



# DEVICE MONITORING STUDIO İLE USB DEBUG

USB Protokolünün karmaşıklığı ve tasarımının zor olması, gerek hataların giderilmesinde gerekse bu karmaşık protokolün adım adım izlenmesinde eksiksiz bir test aracının kullanılmasını zorunlu hale getiriyor. Bu yazıda HHD Software'ın geliştirmiş olduğu Device Monitoring Studio aracı ile USB haberleşmesini detayları ile inceliyoruz.

## Bu makalede

USB Monitör'ün yapılandırılması ve kullanımı  
USB Cihazın Windows tarafından listelenmesini izlemek  
USBTest programıyla gönderilen ve okunan verilerin USB Monitör ile izlenmesi

## Gerekli Araçlar

- ❶ USB Test Kartı
- ❷ USBApplication Test Programı
- ❸ Device Monitoring Studio

USB protokolünün diğer protokollere göre daha karmaşık olduğunu diğer makalelerimizde de belirtmiştik. Bu yüzden bu protokolün daha iyi anlaşılabilmesi ve hataların en alt düzeye indirilmesi için tasarımı test edecek Test Araçlarının kullanılması bir zorunluluktur. Bizde bu makalede HHD Software'in geliştirmiş olduğu Device Monitoring Studio aracını kullanarak tasarladığımız USB donanımı ve USBTest uygulaması arasındaki veri trafiğini gözlemleyeceğiz. Device Monitoring Studio programı ücretli bir program ve full sürümünü elde edebilmeniz için üretici sitesinde belirtilmiş ücreti ödememiz gerekiyor. Fakat yine de 14 günlük bir süre kısıtlaması ile bu ürünü testlerimizde kullanabilmek için <http://www.hhdsoftware.com/Products/home/usb-monitor.html> adresinden indirebilirsiniz. Bu ürünün birçok özelliği mevcut. Windows uygulaması ve USB Host Controller arasındaki data transferleri yakalayabilme, USB Aygıt ve Windows Uygulaması arasında data değişimi'ni yakalayabilme gibi yeteneklerinin yanısıra USB Kamera, USB Kart Okuyucu, USB Klavye, USB Yazıcı, USB Modem, USB Hub, USB Tarayıcı gibi daha bir çok aygıtı test edebilme özelliğinede sahip bulunuyor. Yazılım hakkında daha geniş bilgiyi üreticinin web sitesinden elde edebilirsiniz. Bu yazılımı kullanarak Test aşamasına geçmeden önce bazı hazırlıkları yapmış olmanız gerekmektedir. Önceki makalede verilmiş USB Test Devresini kurmuş, PIC Driver'ını karta yüklemiş olmanız gerekmektedir. Aynı zamanda bu aygıt ile konuşabilmek için C#'da yazılmış USBApplication Test Programını da bilgisayarınızda hazır bulundurmanız gerekmektedir. Bu araç herhangi bir kurulum gerektirmez. Sadece bulunduğu dizinden taşınmamak üzere ya da taşınacaksa ihtiyaç duyduğu USBManagement Library'i (umng.dll) ile aynı dizinde olduğundan emin olmalısınız. Eğer bu aracı PC'nizde sürekli taşımanız gerekiyorsa umng.dll'i GAC'a (Global Assembly Cache) yükleyerek bu zahmetten kurtulabilir, sadece çalıştırılabilir dosyayı istediğiniz dizine tek başına taşıyabilirsiniz. Aynı zamanda bu uygulama DotNET C#'da yazıldığından .NET Framework 2.0 veya üstünü PC'nize kurmanız gerekmektedir. USB Test Kartının devre şemasını, PIC Driver'ını ve USB Test Uygulamasını <http://www.filefactory.com/file/8fd9b4> adresinden temin edebilirsiniz.

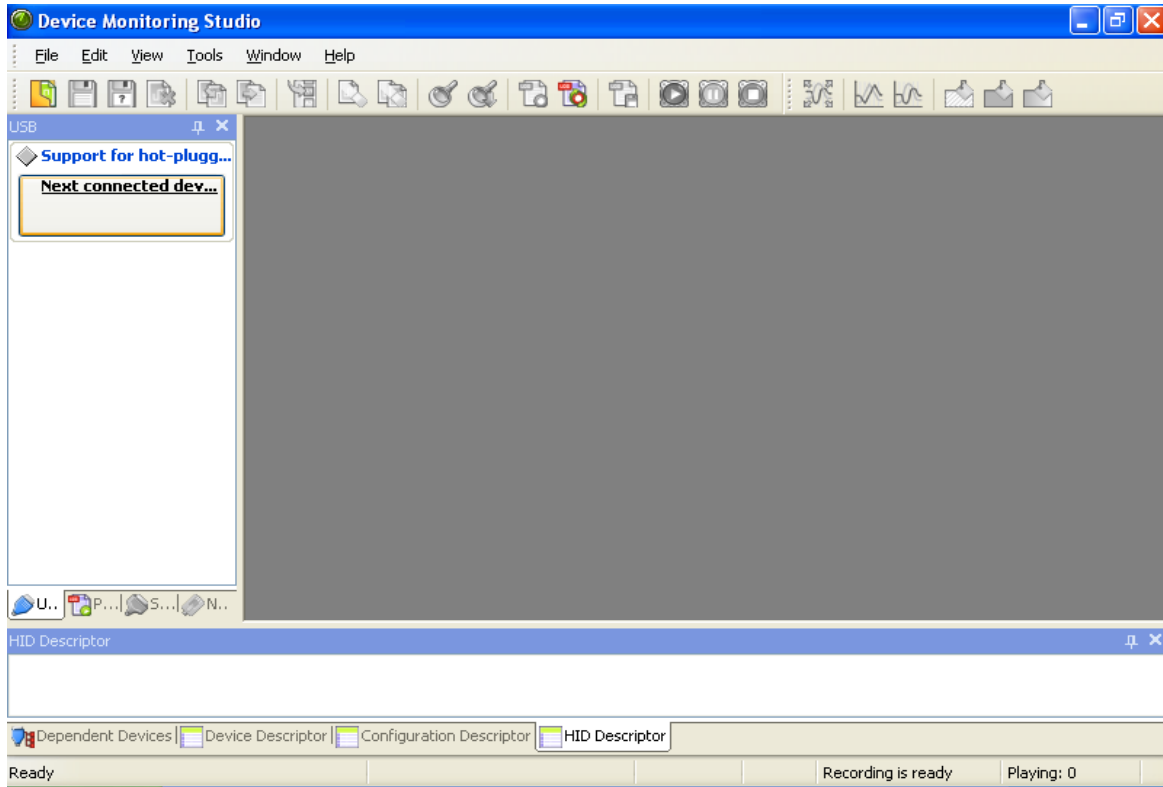
**Not :** Bu makale boyunca USB Device Monitoring programını UDM ismiyle kısaltılmış olarak adlandıracacağız.



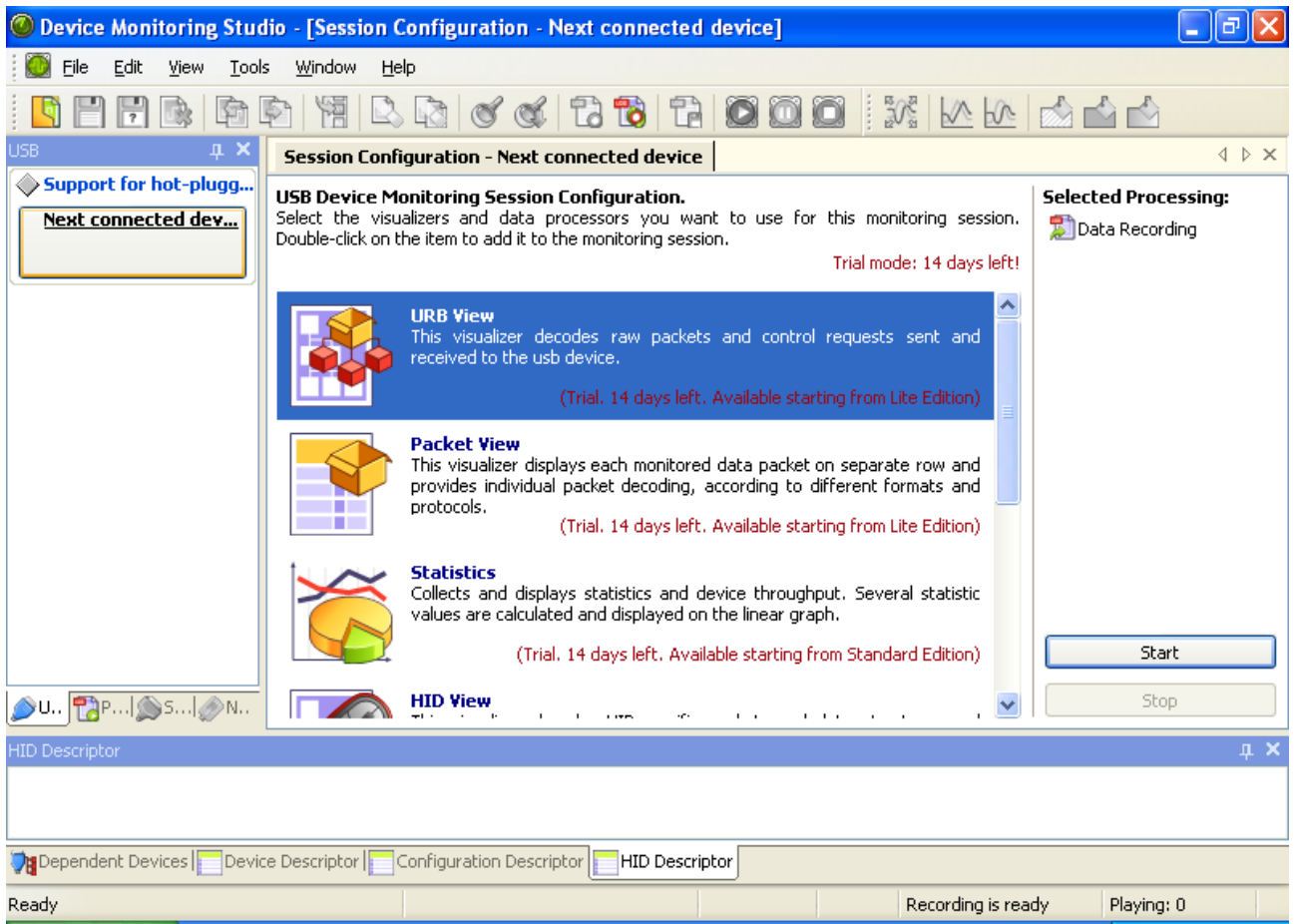
## USB Cihazın Windows'a Tanıtılması

UDM test aracını PC'nize kurduğunuza göre artık test işlemlerine geçebiliriz. Şimdi ilk olarak HID Sınıfı uyumlu USB cihaz'ımızı USB Portlarından herhangi birine taktığımız zaman Host tarafından listeleme işlemi için cihazımıza gönderilen istek paketlerini ve cihazın bu paketlere verdiği yanıtları gözlemleyeceğiz.

UDM aracını ilk çalıştırdığınızda karşınıza aşağıdaki gibi bir pencere gelecektir. USB Cihazımızın listeleme aşamasını gözlemlemek istediğimizden bu aracı çalıştırmadan önce cihazın sistemde takılı olmaması ve daha önce listelenmemiş yani Windows tarafından kaydedilmemiş olması gerekmektedir. Böylece listeleme aşamasında cihaza gönderilen istekleri izleyebileceğiz.



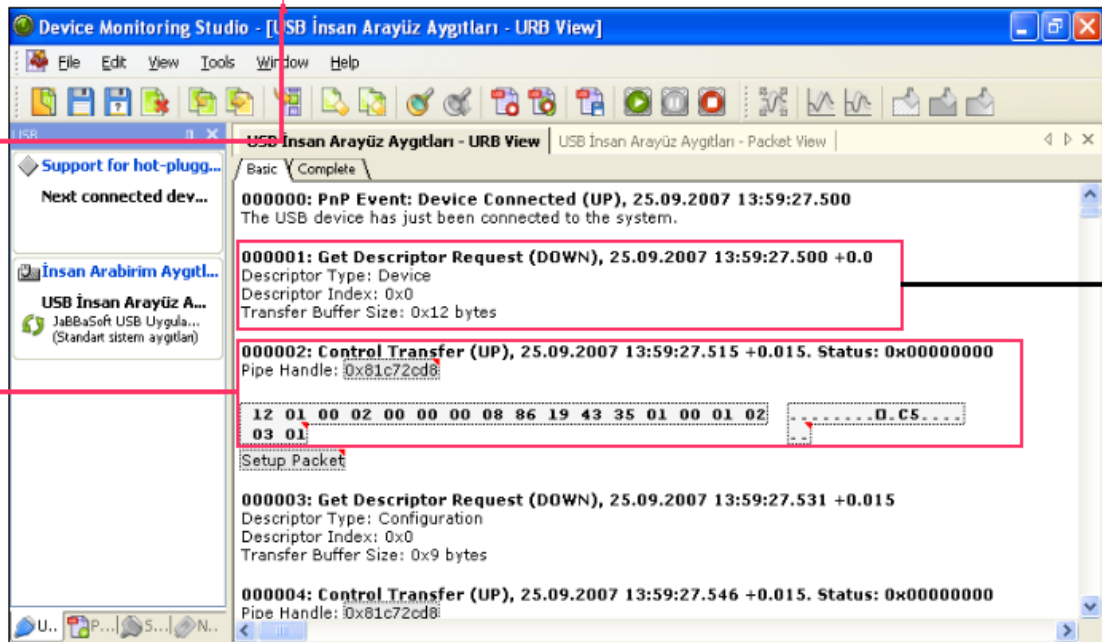
Yukarıdaki resimde dikkat ederseniz, programın alt bölümünde bizim için önemli olan Device Descriptor, Configuration Descriptor ve HID Descriptor tablaları yer almaktadır. Bu tablalar'da USB cihaz listelendikten sonra bu tanımlayıcıların bilgileri yer alacaktır. Programın son bölümünde **Next Connected Device** adında bir buton bulunmaktadır. Bu butona tıkladığımızda sisteme yeni takılacak cihaz'ı seçmiş oluyoruz ve program arayüzü aşağıdaki gibi değişiyor.



Orta bölümde yer alan seçeneklerle, sonra takılacak cihaz için "Oturma Konfigurasyonu" işlemlerini yapabileceğiz. Burada şimdilik bizi ilgilendiren iki bölüm URB View ve Packet View seçenekleri. URB View ile USB cihaza gönderilen ve cihazdan alınan ham paketler ile Kontrol İstekleri Paketlerini gözlemleyebileceğiz. Bu yüzden bu seçeneği çift tıklayarak oturuma dahil ediyoruz. Aynı şekilde Packet View seçeneğinin de çift tıklayarak oturuma dahil ediyoruz. Bundan sonra tek yapmamız gereken sağ tarafta bulunan Start butonuna tıklayıp takılacak cihaz için izleme işlemini başlatmak olacaktır. Bu durumda iken USB Cihazımızı sisteme takıyoruz ve listeleme aşamasının bitmesini bekliyoruz. Listeleme işlemi tamamlandıktan sonra URB View ve Packet View pencereleri birçok bilgi ile dolacaktır. Şimdi bu paketleri inceleyelim.

Cihaz bu işteğe karşılık Aygıt Tanımlayıcısının ilk 18 Byte'ını gönderiyor. Bu byte'ların değerlerini önceki makalemizde açıklamıştık. Bu yüzden Cihaz Tanımlayıcı Tablosunu kontrol edip buradaki değerlerle eşleştirebilirsiniz.  
**12** – Tanımlayıcı Uzunluğu, **01**– Tanımlayıcı Tipi (Sabit Device 0x01), **00 02**– USB Sürüm No (2.0)  
**00** – Sınıf Kodu, **00** – Alt Sınıf Kodu, **00** – Protokol Kodu, **08** – Üçnokta0 Maximum Packet Size .....

**Get Descriptor – Tanımlayıcı AI**  
 Host Cihaz'dan Cihaz Tanımlayıcısının toplam 12byte'ını göndermesini istiyor.

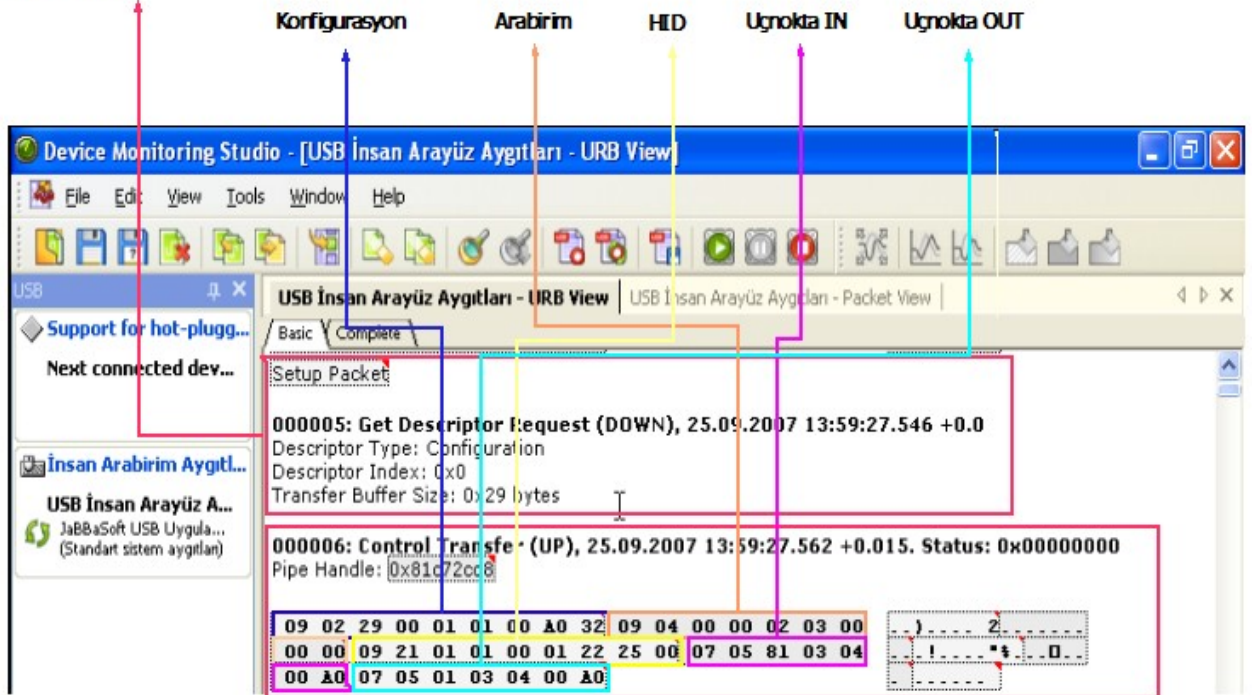


Yukarıdaki resimde de görüldüğü gibi USB cihaz sisteme takıldığında Reset işlemi sonrası, cihaza gönderilen ilk istek paketi Get\_Descriptor paketidir. Cihaz bu isteği almasının ardından paketi değerlendirecek ve istenen Tanımlayıcısının belirtilen uzunluk kadar kısmını Host'a gönderecektir. Burada Host Cihazdan Aygıt Tanımlayıcısının 18 byte'ını (toplam uzunluk) istemiş, cihaz ise istenen bu Tanımlayıcıyı Host'a göndermiştir. Gönderilen her bir byte'ın değerini yukarıdaki resimdeki açıklamadan, ADIM ADIM USB makalesindeki Aygıt Tanımlayıcısından ya da PIC Driver'ından kontrol edebilirsiniz. Bu isteğin yürütülmesi bittikten sonra aşağıdaki resimde görüldüğü gibi tekrar Get\_Descriptor isteği gönderiliyor ve bu sefer Configuration Tanımlayıcısının ilk 9 byte'ı isteniyor.



Gelen paket içeriğini daha detaylı inceleyebilmemiz için 000003 nolu transferin üzerinde görülen Setup Packet yazısının kenarındaki kırmızı ok işaretini çift tıklayın. Hemen altından bu yazının üstündeki "Basic"->"Complete" gibi tablardan belirecektir. Bu tablardan Complete tabını seçin. Böylece Setup Packet içeriğini daha iyi gözlemleyebileceksiniz. Bu istek sonrasında gönderilen byte'ları önceki örnekte yaptığımız gibi Configuration Tanımlayıcısı ile kıyaslayın. Bir sonraki gönderilen istek yine Get\_Descriptor olacağından ve istenen Tanımlayıcı Konfigurasyon Tanımlayıcısı olacağından, genişlettiğimiz Setup Packet içeriğini orada inceleyelim.

Host bu defa cihazdan Konfigurasyon tanımlayıcısına ek olarak arabirim, Hid ve uçnokta tanımlayıcıları ile birlikte toplam konfigürasyon'u istiyor.



Yukarıdaki resimde de görüldüğü gibi Host bu defa tekrar Konfigurasyon Tanımlayıcısını istemiş, fakat bu defa konfigürasyon tanımlayıcısına ek olarak Arabirim, HID ve Uçnokta Tanımlayıcıları ile birlikte toplam uzunluğu istemiştir. Cihaz ise bu isteğe karşılık tüm konfigürasyon Tanımlayıcısını göndermiştir.

Device Monitoring Studio - [USB İnsan Arayüz Aygıtları - URB View]

File Edit View Tools Window Help

USB

Support for hot-plugg...  
Next connected dev...

İnsan Arabirim Aygıtı...  
USB İnsan Arayüz A...  
JaBBSoft USB Uygula...  
(Standart sistem aygıtları)

USB İnsan Arayüz Aygıtları - URB View | USB İnsan Arayüz Aygıtları - Packet View

Basic Complete

Setup Packet  
80 06 00 02 00 00 09 00  
Recipient: Device  
Request Type: Standard  
Direction: Device->Host  
Request: 0x6 (GET\_CONFIGURATION)  
Value: 0x200  
Index: 0x0  
Length: 0x9

000005: Get Descriptor Request (DOWN), 25.09.2007 13:59:27.546 +0.0  
Descriptor Type: Configuration  
Descriptor Index: 0x0  
Transfer Buffer Size: 0x29 bytes

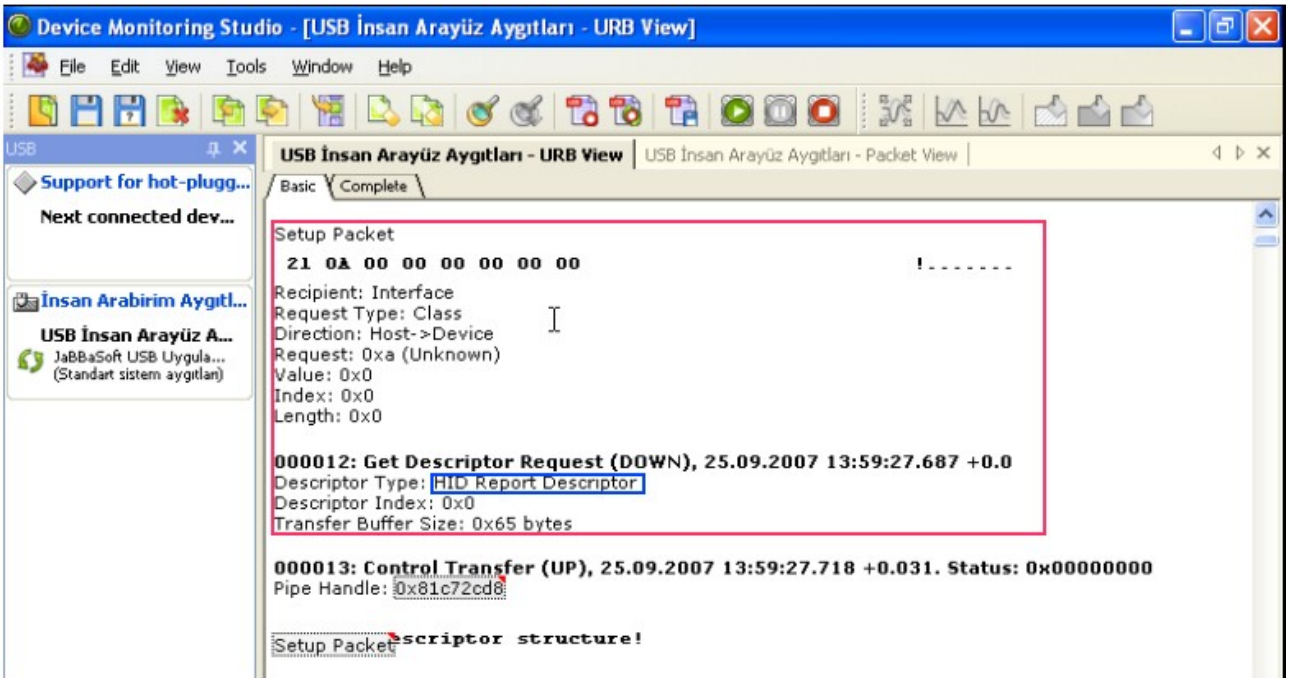
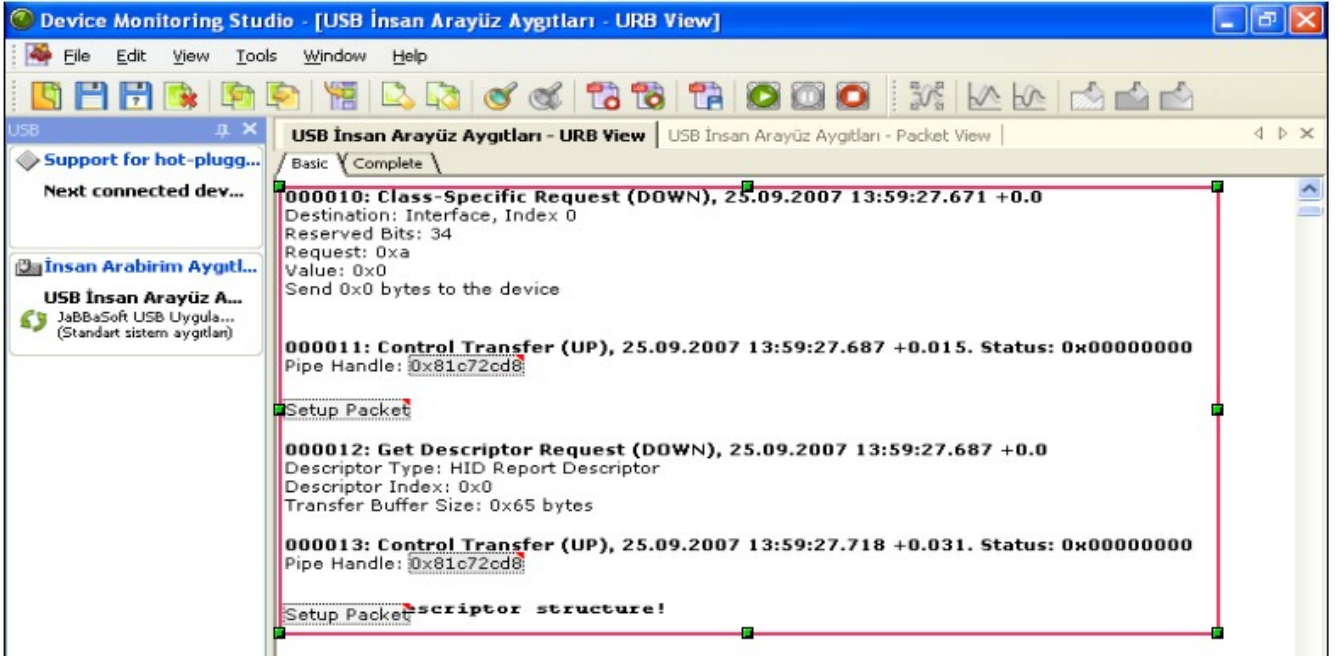
000006: Control Transfer (UP), 25.09.2007 13:59:27.562 +0.015. Status: 0x00000000  
Pipe Handle: 0x81c72cd8

09	02	29	00	01	01	00	A0	32	09	04	00	00	02	03	00
00	00	09	21	01	01	00	01	22	25	00	07	05	81	03	04
00	A0	07	05	01	03	04	00	A0							

The screenshot displays the 'Device Monitoring Studio' interface, specifically the 'USB İnsan Arayüz Aygıtı - URB View' window. The window has a blue title bar and a menu bar with 'File', 'Edit', 'View', 'Tools', 'Window', and 'Help'. Below the menu bar is a toolbar with various icons for file operations and device management. The main area is split into two panes. The left pane, titled 'USB İnsan Arayüz Aygıtı - URB View', shows a list of USB devices. The right pane, titled 'USB İnsan Arayüz Aygıtı - Packet View', shows a list of packets. The 'URB View' pane has a sub-menu with 'Basic' and 'Complete' options. The 'Packet View' pane shows a list of packets with details such as '000007: Select Configuration (DOWN), 25.09.2007 13:59:27.562 +0.0' and '000009: Select Configuration (UP), 25.09.2007 13:59:27.671 +0.0. Status: 0x00000000'.

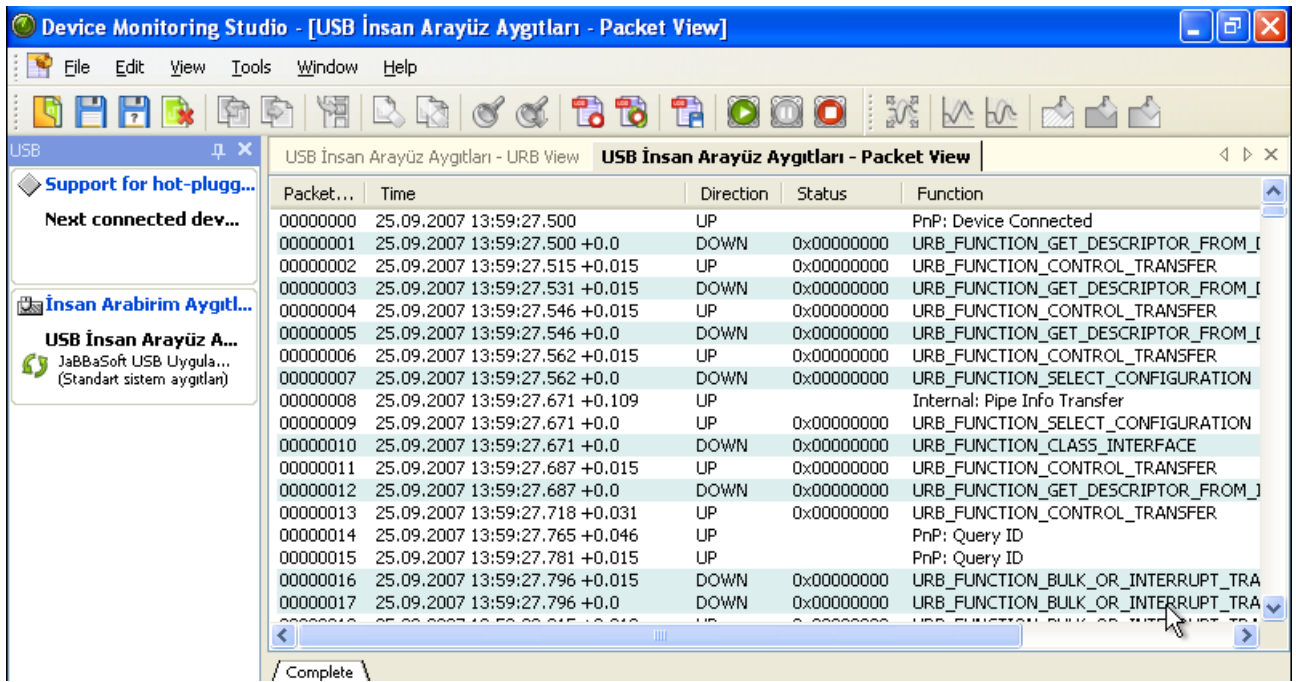


Uygun konfigürasyon, Host tarafından seçildikten sonra bu defa HID Rapor Tanımlayıcısı isteniyor.

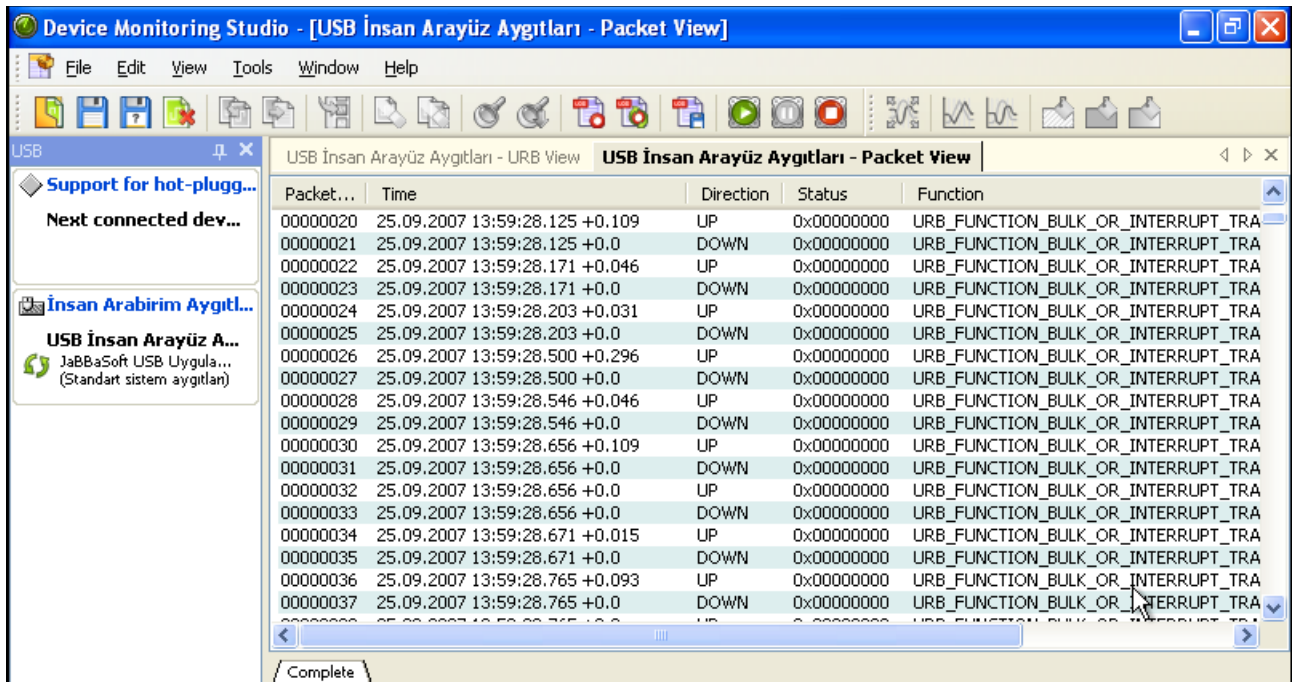


## USB Cihaz İle Veri Değişimi

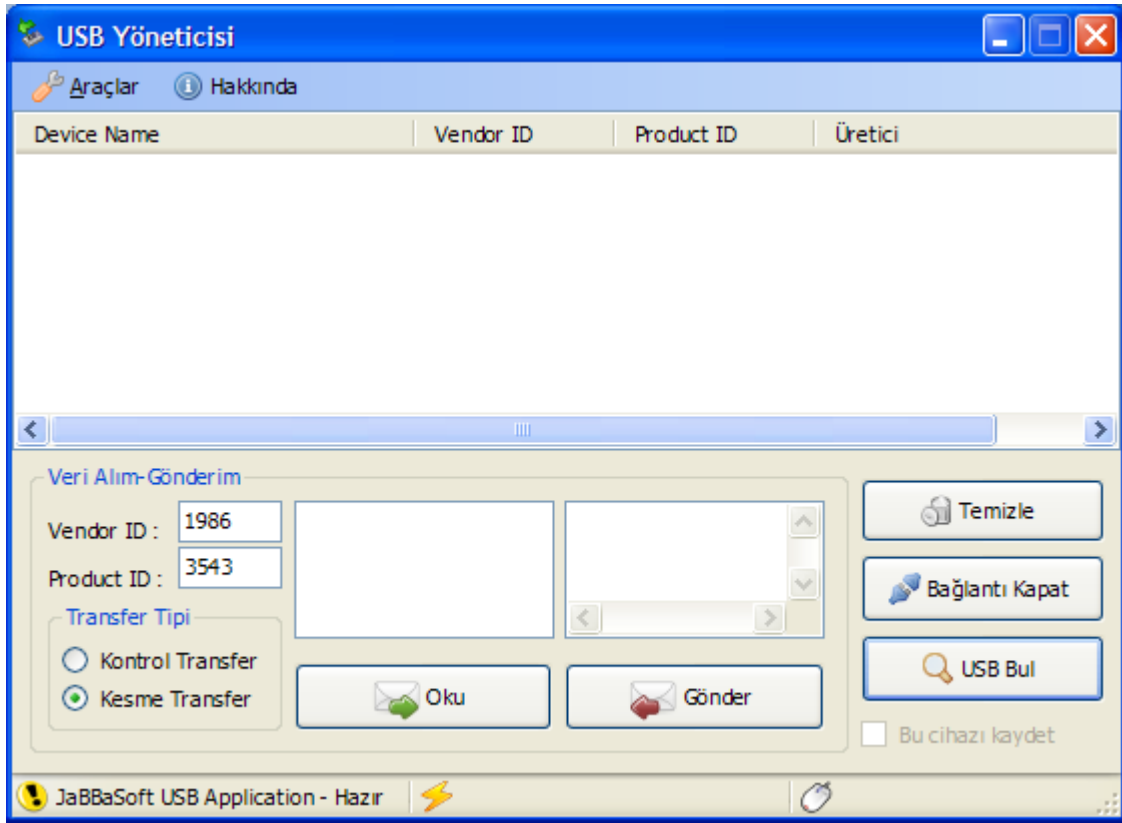
Bu işlemlerden sonra artık cihazımız veri alışverişi için hazır hale geliyor. Aşağıdaki resimde de görüldüğü gibi cihaz ile PC arasında sürekli bir veri alışverişi gerçekleşiyor. Bunun sebebi ise Host'un cihazı ADIM ADIM USB makalesinde de anlatıldığı gibi sürekli olarak yoklamasından kaynaklanıyor. Bu yoklama işlemine cihaz gönderecek verisi olmasa bile uygun bir yanıt vermelidir. Zaten resimde de görüldüğü gibi her yoklama sonrası cihaz Host'a yanıt veriyor. Aşağıdaki resimlerde görülen bazı şeyleri açıklamakta yarar var. Örneğin UDM arayüzündeki Direction bölümünde transferin hangi yöne yapıldığı gösteriliyor. DOWN cihaz'a transferi, UP ise cihazdan Host'a transferi gösteriyor. STATUS bölümü ise her IRP sonrası cihaz driver'ının Dispatch rutini tarafından döndürülen NT\_STATUS kodu olan STATUS\_SUCCESS'dır. Driver lar ile ilgili geniş bilgiyi USB Paketini indirdiğiniz rar içindeki dokümanlardan bulabilirsiniz.



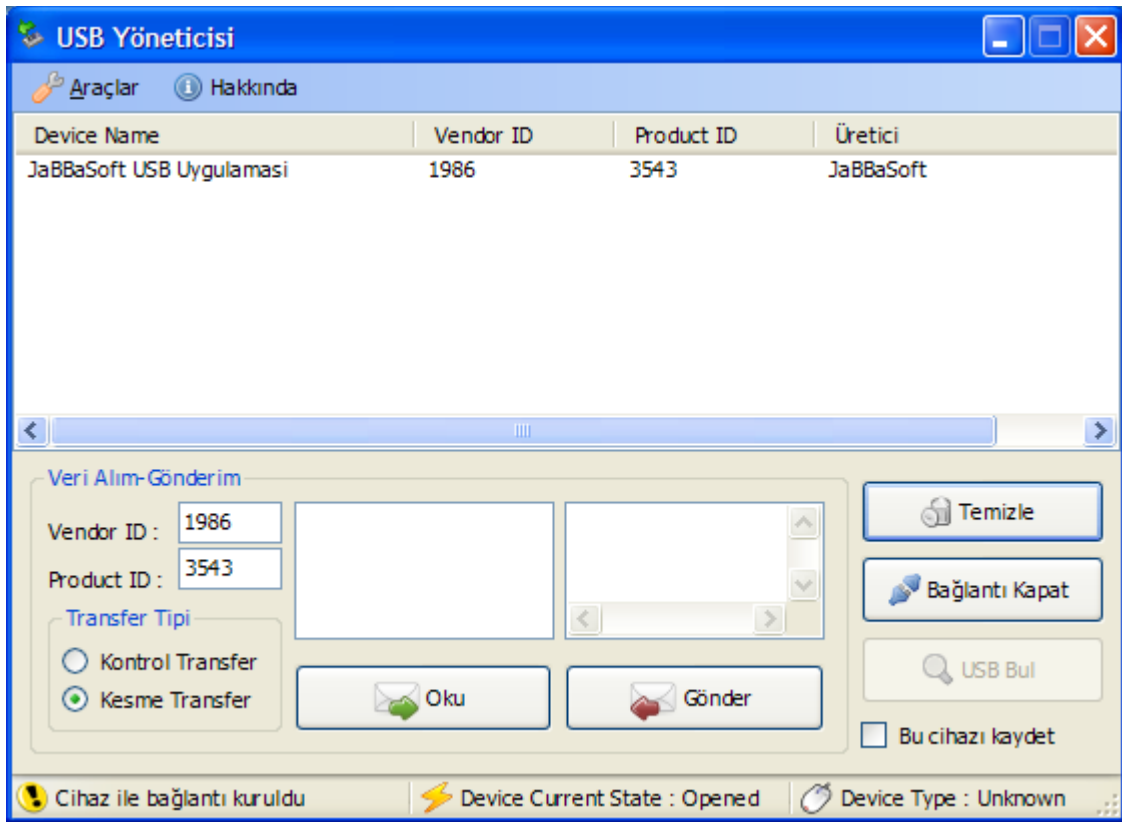
Yukarıdaki resimde, yukarıda anlatılan tüm konuların paketler halinde transfer edildiğini gösteriyor.



Bu resimde ise Host'un cihazı her yoklamasında oluşan transfer işlemlerini, gelen ve giden paketleri ve transfer türlerini gösteriyor. Şimdi bu pencereden, USB TEST UYGULAMASI programını çalıştırdıktan sonra cihaza yazma ve okuma yapıp gelen giden verileri gözlemleyebilirsiniz. Yada şimdi yapacağımız gibi detay tabına geçerek gönderilen ve alınan verileri daha detaylı inceleyebilirsiniz. Öncelikle aşağıdaki resimde görülen USB TEST UYGULAMASINI çalıştırın.



Programı çalıştırdıktan sonra USB Bul adlı butona tıkladığınızda aşağıdaki resimde görüldüğü gibi cihaz program tarafından tespit edilir ve cihaz kodunda tanımlı aygıt ve üretici isimleri okunarak ekrana yazılır.



Bu aşamadan sonra artık cihaz'a veri alma ve gönderme işlemleri yapılabilir ve UDM ile gözlemlenebilir. Şimdi Gönder butonu üzerindeki alana 3-4 byte bilgi yazın ve gönder butonuna tıklayın. Şimdi bu gönderdiğimiz verileri test programında gözlemleyelim.



**001358: Bulk or Interrupt Transfer (UP), 28.09.2007 23:47:10.515 +0.0. Status: 0x00000000**  
Pipe Handle: 0x81cf5504 (Endpoint Address: 0x81)  
Get 0x4 bytes from the device  
31 32 33 34 1234

**001359: Bulk or Interrupt Transfer (DOWN), 28.09.2007 23:47:10.515 +0.0**  
Pipe Handle: 0x81cf5504 (Endpoint Address: 0x81)  
Get 0x4 bytes from the device

USB cihaz'a deneme amaçlı sırası ile 12345 verisi gönderilmiştir.Yukarıdaki resimde de görüldüğü gibi cihaz'a 4 byte gönderilmiş ve bunların hem ASCII hemde normal değerleri gösterilmiştir.Bu görüntüyü Complate tabında görebilirsiniz. Cihaz'a kaç byte gönderirseniz gönderin umng.dll içindeki yazma methodu onu HID Rapor uzunluğundaki değere eşitleyecektir.Şimdi aynı denemeyi cihazdan okuma yapmak için tekrarlayın.Bu kez Oku butonuna tıklayarak cihazdan veri okumayı deneyin.Önceki gönderdiğiniz verileri göreceksiniz çünkü PIC iskelet kodunda alınan byte'ları tutan tampon, okuma sırasında test amaçlı gönderici tampona yazılmaktadır.Gelen giden paketleri daha rahat gözlemleyebilmek için UDM'ün araç çubuğundaki START PAUSE STOP butonlarını kullanarak transferi yönetebilirsiniz.

**Ahmet ATAR**