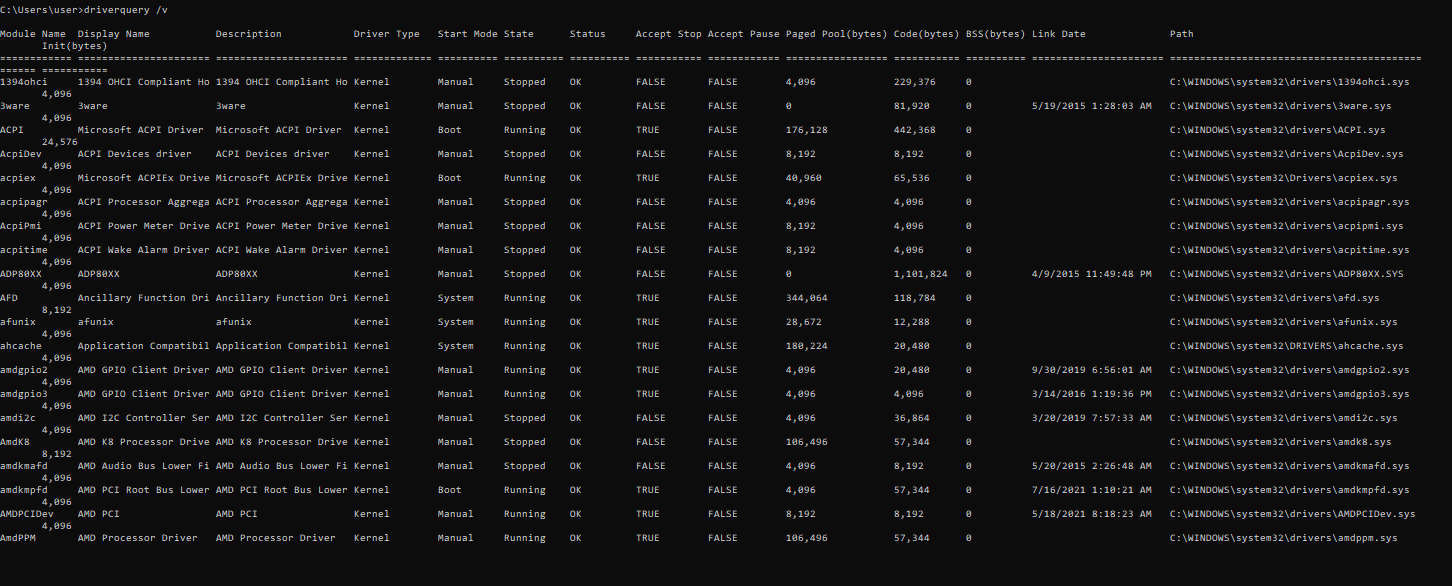
driverquery /FO list /v > D:\FATIH\Self\_Improvement\0\_My\_Works\cSharp\SADTBaslangicEgitimi\DevicesList.txt

* “driverquery /v” komutu bilgisayardaki tüm driverleri listelemektedir.
* “driverquery /v > Devices.txt” komutu ise bu listeyi Devices.txt dosyasına yazmaktır.
* Örnek Bir çıktı aşağıda verilmiş. Aynı zamanda örnek bir çıktının olduğu .txt dosyası verilmiştir.



### DOSYA OKUMA IŞLEMLERI

* Devices.txt dosyasını okuyun ve ilk iki satır dışındaki her satırı bir adet List<Dict<string,string>>’e kaydedin.
* Her satır bir Dict<string,string>’ kaydeilmelidir. Dictionary de key olarak ilgi sütunun başlığı kullanılmalıdır. Eğer atanan list değişkenin ismi “list\_Devices” olarak alırsak ilk elemana ulaşmak için list\_Devices[0][“ Module Name”] kullanılmalıdır. Diğer Key değerleri aşağıda verilmiştir.
  + “Module Name”
  + “Display Name”
  + “Description”
  + “Driver Type”
  + “Start Mode”
  + “State”
  + “Status”
  + “Accept Stop”
  + “Accept Pause”
  + “Paged Pool(bytes)”
  + “Code(bytes)”
  + “BSS(bytes)”
  + “Link Date”
  + “Path”
  + “Init(bytes)”
* Yapılacak olan işlemler bu List üzerinden yapılacak.
* ‘State’i “Running” olan satırları “RunningDevice.txt” isminde bir dosyaya kaydedin. Ama sadece “Module Name”, “Description ”,” Paged Pool(bytes)”, “Code(bytes)” sütunları kaydedilecek ve , “Paged Pool(bytes)”, “Code(bytes)” değerleri hexadecimal olarak yazılacak.

### PROCESS KULLANIMI

DOSYA OKUMA IŞLEMLERI başlığı altında okuğunuz “Devices.txt” dosyasını kendiniz oluşturun.Bunun için içerisinde bu listeyi oluran konutun bulunduğu bir .bat dosyasını çalıştırın. Örnek bir bat dosyası verilmiştir.(DeviceListBat.bat).

Bat dosyasın içerisinde “Devices.txt”nin konumunun verilmesi gerekmektedir. Uygulama taşındığında bu konum değişebileceği için Bat dosyasındaki konumunda güncellenmesi gerekecektir. Bunun için bir şablon Bat dosyası okunup çalıştırılacak olan bat dosyası üretilebilir. Örnek bir şablon bat dosyası verilmiştir.(DeviceListBatTemplate.bat)

### SERİ HABERLEŞME

DOSYA OKUMA IŞLEMLERI başlığı altında okuduğunuz text dosyasındaki bilgilerin bir seri port üzerinden geldiğini düşünün aynı işlemleri bu senaryo için gerçekleştirin.

* Öncelikle seri port üzerinden bu bilgilier gönderen bir uygulamaya ihtiyacınız var. Bu text dosyasını okuyup seri port üzerinden gönderen bir uygulama yazın.
* Seri portu dinleyip ilgili işlemleri gerçekleştirecek uygulama için bir kaç farklı yol izleyebilirsiniz.
  + Seri porttan gelen verilerin tamamın bir txt dosyasına yazıp. Tüm veriyi okuduktan DOSYA OKUMA IŞLEMLERInde yaptıklarınızın aynısını yapabilirsiniz.
  + Seri porttan okuma yaparken veri geldikçe List<String,Stringi> güncelleyebilirsiniz.

### Excel ve .xml dosyalarının kullanımı

Bu dosyaları uygulamanızı konfigüre etmek için kullanabilirsiniz. Bu sayede yaptığınız ufak değişiklikler için uygulamanızı tekrar derlemek zorunda kalmazsınız.

Seri port uygulaması için kullanıcıdan seri port isimi ve baudaret biligilerinin girilmesi isteniyordu. Uygulamaya ufak bir ekleme yapın ve kullanıcı giriş yapmadan önce ön tanımılı değerler bir excel dosyasında okunsun ve kullanıcı girdi yapığında da bu excel dosyası güncellensin ve uygulama her açıldığında son girilen değerler ile kullanıcıyı karşılasın. Örnek bir excel dosyası “SeriPortAyarlar.xlsx” verilmiştir.

Yukarıdaki uygulamayı excel dosyası yerine xml dosyası kullanarak yapın. Örnek bir xml dosyası “SeriPortAyarlar.xml” verilmiştir.

### Parse ve Dönüşüm İşlemleri

* Text -> Double Array

Haberleşme portları üzerinden aldığımız veriler genelde 1er bytelık verilerden oluşuyor. Ama anlamlı verinin boyutu değişebiliyor. Örneğin 12 bitlik bir ADC verisi tek seferde gönderilemediği için MSB ve LSB olarak 2 byte halinde gönderiliyor. Bu verinin kullanılmadan önce işlenebileceği bir formata çevirilmesi gerekiyor.

* + “BinaryBytes.txt” , “DecBytes.txt” ve “HexBytes.txt” dosyalarında 100er adet byte veri bulunmaktadır. Veriler birbirinden farklı olduğu gibi formatları da txt dosyalarının isminde belirtildiği gibidir.
  + Bu dosyaları okuyun ve bir string arraya dönüştürün.(string -> string array)
  + Bu sitring arrayleri byte arraye dönüştürün. Bu dönüşümü yaparken verilerin baselerine dikkat edin(Binary, decimal, hexadecimal). (string array -> byte array)
  + Bu dosyaları okuyun ve aşağıdaki durumları düşünerek double arrayleri oluşturun.
    - (8 bitlik veri):Her byteın bir adet örneğe denk gediğini düşünürek 100 adet elemandan oluşan bir double array oluşturun. Dönüşümleri yaparken verinin signed veya unsigned olduğuna dikkat edilmesi gerekmektedir. O yüzden hem signed hem de unsigned olarak dönüşümü yapın ve farklarını inceleyin.(byte array -> double array)
    - (16 bitlik veri): Her 2 byteın bir adet örneğe denk gediğini düşünürek 50 adet elemandan oluşan bir double array oluşturun. (ilk byte MSB ikinci byte LSB olacak şekilde) Dönüşümleri yaparken verinin signed veya unsigned olduğuna dikkat edilmesi gerekmektedir. O yüzden hem signed hem de unsigned olarak dönüşümü yapın ve farklarını inceleyin. (byte array -> double array)
    - (12 bitlik veri): Her 3 byteın 2 örneğe denk geldiğini düşünerek 66 adet elemandan oluşan bir double array oluşturun.(sadece ilk 99 byteı kullanın). Dönüşümleri yaparken verinin signed veya unsigned olduğuna dikkat edilmesi gerekmektedir. O yüzden hem signed hem de unsigned olarak dönüşümü yapın ve farklarını inceleyin. (byte array -> double array)
* Double array -> Text

Bir önceki örnekteki işlemin tam tersinin de yapılma ihtiyacı doğabilir. İşlenen verinin tekrar geri dönderilmesi veya veri gönderen bir simülatör tasarlanması gerekebilir. Bu kez double,int veya benzer formattaki bir verinin byte arraya dönüştürülmesi gerekebilir.

* + Bir önceki örnekde yapıtğınız işlemleri tersini yaparak ilk başta okuduğunuz txt dosyalarındaki verileri elde edin ve oluşturduğunuz verilerle okudğunuz dosyaları kontrol edin.

### TCP/IP HABERLEŞME

Parse ve Dönüşüm İşlemleri başlığı altındaki işlemleri TCP haberleşme üzerinden gelen veriler için yapın. Txt dosyalarındaki verileri gönderen bir uygulamaya yazın. O uygulamadn gelen verileri bekleyen başka bir uygulama yazın.

### UDP HABERLEŞME

Parse ve Dönüşüm İşlemleri başlığı altındaki işlemleri UDP haberleşme üzerinden gelen veriler için yapın. Txt dosyalarındaki verileri gönderen bir uygulamaya yazın. O uygulamadn gelen verileri bekleyen başka bir uygulama yazın.