Bu protokolda gnderimi bařlatmak iin gnderilen ilk bayt "C" ile deđil "G" olmaktadır. Her paket gnderisi sonrasında <ACK> beklemeden veri gndermeye devam edilmekte bylece iletiřim son derece hız yapılabilir. Ancak HyperTerminal™'de bu protokoln yer almadıđı aklınızda olsun.

Sonuç

XModem-CRC; Basic Tiger™ gibi mikro denetleyicilerle bilgisayarlar arasında dosya aktarımında kullanılabilecek oturmuř bir standarttır.

Ekteki Tiger Basic™ uygulama istenirse kolayca 1K protokoln desteklemek zere revize edilebilir. İletiřimde darbođaz CRC yordamının aldıđı sre olmaktadır. Bu nedenle 1K, yksek hızlı iletiřim (>19200) gibi uygulamalar veri aktarımına dođrusal olarak katkıda bulunmamaktadır.

Kaynaklar:

- 1) Chuck Forsberg - XModem / YModem Protocol Reference (1988)
http://www.commonsoftinc.com/Babylon_Cpp/Documentation/Res/yModem.htm
- 2) Terry Ritter - The Great CRC Mystery
Dr. Dobb's Journal of Software Tools. February. 11(2): 26-34, 76-83
<http://www.ciphersbyritter.com/ARTS/CRCMYST.HTM>
- 3) Ross N. Williams - Everything you wanted to know about CRC algorithms (1996)
http://www.repairfaq.org/filipg/LINK/F_crc_v3.html