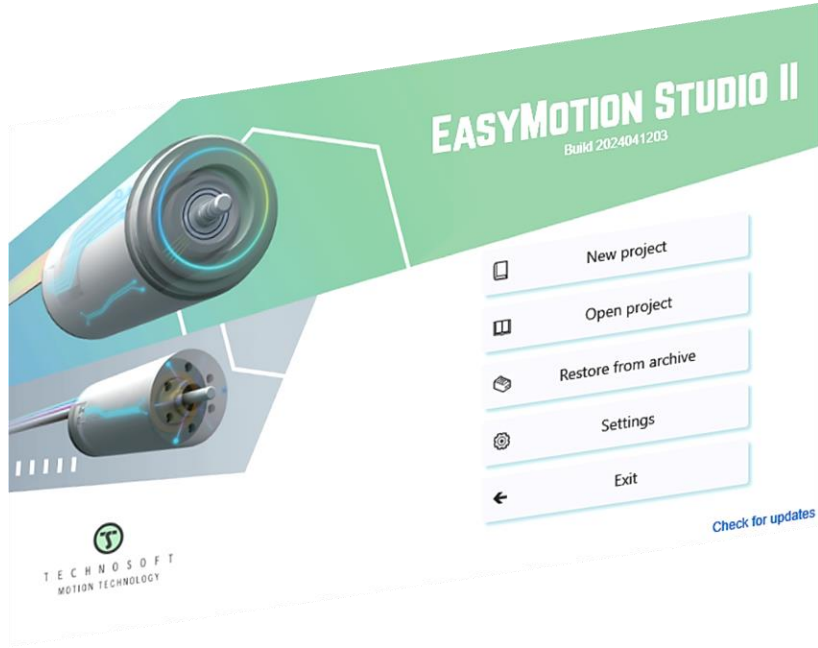


EasyMotion Studio II

Hızlı Kurulum ve
Programlama Kılavuzu



T E C H N O S O F T



Teknik Referans

İçindekiler	3
1 EasyMotion Studio II Hakkında	4
2 Başlangıç	4
3 Yeni Proje Oluştur	6
4 Motor ve Sürücü Ayarları	9
4.1 Mekanik Yapılandırma	10
4.2 Motor Ayarları	11
4.3 Geri Besleme Ayarları	12
4.4 Girişler / Çıkışlar	13
4.5 Kontrol Ayarları	14
4.6 External reference (Harici referans)	14
4.7 Uygulama Ayarları	16
4.8 Hareket Ayarları	18
4.9 Fieldbus Ayarları	19
4.10 Koruma ve Limitler.....	20
4.11 Kontrolörler.....	23
4.11.1.... Otomatik Ayarlama	23
4.11.2.... Manuel Ayarlama & Test	24
4.12 Setup'ı sürücüye yükle	25
5 Sürücü Programlama	26
6 Hareket Değerlendirme ve Uygulama Hata Ayıklama Araçları	29
6.1 Kontrol Panellerinin Kullanımı	29
6.2 Logger Kullanımı	30
6.3 Scope Kullanımı	32
6.4 Uygulama ve Sürücü Fonksiyonlarının Kontrolü veya Hata Ayıklaması için Gelişmiş Araçlar	33
7 Diğer – Proje Arşivleme ve Geri Yükleme	34

1 EasyMotion Studio II Hakkında

EasyMotion Studio II, Technosoft akıllı sürücü ve motorlarının ayarlanması ve hareket programlamasının gerçekleştirilmesi için gerekli yazılım paketidir. EasyMotion Studio II çıktısı; sürücü/motor EEPROM'una indirilebilen veya daha sonra kullanılmak üzere bilgisayara kaydedilebilen bir ayar veri seti (setup) ve bir hareket programından oluşur.

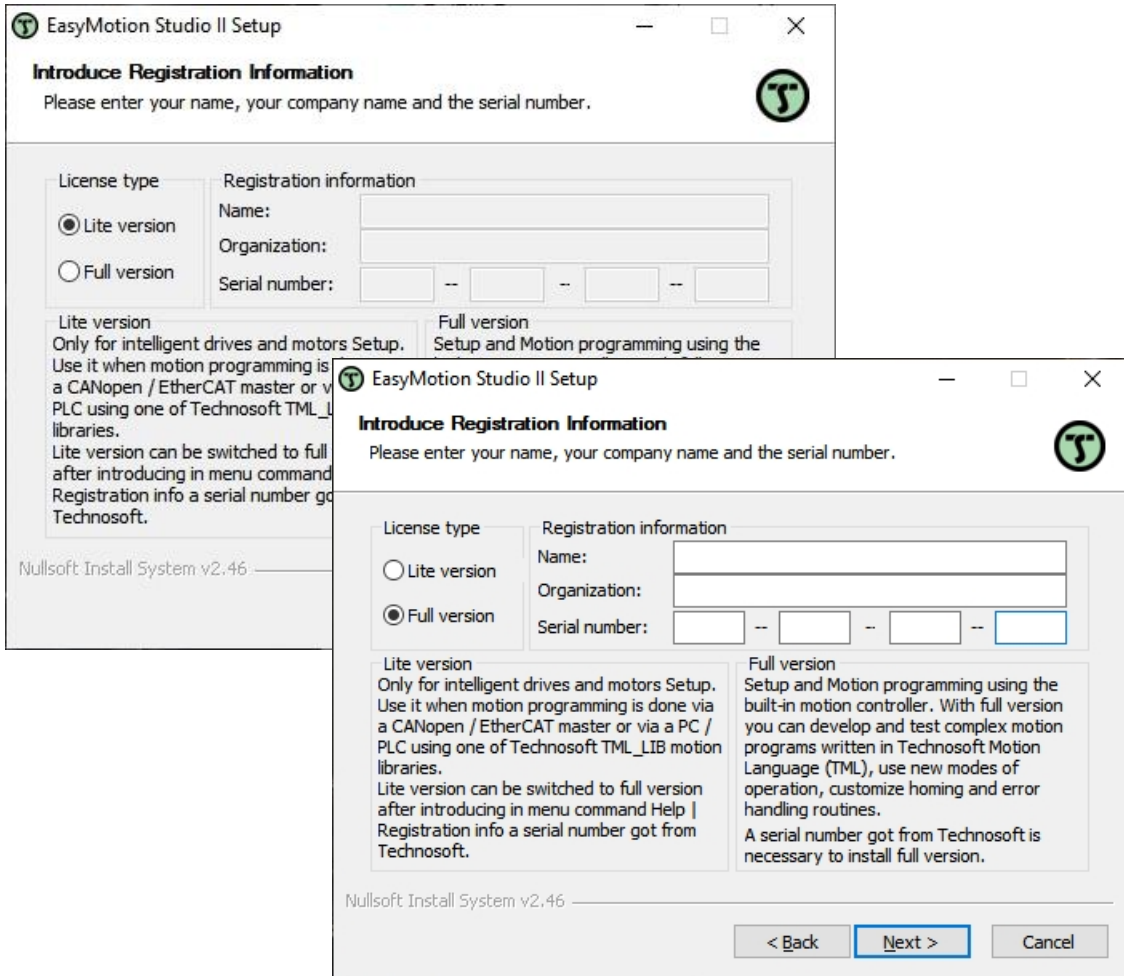
Ayar (setup), bir Technosoft sürücü/motorunun yapılandırılması ve kullanılması için gerekli tüm bilgileri içerir. Bu bilgiler, sürücü/motor EEPROM'unda bir setup tablosu olarak saklanır. Setup tablosu, güç verilmesi (power-on) veya reset sonrasında sürücü/motorun RAM belleğine kopyalanır ve çalışma sırasında (runtime) kullanılır.

Hareket (motion) bileşeni, kullanıcı/uygulama gereksinimlerinin gerçekleştirilmesini içerir. Bu yapı, sürücü/motor içerisinde yer alan dahili hareket kontrolörü tarafından çalıştırılan bir TML (Technosoft Motion Language) programı ile tanımlanır.

EasyMotion Studio II, hareket uygulamasını hızlı bir şekilde test etmek, ölçmek ve analiz etmek için Data Logger, Kontrol Panelleri ve Komut Yorumlayıcı (Command Interpreter) gibi bir dizi değerlendirme aracını da içerir.

2 Başlangıç

EasyMotion Studio II, Technosoft web sitesinden indirilebilir ve Lite ([link](#)) sürüm (ücretsiz) veya Full sürüm (seri numarası gerektirir) olarak kurulabilir.



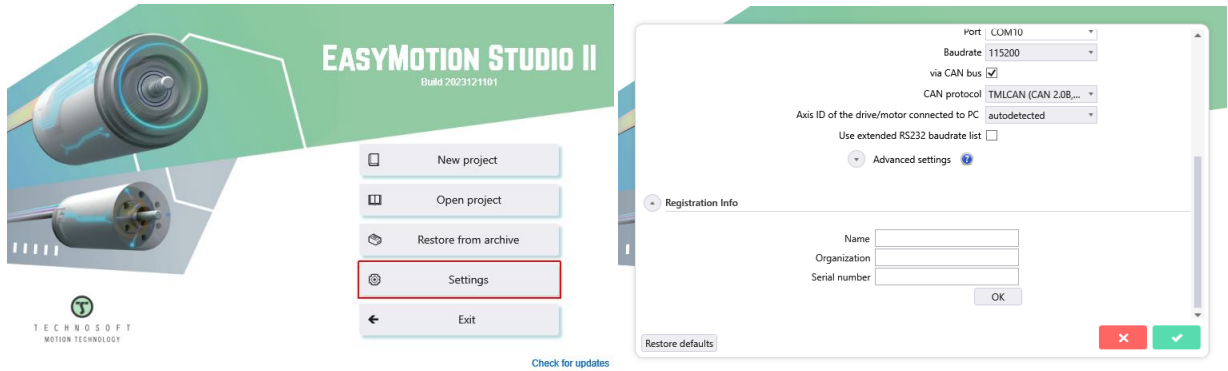
Şekil 2.1 – EasyMotion Studio II – Sürüm Seçimi

Lite sürüm, sürücü/motor devreye alma işlemlerinin gerçekleştirilmesine olanak tanır ve uygulamayı değerlendirmek veya gerektiğinde hata ayıklamak için gerekli araçları (Data Logger, Scope, Kontrol Panelleri vb.) içerir.

Bu sürüm, Technosoft sürücü/motorlarının farklı ağlarda (ör. CANopen ve EtherCAT) basit slave (alt cihaz) olarak kullanıldığı projeler için tasarlanmıştır.

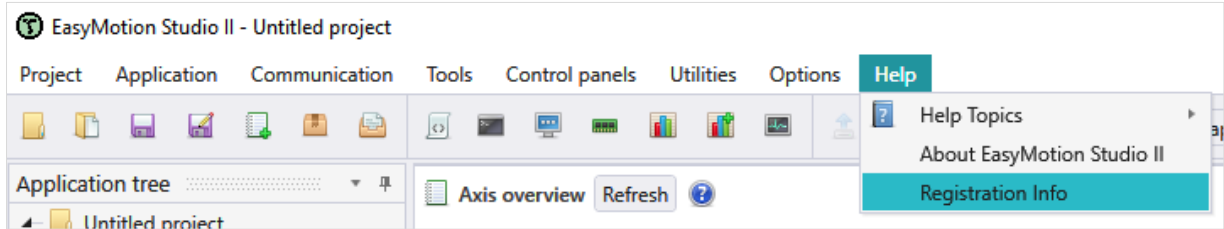
Full sürüm, TML programının sürücüye yüklenmesini mümkün kılar. Bu sayede kullanıcı, sürücü içerisinde bulunan dahili hareket kontrolöründen tam olarak faydalanabilir ve sürücü seviyesinde karmaşık uygulamalar gerçekleştirebilir veya kontrol yapısını master ile sürücü arasında paylaşabilir.

Full sürüm, EasyMotion Studio II açılış ekranından erişilebilen **Ayarlar (Settings)** penceresinde yer alan **kayıt bilgileri (Registration info)** girilerek herhangi bir zamanda aktif hale getirilebilir.



Şekil 2.2 – Ayarlar Penceresinde Kayıt Bilgileri

Registration info (Kayıt Bilgileri) penceresine, proje penceresindeki **Yardım (Help)** menüsü üzerinden de erişilebilir.



Şekil 2.3 – Help menüsü üzerinden Registration info penceresine erişim

EasyMotion Studio II kurulduktan sonra, Technosoft sürücülerinin çalıştırılması için 3 temel adım bulunmaktadır:

Adım 1: Donanım Kurulumu

Donanım kurulumu, her bir sürücü kullanıcı kılavuzunda detaylı olarak açıklanmaktadır.

Adım 2: Sürücü Devreye Alma

Sürücü devreye alma işlemi bu dokümanda detaylı olarak açıklanacaktır.

Adım 3: Hareket Programlama

Programlama, aşağıdaki seçeneklerden biri kullanılarak gerçekleştirilebilir:

- TML kullanılarak EasyMotion Studio II üzerinden;
- Bir EtherCAT master (sürücü modeline bağlı olarak);
- Bir CANopen master;
- PC'ler (Windows veya Linux) için TML_LIB hareket kütüphanesi
- PLC'ler için TML_LIB hareket kütüphanesi
- Yukarıdaki seçenekleri birleştiren dağıtık kontrol yaklaşımı (örneğin, bir host'un sürücüler üzerinde TML ile programlanmış hareket fonksiyonlarını çağırması gibi)

Bu doküman, EasyMotion Studio II ve Technosoft Motion Language (TML) kullanılarak sürücü programlamasını detaylı olarak açıklamaktadır.

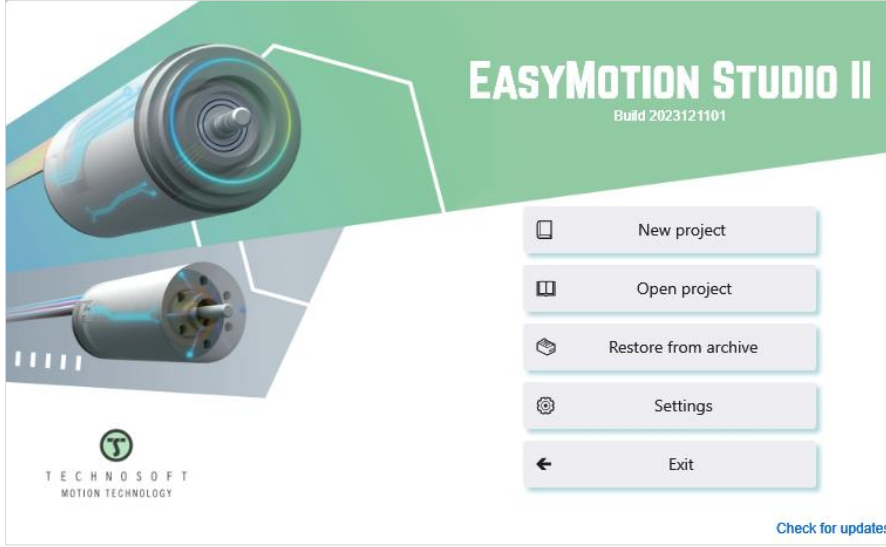
Diğer programlama seçenekleri, aşağıdaki kullanıcı kılavuzlarında detaylı olarak açıklanmaktadır:

- **CANopen Programlama Kılavuzu** – CANopen sürücülerinin programlanmasını açıklar;
- **CoE Programlama Kullanıcı Kılavuzu** – EtherCAT sürücülerinin programlanmasını açıklar;
- **TML_LIB Kullanıcı Kılavuzları** – PC / PLC kütüphaneleri kullanılarak proje geliştirmeyi açıklar.

Yukarıda belirtilen kılavuzlar web sitemizden indirilebilir. ([link](#)).

3 Yeni Proje Oluştur

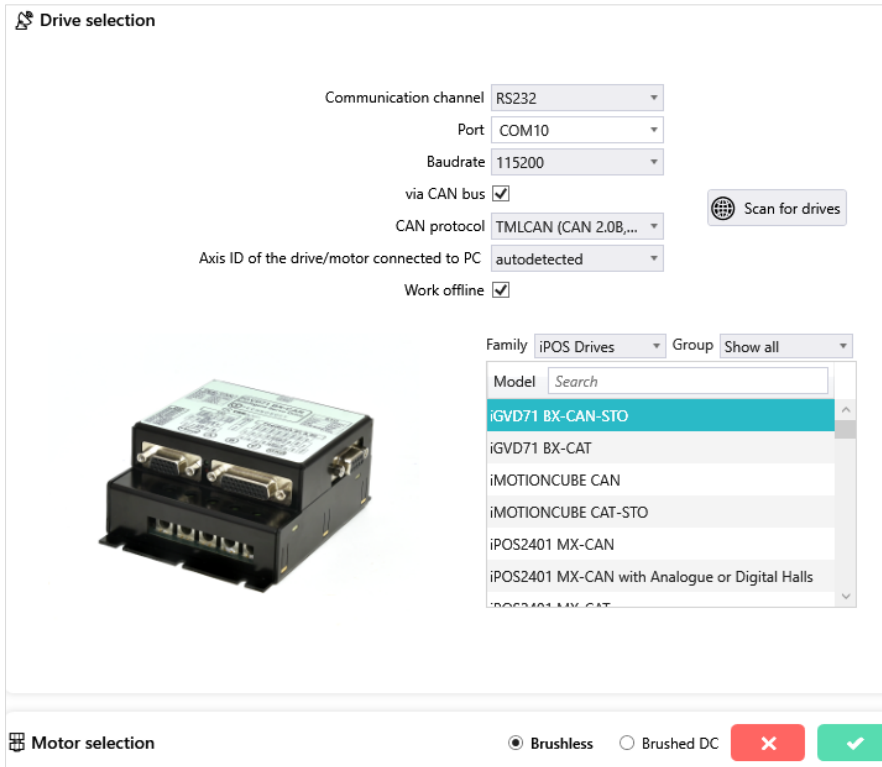
EasyMotion Studio II açılış ekranı, yeni bir proje oluşturmayı, daha önce oluşturulmuş bir projeyi açmayı, bir proje arşivini geri yüklemeyi veya genel ayarları değiştirmeyi (çalışma alanı konumunu belirleme, yazılım dilini değiştirme, global iletişim parametrelerini ayarlama ve registration info bilgisini ekleme/düzenleme) sağlar.



Şekil 3.1 – EasyMotion Studio II Açılış Ekranı

Yeni bir proje oluşturmak için, “**New project**” butonuna tıklamanız yeterlidir. Bu işlem, sürücü modelini ve motor teknolojisini seçmenize olanak tanıyan bir sihirbaz (wizard) penceresini açacaktır.

Sürücü seçimi, herhangi bir sürücü bağlı değilse önceden tanımlı bir listeden veya “**Scan for drives**” butonuna tıklandıktan sonra görüntülenecek olan tespit edilmiş sürücüler tablosundan yapılabilir.



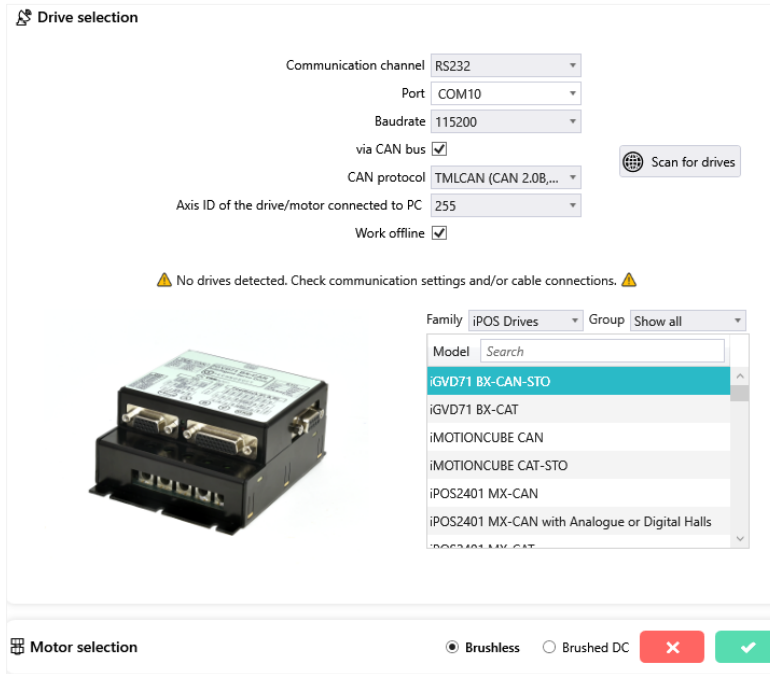
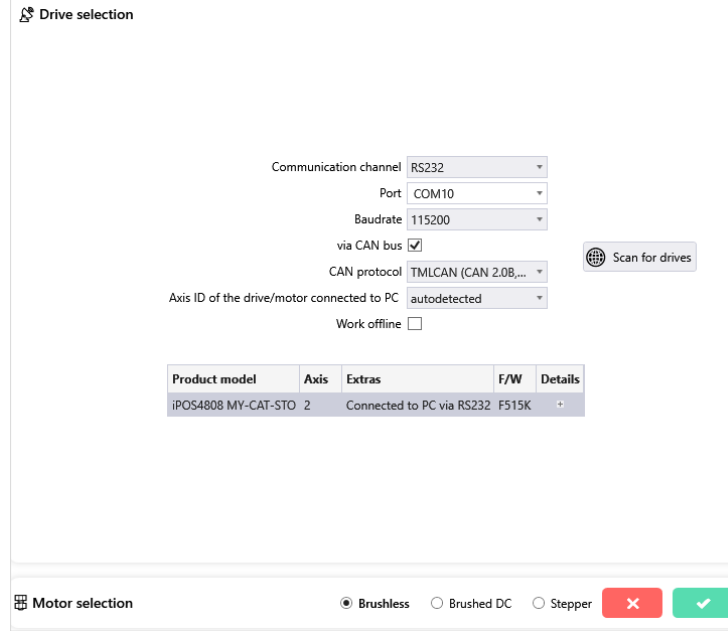
Şekil 3.2 – Sürücü ve Motor Seçim Sihirbazı

“**Scan for drives**” seçeneğinin kullanılabilmesi için iletişim parametrelerinin doğru şekilde ayarlanmış olması gerekir. Bu nedenle, **Sürücü Seçimi (Drive selection)** bölümünde parametrelerin ayarlanmasına veya düzenlenmesine olanak tanıyan bir iletişim ayarları bölümü görüntülenir.

Sürücülerin PC'ye bağlantısı; RS232, CAN, USB, Ethernet veya EtherCAT üzerinden Ethernet (EoE) aracılığıyla aşağıdaki şekillerde gerçekleştirilebilir:

1. RS232 bağlantısı kullanılarak, doğrudan PC'ye veya bir USB-RS232 dönüştürücü aracılığıyla;
2. USB bağlantısı kullanılarak;
3. Technosoft Ethernet-RS232 veya CAN adaptörü kullanılarak;
4. CAN-bus bağlantısı kullanılarak, bir PC-CAN arayüzü üzerinden doğrudan PC'ye bağlanarak;
5. EoE protokolü kullanılarak (yalnızca EtherCAT sürücülerinde mevcuttur);
6. Aynı CAN-bus ağı üzerindeki başka bir sürücü aracılığıyla, bu sürücünün doğrudan PC'ye bağlı olması durumunda.

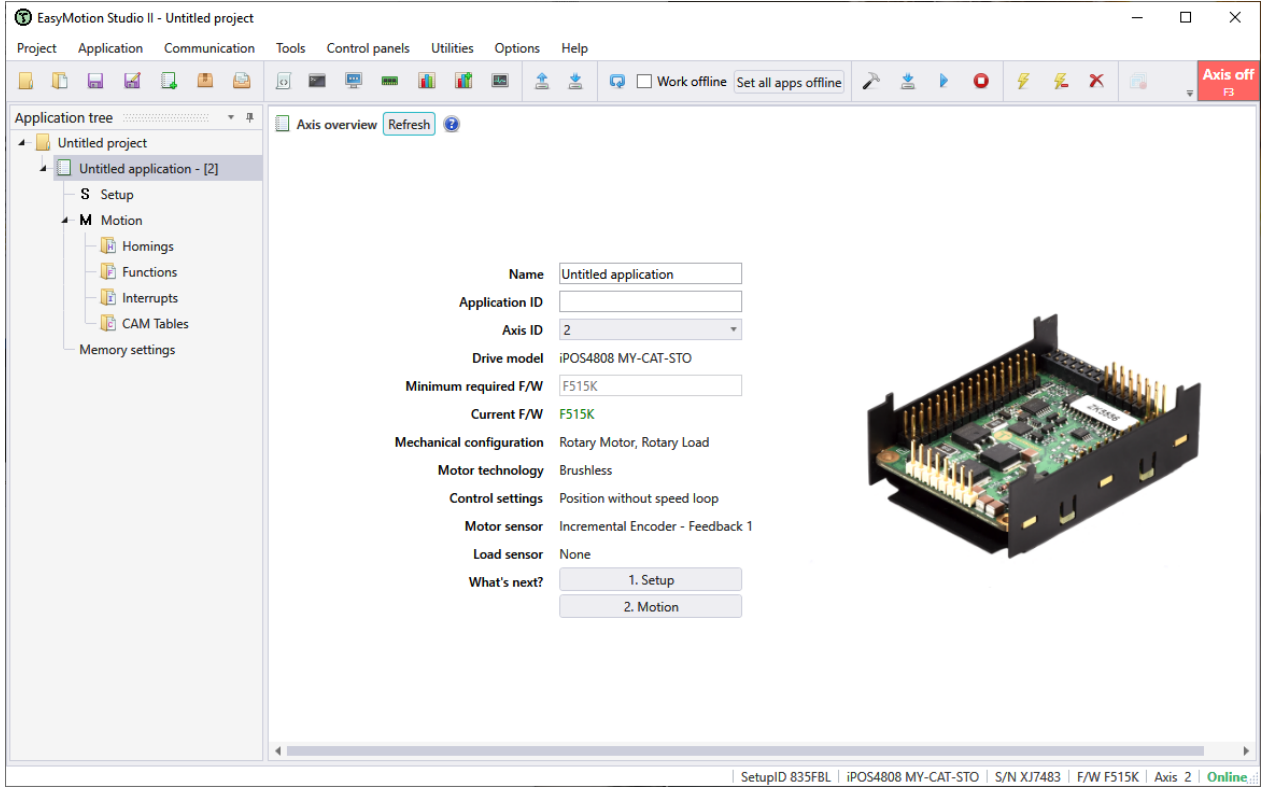
“Scan for drives” işlemi tamamlandığında, EasyMotion Studio II tespit edilen sürücüleri listeler veya herhangi bir sürücü bulunamazsa bir hata mesajı görüntüler.



Şekil 3.2 – Scan for drives sonuçları

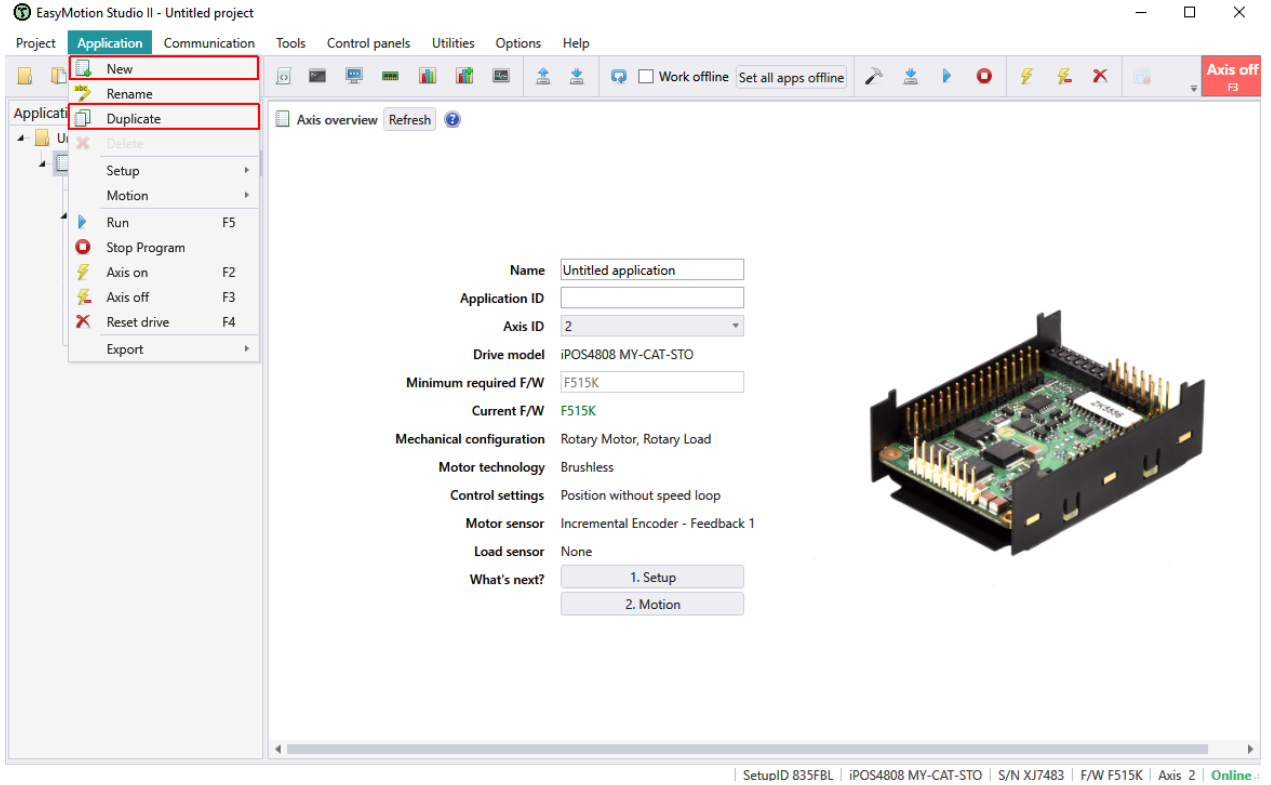
“Scan for drives” seçeneği hata verirse ve PC'ye bağlı bir veya daha fazla sürücü bulunuyorsa; kabloları ve güç kaynağını kontrol edin, iletişim ayarlarının doğru olduğundan emin olun ve işletim sisteminin kullanılan adaptörü tanıyıp tanımadığını ve gerekli sürücülere sahip olup olmadığını kontrol edin.

Sürücü ve motor teknolojisi seçildikten sonra, Motor selection bölümündeki **yeşil onay (tick)** butonuna tıklanarak yeni proje başlatılabilir. Bu işlem sihribazı sonlandırır ve proje penceresini açar.



Şekil 3.3 – Proje Penceresine Genel Bakış

Yeni bir proje başlatıldığında, EasyMotion Studio II otomatik olarak ilk uygulamayı oluşturur. Daha sonra **Application** menüsündeki “New” veya “Duplicate” seçenekleri kullanılarak ek uygulamalar eklenebilir.

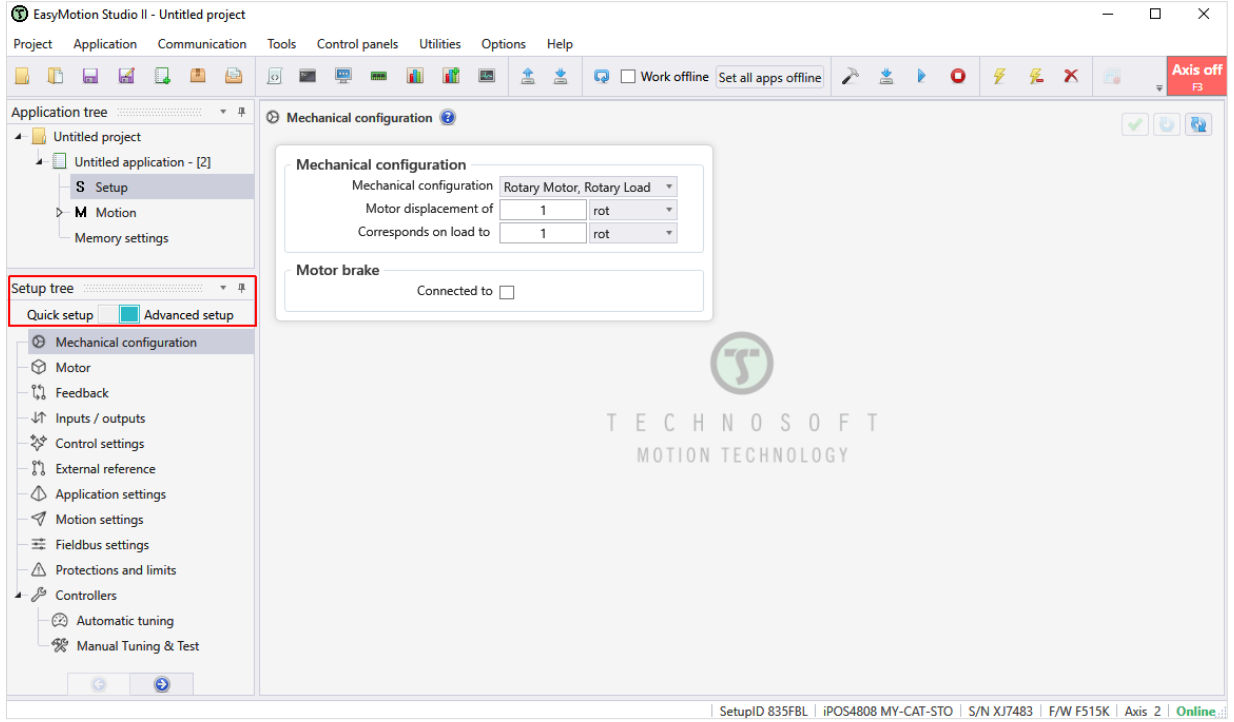


Şekil 3.5 – Yeni bir uygulama ekleme veya mevcut olanı çoğaltma

Her uygulama 3 ana bölüm içerir: **Setup** (sürücü/motor ayarları için), **Motion** (uygulama geliştirme için) ve **Memory Settings** (bellek kullanımına genel bakış ve ayarlar için). Setup ve Motion bölümleri, sonraki bölümlerde detaylı olarak açıklanacaktır.

4 Motor ve Sürücü Ayarları

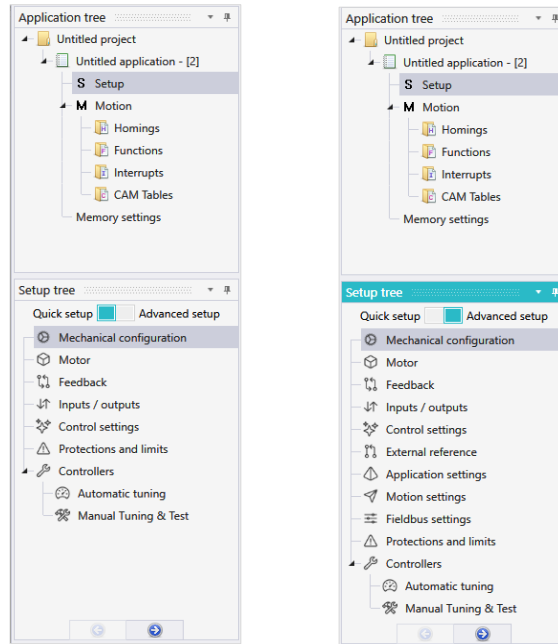
Setup bölümü, sürücü devreye alma işleminin **Quick setup** veya **Advanced setup** yapısı kullanılarak gerçekleştirilmesine olanak tanır. İstenen seçenek, **Setup tree** bölümünün üst kısmında bulunan radyo butonu (radio button) aracılığıyla seçilebilir.



Şekil 4.1 – Setup tree seçenekleri

Quick setup modu, tüm bağlantılar önceden doğrulanmış ve motor ile geri besleme parametreleri iyi biliniyorsa, sürücünün hızlı bir şekilde devreye alınmasına olanak tanır.

Yeni bir proje başlatıldığında, motor, geri besleme ve sürücü bağlantılarının yanı sıra sistemin işlevselliğini doğrulamak için çeşitli testler yapılmasına olanak tanıdığı için **Advanced setup** modunun kullanılması önerilir.



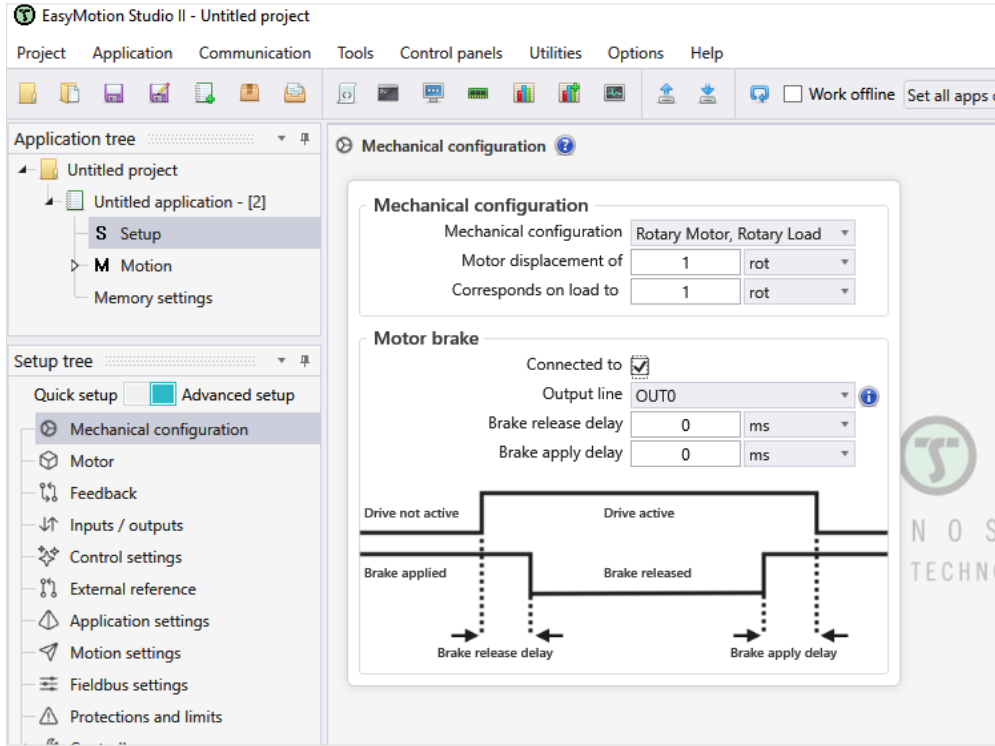
Şekil 4.2 – Quick Setup ve Advanced setup karşılaştırması

Seçilen setup yapısından (quick veya advanced) bağımsız olarak, sürücü devreye alma süreci; Setup tree içindeki tüm bölümlerin yukarıdan aşağıya doğru tek tek kontrol edilmesini, gerekli parametrelerin ayarlanmasını ve uygulamaya ait seçeneklerin etkinleştirilmesini/devre dışı bırakılmasını içerir.

4.1 Mekanik Yapılandırma

Hem *Quick setup* hem de *Advanced setup* modlarında bulunan **Mechanical configuration** bölümü, motor ile yük arasındaki iletim tipinin ve oranının tanımlanmasına olanak tanır. Mevcut seçenekler şunlardır:

- **Rotary Motor, Rotary Load** – döner bir motorun doğrudan döner bir yüke bağlandığı veya bağlantının bir redüktör (gearbox), kasnak (pulley) ya da başka bir dönerden dönere iletim mekanizması üzerinden yapıldığı durumları tanımlar;
- **Rotary Motor, Linear Load** – döner motor hareketinin bir kayış (belt), bilyalı vida (ball-screw), kremayer–pinyon sistemi (rack and pinion) veya başka bir dönerden lineere iletim mekanizması aracılığıyla lineer yük hareketine dönüştürüldüğü sistemleri tanımlamak için kullanılır;
- **Linear Motor, Linear Load** – voice coil veya fırçasız lineer motor kullanılan durumları tanımlar.



Şekil 4.3 – Mechanical configuration

Motor bir elektro-mekanik fren ile donatılmışsa, sürücü bu freni kontrol edecek şekilde yapılandırılabilir. İlgili ayarlar, **Mechanical configuration** bölümündeki **Motor brake** sekmesinde bulunur ve “**Connected to**” onay kutusu seçildiğinde görünür hale gelir.

Sürücü, bu amaç için özel olarak tasarlanmış dijital çıkışlarından biri (genellikle “OUT0” veya “OUT1”) aracılığıyla freni kontrol eder (bu çıkışlar, standart dijital çıkışlara göre daha yüksek akım sağlayabilir) ve çıkış seviyesini güç kademesinin durumuna bağlı olarak otomatik şekilde ayarlar:

- Sürücünün güç kademesi devre dışı bırakıldığında, çıkış pasif seviyeye ayarlanır; bu durum frenin devreye girmesine neden olur ve yük hareketini engeller.;
- Sürücünün güç kademesi etkinleştirildiğinde (motora güç verildiğinde), çıkış durumu aktif seviyeye getirilir; bu da frenin serbest bırakılmasını sağlar.

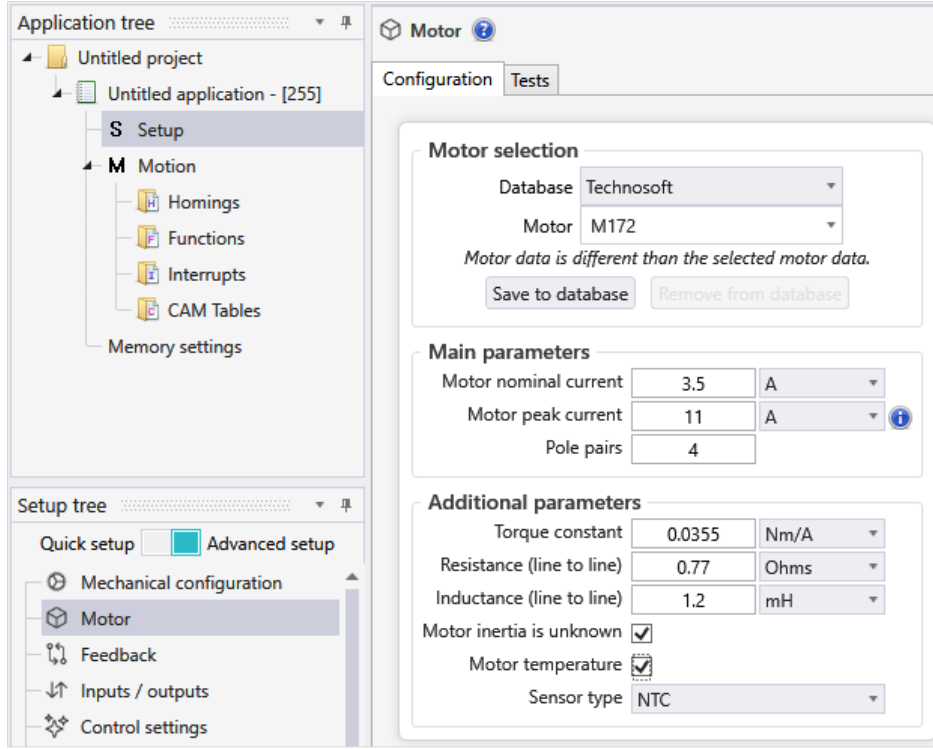
Birçok uygulamada frenin devreye girme ve serbest bırakma gecikmesi gerektiğinden, **Motor brake** bölümü bu parametrelerin ayarlanmasına olanak tanıyan iki alan içerir:

- **Brake release delay (fren bırakma gecikmesi)**, sürücünün güç kademesinin etkinleştirildiği (motora enerji verildiği) andan, fren bırakma komutunun verildiği ana kadar geçen süreyi ifade eder;
- **Brake apply delay (fren uygulama gecikmesi)**, frene enerji verildiği andan, sürücünün güç kademesinin devre dışı bırakıldığı (motora güç verilmediği) ana kadar geçen süreyi ifade eder.

Fren bağlantıları çoğu durumda doğrudan sürücüye yapılabilir. Bağlantı şeması ve diğer önemli detaylar, her bir sürücünün kullanıcı kılavuzunda sunulmaktadır.

4.2 Motor Ayarları

Motor parametreleme sayfası, **Configuration** ve **Tests** olmak üzere iki sekme içerir (Tests sekmesi yalnızca **Advanced setup** modunda kullanılabilir).

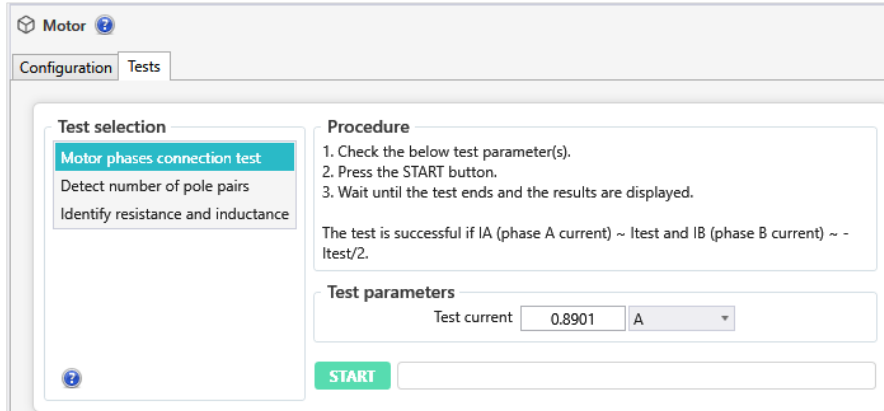


Şekil 4.4 – Motor setup sayfası

Configuration sekmesi, motorun parametrelenmesine olanak tanır ve aşağıdaki grup kutularını içerir:

- **Motor selection** – motor parametrelerinin önceden tanımlanmış motor veritabanından yüklenmesine veya mevcut parametrelerin kullanıcı motor veritabanına kaydedilmesine olanak tanır.
- **Main parameters** – motorun nominal ve tepe akım değerlerinin yanı sıra motor kutup çifti sayısının ayarlanmasına olanak tanır. Bu değerlerin tamamı motor veri sayfasında yer almalıdır.
- **Additional parameters** – motor tork sabiti, faz direnci, faz endüktansı, rotor ataleti ve (varsa) motor sıcaklık sensörü tipinin ayarlanmasına olanak tanır. Bu bilgiler de motor veri sayfasında bulunmalıdır. Bununla birlikte, bu parametreler zorunlu değildir; çünkü bunların belirlenmesi için bazı otomatik veya manuel testler kullanılabilir.

Tests sekmesi, motor faz bağlantılarının test edilmesine, motor kutup çifti sayısının tespit edilmesine ve motor sarğı direnci ile endüktansının belirlenmesine olanak tanır.

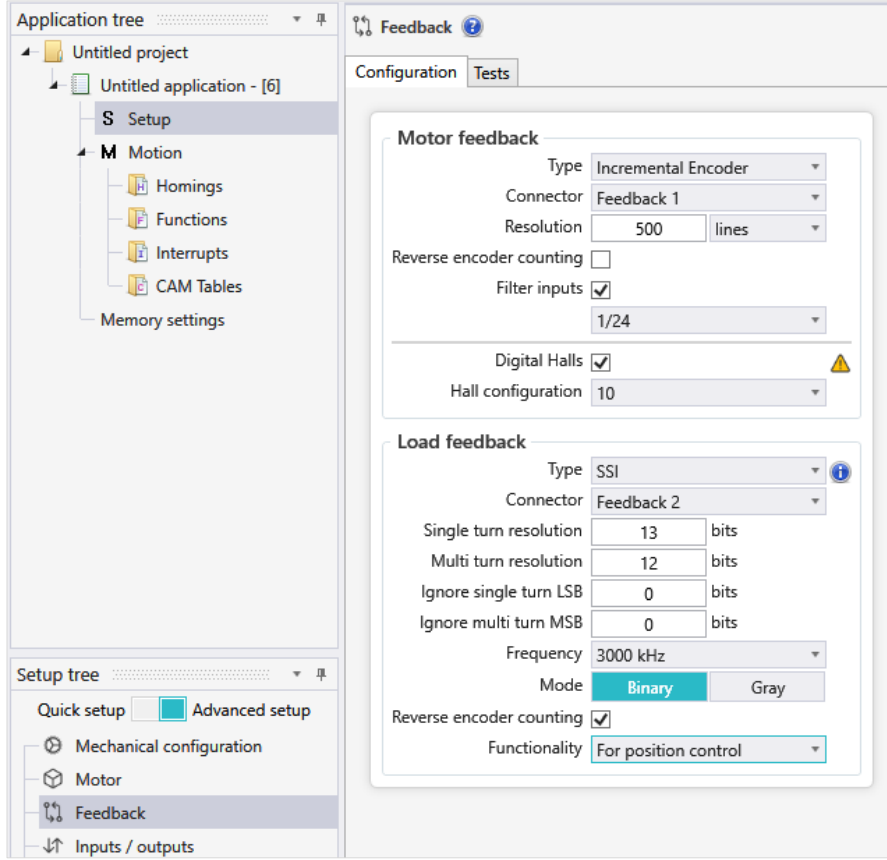


Şekil 4.5 – Tests sekmesi

Yukarıda belirtilen testlerin ayrıntılı açıklaması, EasyMotion Studio II içerisindeki Help menüsünden erişilebilen **Help** topics bölümünde veya test diyaloglarındaki “?” simgesine tıklanarak bulunabilir.

4.3 Geri Besleme Ayarları

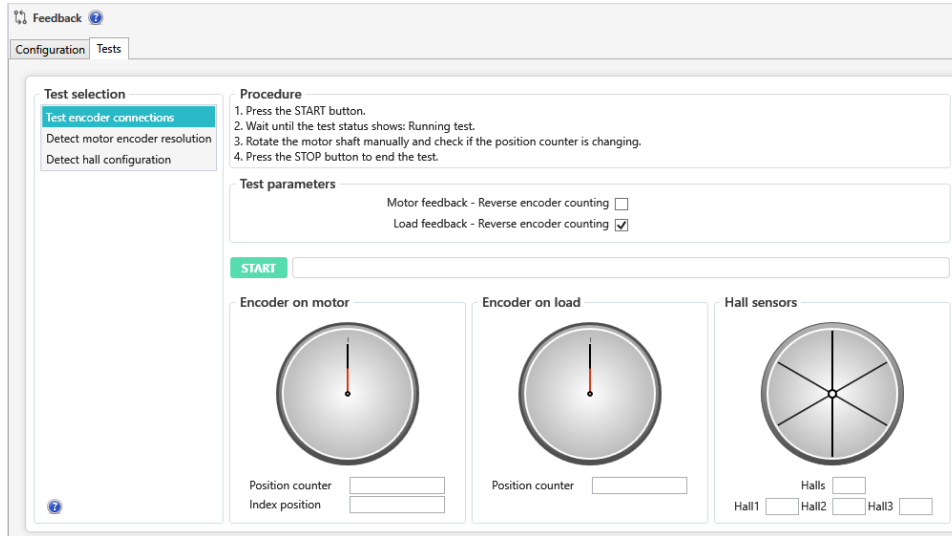
Motor ve yük geri besleme ayarları sayfasına, **Feedback** bölümü üzerinden erişilebilir ve **Configuration** ile **Tests** olmak üzere iki sekme içerir (Tests sekmesi yalnızca **Advanced setup** modunda kullanılabilir).



Şekil 4.6 – Feedback setup bölümü

Configuration sekmesi, kullanılan geri besleme sensörlerinin parametrelenmesine olanak tanır; bu nedenle içeriği, seçilen geri besleme türüne bağlı olarak değişir. Gerekli parametreler sensör veri sayfasında bulunmalıdır.

Tests sekmesi, geri besleme sensörlerinin bağlantısının test edilmesine ve çözünürlük ile diğer önemli sensör parametrelerinin tespit edilmesine olanak tanır.

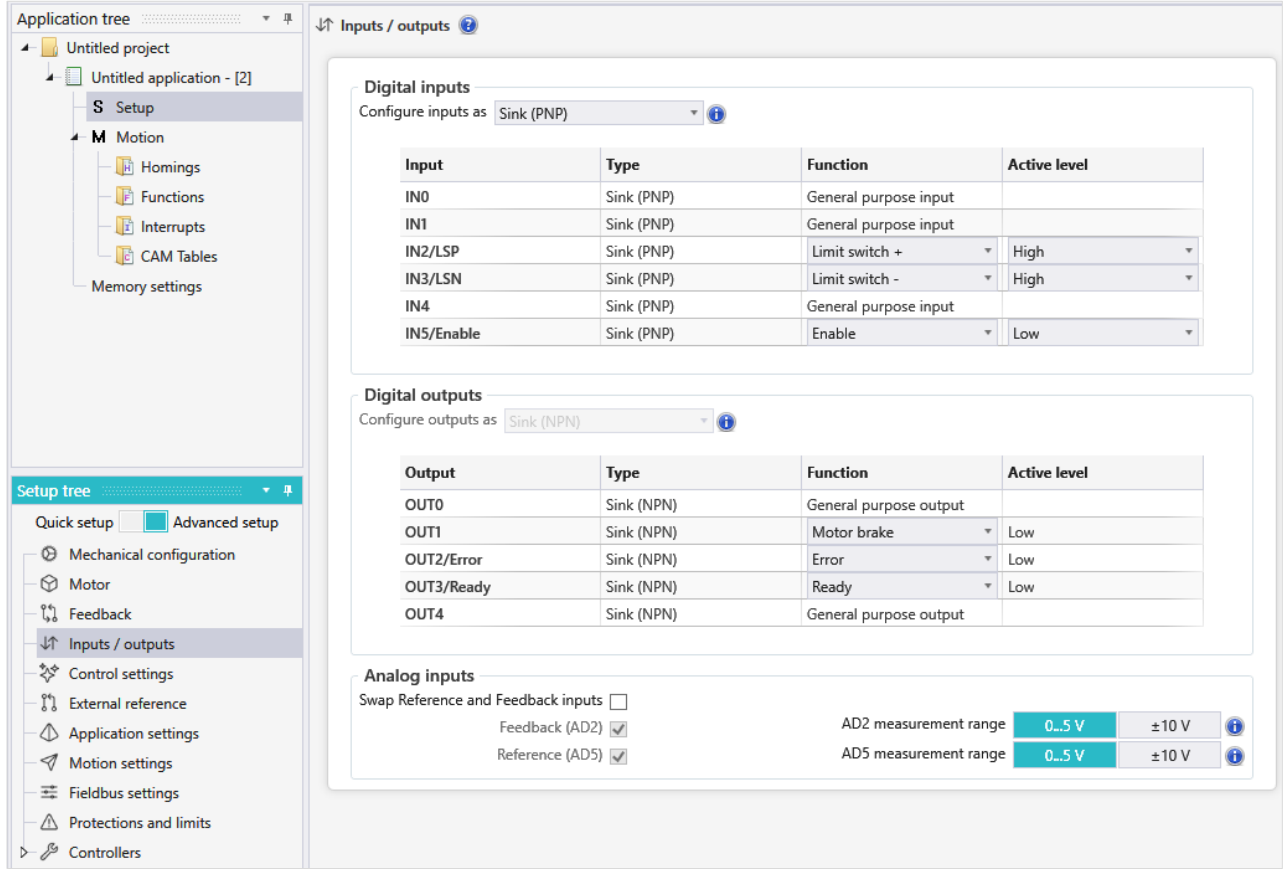


Şekil 4.7 – Tests sekmesi

Configuration ve **Tests** sekmelerindeki parametreler, EasyMotion Studio II içerisindeki **Help** menüsünden erişilebilen Help topics bölümünde veya **Feedback** sayfası ile test diyaloglarındaki “?” simgesine tıklanarak açılabilen yardım konularında detaylı olarak açıklanmaktadır.

4.4 Girişler / Çıkışlar

Sürücünün **Inputs / Outputs** parametrelemesi (tip, fonksiyon ve aktif seviye), Inputs / Outputs sayfası üzerinden kolayca yapılabilir.



Şekil 4.8 – Inputs / Outputs setup sayfası

Varsayılan olarak tüm Technosoft sürücülerini ve akıllı motorlar, genel amaçlı giriş/çıkışların yanı sıra özel işlevlere sahip bazı giriş/çıkışlara da sahiptir:

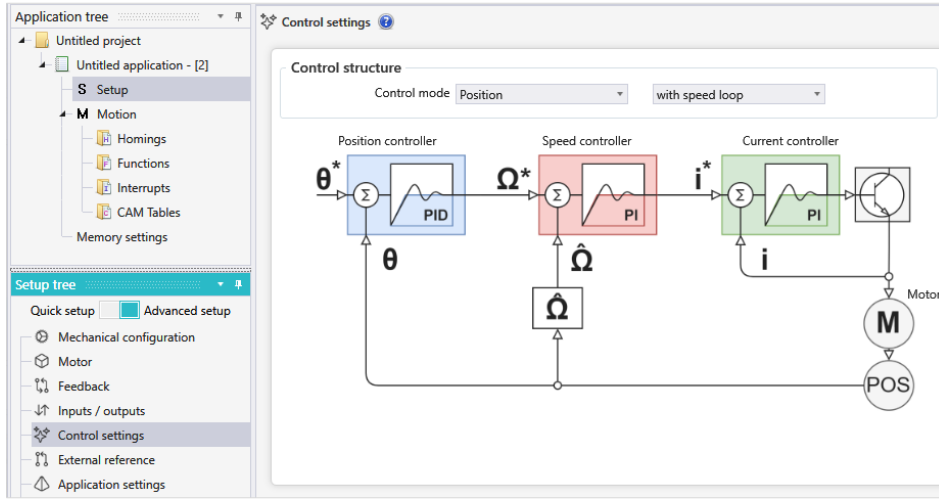
- **IN2 / LSN**
 - Tip: dijital giriş;
 - Varsayılan işlev: aktif olduğunda negatif yöndeki yük hareketlerini engeller.
- **IN3 / LSP**
 - Tip: dijital giriş;
 - Varsayılan işlev: aktif olduğunda pozitif yöndeki yük hareketlerini engeller.
- **IN4 / ENABLE**
 - Tip: dijital giriş;
 - Varsayılan işlev: pasif olduğunda sürücünün güç kademesini devre dışı bırakır.
- **OUT2 / ERROR**
 - Tip: dijital çıkış;
 - Varsayılan işlev: sürücüde bir hata olduğunu bildirir.
- **OUT3 / READY**
 - Tip: dijital çıkış;
 - Varsayılan işlev: sürücünün çalışır durumda olduğunu bildirir.

Giriş/çıkışların özel işlevleri, ilgili giriş/çıkışın **Function** parametresi "**General purpose input/output**" olarak değiştirilerek devre dışı bırakılabilir.

Analog inputs grup kutusu, sürücünün analog gerilim sinyalini (0..5 V veya ±10 V) okuyacak şekilde ayarlanmasına olanak tanır. Bu konu, 5.6 External reference bölümünde detaylı olarak açıklanacaktır.

4.5 Kontrol Ayarları

Bu bölüm, kontrol modunun (**position**, **speed** veya **torque**) seçilmesine ve kontrol yapısının (hız kontrolörü aktifken veya değilken konum kontrolü) ayarlanmasına olanak tanır.



Şekil 4.9 – Control settings sayfası

Kontrol modu **position** olarak ayarlandığında, sürücü dahili akım ve pozisyon döngülerini kullanarak farklı pozisyon profillerine (trapezoidal, S-Curve, PT, PVT, external vb.) olanak tanır.

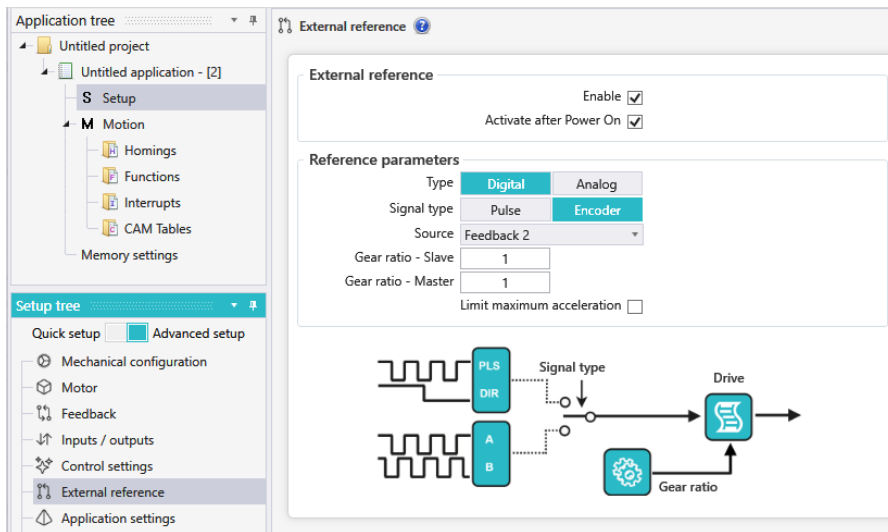
Speed kontrol modu, hız profillerine olanak tanır ve akım ile hız döngülerini kapatır.

Kontrol modunun **torque** olarak ayarlanması, yalnızca akım kontrolörünü aktif tutar ve motor torkunun kontrol edilmesine olanak tanır.

4.6 External reference (Harici referans)

Yalnızca Advanced setup modunda bulunan **External reference** bölümü, sürücünün harici bir sinyali (analog veya dijital) okuyacak ve bunu position, speed veya torque referansına dönüştürecek şekilde ayarlanmasına olanak tanır.

Digital reference seçildiğinde, sürücü **Pulse & Direction** sinyallerini veya bir quadrature **Incremental encoder**'i okuyabilir ve referansı programlanabilir bir dişli oranı (gear ratio) ile hesaplayabilir.



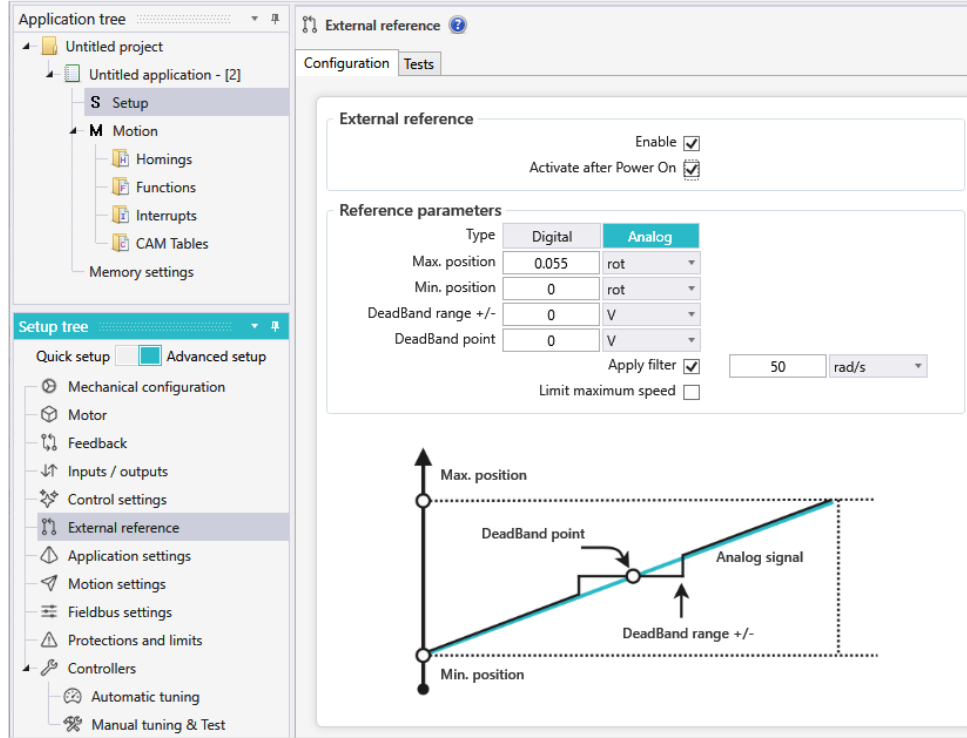
Şekil 4.10 – Digital external reference ayarları

“**Automatically activated after Power On**” seçeneği, AUTORUN modu etkinse ve sürücü belleğinde kayıtlı bir uygulama programı yoksa, sürücüyü güç verildikten sonra External reference modunu otomatik olarak etkinleştirir.

Analog referans seçildiğinde, sürücü 0 V ... 5 V veya ± 10 V gerilim sinyalini okur (desteklenen gerilim aralığı ile ilgili detaylar için sürücü kullanıcı kılavuzuna bakınız) ve seçilen **Control mode**'a bağlı olarak bunu position, speed veya torque referansına dönüştürür.

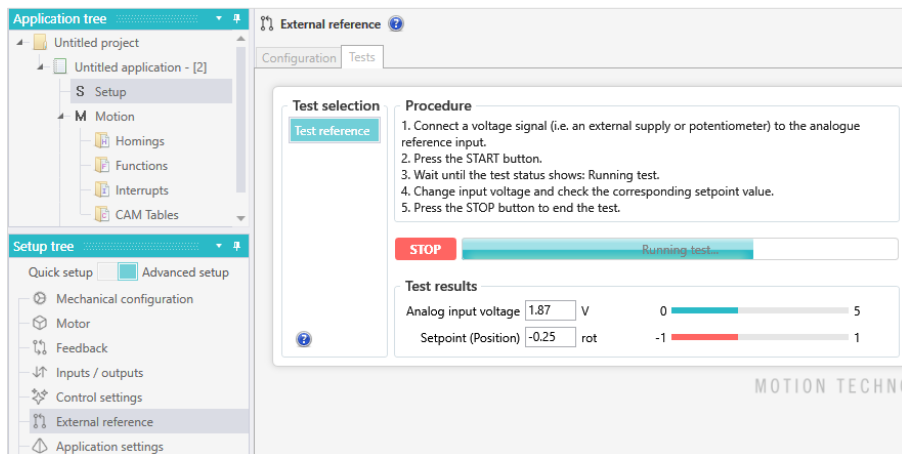
Bu modda External reference bölümü, minimum ve maksimum gerilim referans değerlerine karşılık gelen minimum ve maksimum komut değerlerinin ayarlanmasına olanak tanır (bu iki parametre arasında lineer bir ilişki vardır).

Düşük gerilimli analog sinyaller elektromanyetik gürültülerden kolayca etkilenebildiği ve analogdan dijitale dönüşüm sürecinin kendisi de bir miktar jitter içerdiği için, sürücü bu etkileri telafi etmek amacıyla bazı araçlara sahiptir: referansın daha yumuşak hale getirilmesi için birinci dereceden bir filtre (**first order filter**) ve analog sinyal sabitken motor hareketini/titreşimini önlemek amacıyla referansın sabit tutulacağı bir gerilim aralığı oluşturan "**DeadBand Point**" ve "**DeadBand Range**" seçenekleri.



Şekil 4.11 – Analog external reference ayarları

Tests sekmesi, okunan analog gerilim sinyali üzerindeki referans değişim fonksiyonunun kontrol edilmesine olanak tanır.

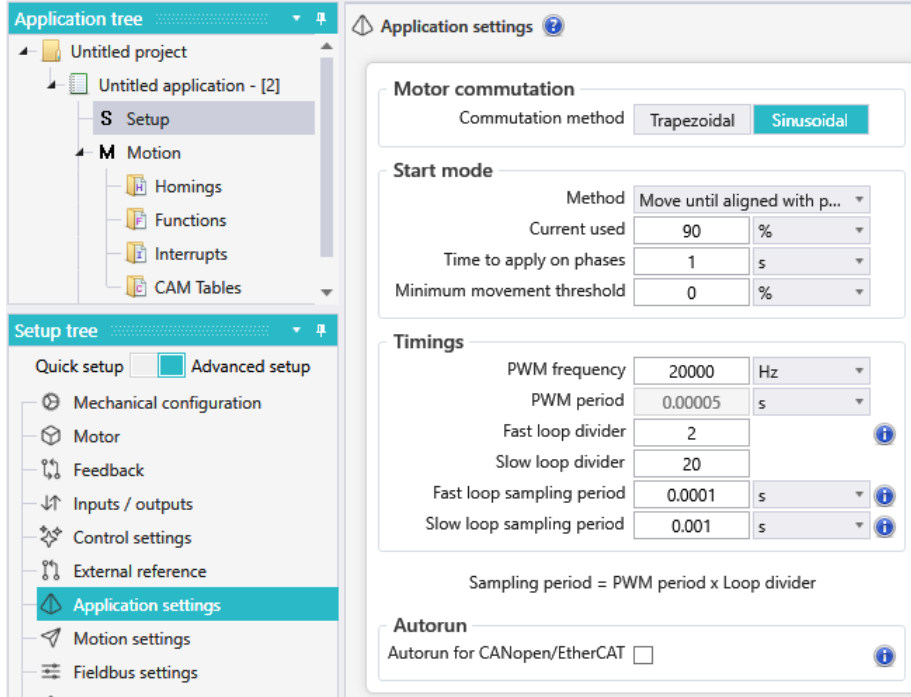


Şekil 4.12 – Analog external reference testleri

Test sırasında, sürücü **Configuration** sekmesindeki **Reference parameters** grup kutusunda daha önce ayarlanmış parametreleri kullanır.

4.7 Uygulama Ayarları

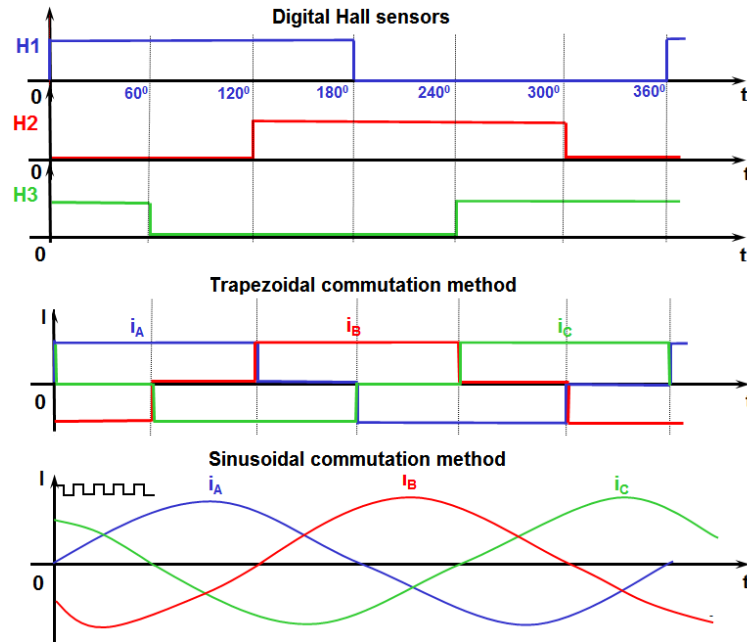
Yalnızca *Advanced setup* modunda bulunan bu bölüm, CANopen/EtherCAT için commutation method, start mode, timings ve AUTORUN modunun yapılandırılmasına olanak tanır.



Şekil 4.13 – Application settings sayfası

Commutation Method bölümü, güç kademesinin komütasyon stratejisinin seçilmesine olanak tanır:

- **Trapezoidal** – motor, komütasyon için Hall sensörleri kullanılarak fırçasız DC (BLDC) motor olarak kontrol edilir. Bu moda motor akımları dikdörtgense, BEMF gerilimleri ise trapezoidal yapıdadır.
- **Sinusoidal** – motor, bir PMSM (Permanent magnet synchronous motor) gibi ele alınır ve FOC (field oriented control) algoritması kullanılarak vektör kontrolü uygulanır. Bu moda motor, üç fazlı sinüzoidal gerilim ve akımlar ile çalışır.

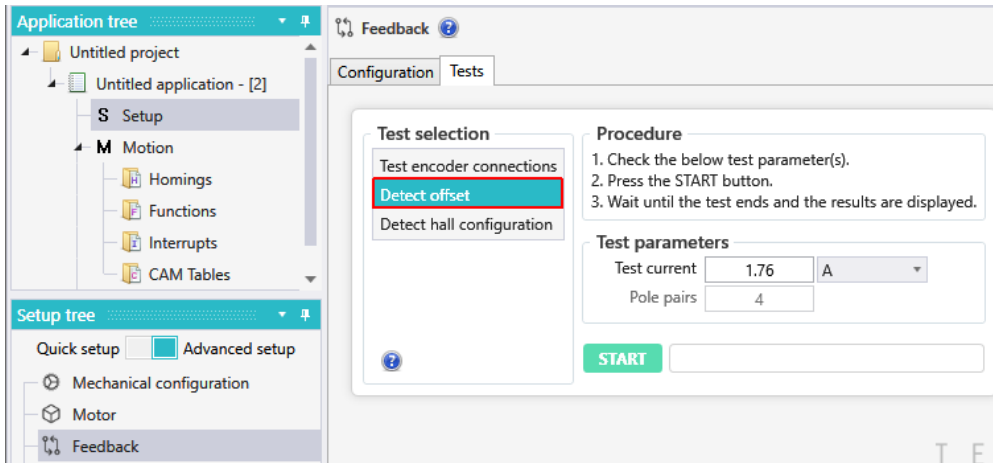


Şekil 4.14 – Trapezoidal ve sinusoidal komütasyon yöntemi karşılaştırması

Not: Sinusoidal mod, özellikle düşük hızlarda daha yumuşak bir çalışma sağlar; buna karşılık trapezoidal mod, daha gürültülü bir hareket karşılığında motora yaklaşık %15 daha fazla tork sağlar.

Field oriented control algoritması, yalnızca rotor konumu hassas şekilde biliniyorsa uygulanabilir. Sürücü, Power On veya Reset sonrasında bunu bir start method aracılığıyla kontrol eder. Bu yöntem, **Start mode** grup kutusu üzerinden seçilebilir ve yapılandırılabilir. Aşağıdaki start method seçenekleri mevcuttur:

- **Move till aligned with phase A**
 - motor, önce B fazı ile, ardından A fazı ile hizalanana kadar hareket eder.
 - bu start method, mekanik sınırların olmadığı ve başlangıçta küçük bir motor sarsıntısının tolere edilebildiği yatay yüklerle sahip tüm uygulamalar için kullanılabilir.
- **BLDC with Hall sensors** (yalnızca motor dijital Hall sensörleri ile donatılmışsa kullanılabilir)
 - motor, trapezoidal moda başlar ve ilk Hall sensörü geçişi algılandıktan sonra sinusoidal moda geçer.
 - bu start mode, dikey yüklerin bulunduğu, motor nominal değerine yakın bir başlangıç torku gerektiren veya başlangıçta dahi kontrolsüz motor hareketine izin verilmeyen uygulamalar için önerilir.
- **PMSM with Hall sensors** (yalnızca motor dijital Hall sensörleri ile donatılmışsa kullanılabilir)
 - BLDC with Hall sensors yöntemine çok benzer; ancak motor trapezoidal yerine doğrudan sinusoidal moda başlar ve ilk Hall sensörü geçişi algılandıktan sonra rotor konumunu ayarlar.
- Motionless start** (yalnızca encoder feedback mevcutsa kullanılabilir)
 - sürücü rotor konumunu belirler; yöntem, başlangıç rotor konumu etrafında yalnızca çok küçük bir hareket içerir;
 - bu start mode, Hall sensörlerinin bulunmadığı ve motor/yükün kontrolsüz hareket edemediği durumlarda kullanılır.
 - en karmaşık yöntemdir ve açı kontrolörünün ayarlanmasını gerektirir.
- **Direct using absolute position sensor** (absolute encoder'lı motorlar için mevcuttur)
 - motorun mutlak konumu bilindiğinden doğrudan çalıştırılmasına olanak tanır;
 - bir offset tespiti gereklidir ve **Feedback** bölümündeki **Test** aracılığıyla yapılabilir.



Şekil 4.15 – Absolute encoder offset tespiti

Timings bölümü, varsayılan PWM frekansının ve kontrol döngülerinin yürütme periyodunun değiştirilmesine olanak tanır.

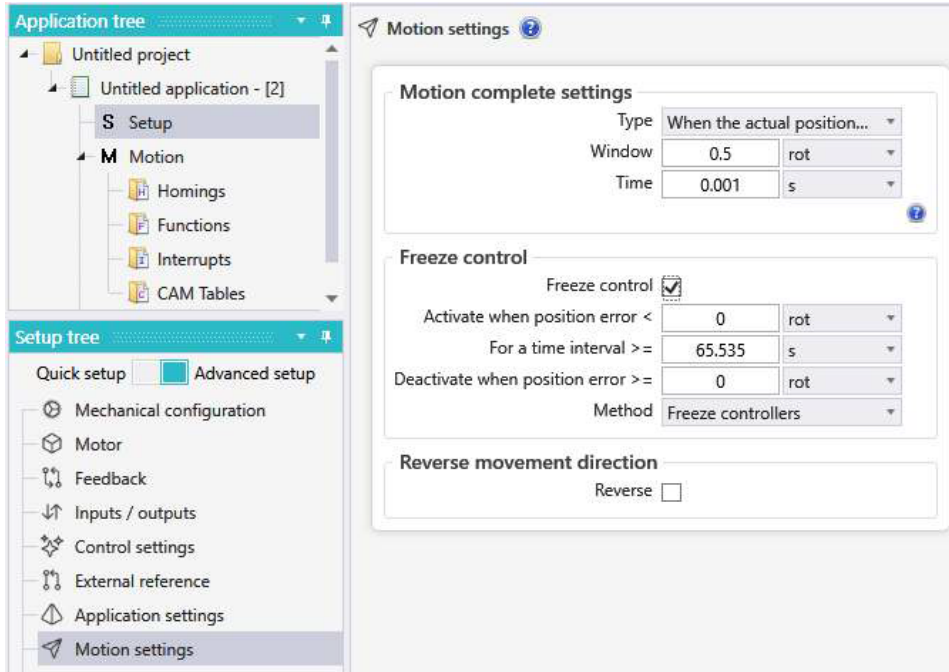
Notlar

- 1) *Güvenli bir çalışma için, PWM frekansının 20 kHz ... 80 kHz aralığında tutulması önerilir;*
- 2) *Hızlı döngü örnekleme periyodunun varsayılan değeri 0.1 ms'dir.*
- 3) *Yavaş döngü örnekleme periyodunun varsayılan değeri 1 ms'dir.*

CANopen/EtherCAT için Autorun seçeneği etkinleştirildiğinde, sürücü Power ON / Reset sırasında TML programını yürütür. Bu durum, yalnızca bir CAN sürücüsü CANopen çalışması için yapılandırıldığında veya bir EtherCAT sürücüsü kullanıldığında geçerlidir.

4.8 Hareket Ayarları

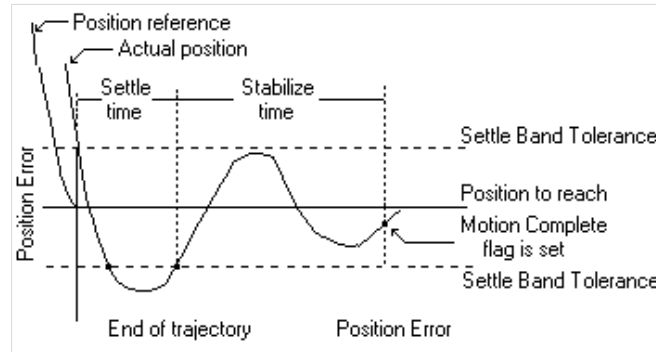
Bu bölüm yalnızca Advanced setup modunda bulunur ve motion complete koşulunun yapılandırılmasına, freeze control mekanizmasının etkinleştirilip ayarlanmasına ve hareket yönünün tersine çevrilmesine olanak tanır.



Şekil 4.16 – Motion settings bölümü

Bir hareket profilinin ne zaman sona erdiğini belirlemek için motion complete koşulu kullanılır. Bu koşul ve ilgili parametreler, **Motion complete settings** grup kutusu üzerinden ayarlanabilir.

Motion complete koşulu, gerçek pozisyonun belirli bir süre boyunca bir settle band içinde kalmasını bekleyecek şekilde ayarlandığında, Motion complete settings grup kutusu; settle band toleransının (position window) ve stabilize süresinin, sistemin imkanları/gereksinimlerine göre ayarlanmasına olanak tanır..



Şekil 4.17 – Motion complete band detayları

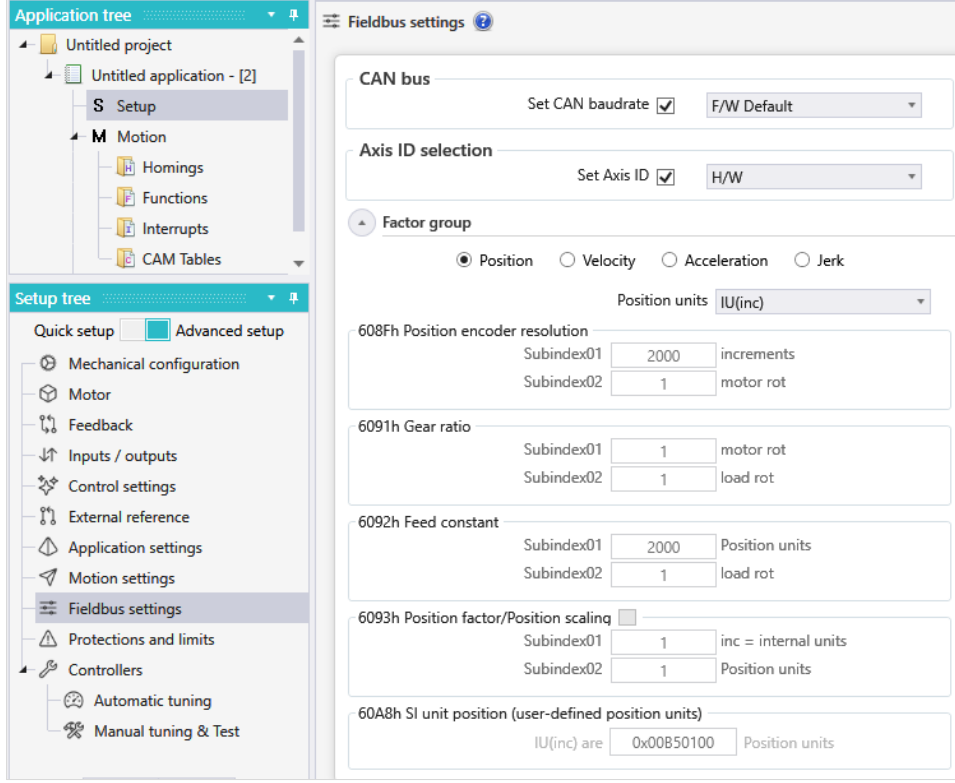
Motion complete on reference koşulu, position kontrolünde pozisyon referansı hedef pozisyona ulaştığında tetiklenir. Speed kontrolünde ise hız referansı 0 olduğunda tetiklenir.

Freeze Control grup kutusu, motor salınımlarını önlemek amacıyla sürücünün kontrolör çıkışlarını donduracak şekilde ayarlanmasına ve son tork komutunun korunmasına olanak tanır. Bu algoritma, gerçek dondurma işleminin etkinleştirilmesi/devre dışı bırakılması için bir pozisyon hata penceresi ve bir zaman parametresinin ayarlanmasını gerektirir.

Motorun dönüş yönü, beklenen yük hareket yönü ile uyuşmadığında, motor bağlantısı değiştirilmelidir (örneğin fırçasız motorlarda iki fazın yerinin değiştirilmesi). Ancak birçok durumda bu işlem zor olduğundan, **Reverse movement direction** seçeneği, motor bağlantılarını değiştirmeden motor hareket yönünü tersine çevirir.

4.9 Fieldbus ayarları

Yalnızca *Advanced setup* modunda bulunan bu bölüm, CAN baudrate, sürücü Axis ID ve CANopen veya EtherCAT factor group parametrelerinin ayarlanmasına olanak tanır.



Şekil 4.18 – Fieldbus settings bölümü

CAN bus grup kutusu, CAN baudrate değerinin ayarlanmasına olanak tanır. Varsayılan değer **F/W default** (firmware default) olup 500 kbps anlamına gelir. Bununla birlikte, 125 kbps, 250 kbps, 500 kbps ve 1 Mbps gibi diğer hız seçenekleri de mevcuttur.

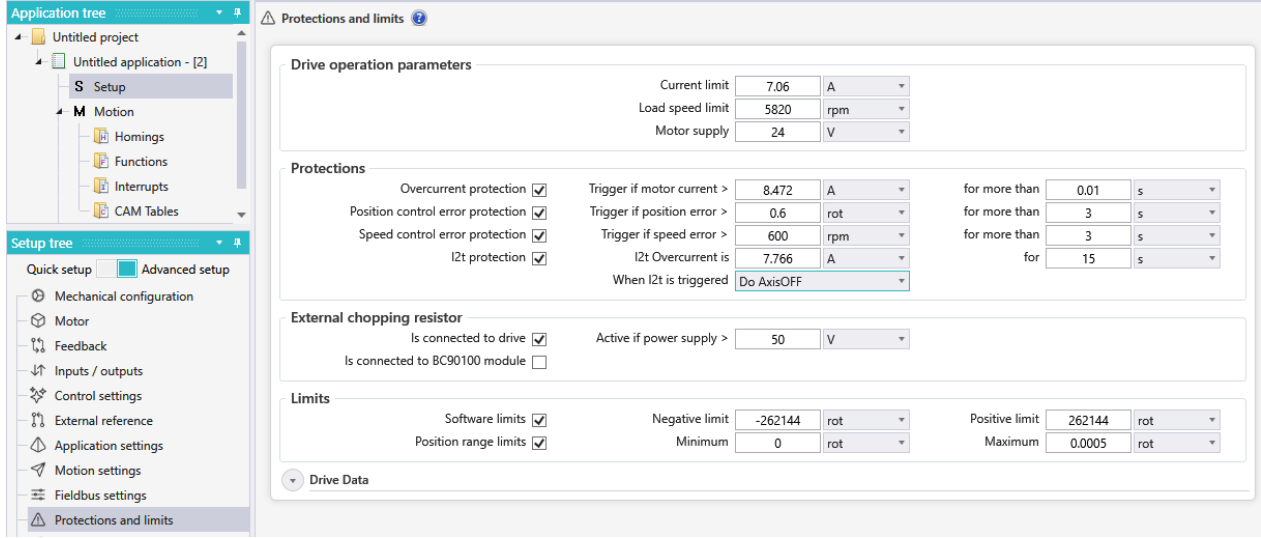
Tüm Technosoft sürücüleri, bir Axis ID ile tanımlanır (CANopen için 1 ile 127 arasında, TMLCAN için ise 1 ile 255 arasında bir değer). Bu değer, donanım üzerinden (detaylar için sürücü kullanıcı kılavuzuna bakınız) veya yazılım üzerinden **Axis ID selection** seçeneği kullanılarak ayarlanabilir.

Bir CAN ağı içerisinde birden fazla sürücü veya akıllı motor birlikte çalışacaksa, aynı CAN baudrate değerine sahip olmaları, aynı CAN protokolünü kullanmaları ve farklı Axis ID değerlerine sahip olmaları gerekir.

CANopen veya EtherCAT uygulamaları için, **Factor group** bölümü; position, speed, acceleration ve jerk değerleri için istenen fiziksel birimin seçilmesine olanak tanır. Bu ayarlar, ilgili nesnelere doğrudan ilişkilidir ve CiA 402 standardına uygundur.

4.10 Korumalar ve Limitler

Bu bölüm, motor korumalarının ve çalışma limitlerinin ayarlanmasına olanak tanır; böylece testler sırasında veya normal çalışma esnasında meydana gelebilecek bozucu etkiler (perturbations) durumunda sistemin (motor, yük, mekanik vb.) zarar görmesi önlenir.



Şekil 4.19 – Protections and limits ayarları

I. Sürücü Çalışma Parametreleri

- 1) **Current limit** - sürücünün motora uygulayacağı maksimum akımı ifade eder. Bu değer, dinamik hareketlere izin vermek için motor nominal akımından biraz daha yüksek, ancak uzun süre uygulanabileceği için motor tepe akımından daha düşük olacak şekilde ayarlanmalıdır.
- 2) **Load speed limit** - hatalı bir komut sonucu tehlikeli hızlara ulaşılmasını önlemek amacıyla yük hızının sınırlandırılmasına olanak tanır; böylece mekanik sistem ve operatörler korunur.
- 3) **Motor supply** – “Vmot” terminaline uygulanan gerilimi ifade eder.

II. Korumalar

- 1) **Over current** - bir akım değeri ve bir zaman aralığı ile tanımlanır ve motor akımı, belirlenen süre boyunca verilen akım limitinin üzerinde kalırsa tetiklenir.

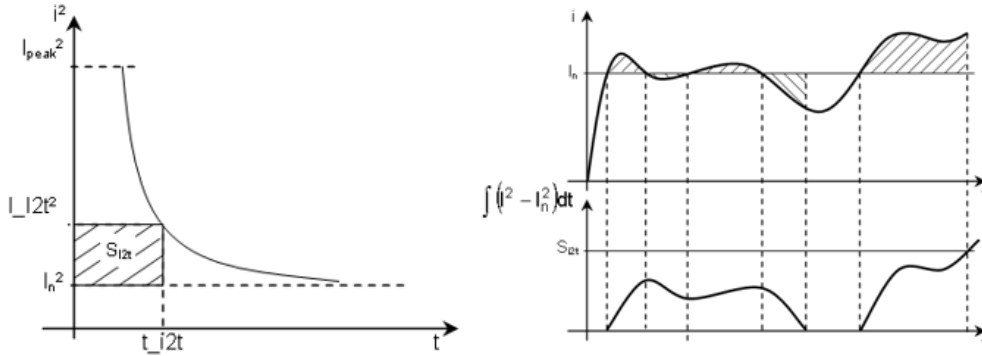
Akım değeri, **current limit** parametresinden biraz daha yüksek ancak motor tepe akımından düşük olacak şekilde ayarlanmalıdır. Over current süresi, motor verilerine uygun olarak belirlenmelidir (motorun tepe akımı taşıyabileceği süreden daha kısa, ancak hatalı tetiklemeleri önleyecek kadar uzun olmalıdır).

- 2) **Control error** - kullanılan kontrol moduna bağlı olarak pozisyon hatasını veya hız hatasını izler ve pozisyon / hız hatası, belirlenen limitin üzerinde, position error time süresinden daha uzun kaldığında tetiklenir.

Pozisyon / hız hatasının mutlak değeri, belirlenen limitin altına düştüğünde position / speed error time sayacının sıfırlandığı unutulmamalıdır.

- 3) **I2t thermal protection** - motorun nominal akımın üzerinde uzun süre çalışması sonucu oluşabilecek hasarı önler. Bu değer, motor I2t eğrisi üzerinde bir nokta seçilerek (motor veri sayfasında bulunmalıdır) ve ilgili alanlara I2t over current değeri ile I2t time değerinin girilmesiyle ayarlanır. Bu noktalar, motor nominal akımı “In” ile birlikte “SI2t” alanını tanımlar (bkz. Şekil 5.20 - sol).

Motor üzerinden geçen akım, nominal akım ("I_n") değerini aştığında, sürücü her akım döngüsünde (fast loop) anlık akımın karesi ile nominal akımın karesi arasındaki farkı hesaplar ve bu değerleri toplar. Bu toplam, ilk şekilde gösterilen "S_{I2t}" alanına eşit veya daha büyük olduğunda, sürücünün I2t koruması tetiklenir (bkz. Şekil 5.20 - sağ).



Şekil 4.20 – I2t alanı ve tetiklenmesi

Tetiklendikten sonra, I2t korumasının varsayılan davranışı motor akımını Motor Setup penceresinde ayarlanan nominal akımın %90'ı ile sınırlamaktır. Dahili koruma değeri 0'a düştüğünde akım, normal akım limitine geri döner. Ayrıca, sürücünün güç kademesini devre dışı bırakacak şekilde de ayarlanabilir ("AXISOFF;").

III. Harici Frenleme Direnci

Motor frenleme yaptığında veya yön değiştirdiğinde, enerjiyi güç kaynağına geri aktarır ve bu durum (güç kaynağı bu enerjiyi absorbe edemiyorsa) DC bara geriliminin yükselmesine neden olur. Aktarılan enerji miktarına ve aktarım hızına bağlı olarak, bara gerilimi aşırı gerilim koruma seviyesini aşabilir ve hatta sürücüye veya güç kaynağına zarar verebilir. Bu istenmeyen durumu önlemek için, aşırı gerilim bir chopping resistor (frenleme direnci) kullanılarak sınırlandırılabilir (frenleme direnci seçimi ve bağlantısı için sürücü kullanıcı kılavuzuna bakınız).

Genellikle, frenleme devresinin devreye girme seviyesi, sürücü besleme gerilimi değerinin biraz üzerinde ve aşırı gerilim koruma seviyesinin altında olacak şekilde ayarlanmalıdır.

IV. Limitler

1) **Software limit switches** - bir limit değerine ulaşıldığında, motor hareketini durdurmak için quick stop profili otomatik olarak uygulanır. İlgili yönde artık harekete izin verilmez.

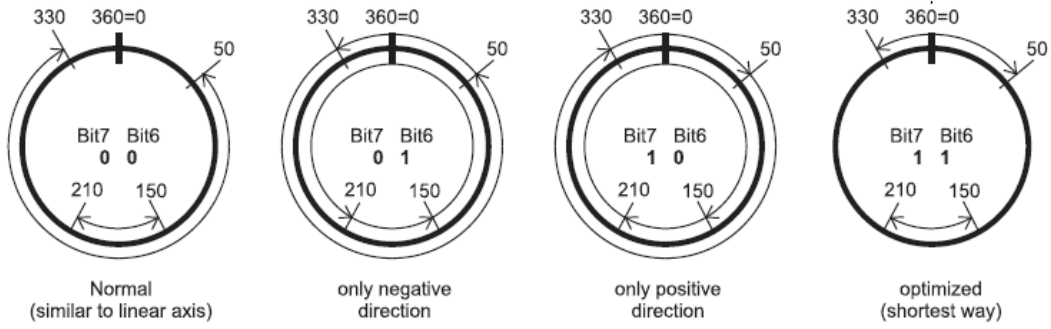
2) **Position range limits** aşağıdaki olanakları sağlar:

- pozisyon göstergesinin iki limit arasında tutulması (varsayılan çalışma modu)
- hedef pozisyona yalnızca negatif veya yalnızca pozitif hareketlerle ulaşılması
- hedef pozisyona en kısa yol kullanılarak ulaşılması

Position range limits davranışı, **POSOPTCODE** register'ındaki 6 ve 7 numaralı bitler üzerinden ayarlanabilir:

Register adı	Bit 7	Bit 6	Açıklama
POSOPTCODE	0	0	Normal positioning (varsayılan seçenek): pozisyon göstergesi, position range limitlerinden birine ulaştığında veya aştığında, aralığın diğer ucuna otomatik olarak sarılır (wrap).
	0	1	Positioning only in negative direction: hedef pozisyon, mevcut pozisyondan büyükse, eksen minimum pozisyon limitini aşarak negatif yönde hareket eder ve hedef pozisyona ulaşır.
	1	0	Positioning only in positive direction: hedef pozisyon, mevcut pozisyondan küçükse, eksen maksimum pozisyon limitini aşarak pozitif yönde hareket eder ve hedef pozisyona ulaşır.
	1	1	Eksen, hedef pozisyona en kısa yolu izleyerek hareket eder. NOT: 360°'lik bir sistemde, mevcut değer ile hedef pozisyon arasındaki fark 180° ise, eksen pozitif yönde hareket eder.

Aşağıdaki şekil, minimum pozisyon aralığı limitinin 0° ve maksimum pozisyon aralığı limitinin 360° olarak ayarlandığı durumda (POSOPTCODE register'ındaki 6 ve 7 numaralı bitlerin ayarlarına bağlı olarak) bazı hareket örneklerini göstermektedir.



Şekil 4.21 – Döner eksen pozisyon aralığı limitleri örnekleri

Limits grup kutusu yalnızca position range limitlerinin ayarlanmasına olanak tanır. Bu limitlerin davranışı (POSOPTCODE register değeri), TML programı içerisinde belirlenir veya bir master varsa, POSOPTCODE register değeri master tarafından uygun şekilde ayarlanabilir.

V. Sürücü Verileri

Drive Data sayfası, sürücünün nominal parametrelerini ve korumalarını içerir. Bu bilgiler önemlidir; çünkü motor korumalarının, sürücü limitleri ile uyumlu olması gerekir.

Protections and limits

Drive operation parameters

Current limit	7.06	A			
Load speed limit	5820	rpm			
Motor supply	24	V			

Protections

Overcurrent protection <input checked="" type="checkbox"/>	Trigger if motor current >	8.472	A		for more than	0.01	s
Position control error protection <input checked="" type="checkbox"/>	Trigger if position error >	0.6	rot		for more than	3	s
Speed control error protection <input checked="" type="checkbox"/>	Trigger if speed error >	600	rpm		for more than	3	s
I2t protection <input checked="" type="checkbox"/>	I2t Overcurrent is	7.766	A		for	15	s
	When I2t is triggered	Do AxisOFF					

External chopping resistor

Is connected to drive <input checked="" type="checkbox"/>	Active if power supply >	50	V		
Is connected to BC90100 module <input type="checkbox"/>					

Limits

Software limits <input checked="" type="checkbox"/>	Negative limit	-262144	rot		Positive limit	262144	rot
Position range limits <input checked="" type="checkbox"/>	Minimum	0	rot		Maximum	0.0005	rot

Drive Data

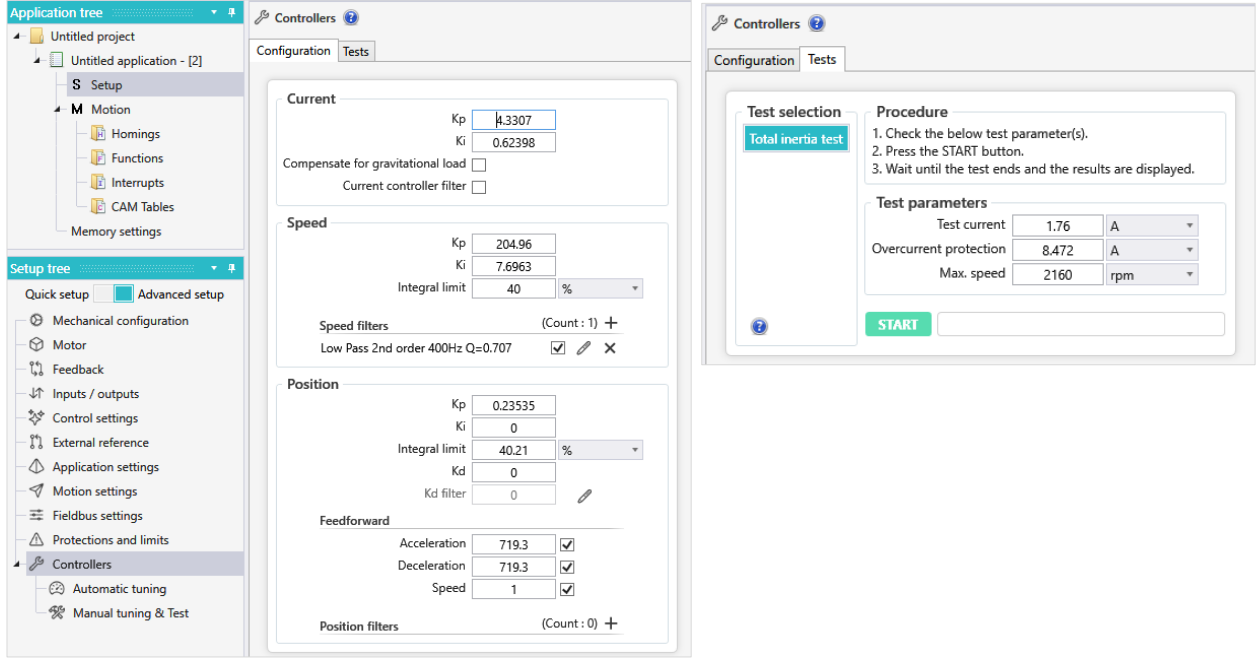
<p>Drive Data</p> <p>Motor supply</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">min. value</td> <td style="width: 10%;">11</td> <td style="width: 10%;">V</td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> </tr> <tr> <td>max. value</td> <td>52</td> <td>V</td> <td></td> <td></td> <td></td> </tr> </table> <p>Logic supply</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">min. value</td> <td style="width: 10%;">8</td> <td style="width: 10%;">V</td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> </tr> <tr> <td>max. value</td> <td>40</td> <td>V</td> <td></td> <td></td> <td></td> </tr> </table> <p>Drive nominal current</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">Drive nominal current</td> <td style="width: 10%;">8</td> <td style="width: 10%;">A</td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> </tr> <tr> <td>Drive peak current</td> <td>20</td> <td>A</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Maximum measurable current</td> <td>20</td> <td>A</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Maximum measurable DC voltage</td> <td>102.3</td> <td>V</td> <td></td> <td></td> <td></td> </tr> <tr> <td>AD2 analogue input range 1</td> <td>5</td> <td>V</td> <td></td> <td></td> <td></td> </tr> <tr> <td>AD2 analogue input range 2</td> <td>10</td> <td>V</td> <td></td> <td></td> <td></td> </tr> <tr> <td>AD5 analogue input range 1</td> <td>5</td> <td>V</td> <td></td> <td></td> <td></td> </tr> <tr> <td>AD5 analogue input range 2</td> <td>10</td> <td>V</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Temperature sensor <input checked="" type="checkbox"/></td> <td colspan="5"></td> </tr> <tr> <td>Sensor gain</td> <td>0.01</td> <td>V/°C</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Output at 0°C</td> <td>0.5</td> <td>V</td> <td></td> <td></td> <td></td> </tr> </table>	min. value	11	V				max. value	52	V				min. value	8	V				max. value	40	V				Drive nominal current	8	A				Drive peak current	20	A				Maximum measurable current	20	A				Maximum measurable DC voltage	102.3	V				AD2 analogue input range 1	5	V				AD2 analogue input range 2	10	V				AD5 analogue input range 1	5	V				AD5 analogue input range 2	10	V				Temperature sensor <input checked="" type="checkbox"/>						Sensor gain	0.01	V/°C				Output at 0°C	0.5	V				<p>Drive protections</p> <p>Overvoltage protection <input checked="" type="checkbox"/></p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">Trigger if motor supply ></td> <td style="width: 10%;">53</td> <td style="width: 10%;">V</td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> </tr> </table> <p>Undervoltage protection <input checked="" type="checkbox"/></p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">Trigger if motor supply <</td> <td style="width: 10%;">9</td> <td style="width: 10%;">V</td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> </tr> </table> <p>I2t protection <input checked="" type="checkbox"/></p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">Allowed overcurrent</td> <td style="width: 10%;">10</td> <td style="width: 10%;">A</td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> </tr> <tr> <td>for</td> <td>15</td> <td>s</td> <td></td> <td></td> <td></td> </tr> </table> <p>Drive temperature protection <input checked="" type="checkbox"/></p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">Trigger if temperature ></td> <td style="width: 10%;">105</td> <td style="width: 10%;">°C</td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> </tr> </table>	Trigger if motor supply >	53	V				Trigger if motor supply <	9	V				Allowed overcurrent	10	A				for	15	s				Trigger if temperature >	105	°C			
min. value	11	V																																																																																																																							
max. value	52	V																																																																																																																							
min. value	8	V																																																																																																																							
max. value	40	V																																																																																																																							
Drive nominal current	8	A																																																																																																																							
Drive peak current	20	A																																																																																																																							
Maximum measurable current	20	A																																																																																																																							
Maximum measurable DC voltage	102.3	V																																																																																																																							
AD2 analogue input range 1	5	V																																																																																																																							
AD2 analogue input range 2	10	V																																																																																																																							
AD5 analogue input range 1	5	V																																																																																																																							
AD5 analogue input range 2	10	V																																																																																																																							
Temperature sensor <input checked="" type="checkbox"/>																																																																																																																									
Sensor gain	0.01	V/°C																																																																																																																							
Output at 0°C	0.5	V																																																																																																																							
Trigger if motor supply >	53	V																																																																																																																							
Trigger if motor supply <	9	V																																																																																																																							
Allowed overcurrent	10	A																																																																																																																							
for	15	s																																																																																																																							
Trigger if temperature >	105	°C																																																																																																																							

Şekil 4.22 – Drive Data sayfası

Not: Varsayılan olarak, **Drive Data** sayfası yalnızca okunabilir ve bilgilendirme amaçlı olduğundan, minimize edilmiş şekilde açılır.

4.11 Kontrolörler

Bu bölüm, kontrolör parametrelerinin düzenlenmesine, toplam sistem ataletinin tespit edilmesine (yalnızca Advanced setup modunda mevcuttur) ve kontrolör ayarlarının (otomatik veya manuel) yapılmasına olanak tanır.

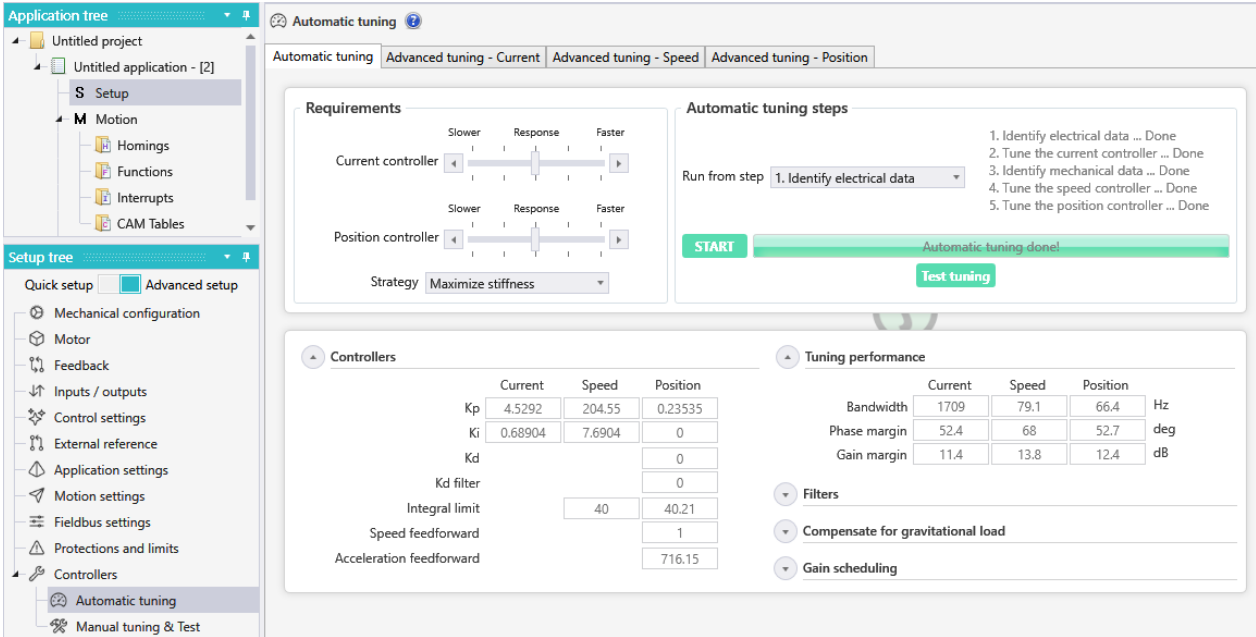


Şekil 4.23 – Controllers Configuration sekmesi

Tahmin edilen atalet, seçilen tuning yöntemi “Set bandwidth” olduğunda, position / speed kontrolör parametrelerinin hesaplanmasında kullanılır (bkz. position ve speed için Advanced tuning sekmelerindeki Tune bölümü).

4.11.1 Otomatik Ayar

Bu seçenek, istenen sistem tepkisine göre kontrolör parametrelerinin kolayca belirlenmesine olanak tanır. Otomatik tuning işlemi, **START** butonuna basıldığında başlatılır.

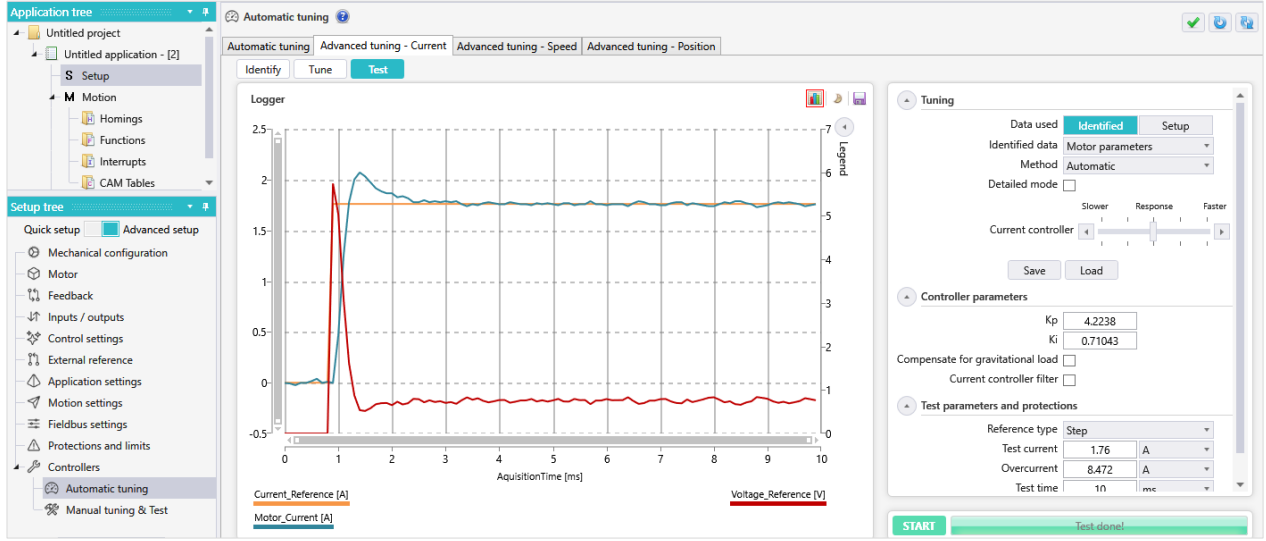


Şekil 4.24 – Automatic tuning sayfası

Tuning işlemi başarıyla tamamlandığında, ilerleme çubuğunda “**Automatic tuning done!**” mesajı görüntülenir. Tuning parametreleri ile birlikte tespit edilen veya hesaplanan diğer parametreler, **Controllers** and **Tuning performances** grup kutusunda gösterilir.

Herhangi bir nedenle otomatik tuning işlemi başarısız olursa, ilerleme çubuğunda “**Automatic tuning failed**” mesajı görüntülenir. İmleç ilerleme çubuğu üzerine getirildiğinde, otomatik tuning işleminin neden başarısız olduğuna dair daha fazla bilgi içeren küçük bir bilgilendirme mesajı açılır.

Tuning parametreleri, **Advanced Tuning** sekmeleri üzerinden test edilebilir ve ayarlanabilir.



Şekil 4.25 – Advanced tuning sekmeleri

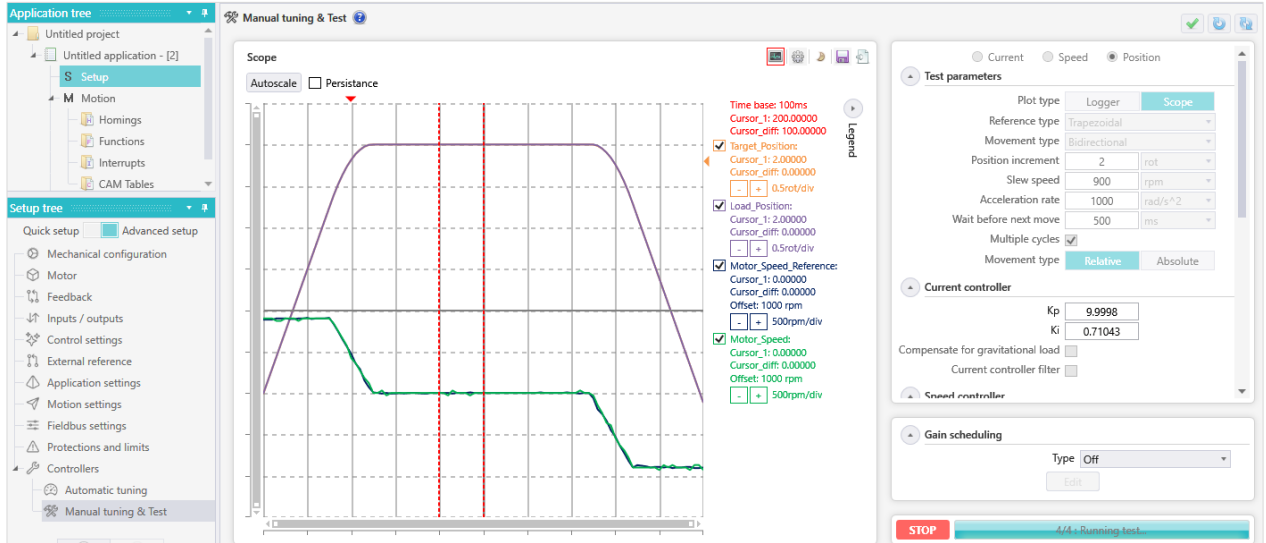
Advanced tuning sekmeleri 3 sayfa içerir:

- **Identify** – ilgili kontrolörün ayarlanmasında kullanılan tespit edilmiş verileri gösterir ve tespit yönteminin değiştirilmesine olanak tanıyarak tespit işleminin yeniden yapılmasını sağlar.
- **Tune** – hesaplanan kontrolör parametrelerini ve tahmini kontrolör tepkisini gösterir. Ayrıca ilgili kontrolör için tuning işleminin yeniden başlatılmasında kullanılır.
- **Test** – ilgili kontrolör tepkisinin test edilmesine ve tuning parametrelerinin değiştirilmesine olanak tanır.

En iyi sistem performansı için, tuning işlemi motor yükü bağlıyken gerçekleştirilmelidir.

4.11.2 Manuel Ayarlama & Test

Bu seçenek, bir akım, hız veya position profili çalıştırılmasına ve logger veya scope araçları üzerinden sistem davranışı izlenirken aktif kontrolör parametrelerinin ayarlanmasına olanak tanır.

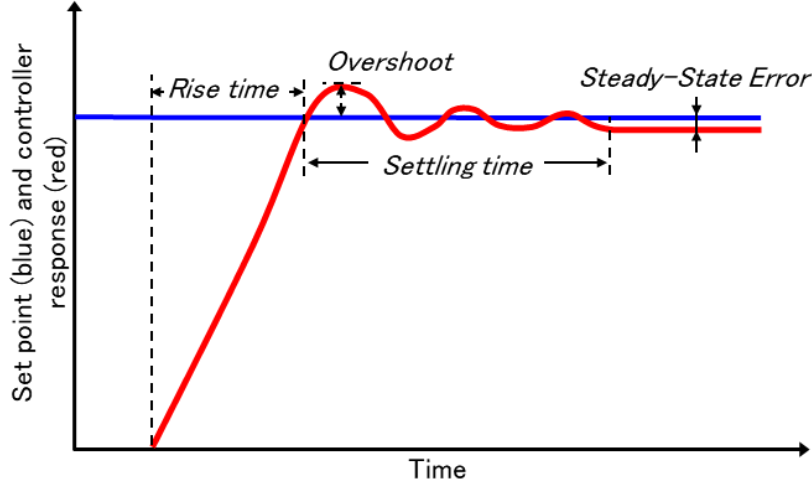


Şekil 4.26 – Manual tuning sayfası

Akım ve hız kontrolörleri PI tipindedir: $(u(t) = kp * \varepsilon(t) + ki * \int_0^t \varepsilon(t)dt)$, Pozisyon kontrolörü ise PID tipindedir: $(u(t) = kp * \varepsilon(t) + ki * \int_0^t \varepsilon(t)dt + kd * \frac{d\varepsilon(t)}{dt})$.

Kontrolör tepkisi karakteristikleri (overshoot, rise time ve steady-state error), K_p , K_i ve K_d sabitleri ile aşağıdaki şekilde ilişkilidir:

- K_p 'nin artırılması, overshoot'un artmasına neden olurken rise time ve steady-state error değerlerini azaltır;
- K_i 'nin artırılması, overshoot'un artmasına ve rise time'in bir miktar azalmasına neden olurken steady-state error'ü önemli ölçüde düşürür; ancak sistemde salınım (oscillation) oluşma olasılığını artırır (ideal bir sistemde, çıkış geriliminin doyumuna girmediği varsayılırsa, K_i pozisyon hatasını tamamen ortadan kaldıracaktır);
- K_d 'nin artırılması, rise time'ı artırırken overshoot'u azaltır (steady-state error, K_d 'den etkilenmez).

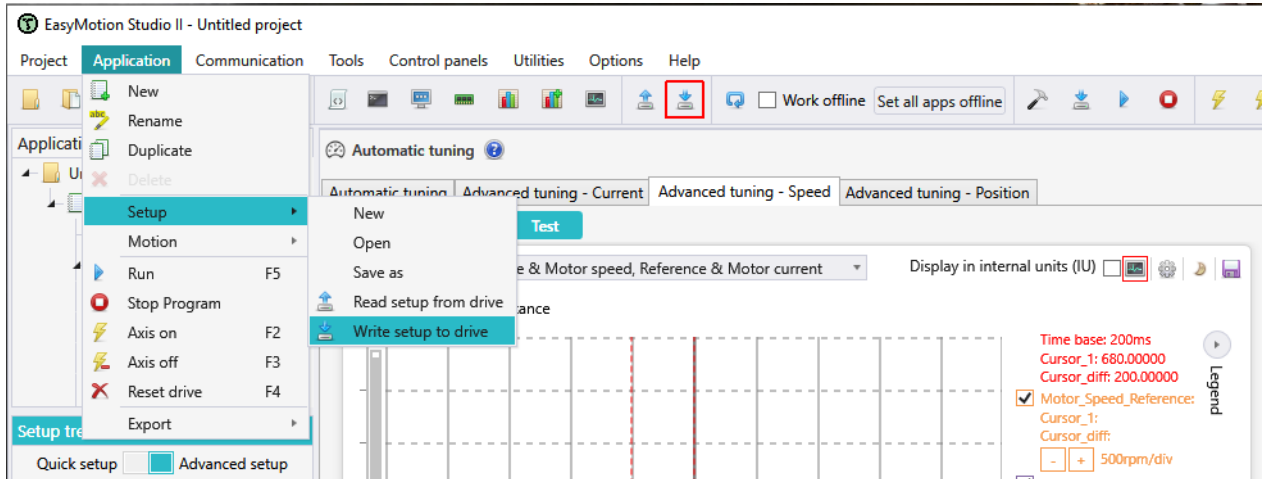


Şekil 4.27 – Kontrolör tepki karakteristikleri

Kontrolör ayarı (tuning), setup sürecinin son adımını oluşturur. Bu işlem tamamlandığında, setup sürücüyü yüklenebilir ve sürücü bir sonraki adım (TML uygulama programlama veya master kontrolü) için hazır hale getirilir.

4.12 Setup'ı sürücüyü yükle

Setup, **Application** menüsündeki “**Write setup to drive**” seçeneği kullanılarak veya aşağıdaki şekilde kırmızı kare ile işaretlenmiş ribbon butonu aracılığıyla sürücüyü yüklenebilir.



Şekil 4.2 – Setup parametrelerinin sürücü belleğine yazılması

Setup, sürücünün kalıcı belleğine (EEPROM) yazıldıktan sonra, etkin hale gelmesi için bir reset gereklidir.

5 Sürücü Programlama

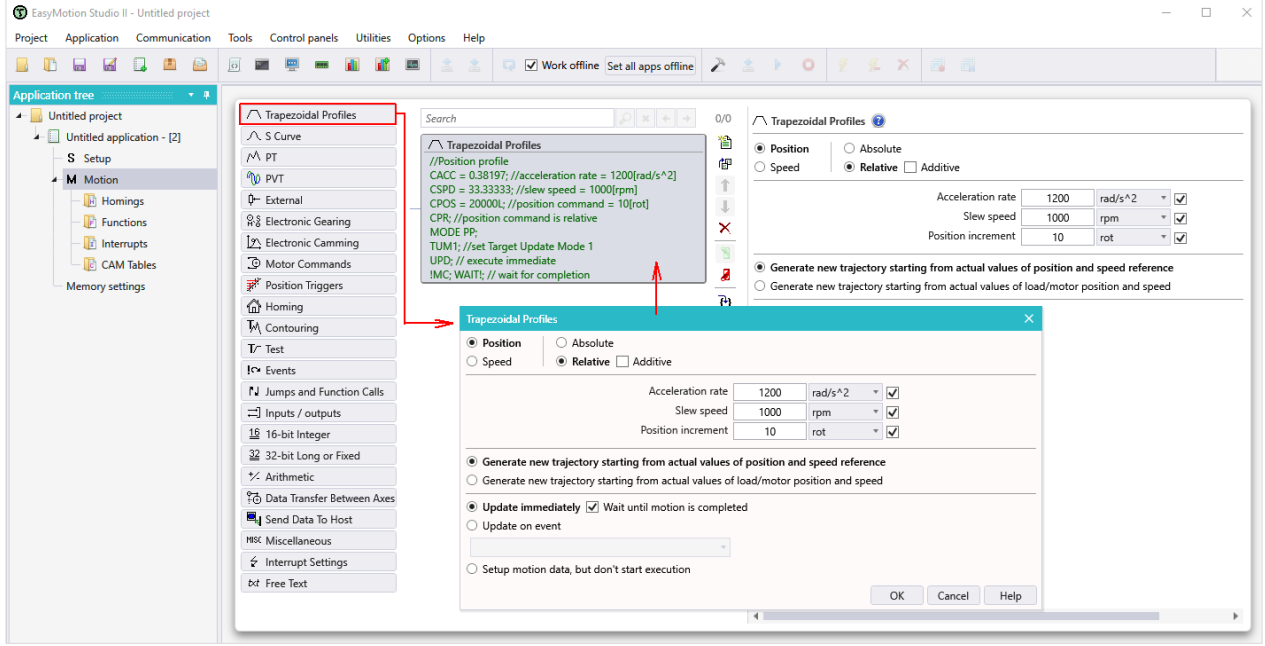
Technosoft sürücüleri ve akıllı motorlar, **bağımsız** cihazlar olarak veya bir **CAN** (TMLCAN veya CANopen) ya da **EtherCAT** ağı içerisinde **slave** olarak çalışabilir.

Sürücü / akıllı motor bağımsız cihaz olarak kullanıldığında veya uygulama fonksiyonelliği master ile Technosoft sürücüleri / akıllı motorlar arasında dağıtıldığında, bir **TML programı** gereklidir.

TML programı, **Motion** bölümündeki **modüller** kullanılarak geliştirilebilir ve sürücüye TML programı veya fonksiyonlar olarak kaydedilebilir. Ayrıca bazı ön tanımlı homing yöntemleri ve interrupt'lar da mevcuttur.

Motion bölümündeki modüller, **Help** topics içerisinde şu yol izlenerek açıklanmıştır: **Help | Help Topics | Application Programming | Motion Programming with built-in Motion Controller**.

EasyMotion Studio ile sürücü / akıllı motor programlama işlemi grafiksel olarak gerçekleştirildiğinden, TML kodu **Motion** bölümündeki modüllerin arkasındaki **sihirbaz (wizard) pencereleri** tarafından otomatik olarak oluşturulur.

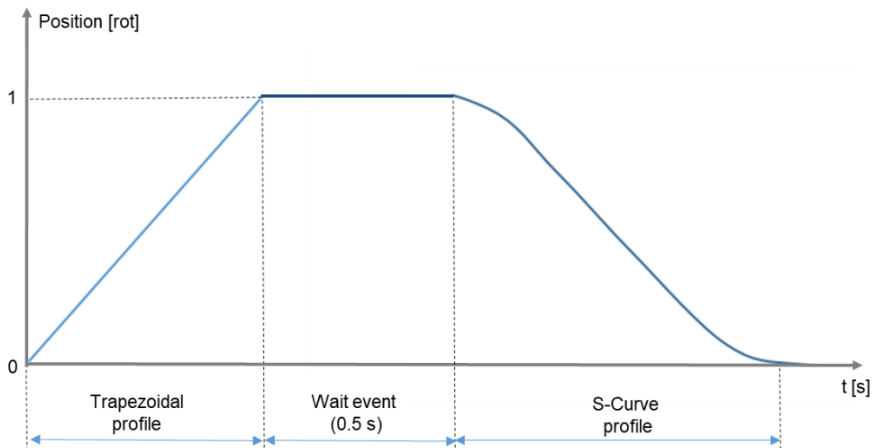


Şekil 5.1 – Trapezoidal profil sihirbazı tarafından oluşturulan TML kodu örneği

TML komutlarına ilişkin daha fazla detay, **Help** topics içerisinde şu yol izlenerek bulunabilir: **Help | Help Topics | Application Programming | Technosoft Motion Language**.

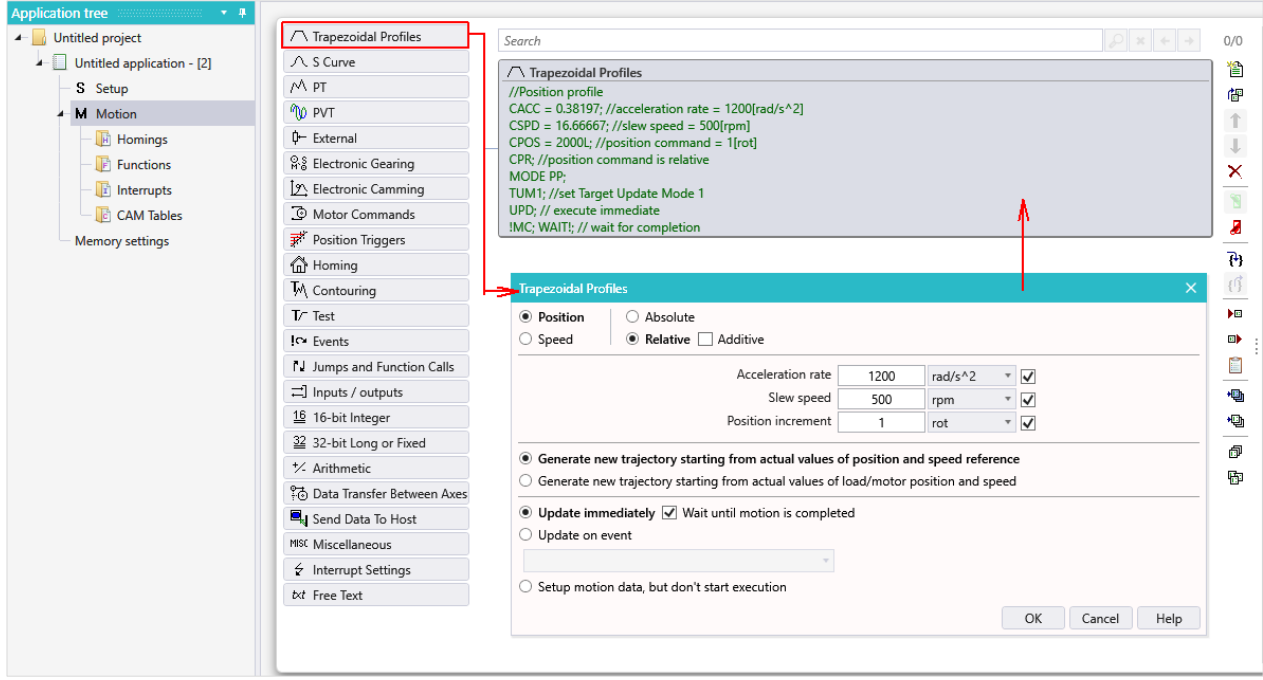
TML Programı Örneği

Yük, trapezoidal position profili kullanılarak relatif olarak 1 tur hareket ettirilir, 0.5 saniye beklenir ve ardından S-Curve profili kullanılarak geri döndürülür.



Şekil 6.2 – Gerekli hareket profili

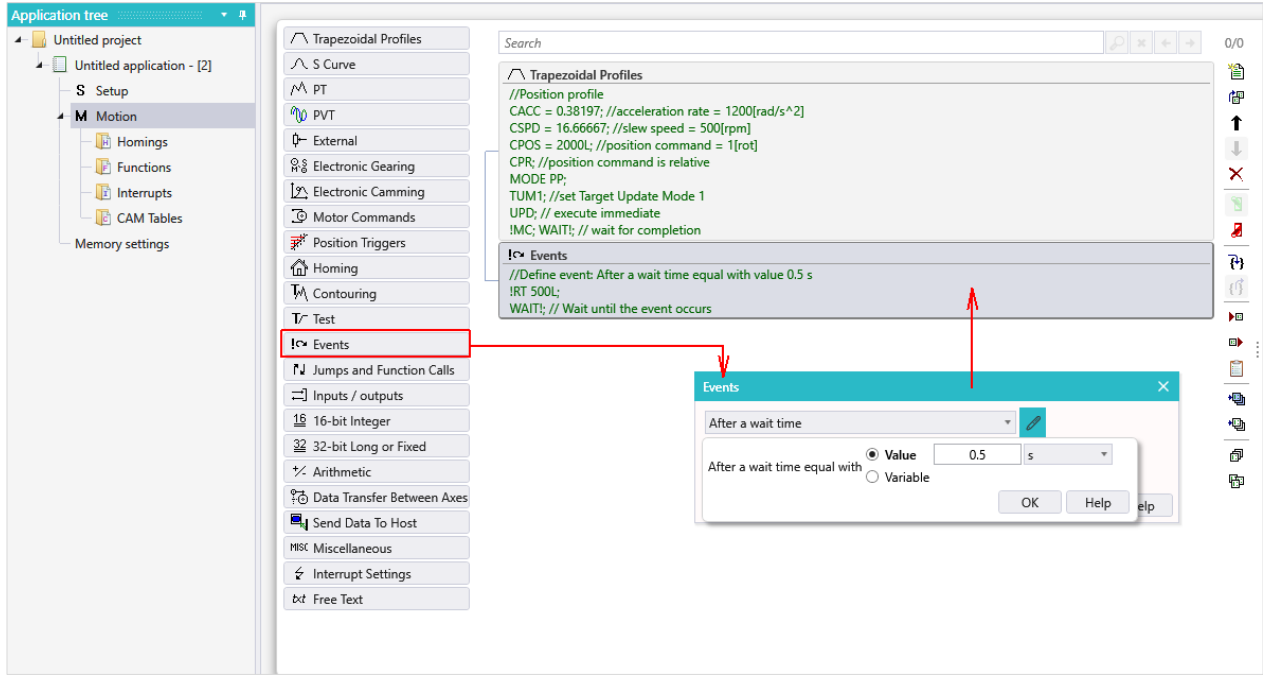
Adım 1: Gerekli pozisyon profilini eklemek için Trapezoidal Profile modülünü kullanın.



Şekil 6.3 – Trapezoidal pozisyon profilinin eklenmesi

Trapezoidal Profile sihirbaz diyalogu, hareket tipinin (absolute veya relative), hareket profili parametrelerinin (ivme, hız ve pozisyon komutları) ve trajektori oluşturma stratejisinin ayarlanmasına olanak tanır.

Adım 2: 0.5 s bekleme süresini eklemek için Events modülünü kullanın.



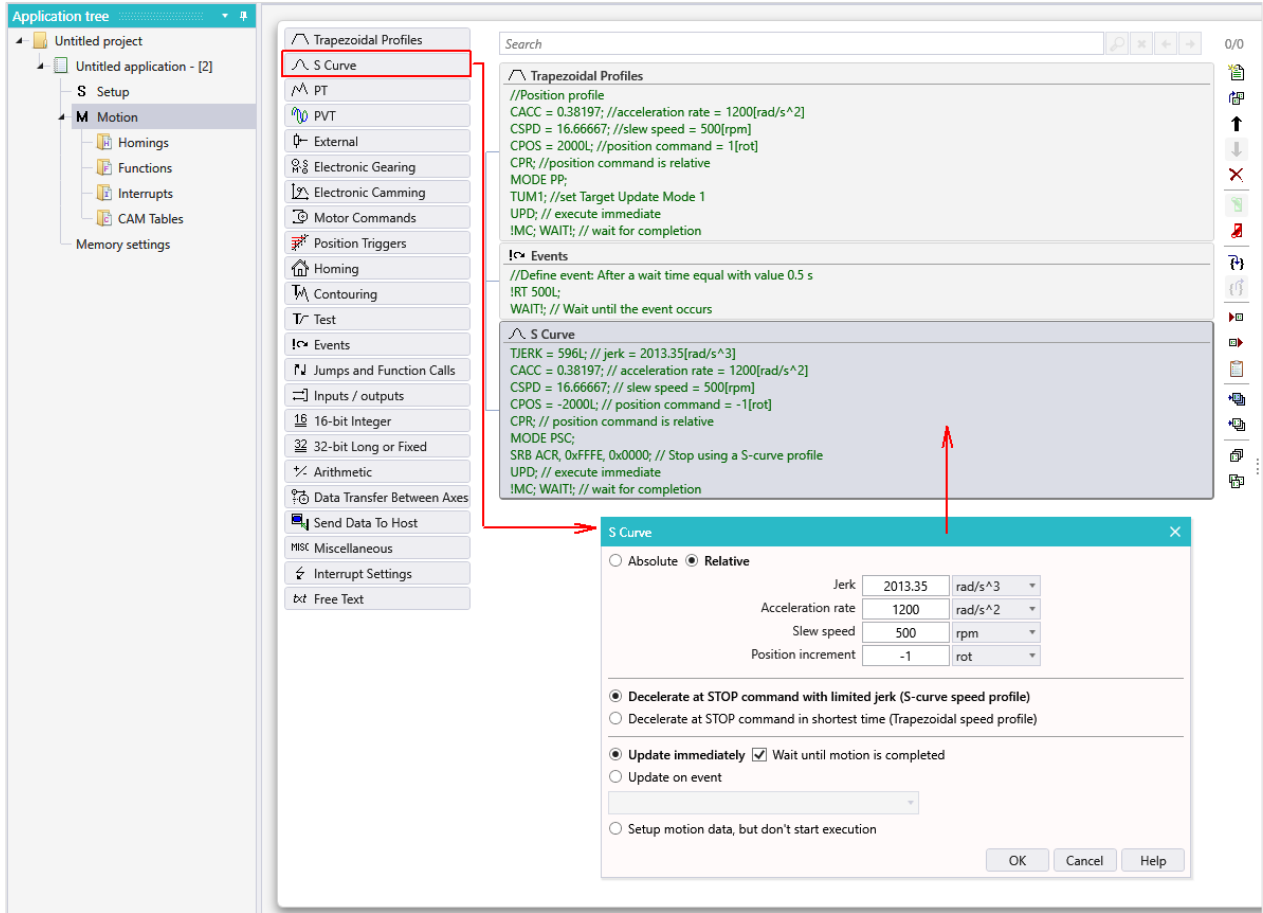
Şekil 6.4 – Zaman olayının eklenmesi

Events diyalogu, sürücünün bir koşul gerçekleşene kadar beklemesini veya ilgili olay meydana geldiğinde hareketi durdurmasını ayarlamaya olanak tanır.

Mevcut olaylar şunlardır:

- Gerçek hareket tamamlandığında
- Motor veya yük pozisyonuna bağlı olarak
- Motor veya yük hızına bağlı olarak
- Belirli bir bekleme süresi sonrasında
- Referans değerine bağlı olarak
- Giriş durumuna bağlı olarak
- 32-bit bir değişken değerine bağlı olarak

Adım 3: Geri dönüş profilini eklemek için S Curve modülünü kullanın.

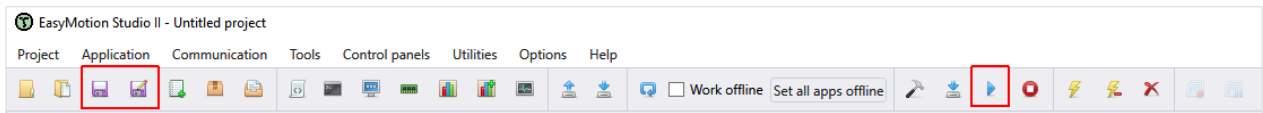


Şekil 6.5 – S Curve profilinin eklenmesi

Trapezoidal position profiline kıyasla, S-Curve profili “jerk” parametresinin de ayarlanmasına olanak tanır. Bu parametre, ivmenin 0 IU’dan belirlenen değere ne kadar hızlı artacağını belirler.

Modül diyaloglarındaki tüm seçenekler ve parametreler için detaylı açıklamalar, ilgili modül diyalogunda bulunan **Help** butonuna tıklanarak açılabilen Help topics içerisinde yer almaktadır.

Adım 4: Projeyi kaydedin ve TML programını çalıştırın.



Şekil 6.1 – Proje “Save” ve TML programı “Run” butonları

Proje, **Project** menüsündeki **Save** seçeneği kullanılarak veya ribbon üzerindeki simge aracılığıyla kaydedilebilir..

TML programı, **Application** menüsündeki **Run** seçeneği kullanılarak veya ribbon üzerindeki buton aracılığıyla başlatılabilir.

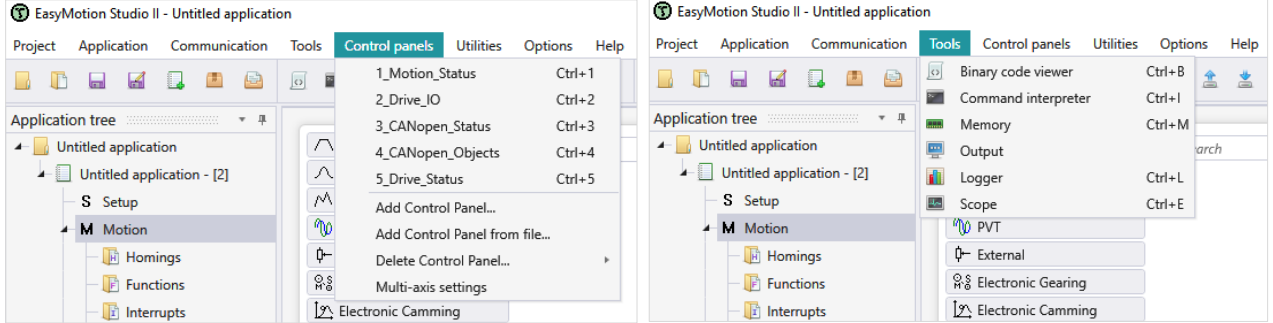
Program çalıştırdıktan sonra, motor programlanan hareket profilini gerçekleştirir (pozitif yönde 1 tur döner, 0.5 s bekler ve ardından negatif yönde 1 tur döner).

Daha fazla örnek için web sitemizdeki Application Notes bölümüne ([link](#)) bakınız.

6 Hareket Değerlendirme ve Uygulama Hata Ayıklama Araçları

EasyMotion Studio II; motor/yük hareketini kontrol etmek veya TML uygulamasını izlemek, kontrol etmek ve hata ayıklamak için kullanılacak bir dizi **kontrol paneli**, **Logger** ve **Scope** araçlarını içerir..

There are also some **tools** (Binary code viewer, Command Interpreter, Memory View and Output dialogue) that can be used for an advanced application/drive functionality check or debug.



Şekil 7.1 – Mevcut kontrol panelleri ve araçlar

Değerlendirme araçlarına ilişkin detaylı açıklamalar, EasyMotion Studio II **Help** topics bölümünde yer almaktadır (Help | Help topics | Evaluation Tools ...).

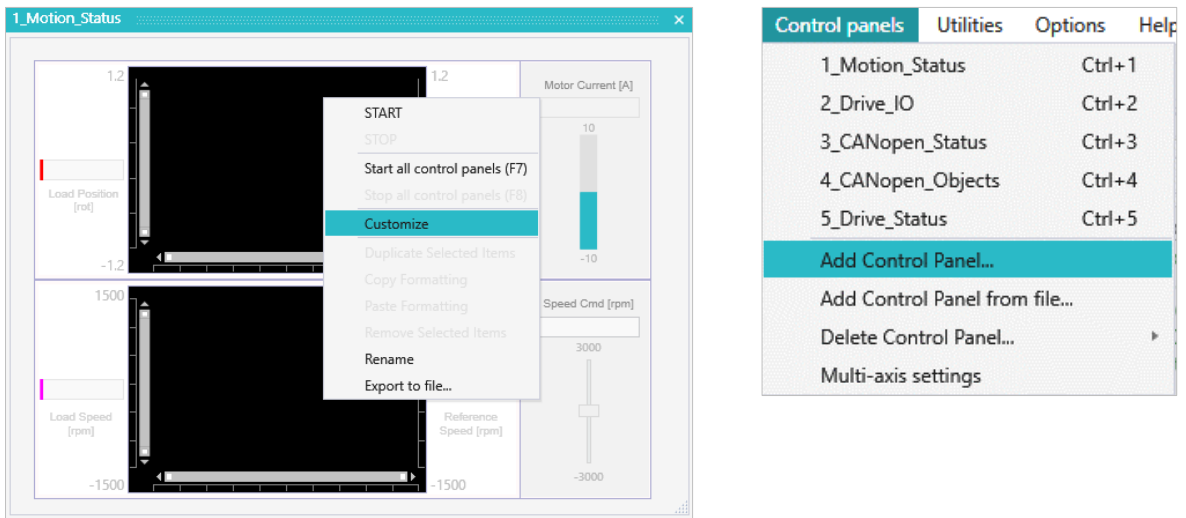
6.1 Kontrol Panellerinin Kullanımı

EasyMotion Studio II, aşağıdaki ön tanımlı kontrol panellerini içerir:

- **Motion Status** - yük pozisyonu ve hızı ile motor akımına ilişkin bilgileri gösterir;
- **Drive IO** – sürücünün dijital giriş/çıkış durumlarını ve analog giriş seviyelerini gösterir;
- **CANopen Status** – CANopen durum makinesinin durumunu gösterir;
- **CANopen Objects** - en önemli CANopen nesnelerinin değerlerini gösterir;
- **Drive Status** - SRL, SRH, MER, DER ve DER2 register'larındaki bit değerlerini gösterir.

Varsayılan kontrol paneli içeriği, panel yüzeyine sağ tıklanıp **Customize** seçeneği seçilerek düzenlenebilir.

Ayrıca, **Control panels** menüsündeki seçenekler kullanılarak sıfırdan yeni bir kontrol paneli oluşturulabilir veya mevcut bir panel içe/dışa aktarılabilir.

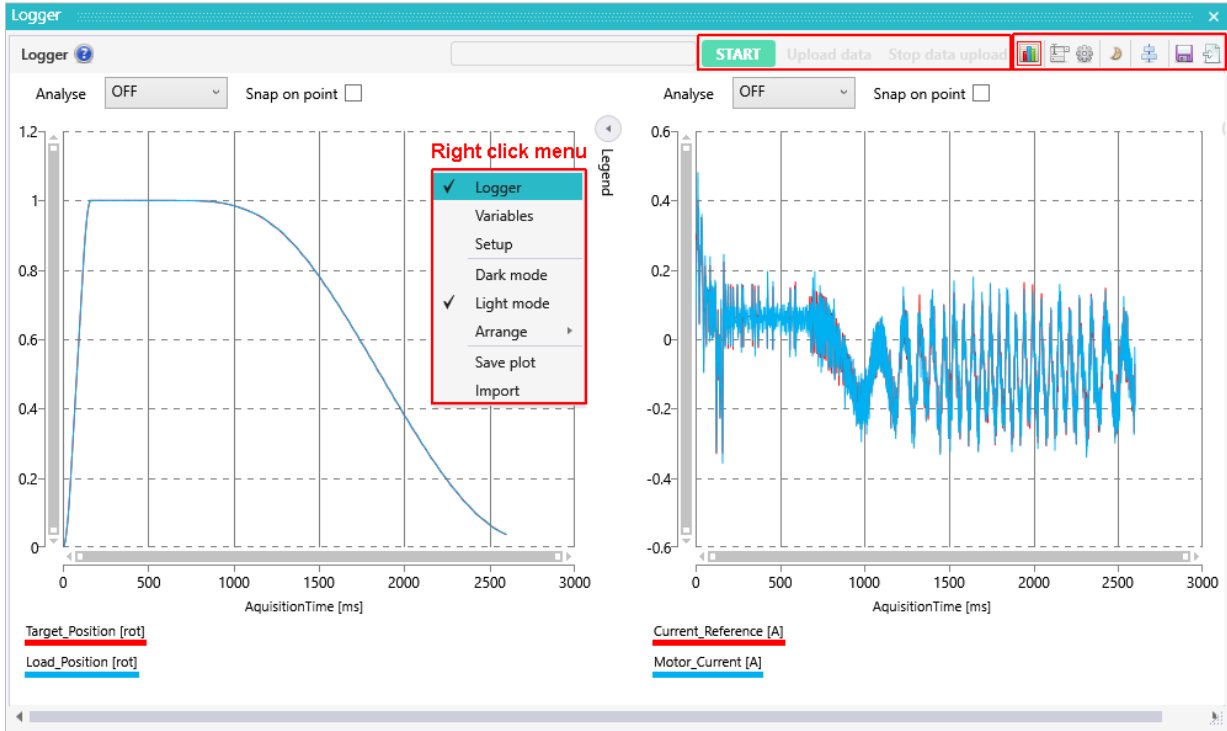


Şekil 7.2 – Control panels seçenekleri

Kontrol panellerinde gösterilen değerler, iletişim üzerinden alındığı için gerçek zamanlı olarak güncellenmez. Gerçek zamanlı ve senkronize sonuçlar için **Logger** veya **Scope** araçlarını kullanın.

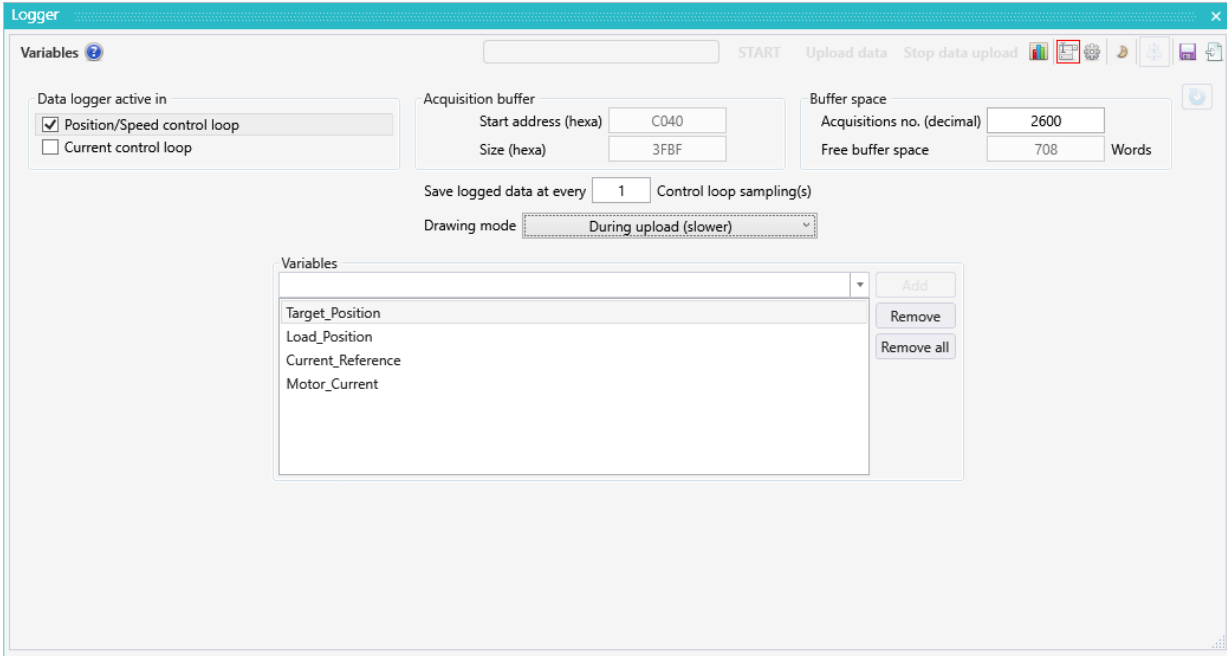
6.2 Logger Kullanımı

Logger yapılandırması ve kullanımı, Logger ribbon üzerindeki butonlar aracılığıyla veya Logger yüzeyine sağ tıklanarak açılan menü üzerinden gerçekleştirilebilir.



Şekil 7.3 – Logger yapılandırma ve kullanım seçenekleri

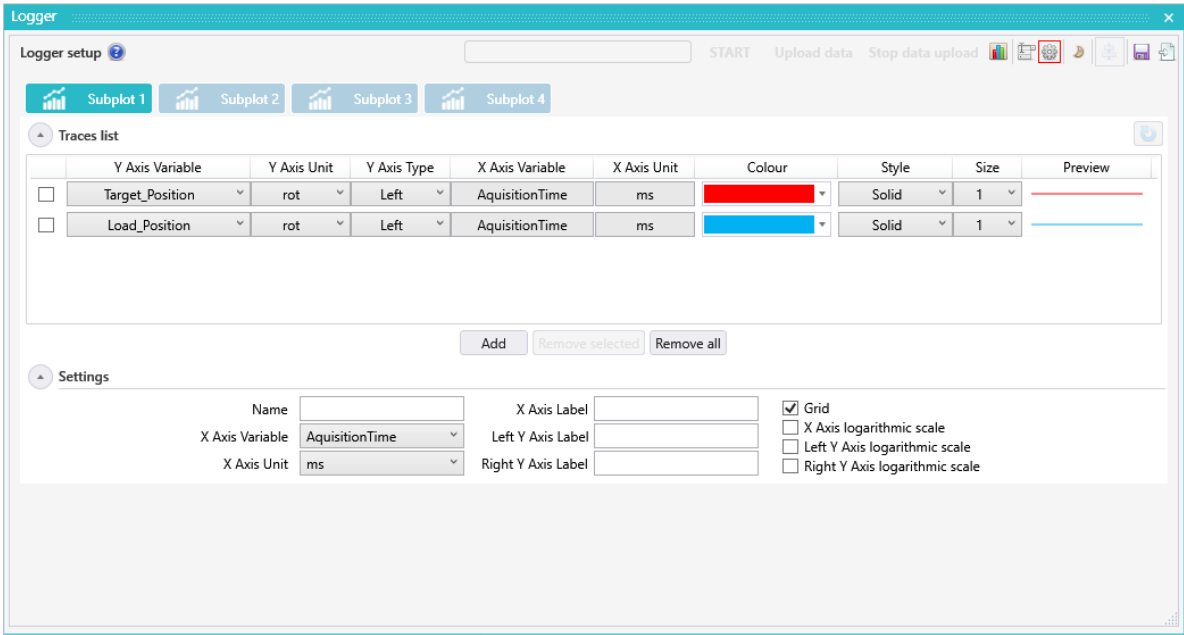
Logger kullanımındaki ilk adım, izlenecek değişkenlerin seçilmesi ve veri toplama (acquisition) parametrelerinin ayarlanmasıdır.



Şekil 7.4 – Logger veri toplama menüsü

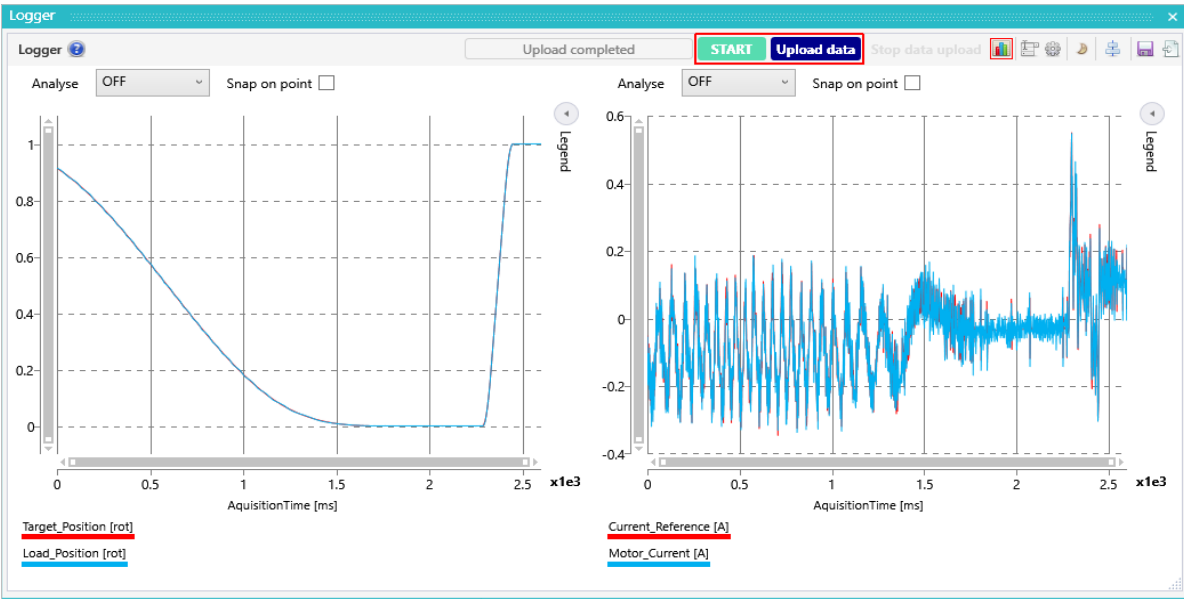
Not: Logger mekanizması, verileri kaydetmek için sürücünün RAM belleğini kullanır; bu nedenle oldukça hassastır. Ancak bellek kapasitesinin sınırlı olması nedeniyle yalnızca sınırlı sayıda veri noktası kaydedebilir ve grafikleyebilir.

Setup seçeneği, her bir **Subplot** için görüntülenecek değişkenlerin seçilmesine ve bu değişkenlerin parametrelenmesine olanak tanır.



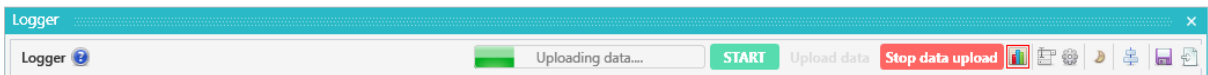
Şekil 7.5 – Logger setup menüsü

Veri toplama işlemi, TML uygulaması çalıştırıldığında otomatik olarak başlar; ancak Logger ribbon üzerindeki **Start** butonu kullanılarak manuel olarak da tetiklenebilir.



Şekil 7.6 – Logger kullanım seçenekleri

Logger ribbon üzerindeki **Upload data** butonu grafikleme (plotting) işlemini başlatır; **Stop data upload** seçeneği ise veri aktarımını durdurur ve Logger üzerinde yalnızca daha önce grafikleymiş olan kayıt kısmını bırakır.



Şekil 7.7 – Logger veri aktarımının iptali

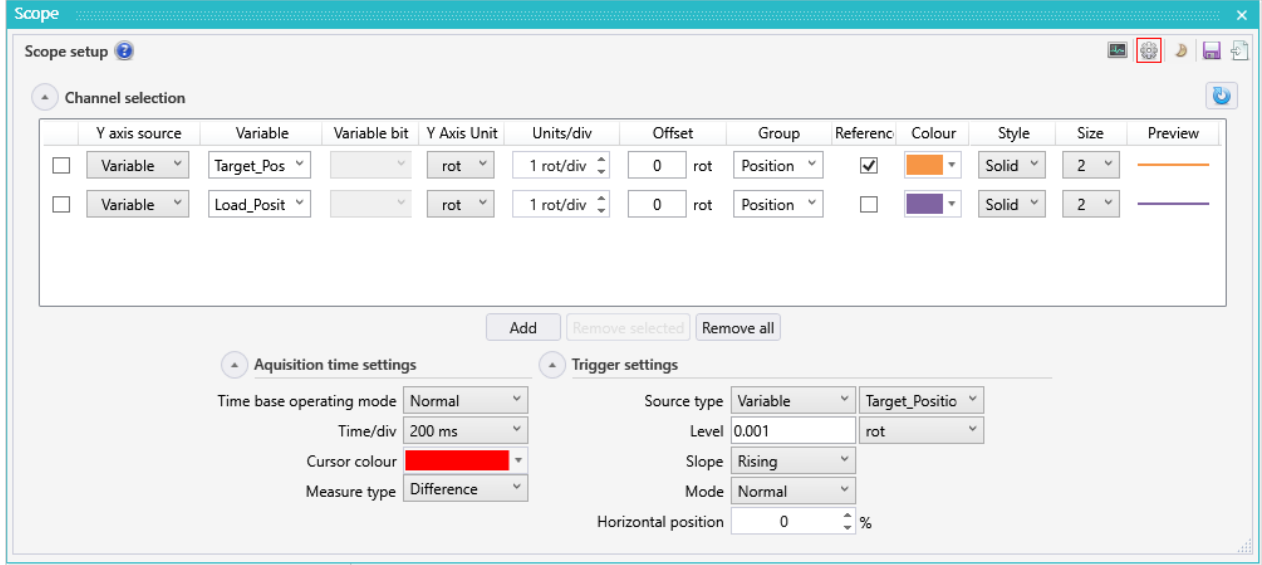
Grafikler, Logger ribbon üzerindeki **Save** butonu kullanılarak Logger dosyası (.lgs), metin dosyası (.txt), görüntü (.png) veya Excel dosyası (.csv) olarak kaydedilebilir.

Logger seçenekleri ve işlevleri, ana **Help** menüsünden erişilebilen ilgili help topics bölümünde veya Logger arayüzündeki "?" simgesine tıklanarak açılabilen yardım içeriklerinde detaylı olarak açıklanmıştır.

6.3 Scope Kullanımı

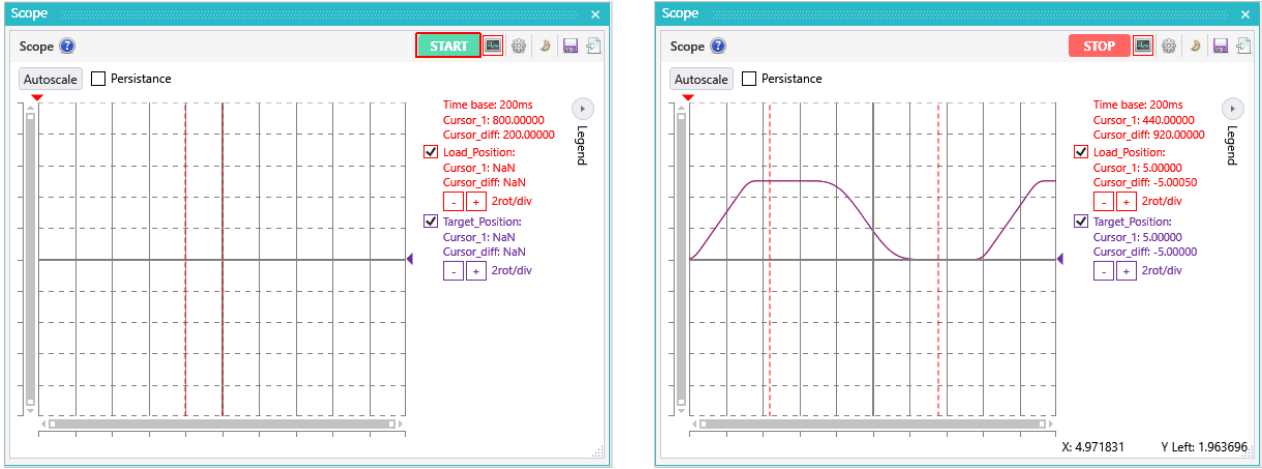
4 kanallı bir osiloskop gibi çalışan **Scope** aracı, farklı değişkenlerin, parametrelerin, register'ların veya giriş/çıkış durumlarının gerçek zamanlı değişimini izlemek üzere ayarlanabilir.

Scope **ayar sayfasına**, Scope ribbon üzerindeki buton aracılığıyla veya sağ tıklama menüsü üzerinden erişilebilir ve bu sayfa, mevcut 4 kanalın yapılandırılmasına ve veri toplama (acquisition) ile tetikleme (triggering) parametrelerinin ayarlanmasına olanak tanır.



Şekil 7.8 – Scope Setup penceresi

Veri toplama işlemi, Scope ribbon üzerindeki START butonuna basıldığında başlar ve Scope ribbon üzerindeki STOP butonu kullanılarak durdurulana kadar bilgiler sürekli olarak güncellenir.



Şekil 7.9 – Scope kullanılarak hareket değerlendirilmesi

Grafik ölçeği, Scope legend üzerindeki “-” ve “+” butonları veya “Autoscale” butonu kullanılarak anlık olarak değiştirilebilir.

Basit ölçümler (değer veya fark) için 1 veya 2 cursor kullanılabilir. Bu cursor'lar, **Scope setup** sayfasındaki **Acquisition time settings** grup kutusu üzerinden etkinleştirilebilir ve ayarlanabilir.

Grafikler, Scope ribbon üzerindeki **Save** butonu veya sağ tıklama menüsündeki **Save plot** seçeneği kullanılarak Scope dosyası (.osc2), metin dosyası (.txt), görüntü (.png) veya Excel dosyası (.csv) olarak kaydedilebilir.

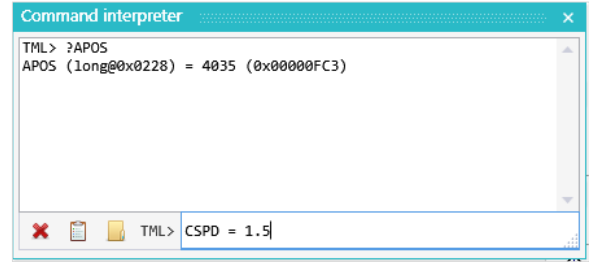
Tüm Scope seçenekleri ve işlevleri, ana **Help** menüsünden erişilebilen help topics bölümünde veya Scope arayüzündeki “?” simgesine tıklanarak açılabilen yardım içeriklerinde detaylı olarak açıklanmıştır.

6.4 Uygulama ve Sürücü Fonksiyonlarının Kontrolü veya Hata Ayıklama için Gelişmiş Araçlar

Komut Yorumlayıcı

Command Interpreter aracı, çevrimiçi sürücülere veya akıllı motorlara on-line TML komutları göndermeye olanak tanır.

Bu araç; TML verilerini (register'lar, parametreler ve değişkenler) ayarlamak veya okumak ya da bellek adreslerini görüntülemek için kullanılabilir.

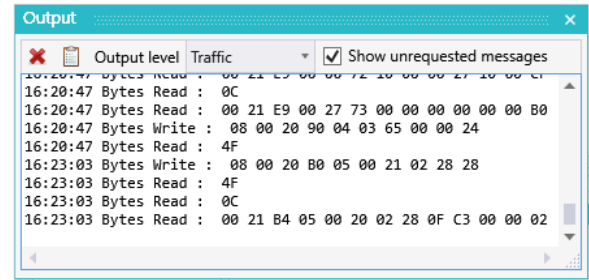


Şekil 7.10 – Command Interpreter genel görünümü

Çıkış Görünümü

Output penceresi, RS232 hattındaki veri trafiğinin izlenmesine ve aşağıdaki mesajların görüntülenmesine olanak tanır:

- hata mesajları
- uyarı mesajları
- RS232 veri trafiği
- istenmeyen mesajlar



Şekil 7.11 – Output view genel görünümü

Bellek

Memory penceresi, Technosoft sürücüsünün ve akıllı motorun EEPROM belleği içeriğinin görüntülenmesine ve hatta değiştirilmesine olanak tanır.

Not: Bu özellik çok düşük seviyeli (low-level) bir fonksiyon olduğundan, değiştirmeyi planladığınız her bir bellek adresinin Technosoft sürücüsü veya akıllı motor tarafından nasıl kullanıldığını derinlemesine bilmeden bellek içeriğinin değiştirilmesi önerilmez.

Address	Data (F12 - Refresh)							
4000	649C	23BF	4011	2398	4009	74C0	09C0	4009
4008	0404	5A57	FFFF	8000	0020	7484	02FB	400D
4010	0102	24A2	7A3B	0000	24A0	5555	0021	249E
4018	2710	0000	5909	FFFF	2000	5909	BFC1	8701
4020	5909	FFFF	4000	0108	700F	0408	70B9	02C2
4028	012C	0000	0408	26D2	0096	0000	24A2	7A3B
4030	0000	24A0	5555	0021	249E	0000	0000	5909
4038	FFFF	2000	5909	FFC1	8707	5B12	FFFE	0000
4040	0108	700F	0408	70B9	02C2	012C	0000	0408
4048	7400	4011	0001	0000	C800	0002	0000	000A
4050	C800	0002	0000	000A	C800	0003	0000	000A

Şekil 7.12 – Memory diyalog genel görünümü

Binary Code Viewer (İkili Kod Görüntüleyici)

Binary Code Viewer aracı, bir Technosoft sürücüsü veya akıllı motor ile mesaj alışverişini yapacak host'un programlanması için hızlı bir yöntem sunar. Bu araç, tüm iletişim tipleri ve desteklenen protokoller için bir TML komutunun nasıl paketleneceğini (encapsulate) gösterir. Ayrıca gönderilmesi gereken mesajların içeriğini ve varsa beklenen yanıtı görüntüler.

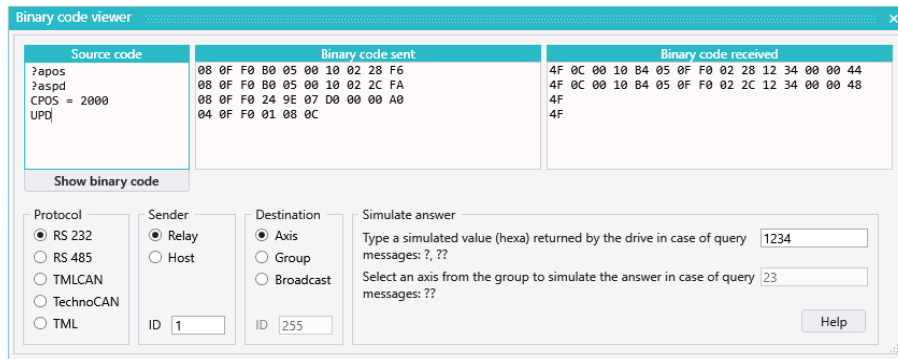
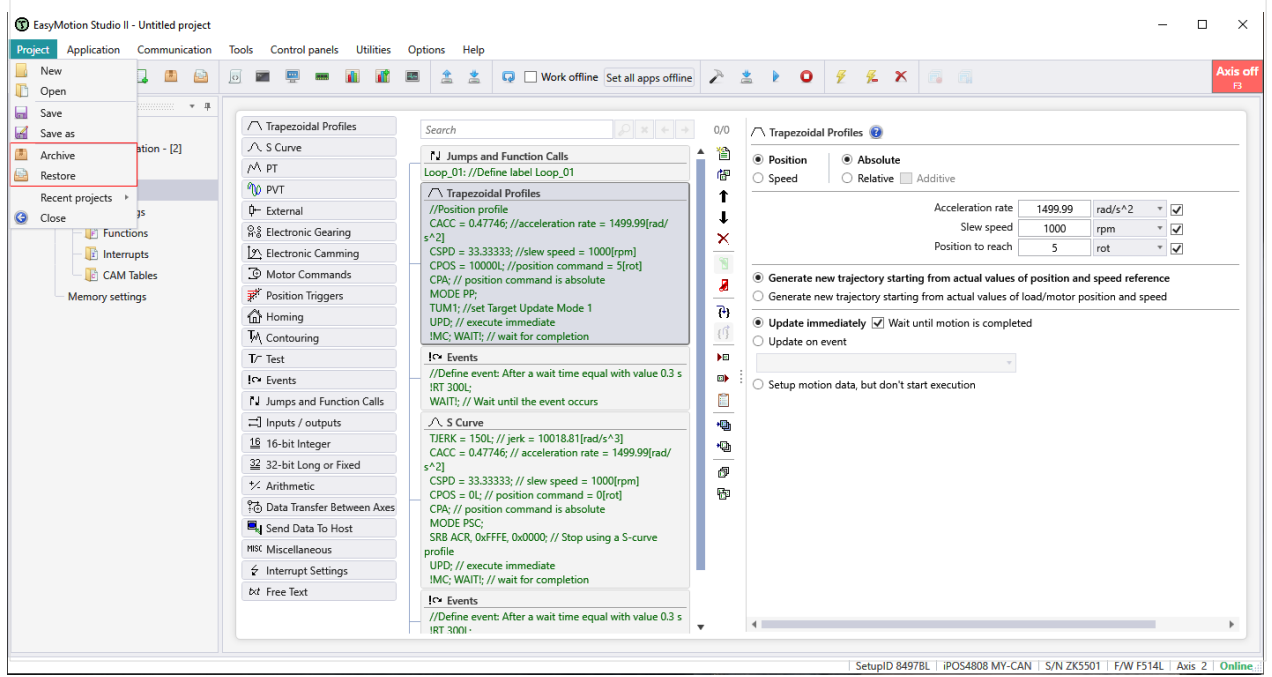


Figure 7.13 – Binary code viewer overview

Gelişmiş araçlar, EasyMotion Studio II Help topics bölümünde detaylı olarak açıklanmıştır (**Help | Help topics | Evaluation tools ...**).

7 Diğer – Proje Arşivleme ve Geri Yükleme

Projenin saklanması veya transfer edilmesi amacıyla, EasyMotion Studio II **Archive** ve **Restore** seçeneklerini içerir.



Şekil 7.1 – “Archive” ve “Restore” seçenekleri

Bir proje arşivi (“*.e2pf” uzantılı dosya), **Project | Archive** seçeneği kullanılarak oluşturulabilir. Proje arşivi; tüm proje bilgilerini, Logger ayarlarını, kontrol panellerini ve diğer özel dosyaları içerir. Bu arşiv, yedekleme yapmak veya projeyi diğer bilgisayarlara aktarmak için kullanılabilir.

Restore seçeneği, “*.e2pf” arşiv içeriğini çıkarır ve tam bir EasyMotion Studio II projesi olarak geri yükler. Geri yüklenen proje, arşivin oluşturulduğu proje ile birebir aynıdır.



T E C H N O S O F T
MOTION TECHNOLOGY

Rottek[®]
T e k n o l o j i