Dotpeenator®

BY SERMATEK

ENTEGRASYON KILAVUZU · TCP/IP VE SERİ PORT

Komut Haberleşme Protokolü.

Dotpeenator® nokta vuruşlu markalama makineleri için TCP/IP veya seri port üzerinden çalışan metin tabanlı komut protokolü. PLC, MES/ERP sistemleri ve özel yazılımların marka içeriği ayarlamasını, dosya yönetmesini, markalamayı ve hareketi kontrol etmesini ve dahili veri listesiyle çalışmasını sağlar.

ARAYÜZ	KODLAMA	SONLANDIRICI	PROTOKOL SÜRÜMÜ
TCP istemci / sunucu · Seri	UTF-8 / GBK / Latin1	\n	V9.0.S0513

Komut adlarında ve yanıtlarda (örneğin `Start`, `!PrinterBusy`, `>Print Stopped`) geçen *print* ve *printer* kelimeleri Dotpeenator® markalama makinesini ifade eder. Komutlar ve yanıtlar yazıldığı gibi aynen kullanılmalıdır.

1 Haberleşme Temelleri

1.1 Bağlantı Modları

MOD	AÇIKLAMA
İstemci Modu	Kontrolcü TCP istemcisi olarak çalışır ve harici bir sunucuya kendisi bağlanır
Sunucu Modu	Kontrolcü TCP sunucusu olarak çalışır ve harici istemcilerin bağlanmasını bekler
Seri Port 1	USART1 üzerinden haberleşme
Seri Port 2	USART2 üzerinden haberleşme

1.2 Veri Formatı

- **Kodlama:** UTF-8 / GBK / Latin1 vb. desteklenir, ayarlardan seçilir
- Komutlar `\n` ile ayrılır; tek gönderimde birden fazla komut gönderilebilir (satır sonlarıyla ayrılarak)

- Kontrolcü yanıtları varsayılan olarak `>` başlangıç sembolüyle başlar (ayarlanabilir) ve `\n` ile biter
- Komut anahtar kelimelerinde büyük/küçük harf ayrımı yoktur

1.3 Genel Yanıt Kuralları

YANIT	ANLAMI
`>data content`	Normal yanıt (`>` başlangıç sembolüdür, ayarlanabilir)
`!PrinterBusy`	Makine meşgul, mevcut komut çalıştırılamıyor
`!BadFormat:...`	Komut formatı hatası
`!BadProperty:...`	Desteklenmeyen özellik adı
`!CanNotLock:...`	Marka nesnesi kilitlenemiyor
`!BadCommand:...`	Tanınmayan komut
`!CodingFailed:...`	Kodlama başarısız
`!SetPropFailed:...`	Özellik ayarlanamadı

1.4 Mantıksal (Boolean) Değer Kuralı

Mantıksal bir özellik **yazılırken** şu değerler "doğru" kabul edilir: `T` / `t` / `1` / `True` (büyük/küçük harf duyarsız); diğer tüm değerler "yanlış"tır.

Mantıksal bir özellik **okunurken** yanıt `T` veya `F` olur.

---

2 Marka İçerik İşlemleri (En Sık Kullanılanlar)

2.1 `@` Marka Özelliği Ayarla

Belirtilen markanın özellik değerini ayarlar. Makine meşgulken `@` önekli komutlar reddedilir.

FORMAT

```
@MarkName Property=Value
@MarkName=Value            # Özellik belirtilmezse Code kabul edilir
```

DESTEKLENEN ÖZELLİKLER

ÖZELLİK	AÇIKLAMA	ÖRNEK
`Code`	Marka içeriği (varsayılan)	`@M1=ABC123` / `@M1 Code=ABC123`
`Pos`	Marka konumu (x,y)	`@M1 Pos=100,200`
`Select` / `Sel`	Seçim durumu	`@M1 Select=T`
`Index` / `Idx`	Marka indeksi	`@M1 Index=2`
`Enable`	Markalamayı etkinleştir	`@M1 Enable=F`
Diğer genişletilmiş özellikler	Marka tipine bağlı	`@M1 FontSize=12`

ÖRNEKLER

```
@M1=ABC123            → M1 içeriğini ABC123 yap
@M1 Code=ABC123       → Yukarıdakiyle aynı, Code açıkça belirtilmiş
@M1=Hello World       → İçerik boşluk içerebilir (= sonrasındaki her şey değerdir)
@M1 Pos=100.5,200.3   → M1 konumunu ayarla
@M1 Pos=100,          → Yalnızca X ayarlanır, Y değişmez
@M1 Select=T          → M1'i seç
@M1 Enable=F          → M1 markalamasını devre dışı bırak
@M1 Index=3           → M1 indeksini 3 yap
```

YANIT: Başarılı olursa marka adını döndürür, örn. `>M1` ; başarısız olursa `!CodingFailed:...` vb. döndürür.

2.2 `Get` Marka Özelliğini Sorgula

FORMAT

```
Get MarkName [Property]
```

Özellik adı belirtilmezse varsayılan olarak `Code` sorgulanır.

DESTEKLENEN ÖZELLİKLER

ÖZELLİK	AÇIKLAMA	DÖNÜŞ DEĞERİ
`Code`	Marka içeriği	Metin (String)
`Pos`	Marka konumu	`x,y`
`Select` / `Sel`	Seçim durumu	`T` veya `F`
`Index` / `Idx`	Marka indeksi	Tam sayı
`Enable`	Markalama etkin mi	`T` veya `F`
Diğer genişletilmiş özellikler	Marka tipine bağlı	Metin (String)

ÖRNEKLER

```
Get M1          → M1 içeriğini sorgula (varsayılan: Code)
Get M1 Code     → Yukarıdakiyle aynı
Get M1 Pos      → M1 konumunu sorgula, örn. >100.5,200.3 döner
Get M1 Select   → M1 seçim durumunu sorgula, >T veya >F döner
Get M1 Enable   → M1 markalamasının etkin olup olmadığını sorgula
Get M1 Index    → M1 indeksini sorgula
```

2.3 Set Marka Özelliği Ayarla (@ ile eşdeğer)

`Set` komutu işlev olarak `@` önekiyle aynıdır, yalnızca komut formatı farklıdır. Makine meşgulken reddedilir.

FORMAT

```
Set MarkName Property=Value
Set MarkName=Value
```

ÖRNEKLER

```
Set M1=ABC123        → M1 içeriğini ayarla
Set M1 Pos=100,200   → M1 konumunu ayarla
```

2.4 Formatlı Veri Girişi — Toplu Marka İçeriği Ayarlama

"Özel formatlı dizeleri almaya izin ver" ve "Formatlı dize parametreleri" ayarlandığında, tek bir formatlı dizeyle tüm marka içerikleri toplu olarak ayarlanabilir.

FORMAT

```
StartSymbolData1SeparatorData2SeparatorData3...
```

Varsayılan başlangıç sembolü `^`, varsayılan ayırıcı `,` karakteridir. Örnek: `^ABC,DEF,,GHI`

KURALLAR

- Veriler markalara önce Index sırasına (artan), sonra ada göre eşleştirilir
- Dolu veri: marka içeriğini ayarlar ve markalamayı etkinleştirir
- Boş veri: o markanın markalamasını devre dışı bırakır
- Veri gönderilmeyen markalar: markalama etkin kalır (orijinal içerik korunur)
- "Formatlı veri alındıktan hemen sonra yazdır" ayarlıysa markalama otomatik başlar

Dokümanda üç marka olduğunu varsayalım: `M1(idx=1)`, `M2(idx=2)`, `M3(idx=3)`:

ÖRNEKLER

```
^ABC,DEF,GHI   → M1=ABC, M2=DEF, M3=GHI, tümü etkin
^ABC,DEF,      → M1=ABC, M2=DEF, M3 markalaması kapalı
^ABC,,GHI      → M1=ABC, M2 kapalı, M3=GHI
^ABC           → M1=ABC, M2/M3 orijinal içeriği korur ve etkin kalır
```