

Bölüm 14 ASCII Çıkış Fonksiyonu Uygulaması

FBs-PLC'lerin ASCII dosya çıkış fonksiyonu, printer ve terminal gibi ASCII çıkış aygıtlarını direk olarak sürmeye izin verir. Ürün raporları, malzeme detayları ve uyarı mesajları gibi ekran görüntülerinin ve İngilizce dataların yazma veya görüntülenmelerini sağlar. ASCII dosya çıkış fonksiyonu uygulanması için, ASCII dosya data çıkışının FUN 94 (ASCWR) komutunun gerekli gördüğü formata uygun hale getirilmelidir. Sonra bu komut kullanılarak Port 1 ile bağlanmış ASCII çıkış aygıtına Port 1 üzerinden data gönderilecektir.

14.1 ASCII Dosyasının Formatı

ASCII dosya datası, sabit, değişmeyen arka fon dosyası ve dinamik olarak değişen değişken datalara bölünmüş olabilir. Arka fon dosya datası, İngilizce karakterler, sayılar, semboller, grafikler v.b. den oluşabilirler. Değişken data ise binary, decimal, hexadecimal olarak yazdırılabilirler.

ASCII kodu toplam 256 baytlık. Bunların ilk 128 baytı (0–127) açıkça anlatılmış ve çoğu ASCII ortamlarında kullanılmışlardır. 128 bayt dan büyük kodlar için her imalatçının farklı tanımları ve grafikleri vardır ve sabit bir yapısı yoktur. FBs-PLC de düzenleme için değil iletim sorumluluğu için FUN94 komutu oluşturulmuştur. Bu çalışma WinProladder yazılım paketinin ASCII editörü ile yapılmıştır. Aşağıda WinProladder yazılım paketi düzenleyicisi tarafından yapılmış düzenlenmiş komut formatı bulunmaktadır.

1. Temel Komut Sembolleri

- | |
|---|
| / |
|---|

Linefeed

Sağdan sola eğimle aşağı indirilmiş bir çizgi, yazdırmanın nerde olduğu önemsenmeden bu sembol uygulanırsa, yazdırma başı veya terminal ekranı bir sonraki çizginin başına hareket edeceği (en sola) ve yazdırma veya görüntülemeye bu noktadan devam edeceği anlamına gelir. "/" serisi, linefeed dizisi oluşturacaktır (bir "/", bir satır ilerleyecektir).

- | |
|---|
| \ |
|---|

Sayfa İlerletme

Soldan sağa doğru eğilmiş bir çizgi sembolü uygulandığında, yazdırmanın başı veya terminal görüntüsü bir sonraki sayfanın başlangıcına (sol üst köşeye) hareket edecek ve yazdırmaya veya görüntülemeye bu noktadan devam edecek demektir. "\ " serisi sayfa linefeed dizisi oluşturacaktır (Bir "\ " bir sayfa ilerletecektir).

- | |
|---|
| , |
|---|

Virgül

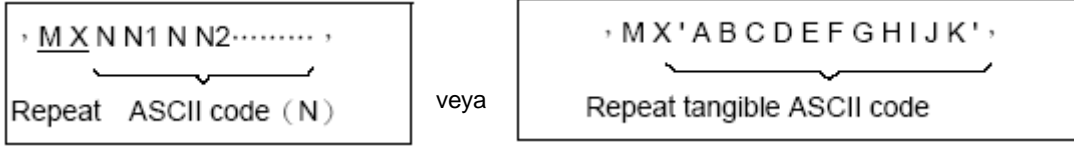
Dosya datasında cümleleri ayırmada kullanılır. İki virgül arasındaki tüm datalar tam ve yürütülür cümlelerdir (bir dosyanın başlangıcı veya bitişi için kullanılmamalıdır). Bir virgölün şekli bir tırnak işaretinin şekli ile aynı olsa da pozisyonları birbirlerinden farklıdır (virgül harfin ortasında bir yerde, tırnak işareti ise üst sağ köşede kullanılır). Temsil ettikleri özellikler de tamamen farklıdır. Lütfen Öğe 2 ye, arka fon data formatları – ifadelerine bakınız.

- | |
|-----|
| END |
|-----|

Dosya Sonu

ASCII dosyasının sonuna ASCII dosyasının bittiğini gösteren END eklenir.

2. Arka plan Data Formatı



- **MX:**

Tekrarların sayısını gösterir. M, 1 ile 999 arasında olabilir. ASCWR komutu tüm hexadecimal ASCII kodlarını veya X ve ilk virgül arasında bulunan gerçek ASCII kod dasetasını M kere gönderebilir. Eğer X' ten sonra data yok ise (örneğin, virgül direk X ten sonra geliyorsa) bu durumda ASCWR M ye boş kodlar gönderecektir. Eğer ASCII kodunu veya gerçek ASCII kodlarını bir kerede göndermek zorundaysanız MX çıkartılabilir.

- **ASCII Kodu data formatı:**

Bu format, bir N iki-basamaklı hexadecimal değere sahiptir. X' in sağ tarafından başlayarak her bitişik hexadecimal sayı bir ASCII kod sayılmıştır. NN, sayısal semboller ve kontrol kodları gibi İngilizce karakterler bitişik veya ayırık ASCII kodları kapsayan herhangi bir ASCII kod olabilir. Ancak asıl kullanımı, kontrol kodları için özel bitişik bir kod olmasıdır. Bu kod bitişik karakter fontları ile simgelenemez veya WinProladder ASCII editör font veya sembolleri bulamaz. Bitişik tuşlar ile ASCII editörlerinde direk olarak simgelenemediği bitişik karakterler veya semboller için orijinal baskı formatını kullanmak daha uygundur. Örneğin eğer "A" karakteri basılmak isteniyorsa orijinal basma yöntemi ile klavyede A basılabilir. Ama ASCII kod kullanmak isteniyorsa, "A" nın ASCII koddaki temsili kontrol edilmelidir ve 41 e basılmalıdır. Bu açıkça daha az uygundur.

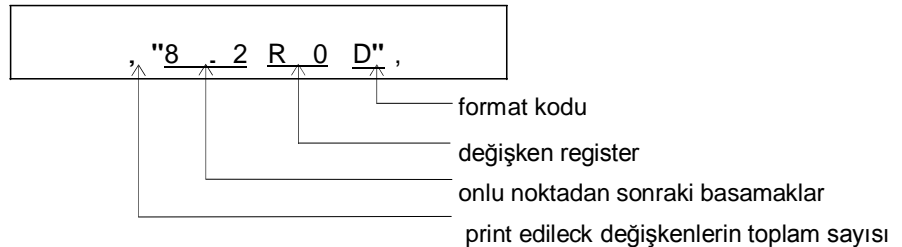
- **Orijinal baskı bitişik ASCII kod data formatı:**

İki tek tırnak işareti ' ile kapsanan şey sadece İngilizce karakterler, rakam, semboller ve grafikler(ASCII editörü tarafından girilen veya bulunan karakterler) gibi bitişik ASCII kodu olabilirler. ASCWR komutu ' ' içinde kapsanan tüm karakterleri basacaktır, bu yüzden tek bir tırnak işareti basılmak istendiğinde iki sıralı tırnak işareti kullanılmalıdır. Örneğin;

'I'M A BOY' şu şekilde bastırılacaktır I' M A BOY

ASCII çıkış cihazının grafikleri veya sembolleri ASCII düzenleyici klavyede bulunamıyorsa doğal olarak bu format kullanılarak giriş yapılamaz demektir. Böyle bir durumda sembol veya grafik için ASCII kodu kontrol edilebilir ve girişte ve baskıda ASCII kodu kullanılabilir.

3. Değişken Data Formatı



İki çift tırnak içindeki data ifadesi, değişken datanın register adresini belirtmek için kullanılmış ve bu tırnak içindeki ister format ister format kodu olsun bastırılacaktır.

- Basılan değişkenlerin toplam sayısı: Bu örnekte, "8" çeşitli (R0) sayısal değerlerin (negatif işaretler dahil) ayrılmış 8 basamak sütununun basılması için kullanılmıştır. Değişken değeri basılan basamakların toplam sayısından fazla ise yüksek basamak çıkartılacaktır. Eğer basamakların sayısı yetersiz ise kalan pozisyonlar boşluklarla doldurulacaktır.

Bu dosyayı düzenlemeden önce , dosya editörüne saklamayı hangi registerden başlayarak yapacağı söylenmelidir. Dosya datasını düzeltirken, düzeltiliş arka plan verisinin mi yoksa değişken verinin mi düzeltileceğine karar verilmelidir. Arka plan datası ASCII karakterleri veya orjinal baskı biçiminin sembolik grafikleri (' ' içindeki kullanılarak) giriş yapılabilir veya direk olarak karakterlerinin ASCII kodları veya sembol grafikleri kullanılabilir. Değişken data bölümünde olduğu gibi, registerlarda depolandığından dolayı yazdırma mesajı, format koduna ek olarak ondalık noktasından sonraki basamakları, karakter numaraları v.b. gibi yazdırma formatı ve register numaraları içerir. Yukarıdaki tablodaki örnekte, ay, yıl, gün verileri ve verimlilik (G) datası, toplam numara (A) gibi dataların hepsi değişken datalardır. Yıl, ay, gün datalarına gerçek zamanlı saat registerı RTCR içinde yıl, ay, gün registerlarından (R4133'den R4131'e) erişir. R0 toplam sayıyı (A) depolar, R1 ürün numarasını (B) depolar vs. Ve R6 verimlilik (G) değerini depolar. Aşağıda bu istatistikî tablo örneği için ASCII dosya datalarıdır;

```

///,28X,'Uretim Raporu,/,28X,'=====', /,
52X,'Date:', "2R4132", '/', "2R4131", '/', "2R4133", //, 16X, 'Toplam Sayı(A)
:', "10R0", ' PCS', //, 16X, 'Urun Sayısı (B) :', "10R1", '
PCS', //, 16X, 'Tamir Edilecek Sayısı (C) :', "10R2", ' PCS', //, 16X, 'Standart Zaman
(D) :', "10.1R3", ' MIN/PCS', //, 16X, 'TOTAL WORKING TIME (E) :', "10R4", '
MIN', //, 16X, 'Gerçek çalışma Zamanı(F) :', "10R5", ' MIN', //, 16X, 'Verimlilik
(G) :', " 10.2R6", ' %', //, //, 22X, 'Acıklama: AXD=E, E/F=G', END

```

* : Üstteki örnekte ' ===== ' 18X' tarafından doldurulabilir.= ' veya 18X3D.

Dosya çıkış işlemi esnasında, çıkış değişken veriye ulaştığında, CPU' da erişecek ve çıkışı sayısal değerlerle adresi " " içinde kapsanan register zamanında yapacaktır. Bu sebeple, eğer bir değişken bir dosyanın hem başında hem de sonunda basılıyorsa, farklı bir sayısal değer elde edilecektir (ortada basıldığında register değeri değişir).

Dosya düzenlemesi tamamlandıktan sonra, FUN94 komutu arka plan ve dinamik dataları yazdırmak için kullanılabilir. Eğer R1000' den başlayarak dosya düzenlenirse (depolanırsa) çıkışında, S sol köşede görülen örnekteki programda olduğu gibi doğru bir çıkış olmadan önce R1000 şeklinde belirtilmelidir. X1 ve X2 0 olduğunda ve X0 0' dan 1' e gider. Bu komut, PLC' nin Port1'inden önceki sayfada olan istatistiksel tabloyu yazdıracaktır.

