Siemens TIA Portal'da Teknoloji Nesneleri ve Simotion Arasındaki Farklar

1. Temel Yaklaşım ve Kullanım Alanları:

* TIA Portal Teknoloji Nesneleri:
  + Kolay Kullanım: S7-1200/1500 PLC'lerde doğrudan çalışır. Hazır fonksiyon blokları (örneğin, "MC\_MoveAbsolute") ile basit/mekanik hareket kontrolü sağlar.
  + Uygulama Alanı: Standart pozisyonlama, hız kontrolü, eksen takip işlemleri gibi temel hareket uygulamaları için ideal.
  + Entegrasyon: PLC programlaması (LAD/FBD/STL) ile doğrudan entegre olur.
* Simotion:
  + Dedike Hareket Kontrolü: Karmaşık senkronizasyon, kam profilleri, elektronik dişli sistemler gibi gelişmiş hareket senaryoları için özel olarak tasarlanmıştır.
  + Programlama Dili: "SCOUT" ortamında "SCL" (Structured Control Language) veya grafiksel bloklarla programlanır.
  + Uygulama Alanı: Yüksek hassasiyetli CNC makineleri, robotik kollar, paketleme makineleri gibi dinamik sistemlerde tercih edilir.

2. Performans ve Esneklik:

* TIA Portal:
  + Sadece 1-2 eksenli basit uygulamalarda yeterlidir.
  + Gerçek zamanlı senkronizasyon sınırlıdır.
  + Donanım tabanlı hız denetimi (örneğin, hızlı giriş/çıkış trigger'ları) sınırlı desteklenir.
* Simotion:
  + Çok eksenli senkronizasyon (örneğin, 6 eksenli robot kol hareketleri) için optimize edilmiştir.
  + Dinamik yörünge hesaplamaları (NURBS interpolasyonu) ve geri besleme döngüleri için yüksek işlem gücü sunar.
  + Gerçek zamanlı haberleşme (PROFIBUS/PROFINET üzerinden) ile düşük gecikmeli kontrol sağlar.

3. Donanım Bağımlılığı:

* TIA Portal: Sadece Siemens S7-1200/1500 PLC'lerinde çalışır.
* Simotion:
  + Bağımsız bir kontrol ünitesi (Simotion C/D/P serisi) olarak çalışabilir.
  + S7-1500 ile entegre kullanıldığında, Simotion'un gücüne erişilebilir (örneğin, S7-1500T CPU ile).

4. Geliştirme ve Bakım Süreci:

* TIA Portal:
  + Daha kısa öğrenme eğrisi.
  + Basit hata ayıklama ve parametreleme imkanı.
* Simotion:
  + Kompleks senaryolar için derin teknik bilgi gerektirir.
  + Simülasyon araçları (örneğin, Drive Simulation) ile test süreci uzar.

5. Maliyet ve Ölçeklenebilirlik:

* TIA Portal: Düşük maliyetli çözümler için uygun. Küçük projelerde tercih edilir.
* Simotion: Yüksek başlangıç maliyeti, ancak büyük ölçekli sistemlerde uzun vadeli verimlilik sağlar.

Aynı Şeyleri Yaptırabilir miyim?

* Evet, ama sınırlı olarak:
  + Basit pozisyonlama, hız kontrolü gibi temel görevler hem TIA hem Simotion ile yapılabilir.
  + Simotion olmadan yapılamayacak şeyler:
    - Çok eksenli senkronize hareket (örneğin, 4 eksenli çizim robotu).
    - Elektronik kam profilleri (örneğin, makine döngüsüne bağlı dinamik hareket).
    - Yüksek hızlı geri besleme döngüleri (örneğin, 1 ms altı güncelleme süresi).

Sonuç:

* TIA Portal Teknoloji Nesneleri: Basit ve orta düzey hareket kontrolü için hızlı, maliyet etkin çözümler sunar.
* Simotion: Endüstriyel otomasyonun en zorlu uygulamalarında esneklik ve performans sağlar.

Proje gereksinimlerine göre seçim yapılmalıdır. Örneğin, bir paketleme makinesinde Simotion gerekliyken, bir taşıyıcı bandın hız kontrolü TIA ile yeterlidir.

Omron ve Delta'nın Çok Eksenli Servo Kontrol Çözümleri

1. Omron: Sysmac Platformu

Ana Ürünler ve Yazılımlar:

* Kontrolörler:
  + NJ/NX Serisi (Sysmac Platformu): Omron'un çok eksenli hareket kontrolü sağlayan entegre PLC+Motion kontrolörleri.
  + CP2E Serisi : Küçük ölçekli uygulamalar için kompakt PLC'ler (az eksenli kontrol).
* Yazılım:
  + Sysmac Studio:
    - PLC programlama (IL, ST, LD), HMI tasarımı ve hareket kontrolünü tek bir ortamda birleştirir.
    - Motion Modülü: Çok eksenli senkronizasyon, kam profilleri, elektronik dişli sistemler gibi gelişmiş hareket senaryoları için kullanılır.
    - Benzetim (Simulation): Gerçek zamanlı hareket senaryolarını test etme imkanı.
* Haberleşme:
  + EtherCAT: Yüksek hızlı (1 ms altı döngü süresi) ve düşük gecikmeli haberleşme.
  + Diğer protokoller: PROFINET, CANopen, MECHATROLINK.

Ana Özellikler:

* Çok Eksenli Kontrol:
  + NJ serisi ile 32-64 eksen senkronizasyonu desteklenir.
  + 3B koordinat dönüşümleri (örneğin, robot kol hareketleri).
* Karmaşık Yörünge Hesapları:
  + Lineer, dairesel interpolasyon, NURBS desteği.
* Entegrasyon:
  + Vision sistemleri (örneğin, Omron FH Serisi) ile senkronize çalışma.

Uygulama Alanları:

* Paketleme makineleri, montaj hatları, test cihazları gibi orta ila yüksek karmaşıklıktaki sistemler.

2. Delta: ASDA ve DMC Platformları

Ana Ürünler ve Yazılımlar:

* Servo Sürücüler:
  + ASDA-A3/A2 Serisi: Yüksek hassasiyetli sürücüler (±1 mikron doğruluk).
  + ASDA-M Serisi: Tek kontrolör ile birden fazla eksen kontrolü (örneğin, ASDA-M-R2 ile 6 eksen).
* Kontrolörler:
  + DVP Serisi PLC'ler: Basit eksen kontrolü için.
  + PC Tabanlı Motion Kartları:
    - DMCNET Serisi: 4-8 eksenli kontrol için PCI/PCIe kartlar (örneğin, DMCNET-01).
    - CANopen/PULSE Serisi: Esnek haberleşme protokolleri.
* Yazılım:
  + ASDA-Soft: Servo motorların parametrelenmesi ve temel hareket kontrolü için.
  + ISPSoft: PLC programlama ve HMI entegrasyonu.
  + DMC Motion Studio: PC tabanlı gelişmiş hareket senaryoları için (kam profilleri, senkronizasyon).

Ana Özellikler:

* Çok Eksenli Kontrol:
  + ASDA-M serisi ile 6 eksenli senkronizasyon.
  + Elektronik dişli (Gearing) ve pozisyon senkronizasyonu.
* Haberleşme:
  + CANopen, MODBUS RTU/TCP, PROFINET.
* Kolay Entegrasyon:
  + Üçüncü parti PLC'lerle uyumluluk (örneğin, Siemens, Allen Bradley).

Uygulama Alanları:

* CNC makineleri, lazer kesim sistemleri, elektronik montaj hatları gibi maliyet odaklı çözümler.

Karşılaştırma ve Özet

| Özellik | Omron (Sysmac) | Delta (ASDA/DMC) |
| --- | --- | --- |
| Haberleşme | EtherCAT (ana tercih), PROFINET | CANopen, MODBUS, PROFINET |
| Çok Eksen Desteği | 32-64 eksen (NJ serisi) | 6 eksen (ASDA-M), 8 eksen (DMC kart) |
| Yazılım Kolaylığı | Tek ortamda entegrasyon (Sysmac Studio) | Modüler yazılımlar (ASDA-Soft, ISPSoft) |
| Karmaşık Hareketler | Kam profilleri, 3B interpolasyon | Elektronik dişli, temel senkronizasyon |
| Maliyet | Üst segment (yüksek performans) | Orta-alt segment (uygun fiyat) |
| Tipik Kullanım Alanı | Endüstriyel otomasyon, robotik | CNC, lazer kesim, montaj hatları |

Sonuç: Aynı Şeyleri Yaptırabilir miyim?

* Evet, ama senaryoya bağlı:
  + Basit Uygulamalar (1-4 eksen): Hem Omron hem Delta yeterlidir (örneğin, taşıyıcı bandın pozisyon kontrolü).
  + Karmaşık Senkronizasyon (6+ eksen, yüksek hız):
    - Omron: Daha gelişmiş senaryolar için uygun (örneğin, 6 eksenli robot kol hareketleri).
    - Delta: Maliyet odaklı çözümler sunar, ancak en yüksek karmaşıklıkta Simotion veya Omron kadar esnek değildir.
* Simotion ile Karşılaştırma:
  + Simotion, daha yüksek hassasiyet (örneğin, 1 µm altı), gerçek zamanlı dinamik yörünge optimizasyonu ve endüstriyel ağırlıklı uygulamalar için tercih edilir.
  + Omron ve Delta, Simotion'un bazı özelliklerini sunar ama maliyet ve ölçeklenebilirlik açısından farklı sektörlere hitap eder.

Öneri:

* Yüksek Performans Gerekiyorsa: Omron Sysmac (NJ serisi + EtherCAT).
* Maliyet Duyarlı Proje: Delta ASDA-M veya DMC kartları.
* Simotion Düzeyi İhtiyaç: Sadece Siemens çözümü uygun olur.

Her iki marka da Siemens'in sunduğu kapsamda yer alır, ancak uygulama alanları ve bütçeye göre seçim yapılmalıdır.

Omron Sysmac Studio ile 3 eksenli servo kontrol simülasyonu yapmak için aşağıdaki adımları takip edebilirsiniz. Bu rehber, yeni bir proje başlatırken dikkat edilmesi gerekenleri ve doğru kontrolcü seçimini detaylandırır.

1. Uygun Kontrolcü Seçimi

Önerilen Model:

* NJ Serisi (örneğin NJ501-1500):
  + 3 eksenli servo kontrol için yeterli kapasiteye sahiptir.
  + EtherCAT haberleşmesi ile yüksek hızlı hareket kontrolü sağlar.
  + Sysmac Studio ile tam entegre çalışır.
* Alternatif: NX1P2 (daha düşük maliyetli, HMI entegrasyonu için uygun).

Neden NJ Serisi?

* EtherCAT Desteği: Servo sürücülerle düşük gecikmeli haberleşme.
* 3 Eksen Senkronizasyonu: Lineer/dairesel interpolasyon gibi gelişmiş hareketler için uygun.
* Esneklik: Gelecekte eksen sayısını artırma imkanı sunar.

2. Gerekli Yazılım ve Sürücüler

* Sysmac Studio:
  + [Omron’in resmi sitesinden](https://www.fa.omron.com.cn/product/software/sysmac-studio) indirin.
  + Sürüm seçimi: NJ serisi için en az Ver. 1.25 önerilir.
* EtherCAT Sürücüsü:
  + Sysmac Studio kurulumunda "EtherCAT Master" bileşenini ekleyin.

3. Proje Oluşturma Adımları

A. Yeni Proje Açma

1. Sysmac Studio'yu Başlatın:
   * "New Project" seçeneğine tıklayın.
2. Kontrolcü Modelini Seçin:
   * "Controller Model Selection" kısmında:
     + Series: NJ
     + Model: NJ501-1500 (veya seçtiğiniz model).
   * "OK" butonuna basın.

B. EtherCAT Cihazlarını Ekleme

1. EtherCAT Master Modülü Ekleyin:
   * Sol paneldeki "Device & Network" sekmesine geçin.
   * "EtherCAT Master (Port 1)" seçeneğini sürükle-bırak ile ekleyin.
2. Servo Sürücüleri Tanımlayın:
   * "Slave Library" sekmesinden Omron R88D-KN sürücüsünü (veya kullandığınız sürücü modelini) seçin.
   * Her eksen için ayrı sürücü ekleyin (toplam 3 adet).
   * Her sürücüye benzersiz Slave ID atayın (örneğin 1, 2, 3).

C. Ağ Ayarlarını Yapılandırma

1. IP Adresi ve Alt Ağ Ayarları:
   * "Device & Network" sekmesinde "Network Configuration" kısmına gidin.
   * Kontrolcünün IP adresini 192.168.0.1 olarak ayarlayın.
   * Servo sürücülerin IP adresleri otomatik atanır (EtherCAT üzerinden).

4. Hareket Kontrolü Yapılandırması

A. Eksen Tanımlama

1. Motion Modülüne Geçin:
   * Sol panelde "Motion" sekmesine tıklayın.
2. Yeni Eksen Oluşturun:
   * "Add Axis" butonuna basarak 3 adet eksen tanımlayın (örneğin Axis1, Axis2, Axis3).
3. Eksen Parametrelerini Ayarlayın:
   * Gear Ratio: Redüktör oranı (örneğin 1:5).
   * Encoder Resolution: Encoder çözünürlüğü (örneğin 10000 pulse/rev).
   * Home Method: Sıfır noktası ayarı (örneğin limit switch veya sensör).

B. Simülasyon Modunu Etkinleştirme

* Gerçek Donanım Olmadan Test Etme:
  + Proje ayarlarında "Simulation Mode" seçeneğini aktif edin.
  + "Virtual I/O" kullanarak servo sürücülerin davranışını simüle edin.
* 5. Programlama (Structured Text - ST)
* Aşağıda 3 eksenli basit bir pozisyonlama örneği verilmiştir:

// Global Değişkenler

VAR

Axis1, Axis2, Axis3: AXIS\_REF;

Pos1, Pos2, Pos3: REAL := 100.0; // Hedef pozisyonlar

Speed: REAL := 1000.0; // Hız (mm/s)

END\_VAR

// Ana Program

PROGRAM PLC\_PRG

VAR

Axis1MoveDone, Axis2MoveDone, Axis3MoveDone: BOOL;

END\_VAR

// Eksen 1 Hareketi

MC\_MoveAbsolute(

AXIS := Axis1,

Position := Pos1,

Velocity := Speed,

Acceleration := 1000,

Deceleration := 1000,

Execute := NOT Axis1MoveDone,

Done => Axis1MoveDone

);

// Eksen 2 Hareketi

MC\_MoveAbsolute(

AXIS := Axis2,

Position := Pos2,

Velocity := Speed,

Acceleration := 1000,

Deceleration := 1000,

Execute := Axis1MoveDone AND NOT Axis2MoveDone,

Done => Axis2MoveDone

);

// Eksen 3 Hareketi

MC\_MoveAbsolute(

AXIS := Axis3,

Position := Pos3,

Velocity := Speed,

Acceleration := 1000,

Deceleration := 1000,

Execute := Axis2MoveDone AND NOT Axis3MoveDone,

Done => Axis3MoveDone

);

6. HMI Entegrasyonu (Opsiyonel)

* Basit Arayüz Tasarımı:
  + "HMI Screen" sekmesine geçin.
  + Butonlar, pozisyon göstergeleri ve hata mesajları ekleyerek kullanıcı arayüzü oluşturun.
  + Örnek:
    - "Start" butonu ile hareket başlatma.
    - Gerçek zamanlı pozisyon göstergeleri (Axis1.ActualPosition).

7. Test ve Hata Ayıklama

1. Simülasyon Modunu Başlatın:
   * "Run Simulation" butonuna tıklayarak programı test edin.
2. Değişken İzleme:
   * "Watch Window" ile eksen pozisyonlarını, hızları ve hata durumlarını izleyin.
3. Hata Kontrolü:
   * MC\_ReadError fonksiyonu ile sürücü hatalarını kontrol edin.

8. Ekstra İpuçları

* Örnek Projeler: Sysmac Studio içindeki örnek projeleri (Help > Sample Projects) inceleyin.
* Donanım Uyumlu Modeller:
  + Servo motor: Omron R88M-KH05030H-S2 (3 kW, 3000 RPM).
  + Sürücü: R88D-KN05H-S2 (EtherCAT destekli).
* Gelecekteki Genişletme: 3 eksenden 6 eksene geçmek için NJ serisi daha yüksek model (örneğin NJ501-1600) seçin.

Sonuç

Bu adımları takip ederek Omron Sysmac Studio ile 3 eksenli servo kontrol simülasyonunu kolayca gerçekleştirebilirsiniz. Gerçek donanım olmadan bile simülasyon modu ile mantıksal testler yapmanız mümkün. İleri düzey uygulamalar için kam profilleri, interpolasyon ve senkronizasyon gibi özelliklere de odaklanabilirsiniz.

Aşağıda Sysmac Studio'da "Axis Basic Settings" penceresindeki temel ayarların ne anlama geldiğini ve nasıl kullanılacağını açıklayacağım:

1. Genel Ayarlar

* Axis number (Eksen Numarası):
  + Eksene bir numara atanır (örneğin 0, 1, 2).
  + Bu numara, programda eksenin tanımlanmasında kullanılır (Axis0, Axis1 vs.).
* Axis use (Eksen Kullanımı):
  + Used axis: Eksen etkinleştirilir ve kontrol edilebilir.
  + Not used: Eksen devre dışı bırakılır (programda kullanılmaz).
* Axis type (Eksen Türü):
  + Virtual servo axis: Sanal (yapay) bir servo eksenidir. Gerçek donanım olmadan simülasyonda test için kullanılır.
  + Real servo axis: Gerçek bir servo motorla entegre edilmiş eksen.
* Feedback control (Geribildirim Kontrolü):
  + No control loop: Geribildirim döngüsü yoktur (açık döngü modu).
  + Position control loop: Pozisyon geribildirim döngüsü etkin (kapanış döngüsü).
  + Speed control loop: Hız geribildirim döngüsü etkin.

2. Girdi/Cıktı Ayarları

* Input device 1-3 (Giriş Cihazı 1-3):
  + Fiziksel girdiler (örneğin limit anahtarı, sensör) ile eşleştirilir.
  + <Not assigned>: Henüz bir cihaz atanmamıştır.
  + Channel: İlgili girdinin bağlandığı I/O kanalı (örneğin DI\_01).
* Output device 1-3 (Çıkış Cihazı 1-3):
  + Fiziksel çıkıslar (örneğin servo etkinleştirme sinyali) ile eşleştirilir.
  + <Not assigned>: Henüz bir cihaz atanmamıştır.
  + Channel: Çıkışın bağlandığı I/O kanalı (örneğin DO\_01).

3. Detay Ayarları (Detailed Settings)

* Function Name (Fonksiyon Adı):
  + Eksene özgü işlevler (örneğin çıkışlar, girdiler, dijital girdiler).
* Device (Cihaz):
  + İşlevin hangi fiziksel cihaza bağlı olduğunu belirtir (örneğin servo sürücüsü).
* Process Data (Proses Verisi):
  + İşlevin veri tipi ve boyutu (örneğin REAL için 4 byte).

4. Uyarı Mesajı

* "Invalid combinations may cause unexpected operations..."
  + MC Function Module'ların kombinasyonları ve proses verileri değiştiğinde, cihazların beklenmedik davranış göstermesine dikkat edilmesi gereklidir.
  + Örneğin: Bir servo eksenine yanlışlıkla hız kontrolü yerine pozisyon kontrolü uyguladığınızda hata oluşabilir.

Simülasyon Konteksi

* Virtual servo axis seçimi, gerçek donanım olmadan da test yapmanızı sağlar.
* No control loop ayarı, simülasyonda geribildirim döngüsünün gerekmediğini belirtir.
* Giriş/Çıkış ayarlarının <Not assigned> olması , sanal eksenin fiziksel cihazlara bağımlı olmadığını gösterir.

Örnek Kullanım Senaryosu

1. Sanal Eksen Testi:
   * Ekseni Virtual servo axis olarak ayarlayıp, programda pozisyonlama komutları yazarsınız.
   * Simülasyon modunda eksenin hareketini izleyebilirsiniz.
2. Gerçek Donanıma Geçiş:
   * Ekseni Real servo axis yaparken, giriş/çıkış cihazlarını (limit anahtarları, encoder) doğru kanallara eşleştirirsiniz.
   * Feedback control 'u Position control loop yaparak kapanış döngüsünü aktifleştirirsiniz.

Bu ayarlar, Omron Sysmac Studio'da 3 eksenli servo kontrol projesi oluştururken temel konfigürasyonun nasıl yapılacağını gösterir. Sanal eksen kullanarak hızlıca test yapabilir, daha sonra gerçek donanıma geçiş yapabilirsiniz.

Aşağıda Sysmac Studio'da "Motion Unit Conversion" ayar panelinin ne işe yaradığını ve nasıl çalıştığını açıklayacağım:

1. Unit of Display (Gösterim Birimi)

* Amaç: Eksenin pozisyonunu hangi birimde göstereceğini seçmenizi sağlar.
  + pulse: Encoder pulsları (örneğin 10.000 pulse/döngü).
  + mm, µm, degree, inch: Fiziksel hareket birimleri.

2. Travel Distance Ayarları

Bu bölümde, motor döngüsü ile fiziksel hareket arasındaki ilişkiyi belirleyen parametreler bulunur.

A. Do Not Use Gearbox (Redüktör Kullanma)

* Kullanım Alanı: Redüktör olmayan sistemler (doğrudan motor → yük).
* Parametreler:
  1. Command Pulse Count per Motor Rotation (Motor Döngüsü başına Komut Pulse Sayısı):
     + Motorun encoder'ının çözünürlüğü (örneğin 10.000 pulse/döngü).
  2. Work Travel Distance per Motor Rotation (Motor Döngüsü başına Yük Hareket Mesafesi):
     + Motorun bir döngüsünde yükün ilerlediği mesafe (örneğin 5 mm).
* Formül:

1

Number of Pulses = (Command Pulse Count) × (Travel Distance) / (Work Travel Distance per Motor Rotation)

* 1. Örnek: 10.000 pulse/döngü ve 5 mm/döngü için:
     + 10 mm → 10.000 × 10 / 5 = 20.000 pulse

B. Use Gearbox (Redüktör Kullanma)

* Kullanım Alanı: Redüktör veya dişli kutusu içeren sistemler.
* Parametreler:
  1. Work Travel Distance per Work Rotation (Yük Döngüsü başına Hareket Mesafesi):
     + Yükün bir döngüsündeki mesafe (örneğin 5 mm/döngü).
  2. Work Gear Ratio (Yük Redüktör Oranı):
     + Yük taraflı dişli sayısı (örneğin 10).
  3. Motor Gear Ratio (Motor Redüktör Oranı):
     + Motor taraflı dişli sayısı (örneğin 2).
* Formül:

1

Number of Pulses = (Command Pulse Count) × (Motor Gear Ratio) × (Travel Distance) / [Work Travel Distance per Work Rotation × Work Gear Ratio]

* 1. Örnek: 10.000 pulse/döngü, Motor Gear=2, Work Gear=10, 5 mm/döngü için:
     + 10 mm → 10.000 × 2 × 10 / (5 × 10) = 40.000 pulse

3. Grafiksel Temsil

* Motor ve Yük Arasındaki İlişki:
  + Sol taraftaki resim, motorun dişli ile yükün dişli arasındaki etkileşimiyi gösterir.
  + M: Motor, W: Yük.
  + (1)-(5): Parametrelerin fiziksel karşılıkları (örneğin, (3) = Work Travel Distance).

4. Modo Türleri (Count Mode)

* Linear Mode (Doğrusal Mod):
  + Sınırsız doğrusal hareket (örneğin, taşıyıcı band).
  + Formül: Number of Pulses = (Command Pulse Count × Motor Gear Ratio × Travel Distance) / (Work Travel Distance × Work Gear Ratio)
* Rotary Mode (Dönel Mod):
  + Sınırlı döngüler (örneğin, devir sayımı).
  + Formül: Number of Pulses = (Command Pulse Count × Motor Gear Ratio × (Modulo Max - Min) × Travel Distance) / (Work Gear Ratio)

5. Pratik Örnek

* Sistem: Motor → Redüktör (2:10) → Yük.
  + Motor Encoder: 10.000 pulse/döngü.
  + Yük Hareketi: 5 mm/döngü.
  + Gösterim Birimi: mm.
* Hedef: 10 mm gitmek için gereken pulse sayısını hesaplayın.

Number of Pulses = (10.000 × 2 × 10) / (5 × 10) = 4.000 pulse

Sonuç

Bu ayarlar, motorun verdiği pulse sayısını fiziksel hareket (mm, derece) ile ilişkilendirmenizi sağlar. Redüktör kullanılıyorsa, dişli oranlarını da dikkate alarak doğru konfigürasyon yapmanız gerekir. Hesaplamalar, programdaki MC\_MoveAbsolute gibi fonksiyonların çalışması için kritiktir.

Redüktör Kullanılan Sistemler ve Doğrudan Motor → Yük Sistemleri

Motorların yük ile nasıl entegre edileceği, uygulamanın fiziksel gereksinimlerine bağlıdır. İşte iki yaklaşımın temel farklılıkları ve örnek uygulamaları:

1. Redüktör Kullanılan Sistemler

Neden Kullanılır?

* Torque Arttırma: Motorun ürettiği torku artırır (speed düşürülerek).
* Hız Düşürme: Motor hızını yük için uygun seviyeye indirir.
* Mekanik Çevirme: Motor döngüsünü doğrusal harekete dönüştürür (örneğin, dişli-zımpa sistemleri).

Örnek Uygulamalar:

* CNC Makineleri: Spindle'ın yüksek tork gerektiren kesim işlemlerinde.
* Taşıyıcı Bandları: Ağır yük taşımak için düşük hiz, yüksek tork gerekir.
* Montaj Robotları: Ağır parçaları taşıyan kollar.
* Elektronik Dişli Sistemleri: Senkronize eksenlerde hız kontrolü.

Redüktör Seçimi:

* Dişli Kutuları: Yüksek verimli, ancak geri besleme var.
* Vites Sistemleri: Hız/tork oranını ayarlamak için.
* Zımpalar: Doğrusal hareket için (örneğin, hidrolik pompa).

2. Doğrudan Motor → Yük Sistemleri

Neden Kullanılır?

* Yüksek Hassasiyet: Geri besleme yok, hızlı yanıt sağlar.
* Düşük Gecikme: Redüktörün ek gecikmesi olmaz.
* Basit Mekanik Tasarım: Entegre edilmiş motorlar (örneğin, lineer servolar).

Örnek Uygulamalar:

* 3B Yazıcılar: Hassas pozisyonlama.
* Paketleme Makineleri: Hızlı ve doğru pozisyonlama.
* Robot Kol Armaları: Hafif ağırlıklarla çalışırken.
* Lazer Kesim Sistemleri: Düzgün çizgiler için hızlı yanıt.

Teknik Avantajlar:

* Etkinlik: Redüktörün kaybı yok (düşük enerji tüketimi).
* Geri Besleme: Motorun direkt geri beslenmesiyle daha az hata.
* Uzun Ömrü: Daha az механический parça, daha az bakımda.

Karşılaştırma Tablosu

| Kriter | Redüktör Kullanma | Doğrudan Motor → Yük |
| --- | --- | --- |
| Torque Gereksinim | Yüksek | Orta/Düşük |
| Hız Gereksinim | Düşük | Yüksek |
| Mekanik Kompleksite | Yüksek (dişli, zımpa vb.) | Düşük |
| Etkinlik | Düşük (redüktör kaybı) | Yüksek |
| Hassasiyet | Orta | Yüksek |
| Maliyet | Yüksek (redüktör maliyeti) | Düşük |
| Uygulama Alanı | Ağır makine, CNC, taşıyıcı band | Robotik kol, paketleme, 3B yazıcı |

Sonuç

* Redüktör Kullanın:
  + Ağır yükler, düşük hız, yüksek tork gereklidir.
  + Mekanik çevirmede (döngü → doğrusal) dönüşüm yapmanız gerekiyorsa.
* Doğrudan Motor → Yük:
  + Hassas pozisyonlama, yüksek hız, düşük gecikme gereklidir.
  + Entegre motorlar (lineer servolar) kullanabilirseniz.

Sysmac Studio'da konfigürasyon yaparken, redüktör kullanılıyorsa "Use gearbox" seçeneğini etkinleştirip dişli oranlarını belirtmeniz gerekir. Aksi takdirde "Do not use gearbox" ile basit formülle hesaplamalar yapabilirsiniz.

Sysmac Studio'da "Unit of Display" (Gösterim Birimi) olarak pulse seçildiğinde, eksen pozisyonu encoder pulsları cinsinden gösterilir. Bu durumda motor encoder değeri aşağıdaki gibi hesaplanır:

1. Temel Bilgiler

* Motor Encoder:
  + Motorun bir devrinde (360°) ürettiği pulse sayısıdır.
  + Örnek: 10.000 pulse/rev (motor bir kez döndüğünde 10.000 pulse üretilir).
* Gearbox Ratio (Redüktör Oranı):
  + Motor dişlisi / Yük dişlisi oranı.
  + Örnek: 2:10 → Motor 2 dişli, yük 10 dişli.
  + Bu oran, motorun kaç kez dönmesi gerektiğini belirler:

Gear Ratio = Motor Dişlisi / Yük Dişlisi = 2 / 10 = 0.2

* Yani, yükün 1 devri için motor 5 kez dönmelidir (1 / 0.2 = 5).

2. Pulse Cinsinden Hesaplama

A. Motorun Bir Devri İçin Pulse Sayısı

* Formül:

Motor Devri Başına Pulse Sayısı = Encoder Çözünürlüğü

Örnek:

Motor 1 devri → 10.000 pulse

B. Yükün Bir Devri İçin Pulse Sayısı

* Formül:

Yük Devri Başına Pulse Sayısı = Encoder Çözünürlüğü × Motor Devri Sayısı (Redüktörden Kaynaklı)

Örnek:

Motor 5 devir → 10.000 × 5 = 50.000 pulse

Yük 1 devri = 50.000 pulse

C. Belirli Bir Mesafe İçin Pulse Sayısı

* Örnek: Yükün 10 mm ilerlemesi isteniyor.
  + Yükün 1 devri → 5 mm hareket.
  + 10 mm → 2 devir.
  + Gerekli pulse:

2 devir × 50.000 pulse/devir = 100.000 pulse

3. Sysmac Studio'da Ayarlar

A. "Motion Unit Conversion" Ayarları

1. Unit of Display:
   * pulse olarak ayarlayın.
2. Use Gearbox Bölümü:
   * Work Travel Distance per Work Rotation:
     + Yükün bir devrindeki hareket mesafesi (örneğin 5 mm).
   * Work Gear Ratio:
     + Yük dişlisi (örneğin 10).
   * Motor Gear Ratio:
     + Motor dişlisi (örneğin 2).
3. Command Pulse Count per Motor Rotation:
   * Motor encoder çözünürlüğü (örneğin 10.000 pulse/rev) olarak girin.

4. Gerçek Zamanlı İzleme

* Watch Window ile Pulse Gözlemi:
  + Programda eksen pozisyonunu Axis.ActualPosition ile izlediğinizde, bu değer pulse cinsinden gösterilecektir.
  + Örnek:

// Eksen 1'in gerçek pozisyonunu pulse cinsinden göster

ActualPulse := Axis1.ActualPosition; // 10.000 = 1 motor devri

6. Anahtar Noktalar

* Encoder Değeri:
  + Sadece motorun fiziksel encoder çözünürlüğüne bağlıdır (örneğin 10.000 pulse/rev).
  + Redüktör oranı, pulse sayısını değil, yükün hareket mesafesini etkiler.
* Redüktörün Rolü:
  + Pulse sayısını artırmaz, ancak aynı pulse sayısı için yükün daha az hareket etmesine neden olur.
  + Örnek:
    - 10.000 pulse → Motor 1 devir, yük 0.2 devir (redüktör 5:1).
* Simülasyon Modunda Test:
  + Gerçek donanım olmadan pulse cinsinden hareketi test edebilirsiniz.

Bu ayarlar sayesinde, pulse cinsinden pozisyonlama yaparken motorun encoder çözünürlüğünü ve redüktör oranlarını doğru şekilde tanımlayabilirsiniz. Redüktör oranı, pulse sayısını değil, yükün fiziksel hareket mesafesini belirler.

Aşağıda Sysmac Studio'da "Operation Settings" penceresindeki ayarların ne işe yaradığını ve nasıl kullanılacağını açıklayacağım:

1. Hız/Aksilasyon/Değişim Hızı Ayarları

Bu bölümde, eksenin hız, ivme ve azalış hızı sınırları tanımlanır.

A. Hız Sınırları

* Maximum velocity (Maksimum Hız):
  + Eksenin izin verilen en yüksek hızı (pulse/s).
  + Örnek: 40.000.000 pulse/s → Motor saniyede 40 milyon pulse ile çalışabilir.
* Start velocity (Başlangıç Hızı):
  + Hareket başladığında ilk anda motorun kullandığı hız (pulse/s).
  + Genellikle 0 olarak ayarlanır (hareket yavaşça başlar).
* Maximum jog velocity (Maksimum Jog Hızı):
  + Manuel kontrol (jog) modunda izin verilen en yüksek hız (pulse/s).
  + Örnek: 1.000.000 pulse/s → Jog sırasında saniyede 1 milyon pulse.

B. İvme/Azalış Hızı Sınırları

* Maximum acceleration (Maksimum İvme):
  + Eksenin hızını artırmak için izin verilen en yüksek ivme (pulse/s²).
* Maximum deceleration (Maksimum Azalış Hızı):
  + Eksenin hızını azaltmak için izin verilen en yüksek azalış hızı (pulse/s²).

C. Uyarı Değerleri

* Velocity warning value (Hız Uyarısı):
  + Hızın % kaçına ulaştığında uyarı verileceği değer.
  + Örnek: 80% → Hız maksimumunun 80%'ine ulaştığında alarm tetrelir.
* Acceleration/Deceleration warning value (İvme/Azalış Hızı Uyarısı):
  + İvme veya azalış hızının % kaçına ulaştığında uyarı verileceği değer.

D. İvme/Azalış Modu

* Acceleration/deceleration over:
  + Use rapid acceleration/deceleration (Blending is changed to Buffered):
    - Hız değişimlerini daha hızlı yapar, ancak interpolasyonu "Buffered" (tamponlu) modda çalıştırır.
    - Diğer seçenek: Use smooth acceleration/deceleration (Blending) → Daha yumuşak geçişler sağlar.

E. Yön Değiştirme Davranışı

* Operation selection at Reversing:
  + Motorun yön değiştirmede ne yapacağını belirtir.
  + Deceleration stop (Azalışla Durdur): Motor önce yavaşlayıp durur, sonra ters yöne geçer (daha güvenli).
  + Immediate reversing (Anlık Ters Çevir): Motor anında ters yöne geçer (daha hızlı, ancak motoruna daha fazla stres uygular).

2. Tork Ayarları

Bu bölümde, motorun tork sınırları ve uyarı değerleri ayarlanır.

* Positive/Negative torque warning value (Poziif/Negatif Tork Uyarısı):
  + Motorun ürettiği torkun % kaçına ulaştığında uyarı verileceği değer.
  + Örnek: 90% → Tork maksimumunun 90%'ine ulaştığında alarm tetrelir.

3. İzleme (Monitor) Ayarları

Bu bölümde, pozisyon ve hız kontrolü için toleranslar ayarlanır.

* In-position range (Pozisyon Toleransı):
  + Motorun hedef pozisyona ne kadar yaklaştığını belirler (pulse cinsinden).
  + Örnek: 10 pulse → Motor hedefe 10 pulse içine girdiğinde "pozisyonda" sayılır.
* Actual velocity filter time constant (Gerçek Hız Filtre Zaman Sabiti):
  + Hız okumalarındaki dalgalanmaları filtrelemek için kullanılan zaman sabiti (ms).
  + 0 ms → Filtre devre dışı.
* Zero position range (Sıfır Pozisyon Toleransı):
  + Motorun sıfır noktasına ne kadar yaklaştığını belirler (pulse cinsinden).
* In-position check time (Pozisyon Kontrol Süresi):
  + Motorun pozisyona ulaştığını teyit etmek için ne kadar süre beklenir (ms).
  + 0 ms → Anında kontrol yapılır.

Pratik Kullanım Önerileri

1. Simülasyon Modunda Test Etme:
   * Gerçek donanım olmadan da hız, ivme ve pozisyon toleranslarını test edebilirsiniz.
2. Maksimum Hız ve İvme Ayarları:
   * Motorun fiziksel kapasitesine göre ayarlayın. Aksi takdirde motor sobe veya bozulabilir.
3. Tork Uyarıları:
   * Motorun taşıyamayacağı yüklerle karşılaştığında tork sınırını aşarsa uyarı almanız önemlidir.
4. Pozisyon Toleransı:
   * Düşük değerler, daha yüksek doğruluk gerektiren uygulamalarda kullanılır (örneğin 1 pulse).
   * Yüksek değerler, hafif yüklerde veya toleranslı sistemlerde tercih edilir.

Örnek Senaryo

* Motorun Maksimum Hızı: 40.000.000 pulse/s
* Jog Hızı: 1.000.000 pulse/s
* İvme: 20.000.000 pulse/s²
* Azalış Hızı: 20.000.000 pulse/s²
* Pozisyon Toleransı: 5 pulse

Bu ayarlar, motorun hızlı hareket etmesini ve hedefe yüksek doğrulukta ulaşmasını sağlar. Ancak motorun fiziksel sınırlamalarına uygun olup olmadığı dikkatlice denetlenmelidir.

Bu ayarlar, Omron Sysmac Studio'da 3 eksenli servo kontrol projesi oluştururken motor davranışını optimize etmenizi sağlar. Her parametre, uygulamanızın gereksinimlerine göre özelleştirilmelidir.

Aşağıda Sysmac Studio'da "Other Operation Settings" penceresindeki ayarların ne işe yaradığını ve nasıl kullanılacağını açıklayacağım:

1. Anında Durdurma Yöntemleri

Bu ayarlar, durdurma sinyalleri alındığında motorun davranışını belirler.

A. Immediate Stop Input Method (Anında Durdurma Sinyali Metodu)

* Seçenekler:
  + Immediate stop (Anında Durdur): Motor anında durur (herhangi bir yavaşlama olmadan).
  + Deceleration stop (Azalışla Durdur): Motor maksimum azalış hızıyla durur.
  + Coast stop (Kaydırarak Durdur): Motoru devre dışı bırakır ve kendi ağırlığıyla durur.

B. Limit Input Stop Method (Sınır Girdisi Durdurma Metodu)

* Aynı seçenekler ile sınır anahtarı gibi fiziksel sınırlayıcı cihazlardan gelen durdurma sinyaline cevap verilir.
* Örnek: Sınır anahtarı aktive edildiğinde motor anında durur.

2. Hata Reset Zamanı ve Tork Sınırları

A. Drive Error Reset Monitoring Time (Motor Hata Reset İzleme Zamanı)

* Amaç: Motorun hata durumdan kurtulması için izin verilen en fazla zaman (ms).
* Örnek: 200 ms → Motor 200 ms içinde hata durumundan çıkamazsa sistem alarm verir.

B. Maximum Positive/Negative Torque Limit (Maksimum Poziif/Negatif Tork Sınırı)

* Amaç: Motorun ürettiği torkun üst sınırlarını belirler (% cinsinden).
* Örnek: 300% → Motorun maksimum torkünün 300%'ine ulaşması durumunda alarm tetrelir veya durdurulur.

3. Giriş Mantığı Ters Çevirme

Bu ayarlar, fiziksel giriş sinyallerinin mantığını tersine çevirmenizi sağlar (aktif-düşük/aktif-yüksek).

A. Immediate Stop Input Logic Inversion (Anında Durdurma Giriş Mantığı Ters Çevirme)

* Seçenekler:
  + Do not invert (Ters Çevirme Yapma): Standart mantık (örneğin, aktivasyon için sinyal HIGH).
  + Invert (Ters Çevir): Sinyal mantığını tersine çevirir (aktivasyon için LOW).

B. Positive/Negative Limit Input Logic Inversion (Poziif/Negatif Sınır Giriş Mantığı Ters Çevirme)

* Aynı mantık ile sınır anahtarları gibi cihazların sinyallerini ayarlayabilirsiniz.

C. Home Proximity Input Logic Inversion (Ana Konum Yakınlığı Giriş Mantığı Ters Çevirme)

* Amaç: Ana konuma yaklaşma sensörlerinin sinyallerini tersine çevirmek.

4. Pratik Kullanım Önerileri

1. Anında Durdurma Senaryoları:
   * Acil durumlar için Immediate stop tercih edin (örneğin, eylemsizlik algılama).
   * Zorunlu durdurma için Deceleration stop daha güvenlidir (motoru zorlamadan durdurur).
2. Tork Sınırları:
   * Motorun fiziksel kapasitesine göre ayarlayın. Aksi takdirde motor sobe veya bozulabilir.
3. Giriş Mantığı Ters Çevirme:
   * Fiziksel cihazların sinyal mantığı farklıysa (örneğin, limit anahtarı aktif-düşük), burada düzeltin.

Örnek Senaryo

* Anında Durdurma: Immediate stop seçili.
* Sınır Durdurma: Deceleration stop seçili.
* Tork Sınırları: 300% (motorun dayanıklılığına göre ayarlandı).
* Giriş Mantığı: Tüm değerler Do not invert (cihazlar standart mantıkta).

Bu ayarlar, motorun güvenli çalışmasını ve fiziksel cihazlarla uyumlu olmasını sağlar. Özellikle acil durum yönetimi ve hata toleransı önemli olan uygulamalarda kritiktir.

Bu ayarlar, Omron Sysmac Studio'da 3 eksenli servo kontrol projesi oluştururken sistemin güvenliğini ve hassasiyetini optimize etmenizi sağlar. Her parametre, uygulamanızın gereksinimlerine göre özelleştirilmelidir.

Aşağıda Sysmac Studio'da "Limit Settings" penceresindeki ayarların ne işe yaradığını ve nasıl kullanılacağını açıklayacağım:

1. Yazılım Sınır Ayarları (Software Limits)

Bu bölümde, eksenin yazılım seviyesinde izin verilen maksimum ve minimum pozisyon sınırları tanımlanır. Fiziksel sınır anahtarlarına ek olarak bir koruma katmanı sunar.

A. Software Limits Seçenekleri

* Seçenekler:
  + Disabled (Devre Dışı): Yazılım sınırları etkin değildir.
  + Enabled (Etkin): Motor, belirtilen pozisyon sınırlarını aşamaz. Aşarsa durur.
  + Enabled with warning (Uyarıyla Etkin): Sınırları aşarsa alarm verir, ancak durmaz.

B. Pozitif/Negatif Yazılım Sınır Değerleri

* Positive software limit (Pozitif Yazılım Sınır):
  + Eksenin ilerleme yönünde izin verilen en büyük pozisyon (pulse cinsinden).
  + Örnek: 2147483647 pulse → Motor bu değeri aşamaz.
* Negative software limit (Negatif Yazılım Sınır):
  + Eksenin geriye doğru izin verilen en küçük pozisyon (pulse cinsinden).
  + Örnek: -2147483648 pulse → Motor bu değeri geçemez.

2. Takip Hatası Ayarları (Following Error)

Bu bölümde, motorun hedef pozisyona ulaşamaması durumundaki hatayı (following error) kontrol edebilirsiniz.

A. Takip Hatası Üzerinde Değer (Following Error Over Value)

* Amaç: Motorun hedef pozisyondan ne kadar uzak kaldığını belirler.
* Örnek: 10 pulse → Motor hedeften 10 pulse uzaklaşırsa sistem durur.

B. Takip Hatası Uyarı Değeri (Following Error Warning Value)

* Amaç: Motorun hedef pozisyondan belirli bir mesafede olduğuunda uyarı verir.
* Örnek: 5 pulse → Motor hedeften 5 pulse uzaklaşırsa alarm tetrelir.

3. Pratik Kullanım Önerileri

1. Yazılım Sınır Ayarları:
   * Fiziksel sınır anahtarlarının güvenilirliği dikkate alınmadığında veya ek koruma istendiğinde etkinleştirin.
   * Örnek: Motorun fiziksel olarak 100 mm gitmesi için 10.000 pulse olarak ayarlayın.
2. Takip Hatası Kontrolü:
   * Motorun sabitlenmesi (stall) veya mekanik bozulmalar tespit edilmesi için kullanılır.
   * Örnek: Following Error Over Value = 10 pulse → Motor 10 pulse üzerinde takip edemiyorsa durur.
3. Varsayılan Değerler:
   * Tüm değerler 0 ise機能は無効です。Gerekli toleranslar dikkate alınarak ayarlanmalıdır.

Örnek Senaryo

* Yazılım Sınır Ayarları:
  + Enabled seçili.
  + Positive Limit = 10.000 pulse , Negative Limit = -10.000 pulse .
  + Motor bu aralıkları aşarsa durur.
* Takip Hatası Ayarları:
  + Following Error Over Value = 5 pulse → Motor 5 pulse üzerinde takip edemiyorsa durur.
  + Following Error Warning Value = 3 pulse → Motor 3 pulse üzerinde takip edemiyorsa alarm verir.

Bu ayarlar, motorun çalışırken hem fiziksel sınırlarını koruyup hem de performansı optimize etmenizi sağlar. Özellikle hata toleransı ve güvenlik kritik olan uygulamalarda önemlidir.

Bu ayarlar, Omron Sysmac Studio'da 3 eksenli servo kontrol projesi oluştururken sistemin güvenliğini ve hassasiyetini artırmaya yardımcı olur. Her parametre, uygulamanızın gereksinimlerine göre özelleştirilmelidir.

Aşağıda Sysmac Studio'da "Homing Settings" penceresindeki ayarların ne işe yaradığını ve nasıl kullanılacağını açıklayacağım:

1. Anahtar Bölümler ve İşlevleri

A. Homing Method (Ana Konumlama Yöntemi)

Bu bölümde, eksenin ana konuma dönmek için kullanılan yöntem ve parametreler tanımlanır.

* Homing method (Ana Konumlama Metodu):
  + Seçenekler:
    - Zero position preset (Sıfır Pozisyon Preset): Sistem, önceden belirtilen bir pozisyona (software'da ayarlanan Home offset) doğru gitmesini sağlar.
    - Use limit input (Sınır Girdisi Kullan): Fiziksel sınır anahtarlarını kullanarak ana konuma gelir.
    - Use Z-phase input (Z-faz Girdisi Kullan): Encoder'ın Z-faz sinyalini (index pulsu) tespit ederek ana konuma alır.
* Home input signal (Ana Konum Girdisi):
  + Ana konum sinyalinin nereden geleceği seçilir (örneğin, Z-faz, limit anahtarı).
* Homing start direction (Ana Konumlama Başlangıç Yönü):
  + Eksenin ana konuma gitmek için hangi yönde hareket edeceğini belirtir (pozitif veya negatif).
* Home input detection direction (Ana Konum Girdisi Tespiti Yönü):
  + Sınır anahtarı veya Z-faz sinyalinin hangi yönde algılanacağını ayarlar.

B. Senaryo Görselleştirme

Ortada yer alan grafik, aşağıdaki sinyallerin zamanla nasıl değişeğini gösterir:

* Home proximity signal (Ana Konum Yakınlık Sinyali): Eksenin ana konuma yaklaştığında aktive olan sinyal.
* Z-phase input (Z-faz Girdisi): Encoder'ın index pulsu (her tam devirde bir kez verir).
* Positive/Negative limit input (Pozitif/Negatif Sınır Girdisi): Fiziksel limit anahtarlarının durumunu gösterir.

Bu grafik, sistemın ana konum sinyallerini nasıl algıladığını test etmek için kullanılır.

C. Hız/Aksilasyon/Değişim Hızı Ayarları

* Homing velocity (Ana Konumlama Hızı):
  + Ana konuma gitme işlemi sırasında motorun hızı (pulse/s).
  + Örnek: 10.000 pulse/s → Motor saniyede 10.000 pulse ile ilerler.
* Homing acceleration (Ana Konumlama İvme):
  + Ana konuma gitme hızının artışı için izin verilen maksimum ivme (pulse/s²).
* Homing jerk (Ana Konumlama Sertlik):
  + İvmenin değişimi için izin verilen maksimum değeri kontrol eder (pulse/s³).
* Homing approach velocity (Ana Konum Yaklaşma Hızı):
  + Ana konuma yakınlaştığında daha yavaşa geçiş yapmak için kullanılan hedef hız.
* Homing deceleration (Ana Konumlama Azalış Hızı):
  + Ana konuma ulaşıldığında motorun ne kadar hızlı duracağı ayarlanır.

D. Diğer Parametreler

* Home input mask distance (Ana Konum Girdisi Maskesi Mesafesi):
  + Sınır anahtarının ne kadar yakından tespit edileceği (pulse cinsinden).
  + Örnek: 10.000 pulse → Sınır anahtarı 10.000 pulse mesafesinde algılanır.
* Homing holding time (Ana Konum Bekleme Süresi):
  + Ana konuma ulaşıldıktan sonra motorun ne kadar süre bekleyeceği (ms).
* Homing compensation value (Ana Konum Düzeltme Değeri):
  + Ana konumdan sonra motorun geriye kaydırılması için kullanılan değer (pulse cinsinden).
* Homing compensation velocity (Ana Konum Düzeltme Hızı):
  + Düzeltme işlemi sırasında motorun hızı (pulse/s).

2. Pratik Kullanım Önerileri

1. Encoder Tabanlı Ana Konumlama:
   * Homing method = Use Z-phase input , Home input signal = Z-phase input olarak ayarlanır.
   * Motor, encoder'ın Z-faz sinyalini algılayarak ana konuma gelir.
2. Fiziksel Sınır Anahtarı Kullanımı:
   * Homing method = Use limit input , Home input signal = Positive limit input olarak ayarlanır.
   * Motor, pozitif yönde ilerleyerek sınır anahtarına çarptığında ana konuma alır.
3. Yaklaşım Hızı Kontrolü:
   * Ana konuma yakınlaştığında (Homing approach velocity) hızı düşürerek daha güvenli çalışmayı sağlar.
4. Maskeli Sınır Algılama:
   * Home input mask distance ile fiziksel anahtarların gürültüye karşı dayanıklılığını artırabilirsiniz.

3. Örnek Senaryo

* Encoder ile Ana Konumlama:
  + Homing method: Use Z-phase input.
  + Home input signal: Z-phase input.
  + Homing start direction: Positive direction.
  + Homing velocity: 5.000 pulse/s.
  + Homing approach velocity: 500 pulse/s.

Motor, Z-faz sinyalini araya katarak pozitif yönde ilerleyerek ana konuma gelir. Yakınlaştığında hızı 500 pulse/s'a düşürür ve Z-faz sinyalini algıladıktan sonra durur.

4. Uyarılar

* Encoder Yoksun Sistemlerde: Z-faz kullanılamazsa, fiziksel sınır anahtarları tercih edilmelidir.
* Hız ve İvme Sınırları: Motorun fiziksel kapasitesine göre ayarlanmalıdır. Aksi takdirde sobe veya bozulma riski doğar.
* Maskeli Mesafe: Sınır anahtarının yanlış tetiklenmesini önlemek için uygun bir değer seçilmelidir.

Bu ayarlar, Omron Sysmac Studio'da 3 eksenli servo kontrol projesi oluştururken eksenin güvenli ve doğru ana konuma gelmesini sağlar. Her parametre, uygulamanızın gereksinimlerine göre özelleştirilmelidir.

Aşağıda Sysmac Studio'da "Position Count Settings" penceresindeki ayarların ne işe yaradığını ve nasıl kullanılacağını açıklayacağım:

1. Anahtar Bölümler ve İşlevleri

A. Sayılama Modu (Count Mode)

* Seçenekler:
  + Linear mode (Doğrusal Mod): Eksenin pozisyonu sonsuz olarak artar/küçülür (örneğin, taşıyıcı band).
  + Rotary mode (Dönel Mod): Eksenin pozisyonu belirli bir aralıkta döngüsel olarak hesaplanır (örneğin, devir sayımı).
* Örnek:
  + Linear mode: Motor 10.000 pulse ilerlediğinde pozisyon +10.000 olur.
  + Rotary mode: Motor 10.000 pulse ile tam bir devir yaparsa pozisyon 0'a döner (modolo işlemiyle).

B. Modulo Maksimum/Minimum Pozisyon Ayarları

Bu değerler, eksenin pozisyonunun döngüsel olarak hesaplandığı Rotary mode için kritiktir.

* Modulo maximum position setting value (Maksimum Pozisyon Değeri):
  + Eksenin pozitif yönde en yüksek değeri (pulse cinsinden).
  + Örnek: 2147483647 pulse → Sistem, bu değeri aşarsa minimuma geri döner.
* Modulo minimum position setting value (Minimum Pozisyon Değeri):
  + Eksenin negatif yönde en düşük değeri (pulse cinsinden).
  + Örnek: -2147483648 pulse → Sistem, bu değeri altına inerse maksimuma atlar.
* Formül:

Actual Position = (Commanded Position) % (Max - Min) + Min

* Örnek: Max=100, Min=-100, Komut=150 → 150 % 200 = 150, Sonuç: 150 - 100 = 50.

C. Encoder Türü (Encoder Type)

* Seçenekler:
  + Incremental encoder (Artimetik Encoder):
    - Sadece hareket yönü ve miktarını verir (başlangıç noktasız).
    - Motorun yeniden başlatılması durumunda pozisyon kayb edilir.
  + Absolute encoder (Mutlak Encoder):
    - Her anki pozisyonu direk verir (başlangıç noktası saklanır).
    - Güç kesintisi sonrası da pozisyon bilgisi korunur.

2. Pratik Kullanım Önerileri

1. Doğrusal Hareket Sistemleri:
   * Linear mode seçilmelidir (örneğin, paketleme makineleri).
   * Modulo değerleri varsayılan (±2.147.483.648) bırakılabilir.
2. Dönel Hareket Sistemleri:
   * Rotary mode seçilmeli ve modolo sınırları tanımlanmalıdır (örneğin, 360° dönen bir taban).
   * Örnek:
     + Max=360 pulse (1 devir), Min=0 pulse.
     + Motor 400 pulse giderse, sonucu 400 % 360 = 40 pulse olur.
3. Encoder Seçimi:
   * Artimetik Encoder: Düşük maliyetli ve basit sistemlerde tercih edilir.
   * Mutlak Encoder: Güç kesintisi veya yeniden başlatma gerektiren uygulamalarda kullanılır.

3. Örnek Senaryo

* Sistem: Dönel bir tahta (1 devir = 360 pulse).
* Ayarlar:
  + Count mode: Rotary.
  + Modulo max: 360 pulse.
  + Modulo min: 0 pulse.
  + Encoder type: Incremental.

Motor 400 pulse ilerlediğinde, pozisyon 40 pulse olarak hesaplanır (360 modolo). Artimetik encoder kullanıldığından, motor yeniden başlatıldığında pozisyon sıfırlanır.

4. Uyarılar

* Rotary mode'da Modulo Sınırları:
  + Sınırların gerçek mekanik sınırlara uygun olduğundan emin olun. Aksi takdirde motor yanlış pozisyona gitmesine neden olur.
* Encoder Türü:
  + Artimetik encoder kullanılıyorsa, her başlatmadan sonra ana konuma almanız gerekir.

Bu ayarlar, Omron Sysmac Studio'da 3 eksenli servo kontrol projesi oluştururken eksenin pozisyonunu doğru şekilde hesaplamasını sağlar. Her parametre, uygulamanızın gereksinimlerine göre özelleştirilmelidir.

Aşağıda Sysmac Studio'da "Servo Drive Settings" penceresindeki ayarların ne işe yaradığını ve nasıl kullanılacağını açıklayacağım:

1. Modulo Pozisyon Ayarları (Modulo Position Settings)

Bu bölümde, servo sürücüsünün pozisyon değerlerini döngüsel olarak hesaplaması için maksimum ve minimum sınırlar tanımlanır.

* Modulo maximum position setting value (Maksimum Pozisyon Değeri):
  + Eksenin pozitif yönde en yüksek pozisyon değeri (pulse cinsinden).
  + Örnek: 2147483647 pulse → 32-bit imzalı bir tamsayı sınırı (en büyük正值).
* Modulo minimum position setting value (Minimum Pozisyon Değeri):
  + Eksenin negatif yönde en düşük pozisyon değeri (pulse cinsinden).
  + Örnek: -2147483648 pulse → 32-bit imzalı bir tamsayı sınırı (en küçük负值).

İşlev:

* Döngüsel Pozisyon Hesaplama:
  + Eğer eksen Rotary mode (dönel mod)da çalışıyorsa, pozisyon bu aralıklar içinde döngüsel olarak hesaplanır.
  + Formül

Actual Position = (Commanded Position) % (Max - Min) + Min

* Örnek: Max=360 pulse, Min=0 pulse, Komut=400 pulse → 400 % 360 = 40 pulse.

2. Detay Ayarları (Detailed Settings)

Bu bölümde, servo sürücüsünün davranışını kontrol eden ek parametreler bulunur.

A. PDS State Control Method (PDS Durum Kontrol Metodu)

* Seçenekler:
  + Switched on by Servo OFF (Servo Kapanışta Aktifleştir):
    - Servo kapatıldığında pozisyon verileri saklanır ve açıldığında yeniden yüklenir.
  + Switched on by Power ON (Güç Açıklıkta Aktifleştir):
    - Güç açıldığında pozisyon verileri kullanılır.
  + Always On (Her Zaman Aktif):
    - Sistem her zaman önceki pozisyon bilgisini korur.
* Amaç:
  + Encoder'ın mutlak (absolute) değilse, servo kapanış/kapandığında pozisyon kaybını önlemek için kullanılır.

B. Main Circuit Power Supply OFF Detection (Ana Devre Gücü Kapanış Tespiti)

* Seçenekler:
  + Detect (Tespit Et): Ana güçün kapanmasını algılar ve kritik verileri kaydeder.
  + Do not detect (Tespit Etme): Ana gücün kapanışına tepki vermez.
* Amaç:
  + Güç kesintisi durumunda veri kaybını önlemek veya sistem güvenliğini artırmak için kullanılır.

3. Pratik Kullanım Önerileri

1. Modulo Pozisyon Ayarları:
   * Dönel Sistemler:
     + Örnek: Bir devir 360 pulse olan taban.
     + Ayarlar: Max=360, Min=0.
   * Doğrusal Sistemler:
     + Varsayılan değerleri (±2.147.483.648) bırakın (sonsuz pozisyon).
2. PDS Kontrol Metodu:
   * Artimetik Encoder Kullanıyorsanız:
     + Switched on by Servo OFF seçin → Servo kapanışta pozisyonu saklar.
   * Mutlak Encoder Kullanıyorsanız:
     + Always On seçin → Herhangi bir kayp yoktur.
3. Ana Güç Kapanış Tespiti:
   * Güç Kesintisi Riski Varsa:
     + Detect seçin → Güç keseceği takdirde veriler kaydedilir.

4. Örnek Senaryo

* Sistem: Dönel bir paketleme makinesi (1 devir = 360 pulse).
* Ayarlar:
  + Modulo max/min: 360 / 0.
  + PDS state control method: Switched on by Servo OFF.
  + Power OFF detection: Detect.

Motor 400 pulse ilerlediğinde pozisyon 40 pulse olarak hesaplanır. Servo kapatılırsa pozisyon saklanır ve açıldığında tekrar 40 pulse'dan devam eder. Güç kesintisi olursa veriler kaydedilir.

5. Uyarılar

* Modulo Sınırları:
  + Gerçek mekanik sınırlara uygun olmalıdır. Aksi takdirde motor yanlış pozisyona gitmesine neden olur.
* PDS ve Güç Kaydı:
  + Artimetik encoder kullanıyorsanız, servoyu kapatmadan önce ana konuma almayı unutmayın.

Bu ayarlar, Omron Sysmac Studio'da 3 eksenli servo kontrol projesi oluştururken sistem güvenliğini ve doğru pozisyon hesaplamasını sağlar. Her parametre, uygulamanızın gereksinimlerine göre özelleştirilmelidir.