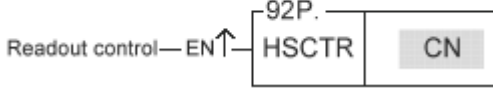


FUN 92 P
HSCTR

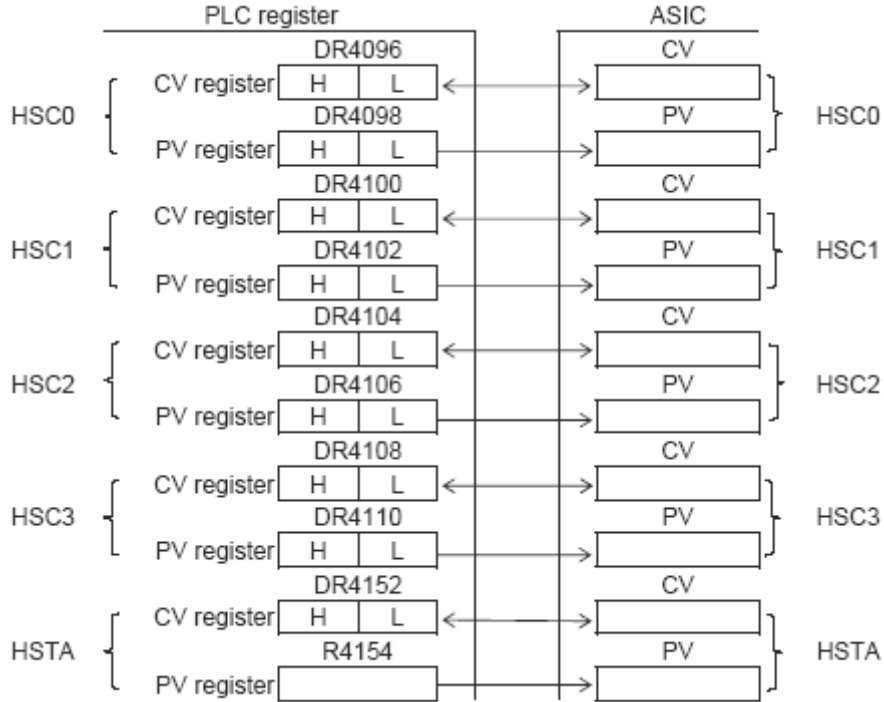
YÜKSEK HIZLI SAYICI GÜNCEL DEĞER (CV) ERIŞİM DONANIMI

FUN 92 P
HSCTRLadder symbol

CN : Hardware high speed counter number



- 0: SC0 or HST0
- 1: SC1 or HST1
- 2: SC2 or HST2
- 3: SC3 or HST3
- 4: STA

- FBs-PLC'nin HSC0~HSC3 sayıcıları, yukarı/aşağı darbeleri, darbe-yönlü, AB-fazlı gibi çeşitli sayma modları ile yüksek hızlı sayıcıların 4 tipidir. 4 yüksek hızlı sayıcının tümü ASIC donanımında yerleşiktir ve sayma, karşılaştırma ve CPU müdahalesi olmadan bağımsız olarak interrupt gönderimi gerçekleştirebilirler. HSC4~HSC7 yüksek hızlı sayıcı yazılımına karşı, CPU işlemi için istenen interrupt yöntemini sağlar. Bundan dolayı, çok fazla sayma sinyali varsa veya sayma frekansı yüksekse PLC performansı (tarama hızı) dramatik bir şekilde azalacaktır. HSC0~HSC3'nin güncel değeri CV, ASIC dahili donanım devresine yerleşik olduğundan dolayı kullanıcı kontrol programına (ladder diyagram) ASIC'den direk olarak erişilemeyecektir. Bu yüzden, kontrol programından erişilerek HSC donanımından CV değerini alıp register içine yerleştirmek bu komut sayesinde yapılır. Aşağıda CV ayarlaması, ASIC'deki PV ve diğer uygun CV, HSC0~HSC3 için PLC'nin PV registerları gösterilmektedir.



- Erişim kontrolü "EN" =1 veya \uparrow "ENP" komutu) 0→1 şeklinde değiştiğinde ASIC'den CN tarafından tasarlanmış HSC'nin güncel değeri alınacak ve HSC'de uygun CV registeri içine koyulacaktır (örn; HSC0'in CV'si okunacak ve DR4096 içine koyulacak veya HSC1'in CV'si okunacak ve DR4100 içine koyulacaktır).
- Her ne kadar ASIC içerisindeki PV CPU'daki uygun PV registerına sahipse, CPU'daki PV registerından gelen ASIC içerisine bu PV değeri için erişmek gerekmemektedir.
- HSTA bir 0.1ms'lik bir zamanlayıcıdır. CV içeriği, 0.1ms im'de geçen zamanı göstermektedir..
- Uygulama detayı için, Bölüm 10'daki "FBs-PLC'nin Yüksek Hızlı Sayıcı ve Yüksek Hızlı Zamanlayıcı" kısmına bakınız.

Gelişmiş Fonksiyon Komutu

HSCTW	DONANIM YÜKSEK HIZLI SAYICI GÜNCEL DEĞERİ VE ÖNCEDEN BELİRLENMİŞ DEĞER YAZIMI	FUN 93  HSCTW
	<p style="text-align: center;"><u>Ladder symbol</u></p>  <p>S : Yazmak için kaynak data</p> <p>CN : Yazılacak olan donanım yüksek hızlı sayıcı</p> <p>0: HSC0 or HST1 1: HSC1 or HST2 2: HSC2 or HST3 3: HSC3 or HST4 4: HSTA</p> <p>D : Yazma hedefi (CV 0, 1 PV 1)</p>	
	<ul style="list-style-type: none"> • CPU'daki PV registerları ve uygun CV'ler ve ASIC içerisinde HSTA ve HSC0~HSC3'ün PV değeri veya CV arasındaki ilişki için FUN92'ye bakınız.. • Yazma kontrolü "EN"=1 veya "EN↑" (P komutu) 0→1 şeklinde değiştiğinde, CPU'nun CN tarafından tasarlanmış yüksek hızlı sayıcısının PV registerı veya CV'nin içeriğini, ASIC ten HSC'de PV veya uygun CV'ye yazar. • Daha fazla program uygulaması için PV değeri sık sık kurulmalıdır. Sayma değeri önceden belirlenen değere ulaştığında, sayıcı hemen interrupt sinyali gönderecektir. Interrupt servis programının dolaylı yolu, pozisyon kontrolü veya duyarlı saymanın farklı çeşitlerini gerçekleştirir. • FBs-PLC için güç kaynağının bir interruptı olduğunda, ASIC içerisine HSC0~HSC3'üngüncel değer registerları CV'nin değerini okuyacak ve CPU'nun otomatik olarak HSC0~HSC3 CV registerları içine yazacaktır. Güç açıldığında, CV değerleri ASIC'e geri yüklenecektir. Ancak, güç ON olduğunda uygulanması isteniyorsa, değerler 0'lanacak veya belirli bir değerden saymaya başlayacaktır daha sonra ASIC'deki HSC için CV değerinde yazmak için bu komutu kullanmalısınız. • Sıfır olmayan bir değer HSTA'nın PV registerına yazıldığında her PV×0.1ms için çalıştırılmış olan HSTAI interrupt subroutineları oluşacaktır. • Uygulama detayları için Bölüm 10'da olan "FBs-PLC'nin Yüksek Hızlı Sayıcı ve Yüksek Hızlı Zamanlayıcı" kısmına bakınız. 	<ul style="list-style-type: none"> ⌘ Soldaki programda, M0 değiştiğinde, FUN93 sayesinde ASIC donanımına yazar.. ⌘ M0, 0 olduğunda güncel sayma değeri okunacaktır. ⌘ M1 0→1 şeklinde değiştiğinde, DR500 DR4098'e taşınır ve önceden belirlenen değer FUN93 sayesinde ASIC donanımına yazılır. ⌘ Her ne zaman güncel değer DR500'e şit olursa, HSC0I interrupt alt pogramı çalışacaktır.
	