EtherCAT Gateway Modülü



SGEAP1-00-00-00-00_1

Genel Özellikler :

* EtherCAT Slave Cihazı
* Sadece Free Run Modu desteği. Senkron modlar (DC Synchronization, Sync0/Sync1 Event Triggering) desteklenmemektedir.
* PDO veri alanları sabit:
  0x1A00 (Giriş PDO, TxPDO)
  0x1600 (Çıkış PDO, RxPDO)
  Her biri 512 byte veri alanı kapasitesine sahiptir.
* PDO, Modbus eşleme (mapping) ve ESI dosyası üretimi için konfigürasyon yazılımı kullanılır.
* EtherCAT Döngü Süresi (Cycle Time): Master tarafından belirlenir, önerilen minimum döngü süresi ≥1 ms
* 2 adet RJ45 Ethernet portu (IN ve OUT), 100BASE-TX, Full-Duplex
* 1 adet RS485 portu ile Modbus RTU Master haberleşme
* Daisy Chain bağlantı desteği (Zincirleme bağlantı imkanı)
* Maksimum 255 WORD giriş ve 255 WORD çıkış verisi desteği
* Sync Manager kullanımı:
  - SM2: PDO Çıkışı (PDO Out)
  - SM3: PDO Girişi (PDO In)
* Slave Stack Versiyonu: SSC 5.13 tabanlı geliştirme altyapısı.
* Mini USB 2.0 üzerinden mapping, parametre ve yazılım güncelleme (boot) imkanı.
* DIP switch üzerinden Explicit Device ID (Belirgin Cihaz Kimliği) ayarlama imkanı
* LED Göstergeleri: MS, NS, RS485, Ethernet bağlantı LED'leri

Teknik Özellikler

Modül Tipi	SGEAP1-00-00-00-00_1
Besleme Gerilimi	24 VDC ±%20 (19,2 VDC - 28,8 VDC)
Güç Tüketimi	2W
USB Aygıtı	Mini USB, USB 2.0
Ethernet	10 / 100 Mbit, Full Duplex 2 adet RJ45 Ethernet soketi (IN/OUT)
Senkronizasyon Modu	
Free Run	Free Run, EtherCAT ağı üzerinde çalışan slave cihazların, gelen veri çerçevelerini senkronizasyon sinyali olmaksızın işlediği bir çalışma modudur. Bu modda, cihazlar belirli bir zamanlama (örneğin Sync0 veya Sync1) beklemeksizin, master cihazdan her yeni veri çerçevesi geldiğinde güncellenir ve çıktı üretir.
Obje Türleri	
PDO (Process Data Object)	PDO EtherCAT ağı üzerinde döngüsel olarak iletilen giriş ve çıkış verilerini temsil eden veri yapılarıdır. PDO yapıları, cihazın çevrimsel veri iletişimini sağlar ve yüksek hızlı haberleşme için optimize edilmiştir. Giriş ve çıkış proses verileri, PDO'lar aracılığıyla slave cihazlarla hızlı ve düzenli şekilde paylaşılır. PDO veri iletimi, herhangi bir ek protokol katmanı olmadan, doğrudan EtherCAT çerçevesi içinde gerçekleştirilir. Bu sayede minimum gecikmeyle yüksek performanslı haberleşme sağlanır. Her PDO, belirli sayıda alt nesneden (sub-index) oluşur ve her biri belirli bir veriyi temsil eder. Hangi verilerin hangi PDO üzerinden iletileceği bilgisi, PDO eşleme (mapping) yoluyla tanımlanır. Bu eşleme yapıları, nesne sözlüğü içinde 0x1600 serisi (RxPDO Mapping) ve 0x1A00 serisi (TxPDO Mapping) nesneleri aracılığıyla tanımlanır. İletilen verilerin kendisi ise çoğunlukla 0x6000–0x6FFF (giriş verileri) ve 0x7000–0x7FFF (çıkış verileri) aralığındaki nesneler üzerinden aktarılır. Bu yapılar, cihazın nesne sözlüğünde tanımlanır ve master cihaz tarafından okunarak çevrimsel veri akışı kurulur.

<table>
<tr><th colspan="2">Protokol</th></tr>
<tr><th colspan="2">SYNC Manager Yapısı</th></tr>
<tr><td>SM2 – Process Data Out (RxPDO)</td><td>Master’dan slave’e gönderilen çevrimsel (PDO) veriler bu kanal üzerinden taşınır. Hangi RxPDO nesnelerinin bu kanaldan iletileceği, 0x1C12 (RxPDO Assign) nesnesi üzerinden tanımlanır.</td></tr>
<tr><td>SM3 – Process Data In (TxPDO)</td><td>Slave’den master’a gönderilen çevrimsel (PDO) veriler bu kanal üzerinden taşınır. Hangi TxPDO nesnelerinin bu kanaldan iletileceği, 0x1C13 (TxPDO Assign) nesnesi üzerinden tanımlanır.</td></tr>
<tr><th colspan="2">Çalışma Modları</th></tr>
<tr><td>Init (Initialization)</td><td>Cihaz açıldığında ilk olarak bu duruma geçer. Haberleşme veri alanları henüz yapılandırılmamıştır. Bu modda yalnızca EEPROM erişimi mümkündür. Master, slave’in temel donanım bilgilerini bu aşamada alabilir.</td></tr>
<tr><td>Pre-Operational (Pre-Op)</td><td>Bu modda slave cihaz, CoE protokolü üzerinden yapılandırılabilir. Master cihaz bu aşamada slave’e parametre yazar, nesne sözlüğünü okur, SDO ile konfigürasyon yapar. PDO iletişimi henüz başlatılamaz.</td></tr>
<tr><td>Safe-Operational (Safe-Op)</td><td>Bu modda, PDO yapılandırması tamamlanmıştır, ancak cihaz çevrimsel veri iletişimine sınırlı şekilde katılır. Slave cihaz, giriş verilerini (TxPDO) üretir ve master bu verileri 0x6000–0x6FFF aralığından okuyabilir. Ancak master, slave'e çıkış verisi gönderemez; RxPDO veri yazımı (0x7000–0x7FFF) pasiftir ve slave'in fiziksel çıkışları aktif hale gelmez. Safe-Op modu, sistemin çalışmaya başlamadan önce güvenliğini kontrol etmek için kullanılan bir aşamadır.</td></tr>
<tr><td>Operational (Op)</td><td>Slave cihaz hem giriş hem çıkış verilerini aktif olarak iletir. PDO haberleşmesi başlamıştır ve cihaz tam olarak çalışır durumdadır.</td></tr>
<tr><th colspan="2">LED Göstergeler</th></tr>
<tr><td>Besleme</td><td>Normal gerilim seviyesinde sürekli yanar, düşük gerilimde 0.1s yanar 0.1s söner.</td></tr>
<tr><td>USB</td><td>USB kablosu takılı iken sürekli yanar, veri transferi sırasında yanıp söner</td></tr>
<tr><td>Ethernet</td><td>Sarı ve Yeşil Kapalı: Ethernet kablosu takılı değil veya takılı olan modülde enerji yok.
Yeşil Sürekli: Ethernet bağlantısı var. Haberleşme henüz başlamadı
<table>
<tr><th>Çalışma Modu</th><th>Status/Run LED Davranışı (Sarı LED)</th><th>Link/Activity LED (Yeşil LED)</th></tr>
<tr><td>Init</td><td>Kapalı</td><td>Hızlı yanıp sönme (~100 ms)</td></tr>
<tr><td>Pre-Op</td><td>Hızlı yanıp sönme (~200 ms)</td><td>Hızlı yanıp sönme (~100 ms)</td></tr>
<tr><td>Safe-Op</td><td>Yavaş yanıp sönme (~1 s)</td><td>Hızlı yanıp sönme (~100 ms)</td></tr>
<tr><td>Op</td><td>Sürekli</td><td>Hızlı yanıp sönme (~100 ms)</td></tr>
</table>
</td></tr>
<tr><td>MS</td><td>Sürekli (Yeşil) : Modül normal şekilde çalışıyor
Blink (Yeşil) : Modül konfigüre edilmedi veya geçersiz konfigürasyon
Sürekli (Kırmızı) : Hata algılandı
Modül boot modundayken kırmızı renkte 1s yanar 0.1s söner</td></tr>
<tr><td>NS</td><td>Kapalı : Ethernet bağlantısı algılanmadı / yok
Sürekli (Yeşil) : EtherCAT bağlantısı kuruldu, Gateway Op modda çalışıyor
Blink 500 ms (Yeşil) : EtherCAT bağlantısı kuruldu, Gateway Safe-Op modda çalışıyor
Blink 300 ms (Yeşil) : EtherCAT bağlantısı var ancak Gateway Safe Op ya da Op modda değil
Blink 300 ms (Kırmızı) : EtherCAT bağlantısı var ama haberleşme hatası</td></tr>
<tr><td>RS485</td><td>Sürekli (Yeşil) : Modül, Modbus Slave modülleriyle haberleşiyor.
Blink 500 ms (Yeşil) : Cihazda yüklü mapping yapılandırması yok.
Blink 100 ms (Kırmızı) : UART haberleşme için donanım seviyesinde hata oluştu.
Blink 500 ms (Kırmızı) : En az 1 Modbus paketi için hata kodu oluştu
Örnek: Bir slave modülle (3 sorgudan sonra) timeout oluştu
Modbus hata kodu (ör. illegal adres) döndü.</td></tr>
<tr><td>Dipswitch</td><td>Gateway cihaz için sabit ve manuel bir Explicit Device ID (Belirgin Cihaz Kimliği) belirlemek amacıyla kullanılır.</td></tr>
<tr><th colspan="2">Çevresel Özellikler</th></tr>
<tr><td>Çalışma Sıcaklığı</td><td>-10°...+60° C</td></tr>
<tr><td>Depolama Sıcaklığı</td><td>-20°...+70° C</td></tr>
<tr><td>İzolasyon</td><td>* Besleme ile USB arasında izolasyon yoktur
* Besleme ile Ethernet arasında izolasyon 1500VAC
* Besleme ile RS485 arasında izolasyon 500 VAC</td></tr>
</table>

Giriş

IPERTU EtherCAT Gateway modülü, seri tabanlı RS485 haberleşme arayüzüne sahip Modbus RTU destekli endüstriyel cihazların, herhangi bir donanım veya yazılım değişikliği gerektirmeden EtherCAT ağına entegre edilmesini sağlayan bir protokol dönüştürücü (Gateway) modüldür. EtherCAT Master ve RS485 üzerindeki Modbus RTU Slave cihazlar arasında çift yönlü veri iletimine olanak tanır.

Sistemde bir veri değişim tamponu görevi üstlenerek iki farklı haberleşme yapısına sahip sistemin yüksek hızlı ve güvenilir şekilde entegre edilmesini sağlar. Bu sayede eski nesil otomasyon cihazlarının modern EtherCAT altyapılarına dahil edilmesi kolaylaşır.

Seri haberleşme sırasında veri uzunluğu (data size) sabit olarak 8 bit şeklindedir. Modül, en fazla 115200 bps baud hızına kadar haberleşmeyi destekler.

Gateway cihazı, 255 adet WORD (16-bit) giriş ve 255 adet WORD çıkış verisini destekler. Giriş ve çıkış PDO veri blokları sabit adreslerde tanımlıdır (0x6000 ve 0x7000) ve EtherCAT master sistemlerine standart PDO yapısıyla sunulur. RS485 hattı üzerinden ise bu veriler Modbus RTU protokolüyle slave cihazlardan okunur ya da bu cihazlara yazılır.

EtherCAT destekli tüm PLC sistemleriyle uyumlu olarak çalışabilir ve Modbus tarafındaki veri eşlemeleri, kullanıcı tarafından yapılandırılabilir. Mapping ve parametre ayarları özel PC yazılımı aracılığıyla yapılır ve USB Mass Storage arayüzü üzerinden cihaza aktarılır.

PDO Eşleme (Mapping) ve Adresleme Yapısı

IPERTU EtherCAT Gateway modülü, EtherCAT master sistemine sabit veri blokları üzerinden çevrimsel veri aktarımı sağlar. Giriş verileri için 0x6000 adres aralığı, çıkış verileri için ise 0x7000 adres aralığı kullanılır. Her iki blok da 255 adet 16-bit WORD'den oluşur ve bu veri alanları, EtherCAT PDO mekanizması ile master cihaza sunulur.

PDO (Process Data Object) yapıları, EtherCAT sistemlerinde çevrimsel veri iletiminin temelini oluşturur. Hangi verilerin EtherCAT master tarafından okunacağı veya yazılacağı bilgisi, PDO eşleme nesneleri olan **RxPDO (0x1600 serisi)** ve **TxPDO (0x1A00 serisi)** üzerinden tanımlanır.

Bu modülde:

- **0x1600 serisi**: EtherCAT master'dan gelen çıkış verilerinin eşlemesini yapar (RxPDO).
- **0x1A00 serisi**: EtherCAT master'a gönderilecek giriş verilerinin eşlemesini yapar (TxPDO).
- **0x7000 serisi**: RxPDO nesneleri tarafından işaret edilen çıkış veri alanıdır.
- **0x6000 serisi**: TxPDO nesneleri tarafından işaret edilen giriş veri alanıdır.

Veri alanlarındaki:

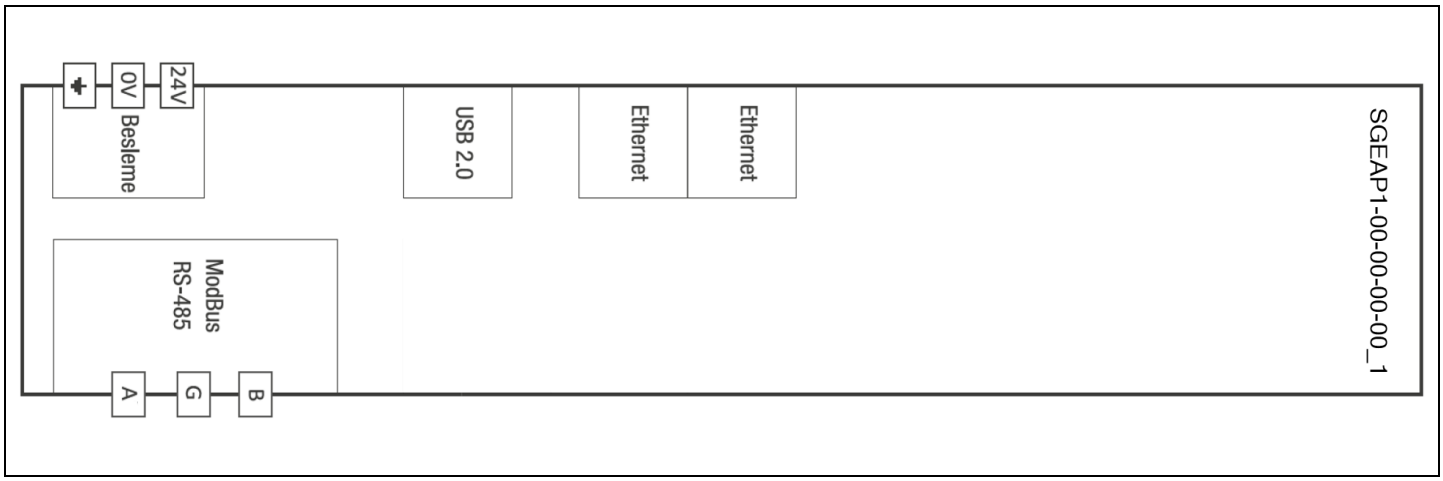
- **Sub-index 01 ve 02**, sistem parametreleri için ayrılmıştır (SYSIN1, SYSIN2, SYSOUT1, SYSOUT2)
- **03–255 arası sub-index'ler**, kullanıcı tarafından tanımlanabilir proses verilerini temsil eder.

PDO eşleme yapısı, sabit değildir. Kullanıcı, özel konfigürasyon yazılımı aracılığıyla hangi sub-index'lerin çevrimsel olarak iletileceğini belirleyebilir. Oluşturulan yapılandırma, USB Mass Storage arayüzü üzerinden cihaza aktarılır. ESI dosyası da aynı yazılım aracıyla otomatik olarak üretilir.

EtherCAT Gateway Modül Parametreleri (6000 ve 7000h)

Adres (Hex)	R/W	Parametre İsmi	Tanımlama	Varsayılan
6000:01	R	SYSIN1 Cihaz Kodu ve UART Hataları	**b0-b7** : modül tipini gösterir. Koda göre modül tipi: SGEAP1-00-00-00-00_1 = 0xC5 **b8** : Düşük besleme gerilimi **b9** : RS485 haberleşme hatası **b10-b13** : UART hatası 0x00- Hata yok. UART sorunsuz çalışıyor. 0x01- Eşlik Biti Hatası: Gönderilen/Alınan veriyle belirlenen eşlik biti (parite) uyuşmuyor. 0x02- Gürültü Hatası: Alım sırasında hatalı seviyeler tespit ediliyor. 0x04- Çerçeveleme Hatası: Stop bit doğru algılanamazsa oluşur. 0x08- Taşma Hatası: UART Rx Buffer taşması durumunda oluşur.	
6000:02	R	SYSIN2 Modbus Haberleşme Hataları	**b15-b8** : Hata tespit edilen mapping sıra numarasını gösterir. **b7-b0** : Modbus hata kodunu gösterir. Örnek: 0- Hata yok. 0x01- İllegal Fonksiyon Kodu 0x02- İllegal Adres 0x03- İllegal Data	
7000:01	W	SYSOUT1	Rezerve	0
7000:02	W	SYSOUT2	3083 değeri yazılarak cihazın yazılımsal reset alması sağlanır. Reset işleminden sonra SYSOUT2'ye 0 yazınız.	0

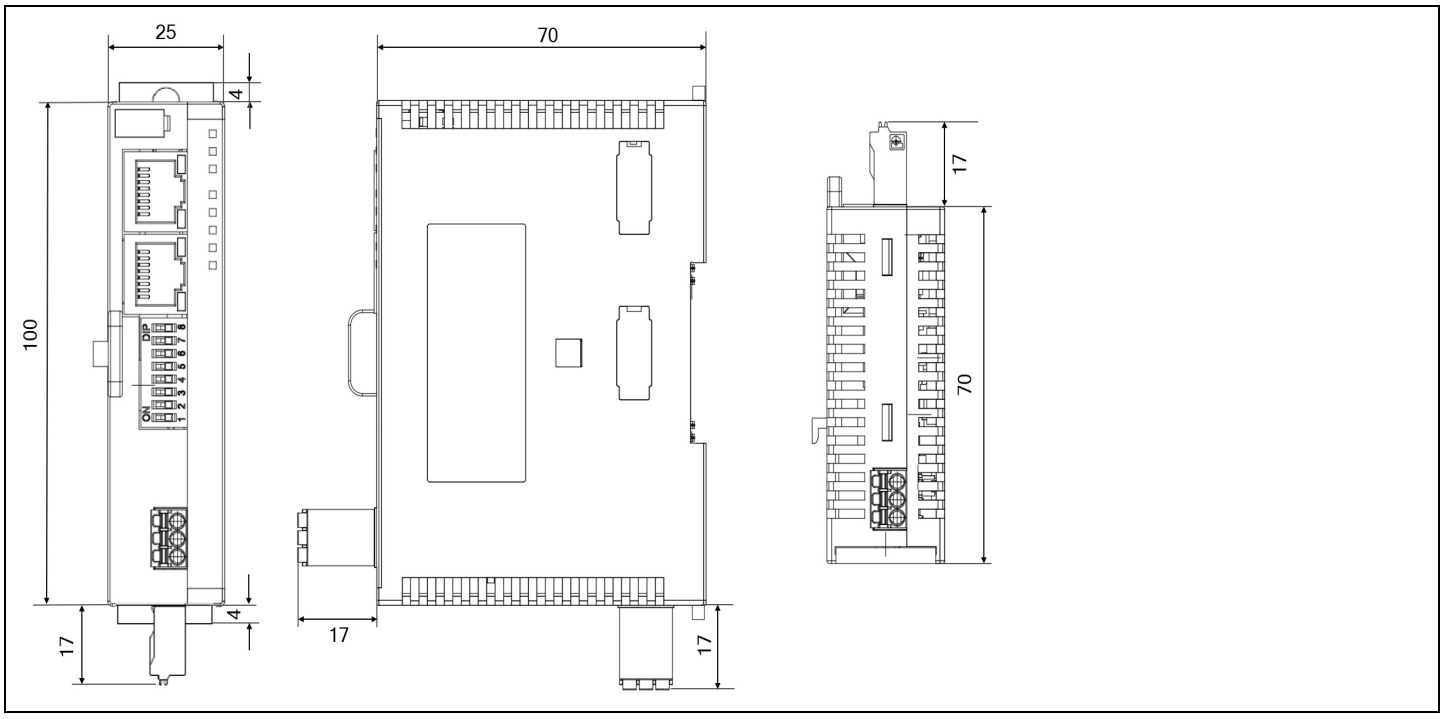
Bağlantı Şeması



** Giriş/Çıkış terminallerine kesinlikle AC besleme bağlamayınız. Aksi halde modül zarar görebilir. Enerji vermeden önce modülün tüm bağlantılarını kontrol ediniz. Elektriksel gürültüyü önlemek için topraklamanın düzgün yapıldığına emin olunuz. Besleme terminalindeki toprak bağlantısını topraklayınız. Enerjili iken modül terminallerine müdahale etmeyiniz.*

** RS485 haberleşme bağlantısı için; Terminal direncini, haberleşme hattının sonundaki modülün A & B uçları arasına 120R olacak şekilde bağlayınız. Haberleşme kablosunu ekranlı (shield) ve sarmal çiftli (twisted-pair) kullanınız. Kablonun ekran bağlantısını topraklayınız.*

Boyutlar



Ürün Sipariş Kodu

Ethernet Gateway Modülü	Ethernet	USB	Dipswitch	RS485
SGEAP1-00-00-00-00_1	2 x 10/100 Mbit	+	+	+

EMKO EtherCAT Gateway Configurator Yazılımı

1. Genel Bakış

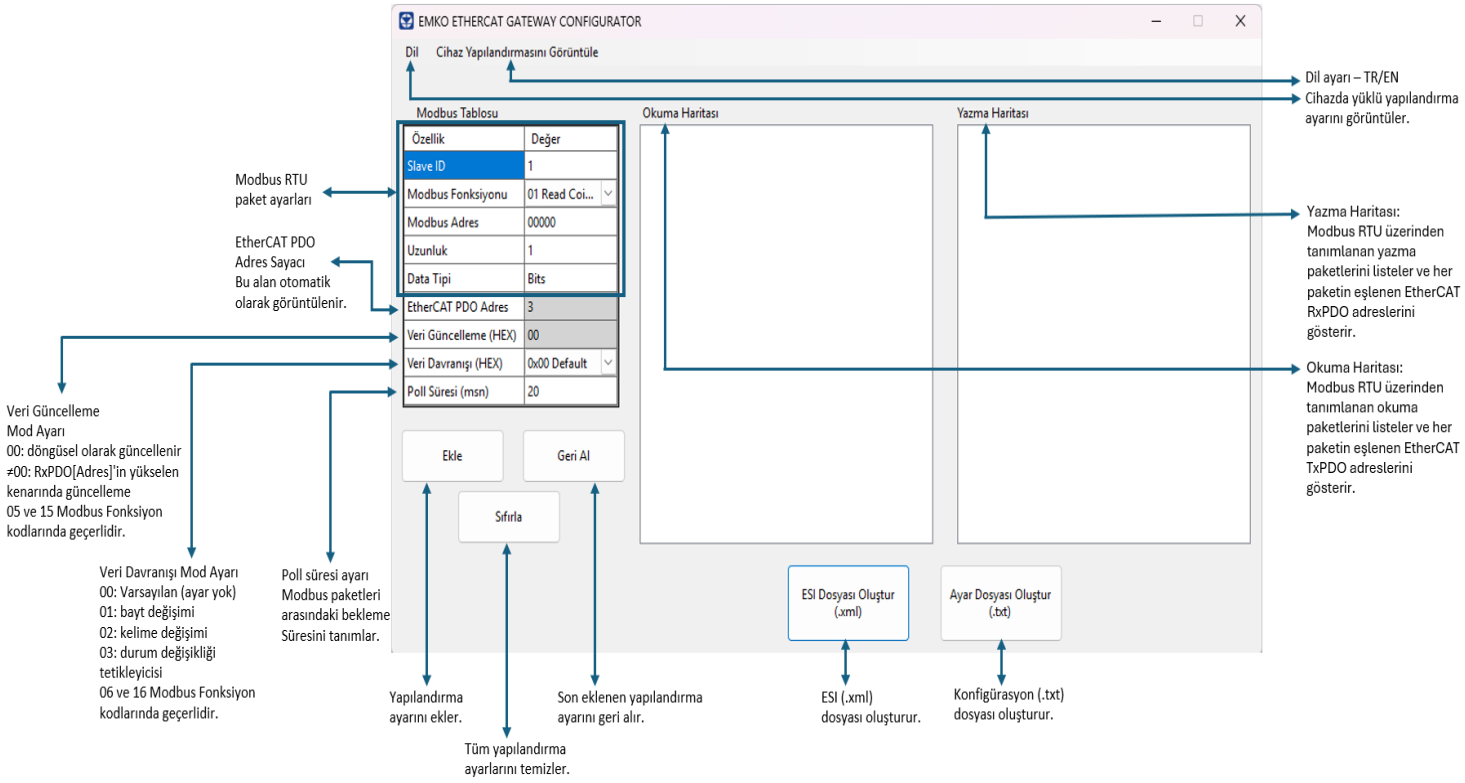
EMKO EtherCAT Gateway Configurator yazılımı, IPERTU EtherCAT Gateway cihazları için geliştirilmiş bir yapılandırma aracıdır. Bu yazılım aracılığıyla, IPERTU EtherCAT Gateway ile RS-485 haberleşen Modbus RTU slave cihazlar arasındaki haberleşme yapısı kullanıcı dostu bir arayüz üzerinden tanımlanabilmekte ve okuma/yazma verilerinin EtherCAT PDO yapısıyla doğru biçimde eşlenmesi sağlanmaktadır.

Temel Özellikler

- **Modbus Mapping Tanımı**:
  Slave ID, fonksiyon kodu (03 Read Holding Registers, 06 Write Single Register vb.), Modbus register adresi, veri uzunluğu, veri güncelleme davranışları ve poll süresi gibi parametreler kullanıcı tarafından ayarlanabilir.
- **EtherCAT PDO Eşleme**:
  Tanımlanan Modbus paket yapıları, otomatik olarak EtherCAT PDO adresleri ile ilişkilendirilir ve EtherCAT master tarafından tanınacak şekilde yapılandırılır.
- **Yazma ve Okuma Haritası Oluşturma**:
  Arayüzde yer alan "Yazma Haritası" ve "Okuma Haritası" ile, hangi verinin hangi EtherCAT PDO adresine ve hangi Modbus slave cihazına karşılık geldiği açıkça takip edilebilir.
- **Dosya Oluşturma Özelliği**:
  Konfigürasyon tamamlandığında, iki farklı dosya oluşturulur:
  - **ESI Dosyası (.xml)**: EtherCAT master sistemin cihaza ait PDO yapılarını tanıyabilmesi için gerekli XML tabanlı açıklama dosyası.
  - **Ayar Dosyası (.txt)**: Modbus ve EtherCAT konfigürasyon parametrelerini içeren, Gateway cihazına USB Mass Storage üzerinden aktarılabilen ayar dosyası.

2. Arayüz

EMKO EtherCAT Gateway Configurator yazılımının arayüzü Şekil 1'de gösterilmektedir. Şekil üzerinde, Modbus RTU paket ayarlarının yapıldığı alan, okuma ve yazma haritaları, dil seçimi, yapılandırma dosyası oluşturma butonları ve diğer temel işlevler açıklanmıştır.



Şekil 1. EMKO EtherCAT Gateway Configurator yazılım arayüzü

3. Modbus Tablosu Bölümü

Modbus RTU paketlerinin tanımlandığı alandır. Burada aşağıdaki parametreler belirlenir:

- **Slave ID:** Hangi Modbus slave cihazı ile haberleşileceğini belirtir.
- **Modbus Fonksiyonu:** Kullanılacak fonksiyon kodu seçimi yapılır.
- **Modbus Adres:** Okuma veya yazma yapılacak başlangıç adres değeri belirtilir.
- **Uzunluk:** Okunacak/yazılacak veri uzunluğu ayarlanır.
- **Data Tipi:** Modbus paketindeki veri tipini gösterir. 01, 02, 05, 15 fonksiyon kodlarıyla okunan/verilen veriler Modbus tarafında bit tabanlıdır; ancak EtherCAT PDO yapısında tüm veriler 16-bit WORD olarak yansıtılır.
- **EtherCAT PDO Adresi:** Modbus verilerinin EtherCAT PDO yapısında eşleneceği sub index değeridir. Otomatik artar.
- **Veri Güncelleme (HEX):** Yazma verisinin ne zaman güncelleneceğini belirleyen mod ayarı.
- **Veri Davranışı (HEX):** Veri değişiminde izlenecek davranışı belirleyen mod ayarı
- **Poll Süresi (msn):** Sorgular arasındaki bekleme süresinin(ms cinsinden) tanımlandığı alan.

3.1. Sınır değerler

- **Slave ID :** 1- 255 değer aralığı
- **Modbus Fonksiyonu:** Aşağıda belirtilen modbus fonksiyonları kullanılabilmektedir.
    - 01 Read Coil Status
    - 02 Read Input Status
    - 03 Read Holding Registers
    - 04 Read Input Registers
    - 05 Write Single Coil
    - 06 Write Single Register
    - 15 Write Multiple Coils
    - 16 Write Holding Registers
- **Modbus Adres :** 0-65535 değer aralığı
- **Uzunluk:** Bir modbus paketi için izin verilen maksimum uzunluk değerleri, kullanılan Modbus fonksiyon koduna göre belirlenmiştir ve Modbus RTU standardına uygundur.
    - 01 Read Coil Status – 2000 bit
    - 02 Read Input Status – 2000 bit
    - 03 Read Holding Registers – 125 kelime
    - 04 Read Input Registers – 125 kelime
    - 05 Write Single Coil – 1 bit
    - 06 Write Single Register – 1 kelime
    - 15 Write Multiple Coils – 1968 bit
    - 16 Write Holding Registers – 123 kelime
- **EtherCAT PDO Adresi :** Modbus verilerinin EtherCAT PDO yapısındaki karşılığını belirtir. Adresleme otomatik olarak artar
    - Cihaz, RxPDO (0x1600) ve TxPDO (0x1A00) alanlarında toplam 255 WORD giriş/çıkış verisini destekler.
    - Her iki PDO'da da 2 WORD sistem tanılama alanı ayrıldığından, kullanıcı tanımlı veri için maksimum 253 WORD kullanılabilir.
    - Bit tabanlı Modbus fonksiyonları (01, 02, 05, 15) EtherCAT tarafına WORD olarak aktarılır. Bit sayısı, gerekli WORD sayısına dönüştürülür.
- **Veri Güncelleme (HEX) :** 0x00-0xFF değer aralığı
- **Veri Davranışı (HEX) :** Aşağıda belirtilen veri davranış modları kullanılabilmektedir.
    - 0x00 Default
    - 0x01 Byte Swap
    - 0x02 Word Swap
    - 0x03 State Change
- **Poll Süresi (msn) :** 1-255 değer aralığı

3.2.Veri Güncelleme Modu

Bu alan, bit tabanlı Modbus yazma fonksiyonlarında (05 – Write Single Coil, 15 – Write Multiple Coils) kullanılan, hangi bitin 0'dan 1'e yükseldiğinde (rising edge) veri gönderileceğini belirleyen bir maskeleme mekanizmasıdır. Kullanıcı, ilgili biti temsil eden hexadecimal maskeleme değeri girer.

Çalışma Prensibi

- **0x00 (Varsayılan Değer):** Paket her Modbus döngüsünde çalıştırılır ve sürekli olarak iletilir.
- Girilen maskeleme değeri, gönderilecek Modbus verisi ile **mantıksal AND (bitwise AND)** işlemine tabi tutulur.
- Eğer maskeleme yapılan bit **0'dan 1'e yükseldiyse**, yani **rising edge** algılandıysa, ilgili Modbus komutu paketlenir ve Modbus hattına gönderilir.
- **Sürekli veri gönderimi yapılmaz.** Bu sayede haberleşme hattı gereksiz veri trafiğinden korunur.
- **Tekrar tetikleme için:** İlgili bitin önce **0** seviyesine düşürülmesi, ardından tekrar **1** yapılması gerekir.

Kullanım Örneği – Fonksiyon 05 (Write Single Coil)

- Kullanıcı **Veri Güncelleme** alanına 0x01 değerini girer (maskeleme bit 0).
- Modbus'a gönderilecek veri **01 05 02 00 FF 00** ise (FF = çıkış 1 seviyesinde), paket **çalıştırılır** ve Modbus hattına gönderilir.
- Çıkış yeniden tetiklenmek istenirse, önce 00 (0 seviye), sonra tekrar FF (1 seviye) gönderilmelidir.

Avantajları

- Haberleşme hattını gereksiz veri yükünden korur.
- Sadece olay bazlı (event-driven) veri aktarımı sağlar.

3.3.Veri Davranışı Modu

Bu alan, Modbus yazma fonksiyonları 06 (Write Single Register) ve 16 (Write Multiple Registers) için kullanılabilir ve EtherCAT'ten alınan verilerin Modbus hattına nasıl aktarılacağını belirler:

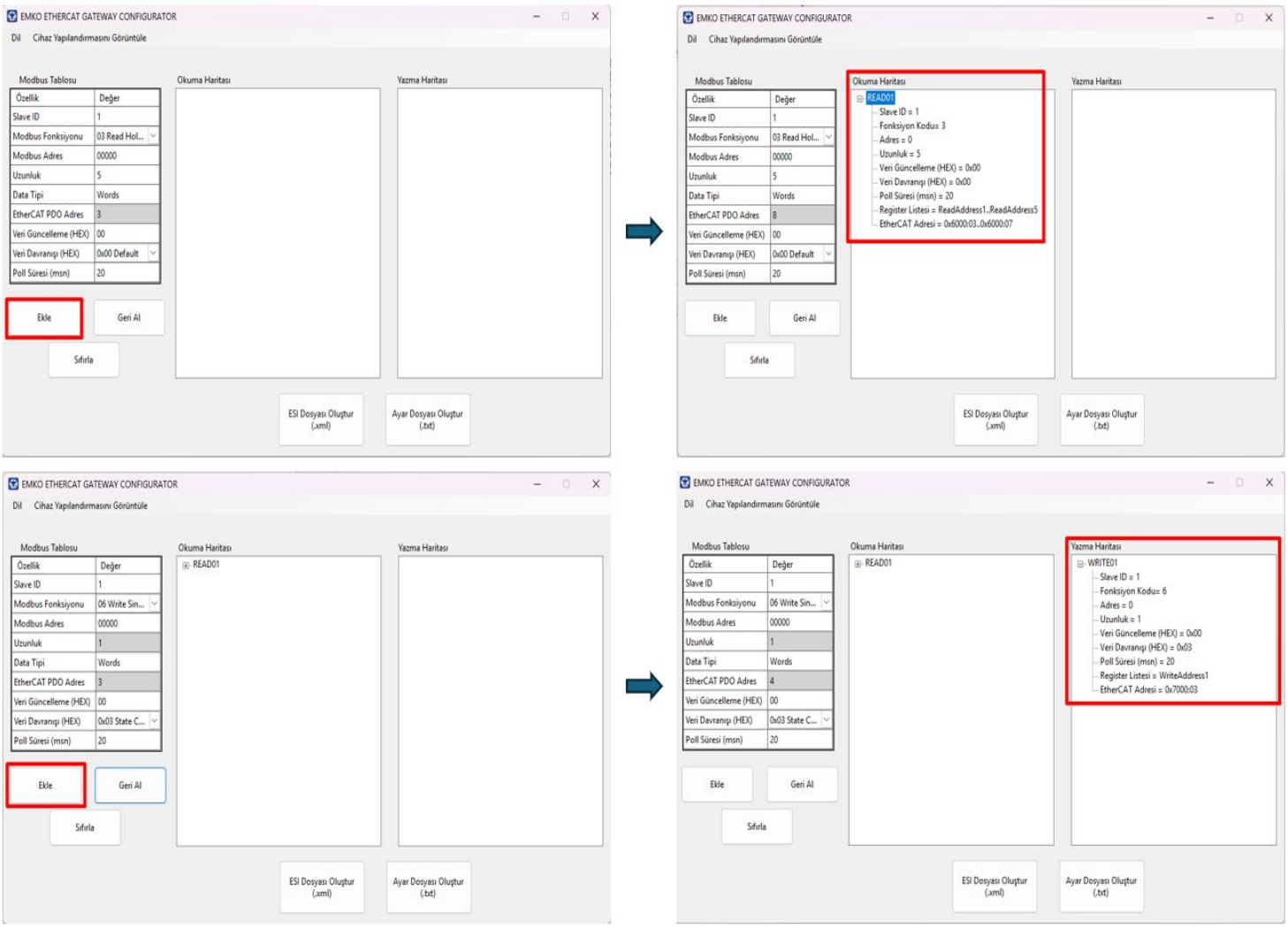
- **0x00 – Varsayılan (Default):** Veriler doğrudan aktarılır, herhangi bir ek işlem yapılmaz.
- **0x01 – Byte Swap:** Her 16-bit veri içindeki byte sırası değiştirilir. (Örn. 0x1234 → 0x3412)
- **0x02 – Word Swap:** Yalnızca Fonksiyon 16 ile ve **uzunluğun 2'nin katı olduğu durumlarda** geçerlidir. 32-bit (veya daha uzun) veri bloklarında, 16-bit word'lerin sırası değiştirilir. (Örn. [0x1111, 0x2222] → [0x2222, 0x1111])
- **0x03 – State Change:** EtherCAT tarafındaki RxPDO verisi değiştiğinde, Modbus paketi **yalnızca bir kez** gönderilir. Aynı veri tekrar edilmez; yeni bir değişiklik olduğunda tekrar iletilir. Bu sayede sürekli yazma işlemleri engellenir ve Modbus hattı gereksiz yükten korunur.

4. Okuma/Yazma Haritaları Bölümü

Okuma ve Yazma Haritaları, Modbus Tablosu üzerinden tanımlanan paketlerin EtherCAT PDO alanlarına nasıl eşlendiğini gösterir.

- **Harita Oluşturma:** Modbus Tablosu'nda gerekli parametreler (Slave ID, fonksiyon kodu, adres, uzunluk vb.) tanımlandıktan sonra **Ekle** butonuna basıldığında, ilgili paket otomatik olarak **Okuma Haritası (Read Map)** veya **Yazma Haritası (Write Map)** alanına eklenir. Fonksiyon koduna göre, paket **TxPDO** (okuma) veya **RxPDO** (yazma) grubuna yerleştirilir.
- **Register Listesi:** Eklenen her veri seti, otomatik olarak **ReadAddressX** veya **WriteAddressX** biçiminde isimlendirilir. Bu isimlendirme, veri sırasına göre yapılır. Örneğin, 5 kelimelik bir okuma paketi **ReadAddress1...ReadAddress5** olarak listelenir.
- **EtherCAT PDO Adresleri:** Her paket için EtherCAT tarafında kullanılan PDO adres aralığı da otomatik oluşturulur. Örneğin, **ReadAddress1...ReadAddress5** aralığı **0x6000:03...0x6000:07** olarak gösterilir. PDO adreslemesinde ilk iki alt indeks (sub-index) sistem tanıma bilgilerine ayrıldığından kullanıcı verileri üçüncü indeksten itibaren başlar.
- **Görselleştirme:** Harita üzerinde seçilen bir paketin ayrıntıları (Slave ID, fonksiyon kodu, Modbus adresi, uzunluk, güncelleme modu, PDO adresi vb.) detaylı biçimde listelenir. Bu yapı sayesinde, hangi Modbus verisinin EtherCAT üzerinde hangi PDO adresine denk geldiği açıkça görülebilir.

Örnek bir ekleme işlemi ve eklenen paketin ayarlarının Okuma/Yazma Haritalarında nasıl listelendiği, ayrıca ilgili verilerin EtherCAT PDO adreslerine nasıl eşlendiği Şekil 2'de gösterilmektedir.



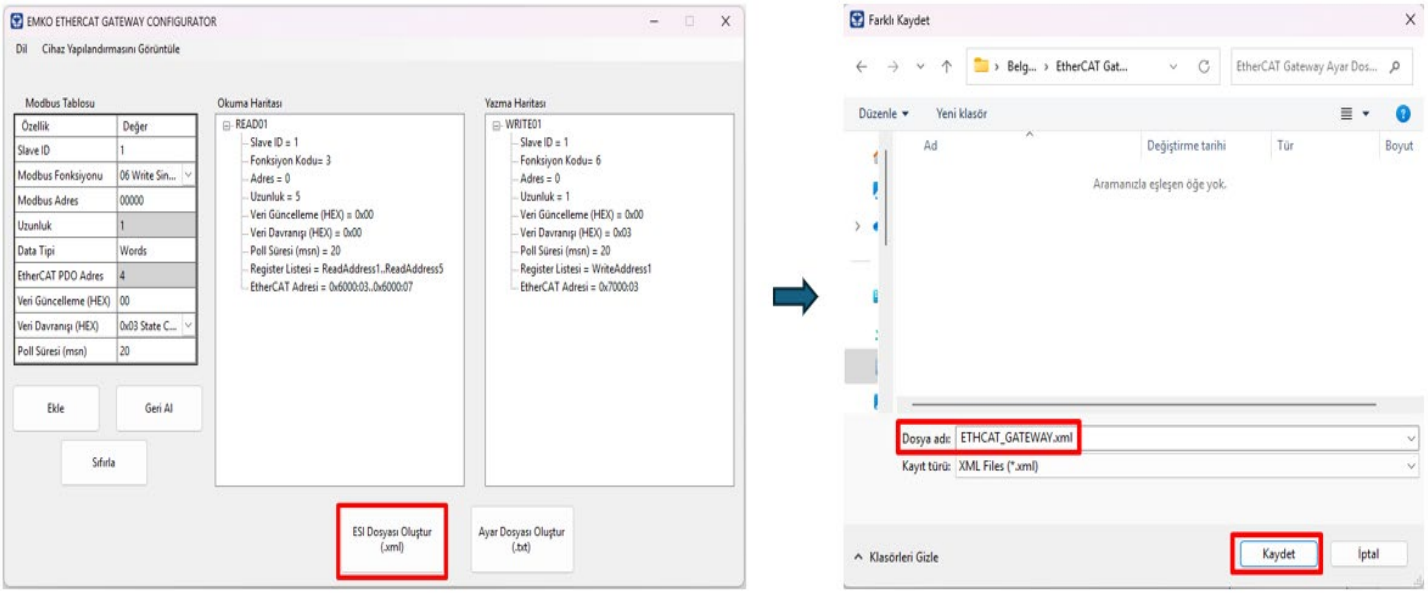
Şekil 2. EMKO EtherCAT Gateway Configurator yazılımında örnek bir paket ekleme işlemi ve Okuma/Yazma Haritalarında görüntülenmesi

5. Dosya Oluşturma

Konfigürasyon tamamlandıktan sonra, ayarların EtherCAT master tarafından tanınması ve EtherCAT Gateway cihazına uygulanması için iki farklı dosya oluşturulmaktadır:

- **ESI Dosyası (.xml)**:
  Modbus tablosunda tanımlanan okuma ve yazma paketlerine göre EtherCAT PDO alanları otomatik olarak düzenlenir ve bu bilgiler XML tabanlı ESI dosyasına eklenir. Oluşturulan ESI dosyası, EtherCAT master sistemlerin cihazın PDO yapılarını tanımasını sağlar.

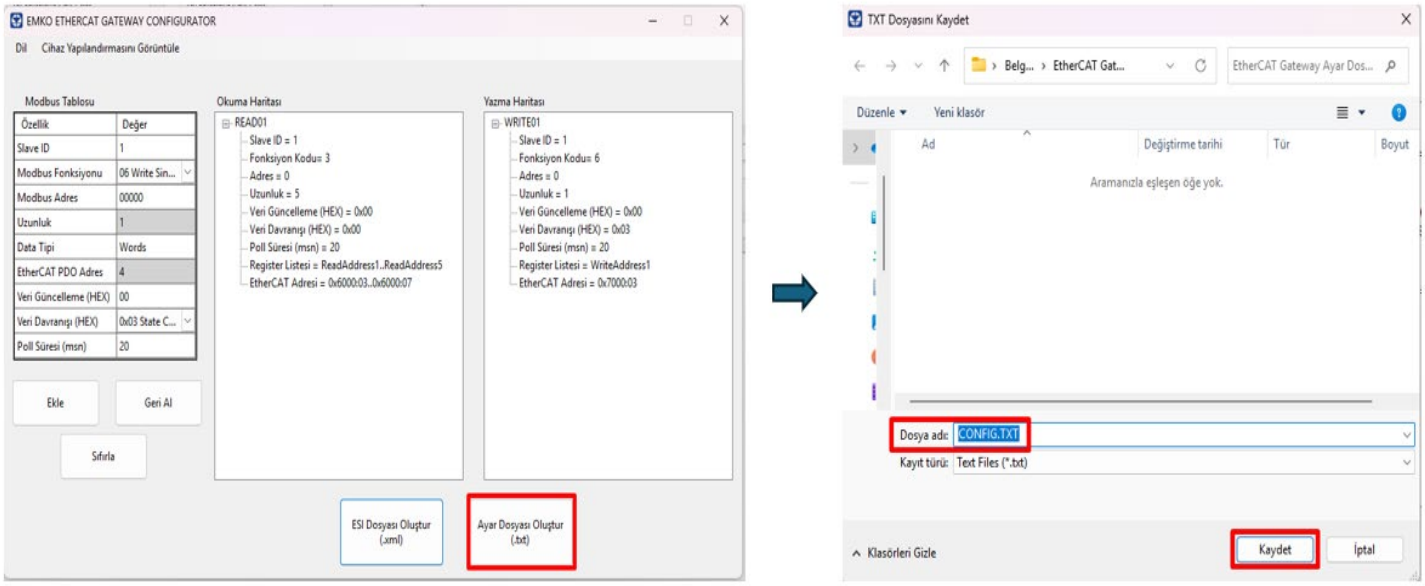
  ESI Dosyası Oluştur (.xml) butonu ile dosya oluşturma işlemi gerçekleştirilmektedir. İlgili butona tıklanıp bir dosya kaydetme penceresine erişilirerek dosyanın kaydedileceği konum belirlenir ve Kaydet seçeneği ile işlem tamamlanır. ESI dosyası, varsayılan olarak ETHCAT_GATEWAY.xml adıyla oluşturulur. Örnek bir ESI dosya oluşturma süreci Şekil 3'te gösterilmiştir.



Şekil 3. EMKO EtherCAT Gateway Configurator yazılımında örnek bir ESI dosyası oluşturma süreci

- **Ayar Dosyası (.txt)**:
  Konfigürasyon yazılımında yapılan tüm ayarlar (Modbus tabloları, EtherCAT PDO eşlemeleri, veri güncelleme ve veri davranışı modları dahil) güvenli bir biçimde .txt formatında saklanır.

  Ayar Dosyası Oluştur (.txt) butonu ile dosya oluşturma işlemi gerçekleştirilmektedir. İlgili butona tıklanıp bir dosya kaydetme penceresine erişilirerek dosyanın kaydedileceği konum belirlenir ve Kaydet seçeneği ile işlem tamamlanır. Ayar dosyası, varsayılan olarak CONFIG.TXT adıyla kaydedilir. Örnek bir ayar dosyası oluşturma süreci Şekil 4'te gösterilmiştir.



Şekil 4. EMKO EtherCAT Gateway Configurator yazılımında örnek bir ayar dosyası oluşturma süreci

Bu dosya, **EtherCAT Gateway cihazına uygulanacak ayarların** yüklenmesi için kullanılır. Dosyanın cihaza aktarımı, **Mini USB ile Parametre Konfigürasyonu** başlığı altında detaylı olarak açıklanmaktadır.

6. Cihaz Yapılandırma Ayarlarını Görüntüleme

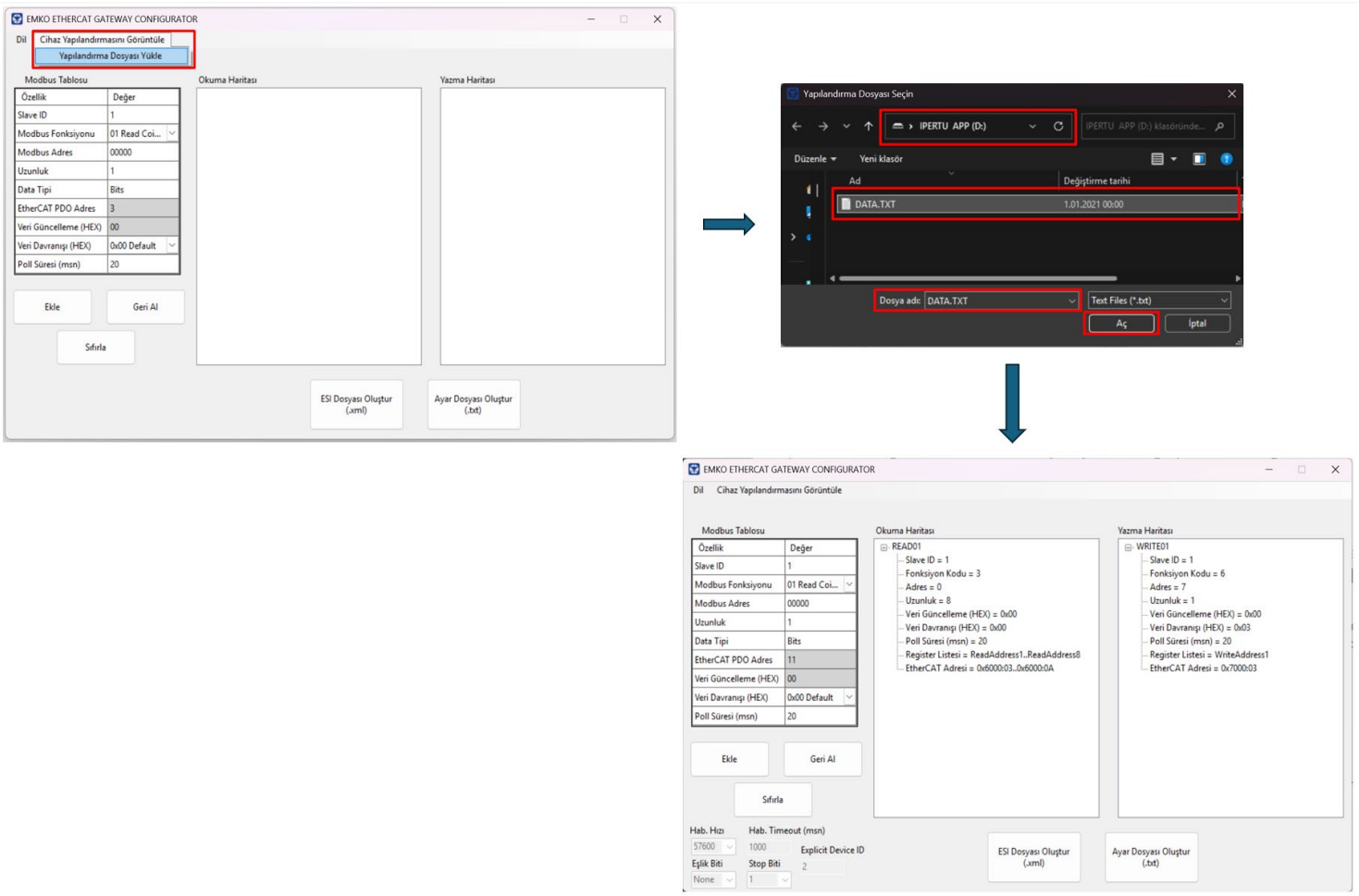
Cihazda yüklü olan mevcut Modbus mapping ve haberleşme ayarları, **EMKO ETHERCAT GATEWAY CONFIGURATOR** yazılımı üzerinden görüntülenebilmektedir. Bu işlem için cihazın bilgisayara **Mini USB bağlantısı** üzerinden bağlanması gereklidir.

Yazılımda **Cihaz Yapılandırmasını Görüntüle** menüsü altındaki **Yapılandırma Dosyası Yükle** seçeneği kullanılır. Açılan dosya seçim penceresinde, cihazın yığın bellek aygıtında bulunan **"DATA.TXT"** dosyası seçilir ve **Aç** butonuna tıklanır.

Bu işlem sonucunda cihazda hali hazırda kayıtlı bulunan:

- Modbus tablosu (Slave ID, fonksiyon kodu, adres, uzunluk, veri davranışı vb.),
- EtherCAT PDO ayarları,
- Haberleşme parametreleri (baudrate, parity, stop bit, timeout),
- Explicit Cihaz ID değeri

yazılım arayüzünde listelenir. Böylece, cihazda kullanılan geçerli konfigürasyonu görüntülenebilir ve gerekirse güncelleme yapmadan önce mevcut ayarlar kontrol edilebilir. İlgili işlem adımları Şekil 5'te gösterilmiştir.



Şekil 5. EMKO EtherCAT Gateway Configurator üzerinden cihazda yüklü yapılandırmanın görüntülenmesi

Mini USB ile Parametre Konfigürasyonu

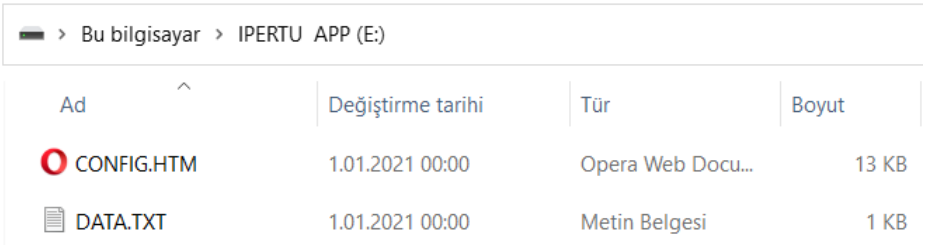
IPERTU EtherCAT Gateway modülünde, Mini USB bağlantısı üzerinden bilgisayar aracılığıyla yığın bellek özelliği kullanılarak parametre konfigürasyonu yapılabilmektedir. Bu özellik ile aşağıda belirtilen işlemler gerçekleştirilebilmektedir:

- Haberleşme hızı (baud rate), eşlik (parity) biti, durma (stop) biti ve timeout süresi gibi parametreler görüntülenmekte ve değiştirilebilmekte,
- Explicit cihaz ID değeri görüntülenmekte ve değiştirilebilmekte,
- EMKO EtherCAT Gateway Configurator arayüzüyle oluşturulan Modbus mapping yapıları Mini USB ile yüklenebilmekte veya hali hazırdaki yapı okunarak incelenebilmektedir.

Bu işlem sırasında IPERTU EtherCAT Gateway modülü harici bir güç kaynağına ihtiyaç duymamakta olup, USB bağlantısından alacağı enerji ile konfigürasyon işlemlerini gerçekleştirebilmektedir. İşlem adımları aşağıda detaylı bir şekilde açıklanmaktadır.

Adım 1: Modülü Bilgisayara Bağlayın

Modülü, Mini USB kablosu kullanarak bilgisayarınıza bağlayın. Bilgisayarınız, Şekil 6'da görüldüğü gibi modülü bir yığın bellek cihazı olarak tanıyacak ve otomatik olarak dosya gezginini açacaktır.



Şekil 6. Bilgisayar bağlantısı sağlanan IPERTU EtherCAT Gateway modülünün yığın bellek cihazı olarak algılanması

Adım 2: Konfigürasyon Arayüzünü Açın

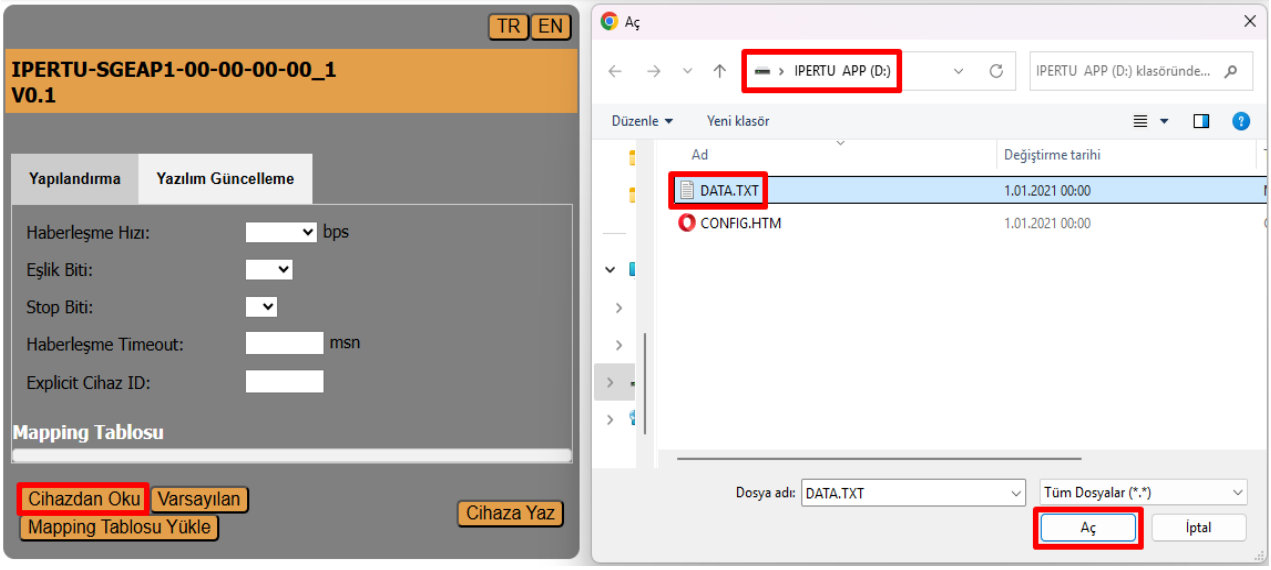
Yığın bellek cihazının içerisinde yer alan, Şekil 6'da de görülen CONFIG.HTM dosyasını açarak Şekil 7'de gösterilen konfigürasyon arayüzünü başlatın. Bu arayüz ile modülün çeşitli parametreleri görüntülenebilmekte ve değişiklik gerçekleştirilebilmektedir.



Şekil 7. IPERTU EtherCAT Gateway modülü parametre konfigürasyon arayüzü

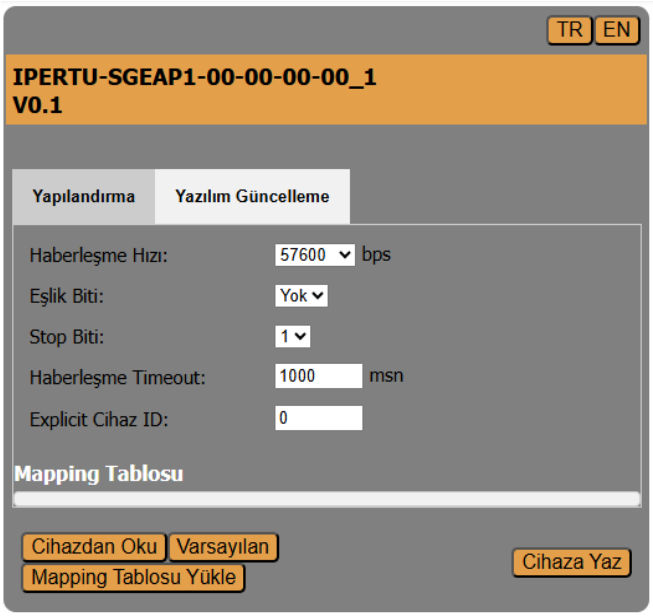
Adım 3: Parametreleri ve Mapping Tablosunu Okuyun

Konfigürasyon arayüzünde, modülün Şekil 7'de belirtilen parametrelerini ve modbus haritasını okumak için öncelikle "Cihazdan Oku" seçeneği kullanılır ve dosya seçim menüsüne erişilir. Burada, yığın bellek cihazı içerisinde yer alan "DATA.TXT" dosyası seçilip açılarak mevcut parametre değerleri arayüze aktarılır. Bu işlem Şekil 8'de gösterilmektedir.



Şekil 8. Modül mevcut parametre değerlerinin ve mapping tablosunun arayüze aktarılması

İşlemin gerçekleştirilmesinin ardından parametreler, Şekil 9'daki gibi arayüzde görüntülenmektedir. Modülde başlangıçta yüklü herhangi bir mapping tablosu yer almadığından ilgili kısım boş görünmektedir.

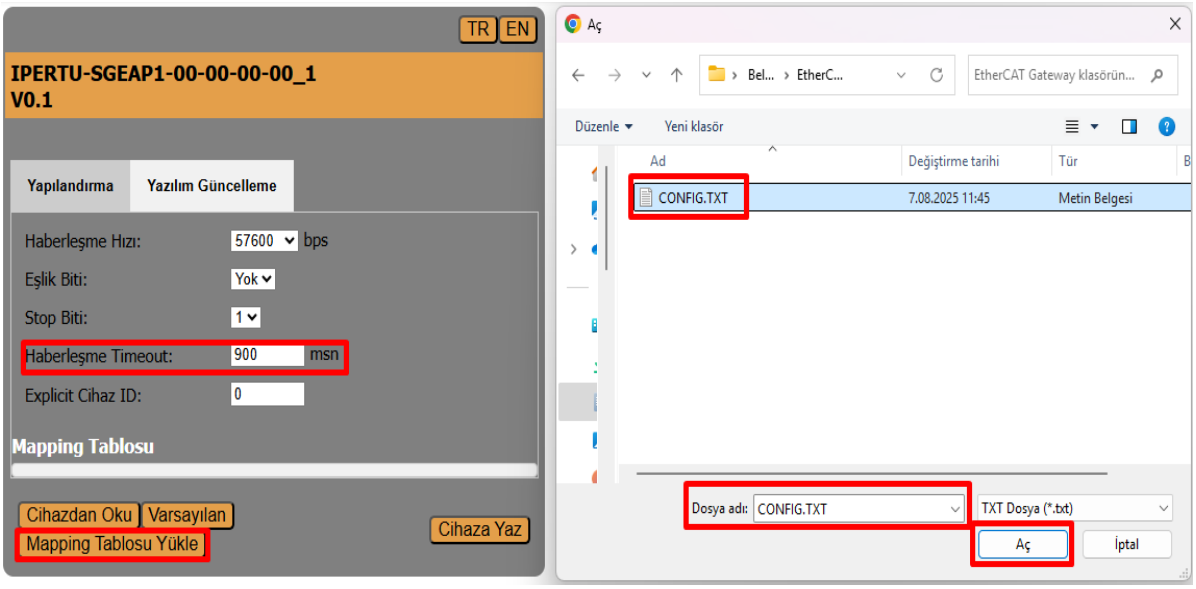


Şekil 9. Modül mevcut parametre değerlerinin arayüzde görüntülenmesi

Adım 4: Parametre Ayarı Yapın ve Mapping Tablosunu Yükleyin

Modül parametreleri, arayüz üzerinden değiştirilebilmekte ve cihaza yazılabilmektedir. Ayrıca EMKO ETHERCAT GATEWAY CONFIGURATOR yazılımında gerçekleştirilen Modbus mapping ayarları da bu arayüz üzerinden cihaza yazılmaktadır. Yazılan değerler modülün EEPROM'unda saklanmaktadır.

Parametre ayarı için değiştirilmek istenen parametre değerinin karşısında yer alan metin kutusuna yeni değer girilir. Mapping tablosunun yüklenmesi için ise arayüzde yer alan **Mapping Tablosu Yükle** butonuna tıklanır[1]. Dosya seçim penceresinde daha önce EMKO ETHERCAT GATEWAY CONFIGURATOR yazılımından üretilen **CONFIG.TXT** dosyası seçilir ve **AÇ** butonuna tıklanarak ayar dosyası arayüze yüklenir. İşlem adımları Şekil 10'da gösterilmiştir.



Şekil 10. Mapping tablosunun HTML arayüze yüklenmesi

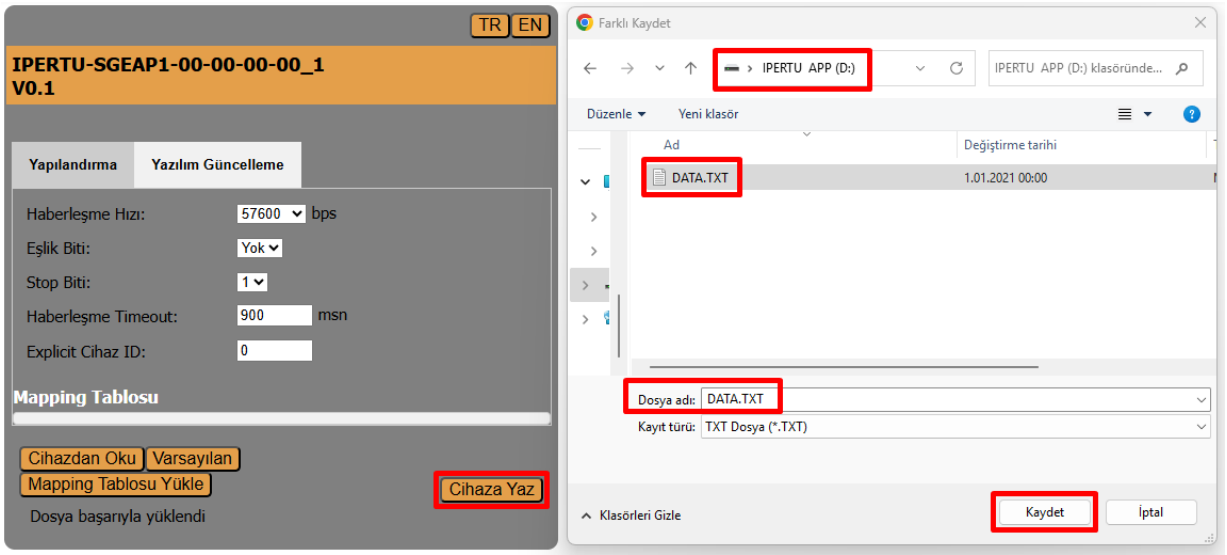
Doğru dosyanın seçilip yüklenmesi durumunda Şekil 11'de görüldüğü gibi arayüzde Dosya başarıyla yüklendi ibaresi görülür. Yanlış dosya seçim durumunda Pop-up ekranında hata gösterilerek işlem gerçekleştirilmez.



Şekil 11. Mapping tablosunun HTML arayüze yüklenmessi sonrası kullanıcının bilgilendirilmesi

Adım 5: Yapılandırmanın Modüle Yazılması

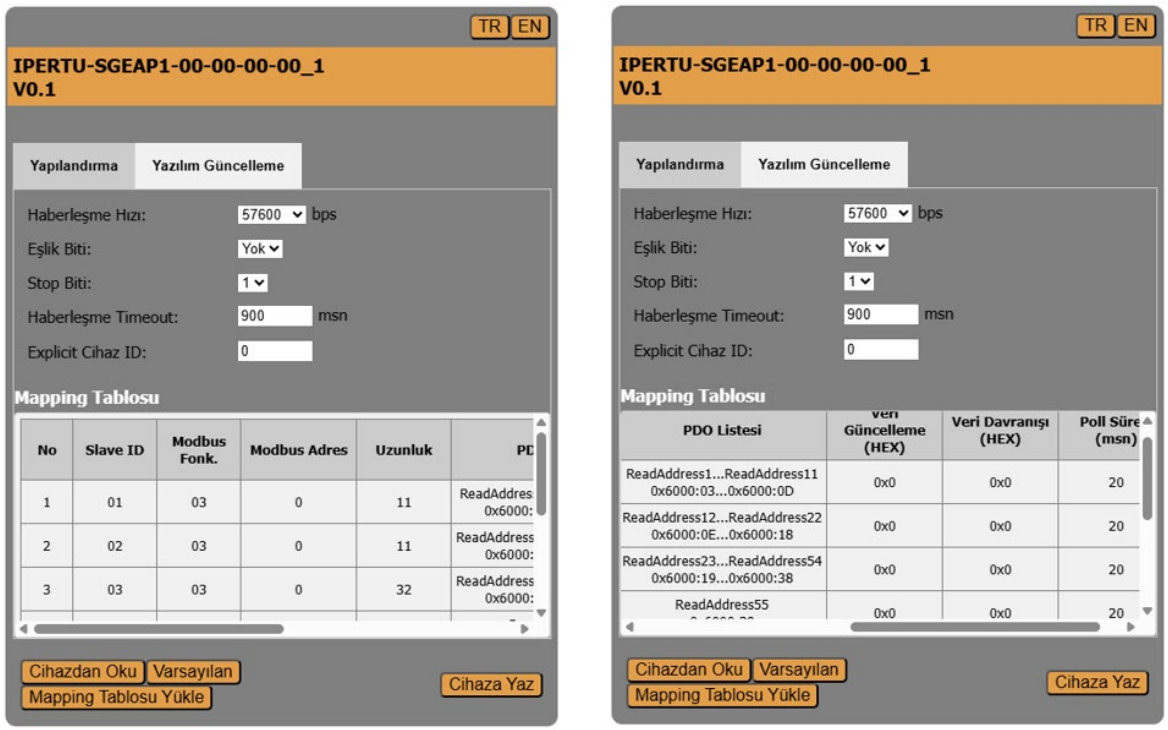
Yapılandırma ayarları ve mapping tablosu işlemleri gerçekleştirildikten sonra "Cihaza Yaz" seçeneği kullanılarak kaydetme menüsüne erişilir. Burada kaydedilecek dosyanın adının "DATA.TXT" olmasına ve dosya yolunun da yığın bellek cihazı olmasına dikkat edilmelidir. Bahsedilen süreçler Şekil 12'de gösterilmektedir.



Şekil 12. Yapılandırma ayarlarının ve mapping tablosunun modüle aktarılması

"Kaydet" seçeneğiyle parametrelerin modüle yazım işlemi başlatılmaktadır[2]. İşlem tamamlandığında ilgili ayarlar modül kalıcı hafızasına kaydedilecek, modül kendini otomatik olarak yeniden başlatacak ve Şekil 6'da gösterilen yığın bellek cihazı dosya gezgini tekrar açılacaktır. Adım 2 ve Adım 3'ü tekrarlayarak parametreleri okuyup değişiklikleri görebilirsiniz.

Şekil 13'te kayıt sonrası tekrar cihazdan oku ile okunan değerler görülmektedir. Burada görüldüğü üzere timeout değeri 900 msn olarak ayarlanmıştır. Mapping tablosu altında da ayarlanan Modbus paket yapıları görüntülenmektedir. Tablodaki tüm parametreler scrool yardımı ile görüntülenebilmektedir.



Şekil 13. Yapılandırma sonrası modül ayarlarının yeniden görüntülenmesi

Not 1: Sadece parametre değerleri ile ilgili değişiklik yapmak istenildiğinde Mapping Tablosu yükleme işlemi zorunlu değildir. İstenilen parametrede ayar yapılarak Adım 5 uygulanıp modüle yazma işlemi gerçekleştirilebilir.

Not 2: Uygun olmayan değer girilmesi halinde yazılan değerler cihaz tarafından reddedilir ve eski değerlerin kullanımına devam edilir.

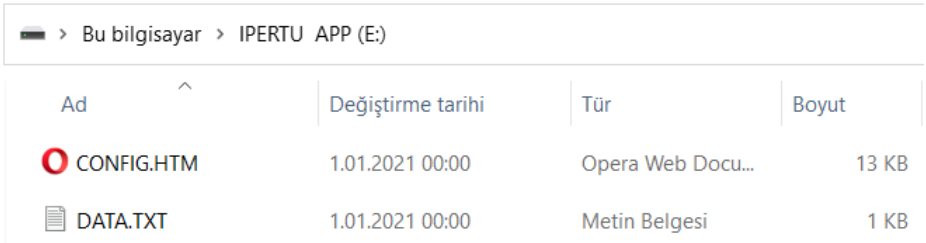
Haberleşme Timeout Sınır : 0 – 32767

Mini USB ile Cihaz Yazılımı Güncelleme (Bootloader)

IPERTU EtherCAT Gateway modülünde, Mini USB bağlantısı üzerinden bilgisayar aracılığıyla yığın bellek ve bootlader özelliği kullanılarak yazılım güncellemesi yapılabilmektedir. Bu işlem sırasında IPERTU EtherCAT Gateway modülü harici bir güç kaynağına ihtiyaç duymamakta olup, USB bağlantısından alacağı enerji ile yazılım güncelleme işlemi gerçekleştirebilmektedir. İşlem adımları aşağıda detaylı bir şekilde açıklanmaktadır.

Adım 1: Cihazı Bilgisayara Bağlayın

Modülü, Mini USB kablosu kullanarak bilgisayarınıza bağlayın. Bilgisayarınız, Şekil 14'de görüldüğü gibi modülü bir yığın bellek cihazı olarak tanıyacak ve otomatik olarak dosya gezginini açacaktır.



Şekil 14. Bilgisayar bağlantısı sağlanan IPERTU EtherCAT Gateway modülünün yığın bellek cihazı olarak algılanması

Adım 2: Konfigürasyon Arayüzünü Açın

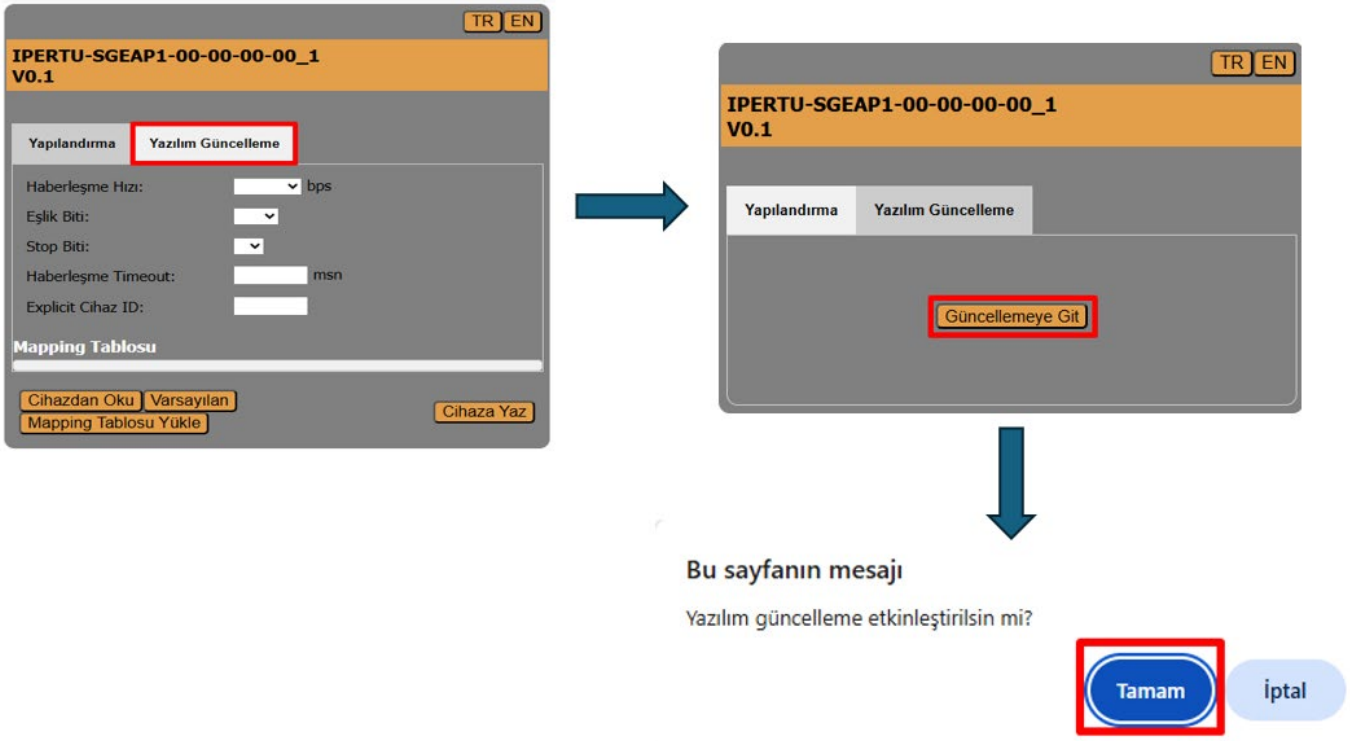
Yığın bellek cihazının içerisinde yer alan Şekil 14'te de görülen CONFIG.HTM dosyasını açarak Şekil 15'te gösterilen konfigürasyon arayüzünü başlatın. Bu arayüz üzerinden yazılım güncelleme etkinleştirme işlemi gerçekleştirilebilmektedir.



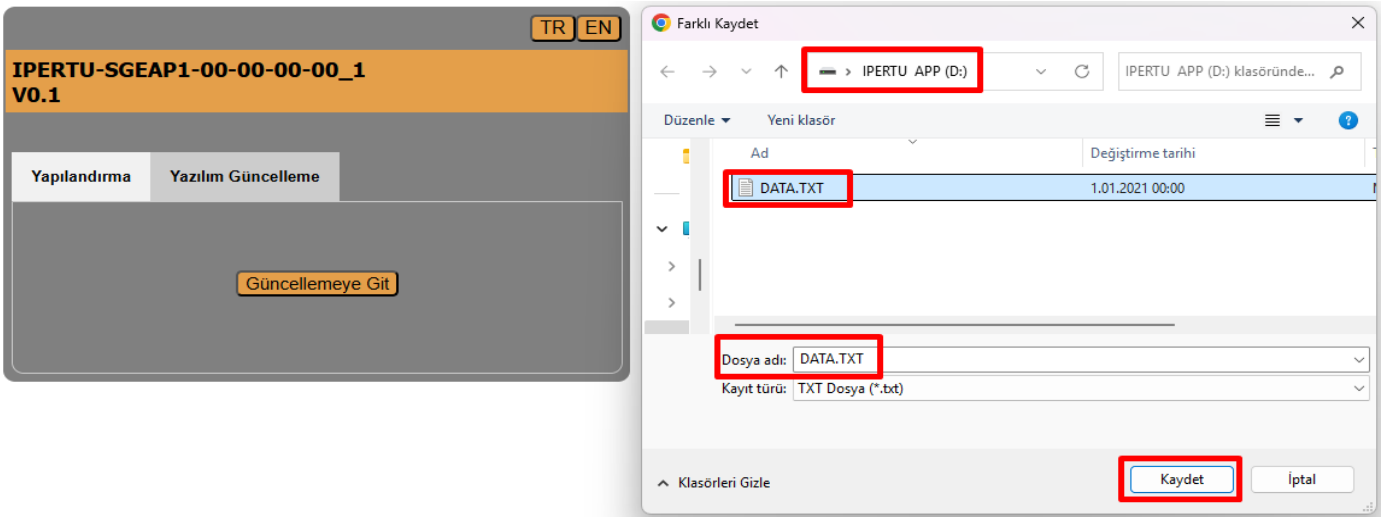
Şekil 15. IPERTU EtherCAT Gateway modülü konfigürasyon arayüzü

Adım 3: Yazılım Güncellemeyi Etkinleştirin

Modül yazılımının güncellenebilmesi için öncelikle arayüz üzerinden yazılım güncellemenin etkinleştirilmesi gerekmektedir. Bu işlem için "Yazılım Güncelleme" sekmesi kullanılır ve güncellemenin etkinleştirileceği sayfaya geçilir. Burada "Güncellemeye Git" seçeneği kullanılır. Ardından, açılan "Yazılım güncelleme etkinleştirilmesin mi?" pop-up'ında "Tamam" seçeneği kullanılarak kaydetme menüsüne erişilir. Burada, kaydedilecek dosyanın adının "DATA.TXT" olmasına ve dosya yolunun da yığın bellek cihazı olmasına dikkat edilmelidir. Bu işlemler sırasıyla Şekil 16 ve 17'de gösterilmektedir.

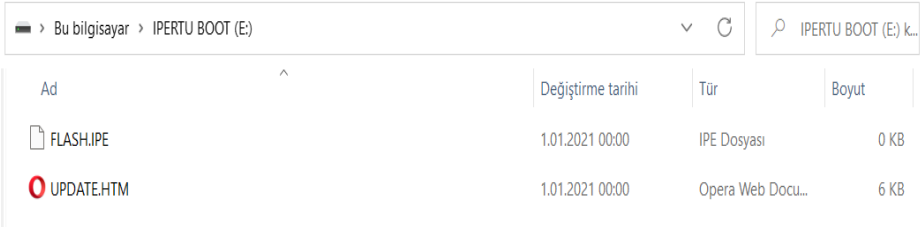


Şekil 16. Arayüz üzerinde yazılım güncellemenin etkinleştirilmesi



Şekil 17. Modülün yazılım güncelleme moduna alınması için yazma işleminin gerçekleştirilmesi

"Kaydet" seçeneğiyle modül yazılım güncelleme moduna alınmaktadır. Bu işlemin ardından modül kendini otomatik olarak yazılım güncelleme modunda yeniden başlatacak ve Şekil 18'de gösterilen yığın bellek cihazı dosya gezgini tekrar açılacaktır.

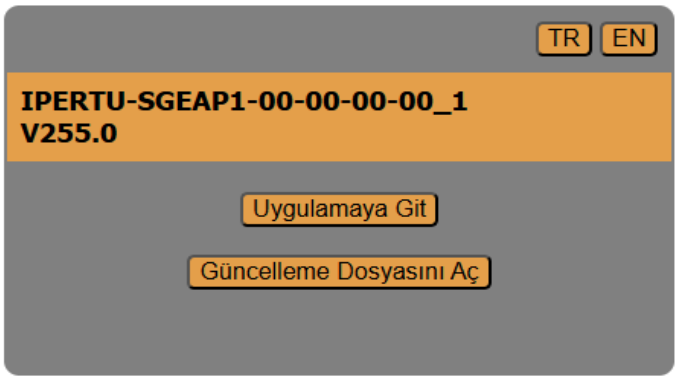


Şekil 18. Yazılım güncelleme moduna alınan IPERTU EtherCAT Gateway modülünün yığın bellek cihazı olarak algılanması

Şekil 18'de de görüldüğü üzere modülün bağlı bulunduğu sürücünün adı "IPERTU BOOT" olarak değişmiş ve modül yazılım güncelleme moduna alınmıştır. Aynı zamanda yazılım güncelleme modunda MS LED'i kırmızı renkte 1s yanmakta 0.1s sönmektedir.

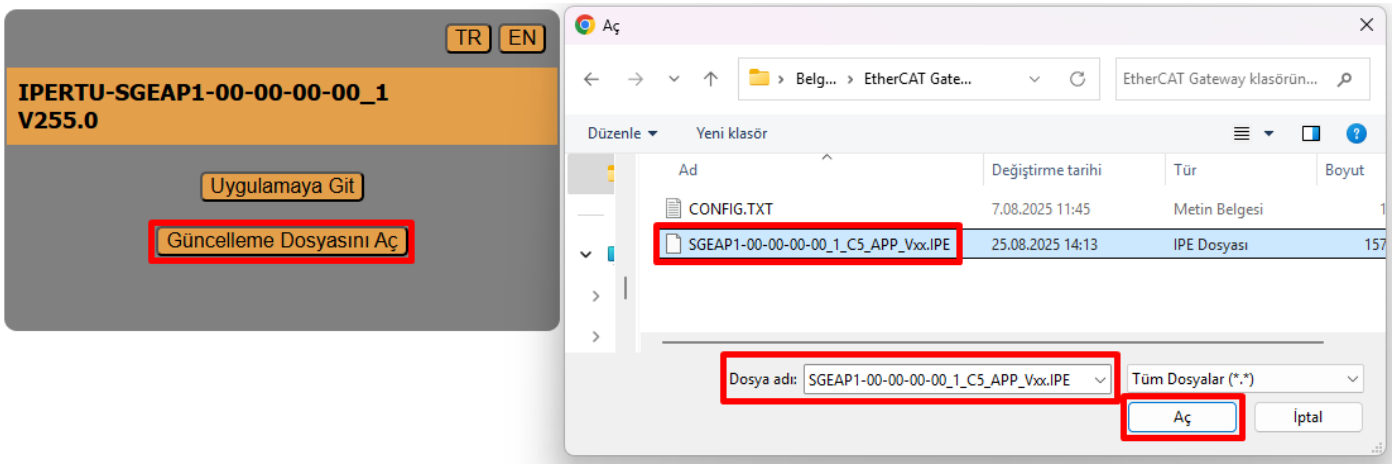
Adım 4: Güncelleme Dosyasını Arayüze Aktarın

Yığın bellek cihazının içerisinde yer alan, Şekil 18'de de görülen UPTADE.HTM dosyasını açarak Şekil 19'da gösterilen yazılım güncelleme arayüzünü başlatın. Bu arayüz üzerinden yazılım güncelleme işlemi gerçekleştirilebilmektedir.



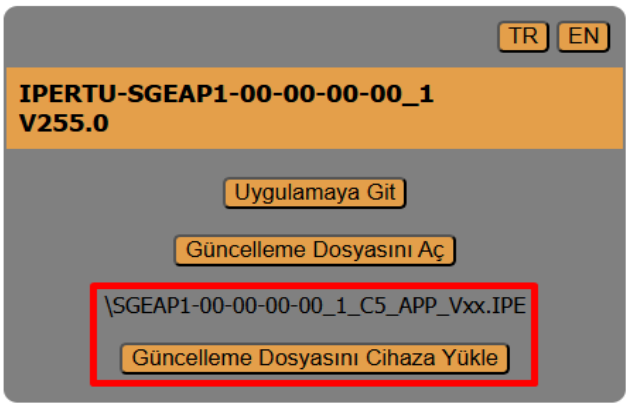
Şekil 19. IPERTU EtherCAT Gateway modülü yazılım güncelleme arayüzü

Burada, modülün yazılımını güncellemek için öncelikle "Güncelleme Dosyasını Aç" seçeneği kullanılır ve dosya seçim menüsüne erişilir. Burada, bilgisayarınızda yer alan, bizim size göndermiş olduğumuz "SGEAP1-00-00-00-00_1_C5_APP_Vxx.IPE" dosyası seçilip açılarak güncelleme dosyası arayüze aktarılır. Bu işlem, Şekil 20'de gösterilmektedir.



Şekil 20. Güncelleme dosyasının yazılım güncelleme arayüzüne aktarılması

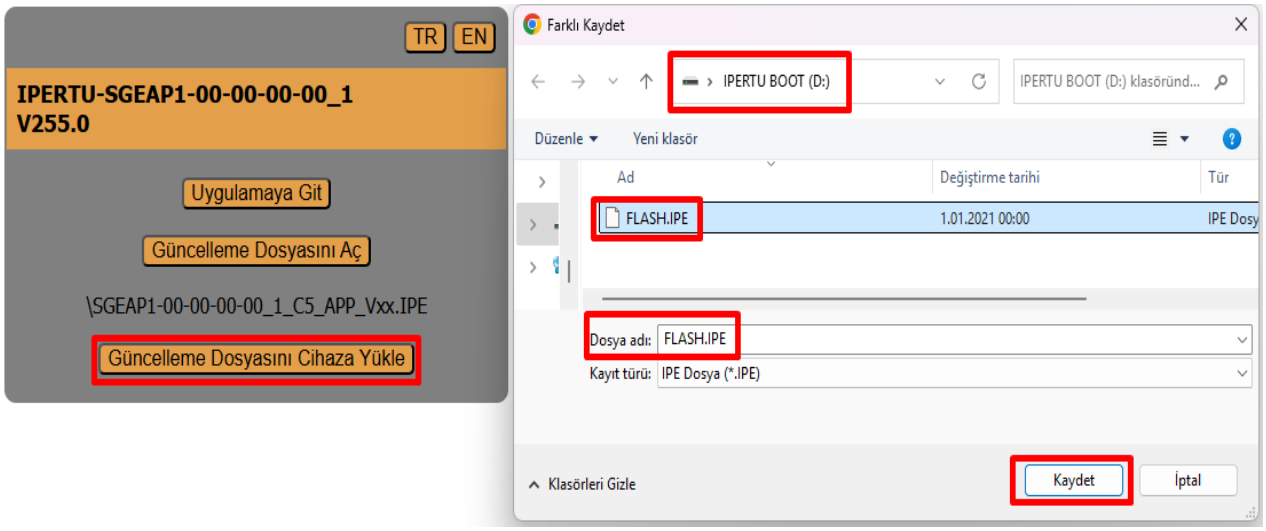
İşlemin gerçekleştirilmesinin ardından güncelleme dosyası adı ve "Güncelleme Dosyasını Cihaza Yükle Butonu", Şekil 21'deki gibi arayüzde görüntülenmektedir.



Şekil 21. Güncelleme dosyasının arayüzde görüntülenmesi

Adım 5: Güncelleme Dosyasını Cihaza Yükleyin

Güncelleme işlemi için öncelikle "Güncelleme Dosyasını Cihaza Yükle" butonu kullanılarak kaydetme menüsüne erişilir. Burada, kaydedilecek dosyanın adının "FLASH.IPE" olmasına ve dosya yolunun da yığın bellek cihazı olmasına dikkat edilmelidir. Bahsedilen süreçler Şekil 22'de gösterilmektedir.



Şekil 22. Güncelleme dosyasının modüle yazılarak yazılım güncelleme işleminin tamamlanması

Ayarlamalar gerçekleştirildikten sonra “Kaydet” seçeneğiyle güncelleme dosyasının modüle yazım işlemi başlatılmaktadır. İşlem tamamlandığında modül kendini otomatik olarak uygulama modunda yeniden başlatacak ve Şekil 6’da gösterilen yığın bellek cihazı dosya gezgini tekrar açılacaktır. Modül üzerinde yer alan LED’ler de normal cihaz çalışma modundaki durumlarına dönecektir.

Explicit Cihaz ID'nin Ayarlanması

EtherCAT ağlarında cihazlar genellikle topolojideki fiziksel sıralamalarına göre adreslenir. Ancak bazı uygulamalarda, cihazların ağ üzerindeki konumu değişse bile aynı kimlikle tanınması gerekebilir. Bu durumda Explicit Cihaz ID devreye girer. Explicit Cihaz ID, bir EtherCAT slave cihazına manuel olarak atanmış sabit bir kimlik numarasıdır. Bu kimlik, cihazın EtherCAT master tarafından fiziksel konumundan bağımsız olarak tanınmasına ve adreslenmesine olanak sağlar.

Bu özellik özellikle Hot Connect destekli sistemlerde önemlidir. Hot Connect, EtherCAT ağı çalışır durumdayken belirli cihazların çıkarılıp takılmasına izin verir. Master, Explicit Cihaz ID sayesinde hangi cihazın bağlandığını veya çıkarıldığını anlar ve uygun yapılandırmayı yükleyebilir. Böylece esnek, yeniden yapılandırılabilir sistemler oluşturulabilir ve bakım sırasında minimum kesinti sağlanır.

Explicit Cihaz ID iki farklı prosedür kullanılarak ayarlanabilir. Bu prosedürler Çizelge 1'de açıklanmaktadır.

Çizelge 1. Explicit Cihaz ID'nin ayarlanmasında kullanılan prosedürler

Prosedür	Açıklama	Gerekli bileşenler
Dipswitch	Dipswitch kullanarak ID atama	-
Yığın Bellek	HTML arayüzü üzerinden ID Atama	Mini USB kablosu ve Bilgisayar

Kullanılacak ayar prosedürü, cihaza enerji verildiği sırada Dipswitch değeri kontrol edilerek seçilmektedir. Bu durum Çizelge 2'de açıklanmaktadır.

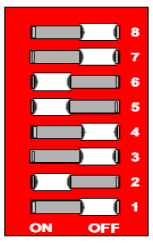
Çizelge 2. Explicit Cihaz ID ayar modları

Dipswitch Değeri	Açıklama
0 - 254	Explicit Cihaz ID, 1-254 arasında sabit bir değere ayarlanır. Bu durumda yığın bellek üzerinden ayarlanmış olan değer yok sayılır.
255	Explicit Cihaz ID, yığın bellekte yer alan HTML arayüzü üzerinden ayarlanmış değere ayarlanır.

1. Dipswitch ile Explicit Cihaz ID ayarlama

Dipswitch kullanılarak, Explicit Cihaz ID, 0 ile 254 arasında ikili kodlanmış değerlerle ayarlanmaktadır. Ayar, cihaza enerji verilmeden önce gerçekleştirilmelidir.

Örnek:



0 0 1 1 0 0 1 0

Bit	=	1	2	3	4	5	6	7	8
Durum	=	OFF	ON	OFF	OFF	ON	ON	OFF	OFF
Çarpan	=	1	2	4	8	16	32	64	128
Değer	=	0 +	2 +	0 +	0 +	16 +	32 +	0 +	0 = **50**

2. Yığın Bellek Özelliği Kullanılarak Explicit Cihaz ID'nin Ayarlanması

IPERTU EtherCAT Gateway modülünde, Mini USB bağlantısı üzerinden bilgisayar aracılığıyla yığın bellek özelliği kullanılarak Explicit Cihaz ID konfigürasyonu yapılabilmektedir. Bu işlem sırasında modül, harici bir güç kaynağına ihtiyaç duymamakta olup, USB bağlantısından alacağı enerji ile konfigürasyon işlemlerini gerçekleştirebilmektedir.

İşlemi gerçekleştirebilmek için öncelikle "Mini USB ile Parametre Konfigürasyonu" başlığında anlatılan Adım 1, Adım 2 ve Adım 3'ü uygularak parametreleri okuyunuz. Daha sonra, Explicit Cihaz ID parametresinin karşısında yer alan metin kutusuna yeni değer girilir[3] ve Adım 5'te anlatıldığı gibi cihaza yazma işlemi gerçekleştirilir.

Not 3: İzin verilen Explicit Cihaz ID değer aralığı : 0 – 32767

Cihaz Hata Durumları

EtherCAT Gateway modülünde meydana gelebilecek hata durumları, cihazın dahili tanılama sistemi tarafından izlenmekte ve PDO'lar üzerinden erişilebilmektedir. Bu kapsamda, oluşan hata kodları 0x1A00 TxPDO'da yer alan 0x6000:01 adresindeki SYSIN1 ve alan 0x6000:02 adresindeki SYSIN2 isimli değişkenler üzerinden okunabilmektedir. SYSIN1 donanımsal hatalarla ilgili olup, SYSIN2 ise modbus haberleşme hatalarını temsil etmektedir. Hataların türleri, olası nedenleri, cihazın bu durumlarda verdiği tepkiler ve çözüm yöntemleri Çizelge 3'te ayrıntılı olarak sunulmuştur.

Çizelge 3. EtherCAT Gateway Modülü Hata Durumları, Nedenleri, Cihaz Davranışı ve Çözüm Yöntemi

Hata Açıklaması	Muhtemel Oluşma Nedeni	Cihaz Davranışı	Çözüm Yöntemi
Düşük besleme gerilimi	- Besleme geriliminin 17V'un altına düşmesi durumunda bu hata oluşur.	- Besleme LED'inde 0.1s'lik yanıp sönme gözlemlenir. - Cihaz RUN durumuna geçmez. Bu nedenle işlevlerini yerine getirmez. - 0x6000:01 PDO adresindeki SYSIN1 değişkeninin 8. Biti SET'lenir.	- Besleme kaynağını ve cihaz besleme bağlantılarını kontrol ediniz. Cihazı 24V ile beslediğinize emin olunuz.
Eşlik (Parity) biti hatası	- Alınan verinin eşlik (parity) biti beklenen değerle uyuşmadığı durumda bu hata oluşur. - Alıcı ve verici eşlik biti ayarları farklı olabilir. - Hat üzerinde gürültü/elektromanyetik parazitten kaynaklı bit bozulması olabilir. - Alıcı ve verici farklı haberleşme hızlarında çalışıyorsa ya da hızda kayma olduğu durumda eşlik biti yanlış okunabilir. - Haberleşme hattında (TX/RX) sorun olabilir	- RS485 LED'inde 100 ms'lik kırmızı renkte bir yanıp sönme gözlemlenir. - Verinin geldiği Slave modülle haberleşme gerçekleştirilemez. - 0x6000:01 PDO adresindeki SYSIN1 değişkeninin 9. ve 10. Biti SET'lenir.	- Alıcı ve verici haberleşme ayarlarını kontrol ediniz. - Haberleşme hattını ekranlı kablo ve sonlandırma direnci kullanarak iyileştirin. - 0x7000:02 PDO adresine 3083 değerini yazıp modülü yeniden başlatarak hata bitini temizleyiniz. *Not: 0x7000:02 PDO adresindeki değeri tekrar 0 yapınız.*
Gürültü hatası	- Haberleşme hattı üzerinde gürültü/parazit algıladığında veya sinyal seviyesi belirsizleştiğinde bu hata oluşur.	- RS485 LED'inde 100 ms'lik kırmızı renkte bir yanıp sönme gözlemlenir. - Verinin geldiği Slave modülle haberleşme gerçekleştirilemez. - 0x6000:01 PDO adresindeki SYSIN1 değişkeninin 9. ve 11. Biti SET'lenir.	- Haberleşme hattını ekranlı kablo ve sonlandırma direnci kullanarak iyileştirin - Topraklama (GND) bağlantılarını kontrol edin. - Haberleşme Kablo uzunluğunuzu kontrol edin. - Haberleşme kablosuna ferit çekirdek yerleştirin. - 0x7000:02 PDO adresine 3083 değerini yazıp modülü yeninden başlatarak hata bitini temizleyiniz. *Not: 0x7000:02 PDO adresindeki değeri tekrar 0 yapınız.*
Çerçeveleme hatası	- Alınan veride beklenen yerde stop bit bulunamazsa veya durma (stop) bit seviyesi yanlışsa bu hata oluşur. - Alıcı ve verici durma biti ayarları farklı olabilir. - Alıcı ve verici farklı haberleşme hızlarında çalışıyorsa ya da hızda kayma olduğu durumda stop biti yanlış okunabilir. - Data bit sayısı uyuşmazlığı durumunda oluşabilir - Sinyal bozulması, gürültü, zayıf bağlantı veya kablo sorunlarında oluşabilir	- RS485 LED'inde 100 ms'lik kırmızı renkte bir yanıp sönme gözlemlenir. - Verinin geldiği Slave modülle haberleşme gerçekleştirilemez. - 0x6000:01 PDO adresindeki SYSIN1 değişkeninin 9. ve 12. Biti SET'lenir.	- Alıcı ve verici haberleşme ayarlarını kontrol ediniz. Haberleşme hattını ekranlı kablo ve sonlandırma direnci kullanarak iyileştirin. - 0x7000:02 PDO adresine 3083 değerini yazıp modülü yeninden başlatarak hata bitini temizleyiniz. *Not: 0x7000:02 PDO adresindeki değeri tekrar 0 yapınız.*

Hata Açıklaması	Muhtemel Oluşma Nedeni	Cihaz Davranışı	Çözüm Yöntemi
Taşma hatası	- Yeni veri geldiğinde önceki veri henüz okunmamışsa bu hata oluşur. - RX buffer dolduğunda bu hata oluşur. - Modülde donanımsal bir problem oluşmuş olabilir.	- RS485 LED'inde 100 ms'lik kırmızı renkte bir yanıp sönme gözlemlenir. - Verinin geldiği Slave modülle haberleşme gerçekleştirilemez. - 0x6000:01 PDO adresindeki SYSIN1 değişkeninin 9. ve 13. Biti SET'lenir.	- Alıcı ve verici haberleşme ayarlarını kontrol ediniz. - 0x7000:02 PDO adresine 3083 değerini yazıp modülü yeniden başlatarak hata bitini temizleyiniz. *Not: 0x7000:02 PDO adresindeki değeri tekrar 0 yapınız.* - Sorunun devam etmesi durumunda teknik servisle iletişime geçin.
Modbus Haberleşme Hatası	- Slave cihaz, desteklenmeyen bir fonksiyon kodu ile sorgulandığında bu hata oluşur. - Belirtilen modbus adres aralığı slave cihazda geçersiz olduğu durumda bu hata oluşur. - Haberleşme timeout durumuna düşerse bu hata oluşur. - Gönderilen veri yapısı veya boyutu slave cihazın beklentisine uygun olmadığı durumda bu hata oluşur. - Belirtilen modbus adres aralığı slave cihazda geçersiz olduğu durumda bu hata oluşur.	- RS485 LED'inde 500 ms'lik kırmızı renkte bir yanıp sönme gözlemlenir. - İlgili Slave modülle haberleşme gerçekleştirilemez. - 0x6000:01 PDO adresindeki SYSIN değişkeninin 9. Biti SET'lenir. - 0x6000:02 PDO adresindeki SYSIN2 değişkeninin 0-7. Bitleri hata kodunu , 8-15. Bitleri hatanın gerçekleştiği mapping sıra numarasını gösterir.	- Mapping ayarlarını kontrol ediniz ve slave cihaza göre düzeltmelerini gerçekleştirerek ayarları modüle yeniden yükleyiniz

Uygulama Notu 1 : TwinCAT Üzerinden EtherCAT Gateway ile IPERTU Genişleme Modüllerinin Kontrolü

Bu uygulama notunda, EtherCAT RS485 dönüşümü ile IPERTU genişleme modüllerinin dijital/analog giriş-çıkış ve parametre kontrollerinin gerçekleştirilmesi amaçlanmaktadır.

Örnek bir senaryo kapsamında, birbirine fiziksel olarak bağlı üç IPERTU genişleme modülünün en başındaki modül ile EtherCAT Gateway modülü arasında RS485 bağlantısı sağlanmıştır. Kullanılan modüllerin model kodları ve bağlantı yapısı aşağıda açıklanmıştır:

- **EtherCAT Gateway Modülü**:

  SGEAP1-00-00-00-00_1 – EtherCAT master ile haberleşmeyi sağlayan ana modüldür.
  - Modülün ön yüzünde yer alan "IN" etiketli Ethernet portu, master cihaz (örneğin bir PC) ile haberleşmenin kurulduğu bağlantı noktasıdır.
  - Genişleme modül grubunun en başındaki modüle ile RS485 bağlantısı sağlanmıştır.
- **1. Genişleme Modülü**:

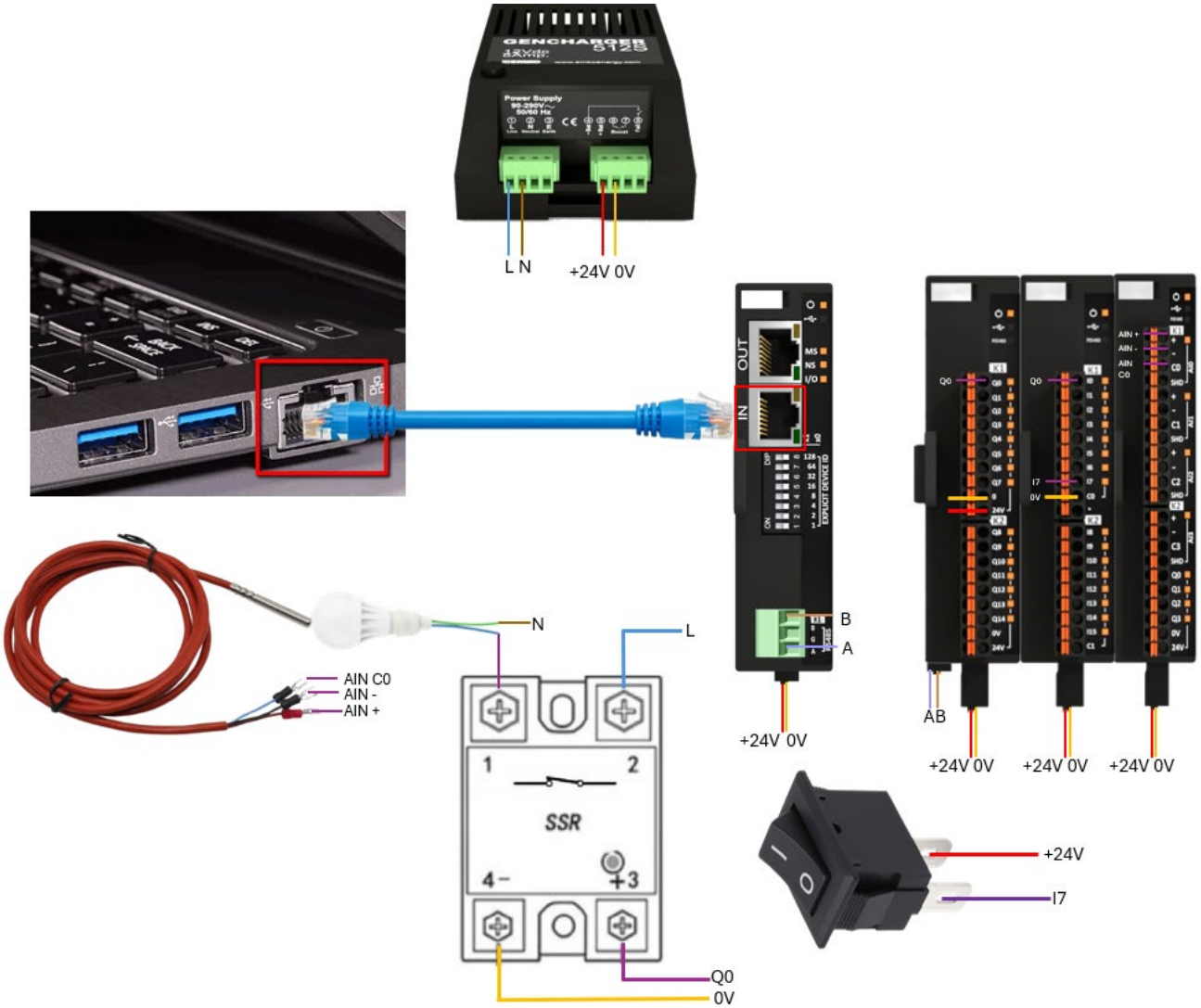
  SMR1P1-00-00-01-00_1 (15xTO) – 15 adet dijital transistör çıkışına sahiptir.
  - İlk sırada yer aldığından master görevi üstlenir ve RS485'ten gelen verilerin iç bus haberleşmeye dönüştürülüp diğer genişleme modüllerine aktarılmasından sorumludur.
  - Slave ID değeri 1'dir. Master olduğundan yanına bağlanan modüllere artan sırayla otomatik olarak ID değeri atar.
  - Transistör çıkışları için gerekli olan modül üzerinde yer alan 0V – 24V bağlantısı sağlanmıştır.
  - **Q0 çıkışı**, bir **AC lambayı** kontrol eder. Lamba, bir solid state röle (SSR) üzerinden sürülmektedir. Q0 çıkışı, SSR'nin pozitif tetikleme girişine bağlanmıştır. SSR'nin negatif tetikleme ucu ise 0 V'a bağlanmıştır.
  - Q0 çıkışı aktif edildiğinde, SSR tetiklenir ve AC lamba çalışır.
- **2. Genişleme Modülü**:

  SMR1P1-01-00-00-00_1 (16xDI) – 16 dijital giriş içerir.
  - **I0 girişi**, Q0 çıkışıyla bağlantılıdır.
  - **I7 girişi**ne bir manuel anahtar bağlanmıştır (bir ucu 24 V'a, diğer ucu I7'ye). Anahtar kapatıldığında giriş aktif olur.
  - Modülün C0 (Common) hattı, sistemin 0 V referansına bağlanmıştır.
- **3. Genişleme Modülü**:

  SMR1P1-00-01-04-00_1 (4xAI-SU, 4xTO) – 4x analog giriş, 4xdijital çıkış içerir
  Bu modüle bir PT-100 sıcaklık sensörü bağlanmıştır. Sensörün:
  - + ucu, AI0+ terminaline,
  - Diğer iki ucu ise AI0− ve C0 (common) girişlerine bağlanmıştır.

Not: Tüm genişleme modülleri 57600 baud hızında çalışmakta olup, eşlik biti kullanılmaz ve durma biti değeri 1'dir.

Sistemin fiziksel bağlantı ve donanım yerleşimi Şekil 23'te gösterilmektedir.



Şekil 23. Örnek kuruluma ait donanım bağlantı diyagramı: Güç kaynağı, EtherCAT Gateway modülü, genişleme modülleri, SSR, AC lamba, PT-100 sensörü, Anahtar ve Ethernet bağlantısı

Sistemin çalışabilmesi için ilk aşamada aşağıdaki iki temel yapılandırmanın gerçekleştirilmesi gereklidir:

- **Modbus Mapping Yapılandırması**: Bu aşamada, her bir genişleme modülüne ait slave ID, haberleşme fonksiyon kodları, başlangıç adresleri ve veri uzunlukları gibi bilgiler tanımlanır. Bu bilgiler, EtherCAT PDO yapısı ile eşleştirilerek doğru veri akışı sağlanır.
- **RS485 Haberleşme Ayarları**: Gateway modülünün genişleme modülleri ile doğru şekilde iletişim kurabilmesi için RS485 parametrelerinin (baudrate, parity, stop bit vb.) yapılandırılması gerekmektedir.

Bu yapılandırmalar, **EMKO ETHERCAT GATEWAY CONFIGURATOR** yazılımı veya cihazda yer alan **USB Yığın Bellek (Mass Storage)** arayüzündeki HTML tabanlı konfigürasyon sayfası aracılığıyla gerçekleştirilir.

Modbus Haberleşme Senaryosu

Bu uygulama senaryosu kapsamında, genişleme modüllerine ait datasheet'ler incelenerek her bir modül için uygun Modbus adresleri ve erişim yöntemleri belirlenmiştir.

Okuma işlemleri için:

- 1. genişleme modülünden, adres 0'dan başlayarak uzunluğu 11 olan veri alanı, Modbus Fonksiyon Kodu 0x03 ile okunacaktır.
- 2. genişleme modülünden, adres 0'dan başlayarak uzunluğu 11 olan veri alanı, yine Fonksiyon Kodu 0x03 ile okunacaktır.
- 3. genişleme modülünden, adres 0'dan başlayarak uzunluğu 32 olan veri alanı ve ayrıca adres 128'den başlayarak uzunluğu 1 olan alan, Fonksiyon Kodu 0x03 ile okunacaktır.

Yazma işlemleri için:

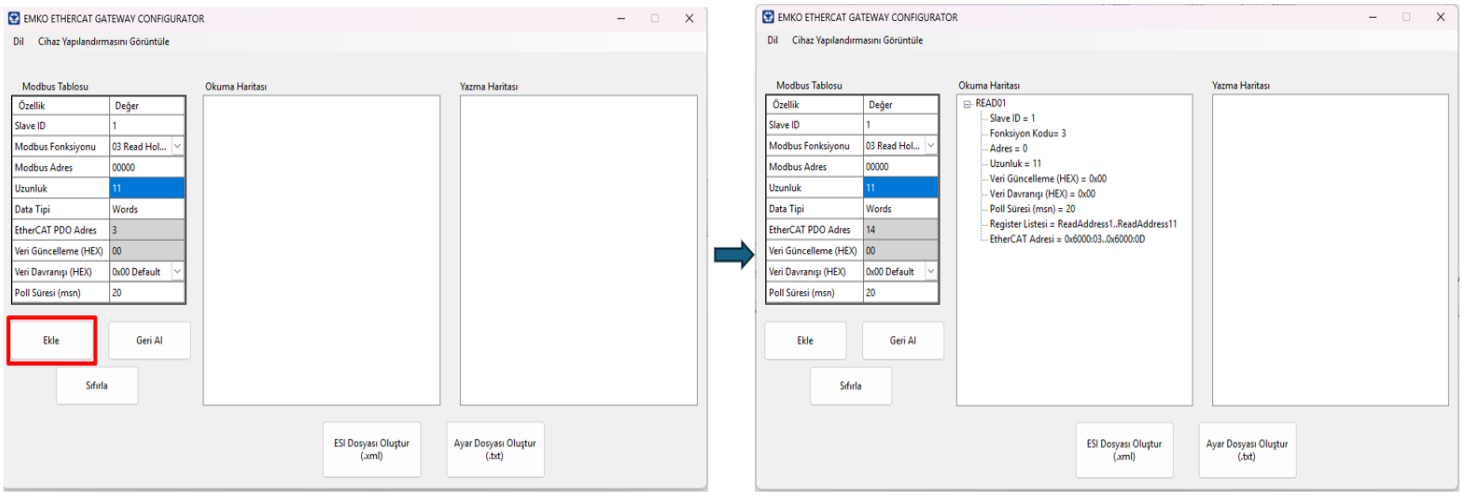
- 1. Genişleme modülünde adres 7'den başlayarak uzunluğu 1 olan veri alanına ve ayrıca adres 10'dan başlayarak uzunluğu 1 olan veri alanına, Modbus Fonksiyon Kodu 0x06 ile yazılacaktır.
- 3. Genişleme modülünde adres 11'den başlayarak uzunluğu 1 olan veri alanına ve adres 128'den başlayarak uzunluğu 1 olan veri alanına Modbus Fonksiyon Kodu 0x06 ile yazılacaktır.

Yazma işlemlerinde Veri Davranışı için **"State Change" (Durum Değişimi)** özelliği kullanılacaktır. Bu durumda, sadece veri değeri önceki değerden farklıysa ilgili yazma komutu Modbus haberleşme trafiğine dahil edilir.

Konfigürasyon Yazılımı ile Yapılandırma Ayarlarının Gerçekleştirilmesi

Planlanan haberleşmedeki okuma ve yazma paketlerinin yapılandırılması, bu paketlerin EtherCAT PDO yapısıyla eşleştirilmesi (RxPDO/TxPDO) ve buna bağlı ESI (EtherCAT Slave Information) ile yapılandırma dosyalarının oluşturulması için **EMKO EtherCAT Gateway Configurator** yazılımı kullanılır.

Şekil 24'te 1. Slave modül (Slave ID : 1) için örnek bir mapping işlemi gösterilmektedir.

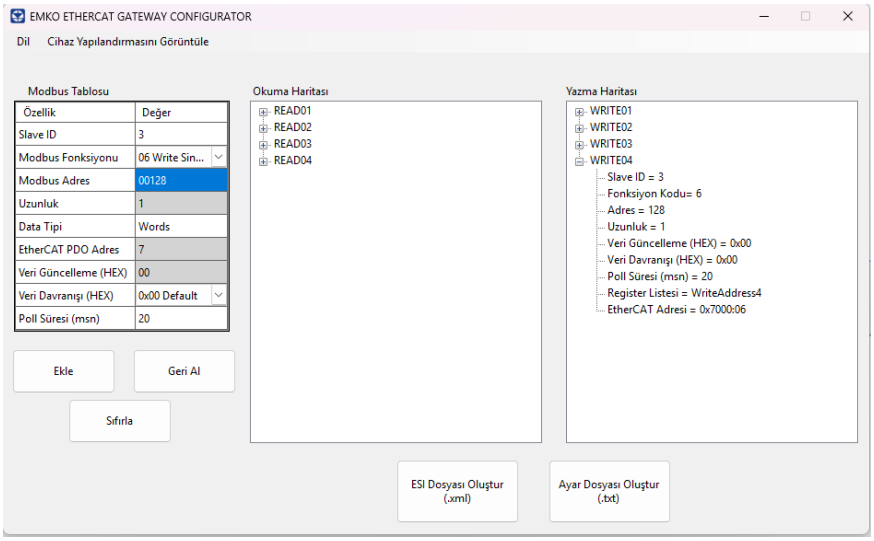


Şekil 24. 1. Genişleme modülü için Modbus okuma paketinin arayüzde eklenmesi

Şekilde de görüldüğü üzere, "Ekle" butonuna basılarak tanımlanan Modbus okuma paketi, "Okuma Haritası" bölümüne Read01 adıyla eklenmiştir. Bu pakete ait tüm haberleşme parametreleri – Slave ID, fonksiyon kodu, Modbus adresi, uzunluk, veri güncelleme süresi, veri davranışı ve polling süresi – ayrıntılı olarak listelenmiştir.

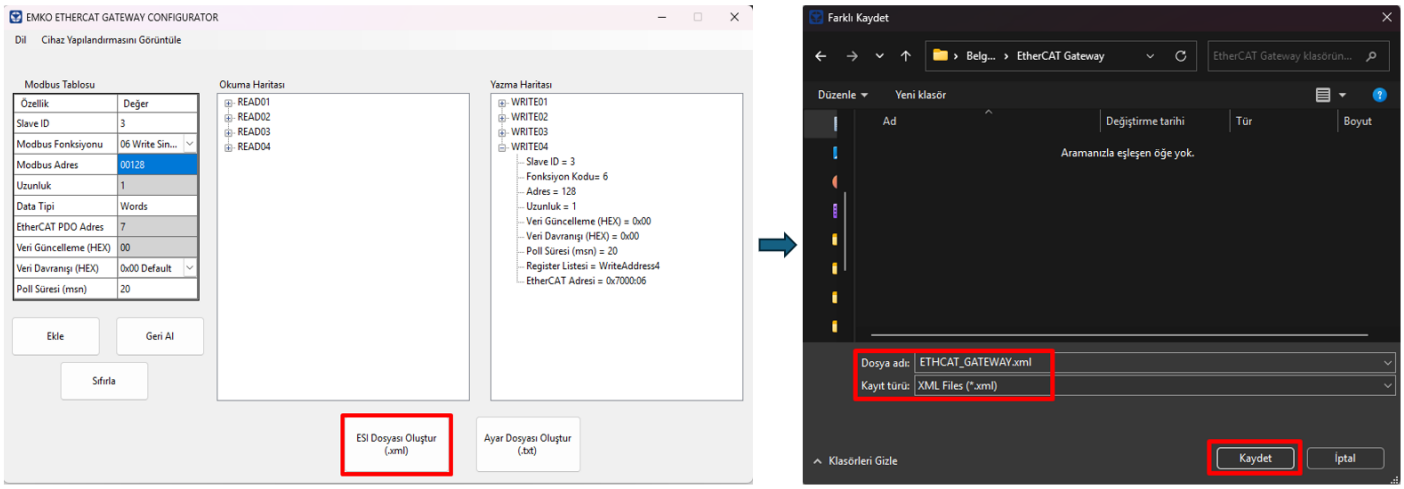
Paket altında yer alan "Register Listesi", bu okuma işlemi sonucunda elde edilen Modbus verilerinin EtherCAT tarafında hangi register adreslerine yerleştirileceğini belirtir. Bu örnekte, ReadAddress1 ile başlayan ve 11 register uzunluğundaki bir blok, EtherCAT PDO adres alanına eşleştirilmiştir. Bu alan, gateway cihaz tarafından otomatik olarak oluşturulan ve sıralı şekilde artan ReadAddressX etiketleri ile temsil edilir.

Devamında planlanan tüm paket yapıları istenilen sıralamada eklenerek haritalara eklenmiş ve Şekil 25'te sunulmuştur. Burada da görüleceği üzere yazma paketlerinde Veri Davranışı 0x03 State Change olarak ayarlanmıştır.



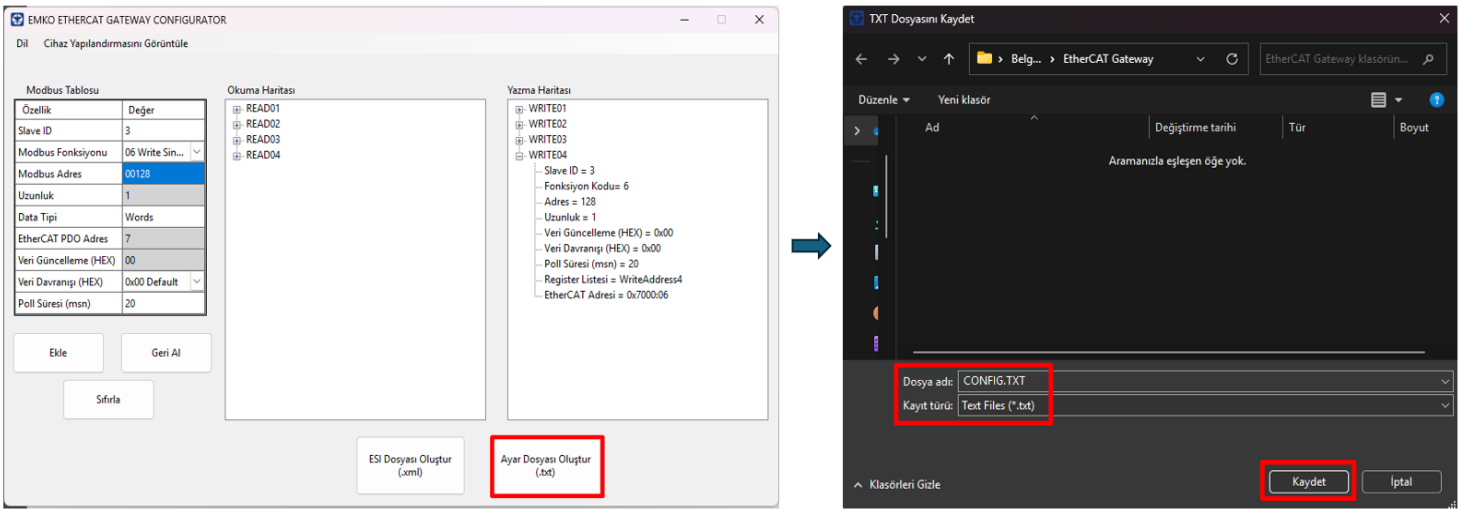
Şekil 25. EMKO ETHERCAT GATEWAY CONFIGURATOR yazılımında haberleşme haritalarının ayarlanması

Bu işlemlerin ardından modül haberleşme yapısının EtherCAT Master tarafından doğru şekilde tanınabilmesi için ESI dosyası oluşturma adımı gerçekleştirilecektir. Bu işlem için ESI Dosyası Oluştur butonuna tıklanarak kayıt penceresine erişilir. Burda istenilen konuma ESI dosyası (ETHCAT_GATEWAY.xml) kaydedilir. İlgili işlem adımları Şekil 26'da gösterilmiştir.



Şekil 26. EMKO ETHERCAT GATEWAY CONFIGURATOR yazılımında ESI dosyası oluşturma

İlgili işlemin ardından haritalama ayarlarının EtherCAT Gateway modülüne yüklenebilmesi için bir ayar dosyası oluşturulması gerekmektedir. Bu işlem için de arayüzde Ayar Dosyası Oluştur butonuna tıklanarak kayıt penceresine erişilir. Burda istenilen konuma ayar dosyası (CONFIG.TXT) kaydedilir. İlgili işlem adımları Şekil 27'de gösterilmiştir.



Şekil 27. EMKO ETHERCAT GATEWAY CONFIGURATOR yazılımında ayar dosyası oluşturma

Mini USB ile Yığın Bellek Üzerinden EtherCAT Gateway Modülünün Yapılandırılması

Modülün yapılandırılması için bu dökümanda “Mini USB ile Parametre Konfigürasyonu” başlığında anlatılan adımlar uygulanır.

İlgili başlıkta anlatılan tüm adımları uygulanmaktadır.

Slave modüllerin haberleşme ayarları: Baud hızı 57600 bps, eşlik biti yok, durma biti 1’dir. Modül parametreleri belirtilen şekilde ayarlanıp EMKO ETHERCAT GATEWAY CONFIGURATOR yazılımından üretilen CONFIG.TXT dosyası arayüze yüklenerek cihaza yazılır.

TwinCAT üzerinden EtherCAT Gateway Modülü ile Haberleşme

Bu uygulama notunda, EtherCAT haberleşmesinin doğrulanması ve modül konfigürasyonlarının test edilmesi amacıyla Beckhoff’un EtherCAT master yazılımı olarak **TcXaeShell** kullanılmıştır. TcXaeShell, Beckhoff’un eXtended Automation Engineering (XAE) tabanlı mühendislik arayüzüdür ve bu uygulama, TwinCAT 3 ile birlikte gelmektedir

Bu uygulama notunda **TwinCAT 3 Build 4024** kullanılmıştır. TwinCAT, EtherCAT ağına bağlı cihazların yapılandırılması, PDO/SDO parametrelerinin görüntülenmesi, çevrimsel veri transferlerinin izlenmesi ve tanılama işlemleri için ortam sunar.

TwinCAT 3 yazılımı Beckhoff'un resmi web sitesinden ücretsiz olarak indirilebilmektedir:
https://www.beckhoff.com/twincat3

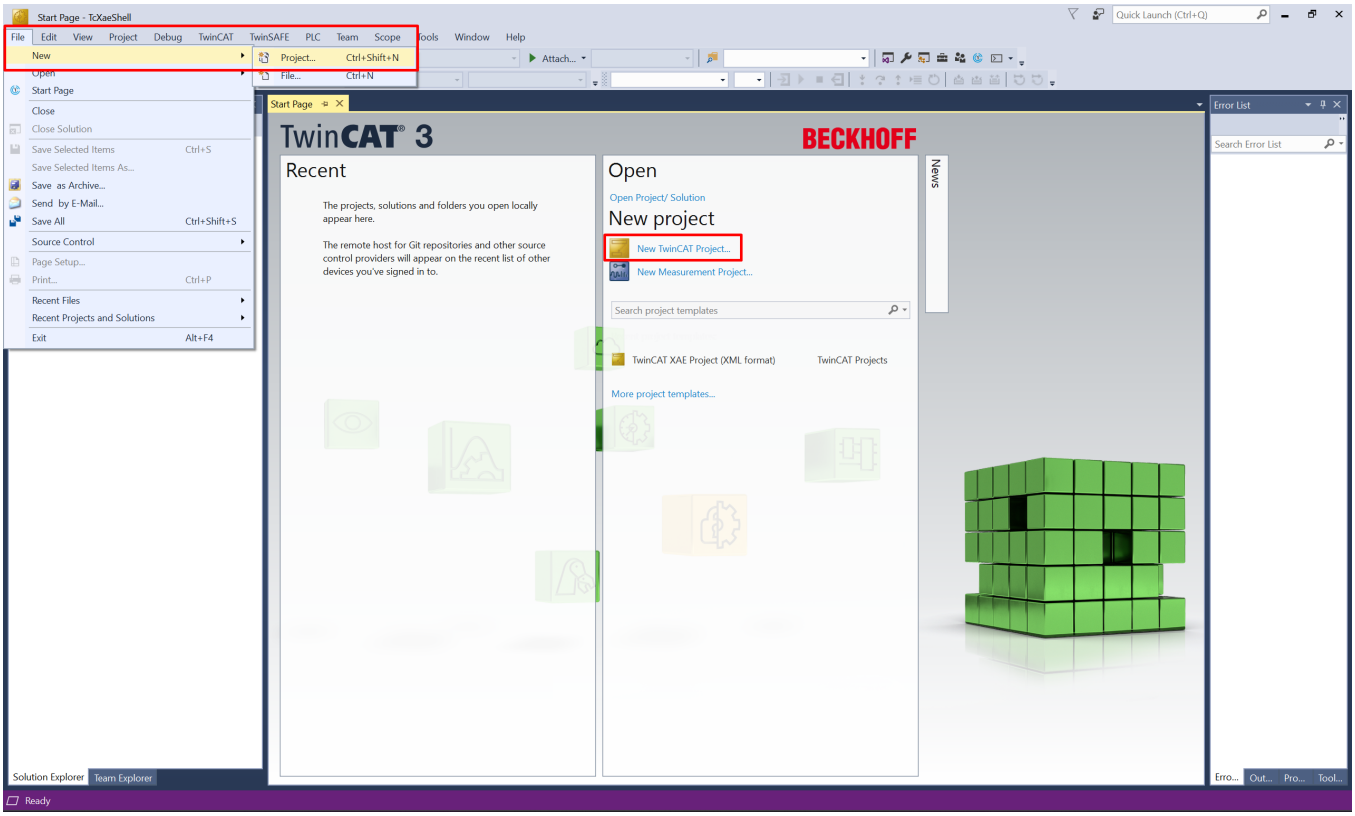
Kurulum gerçekleştirildikten sonra, cihazın EtherCAT master tarafından doğru şekilde tanınabilmesi ve yapılandırılabilmesi için, EMKO ETHERCAT GATEWAY CONFIGURATOR yazılımı üzerinden oluşturulan ESI (EtherCAT Slave Information) dosyasının (ETHCAT_GATEWAY.xml) kütüphaneye eklenmesi gerekmektedir. ESI dosyası, slave cihaza ait tanımlayıcı bilgileri ve PDO haritasını içermektedir.

ESI dosyasının TwinCAT sistemine tanıtılması için dosyanın kütüphaneye eklenmesi gerekmektedir. Bunun için ESI dosyası, uygulamanın yüklü olduğu dizindeki EtherCAT klasörüne eklenmelidir. Varsayılan kurulum yolu kullanıldığında ESI dosyası aşağıda belirtilen yere yüklenmelidir.

C:\TwinCAT\3.1\Config\Io\EtherCAT

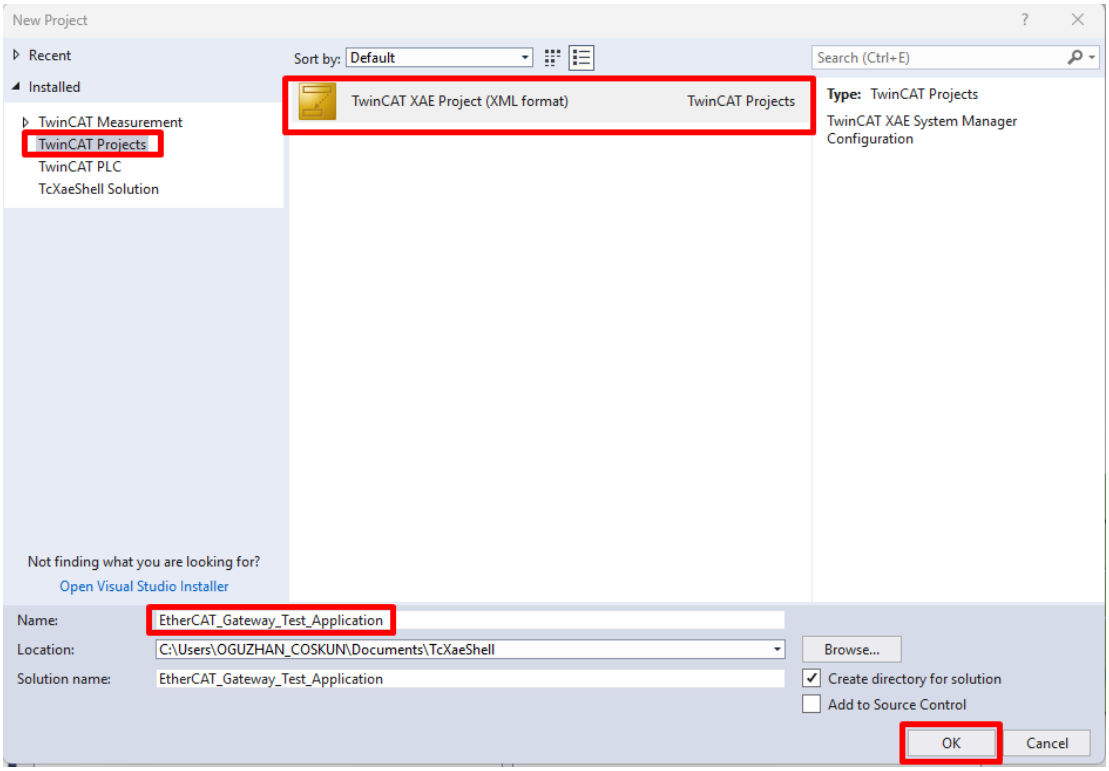
Not: TwinCAT farklı bir dizine kurulduysa, ilgili Io\EtherCAT klasörü kendi kurulum dizini altında yer alacaktır.

Adım 1 : Proje Oluşturma
Kurulum tamamlandıktan sonra, yapılandırma işlemleri için **TcXaeShell** uygulaması açılır. Burada **File** menüsünden **New** ve **Project** yollarını izleyerek veya arayüzde kısayol olarak yer alan **New TwinCAT Project** sekmesinden proje oluşturulmalıdır. İlgili arayüz ve sekmeler Şekil 28’de gösterilmektedir.



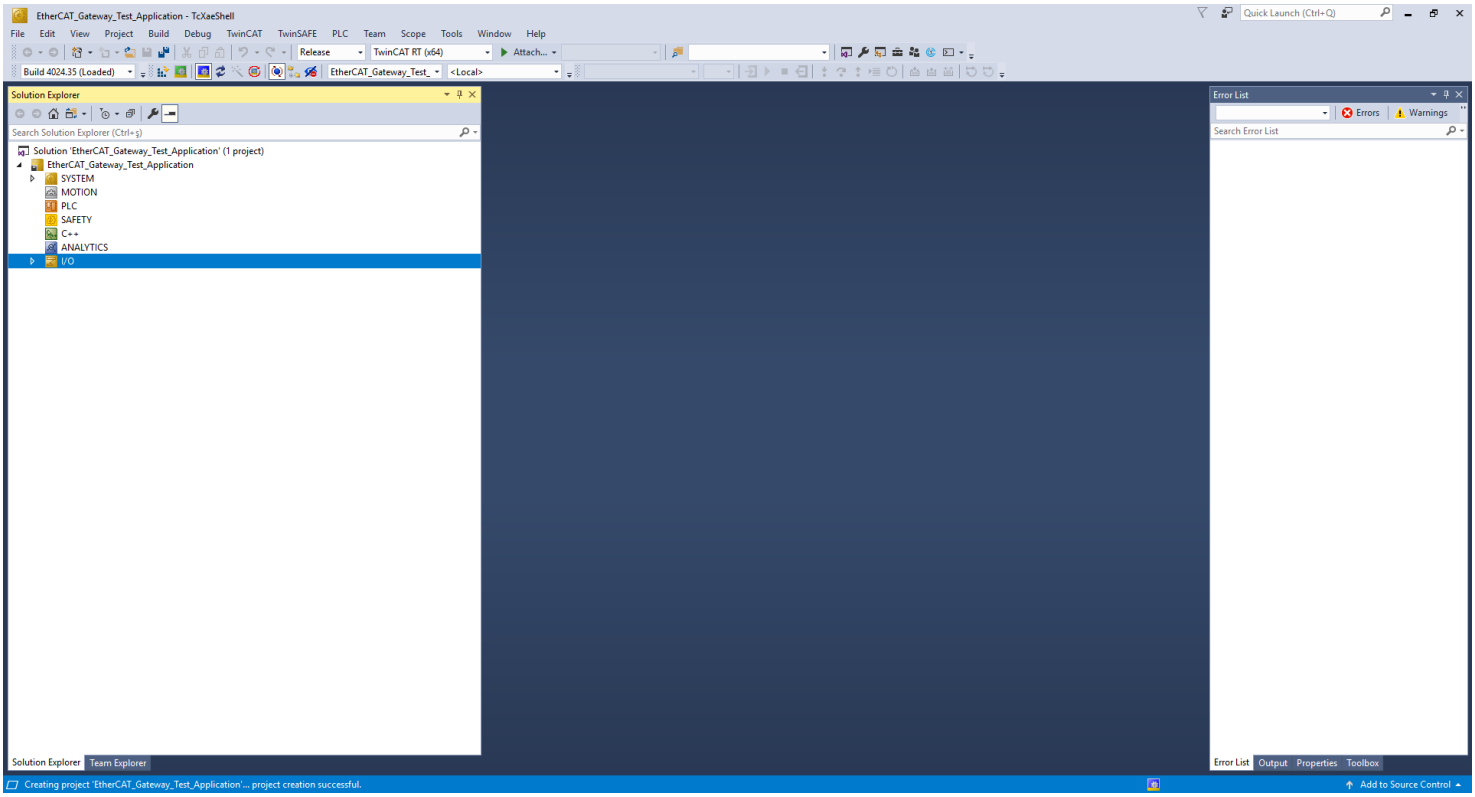
Şekil 28. TcXaeShell arayüzü ve proje oluşturma sekmeleri

Daha sonra **TwinCAT Projects** seçilir ve proje adı ile dosyanın kaydedileceği yer için ayarlar gerçekleştirilir. İşlemlerin ardından **OK** butonuna tıklanarak proje oluşturma adımı tamamlanır. İlgili işlem adımı Şekil 29'da gösterilmektedir.



Şekil 29. Proje oluşturma ayarları

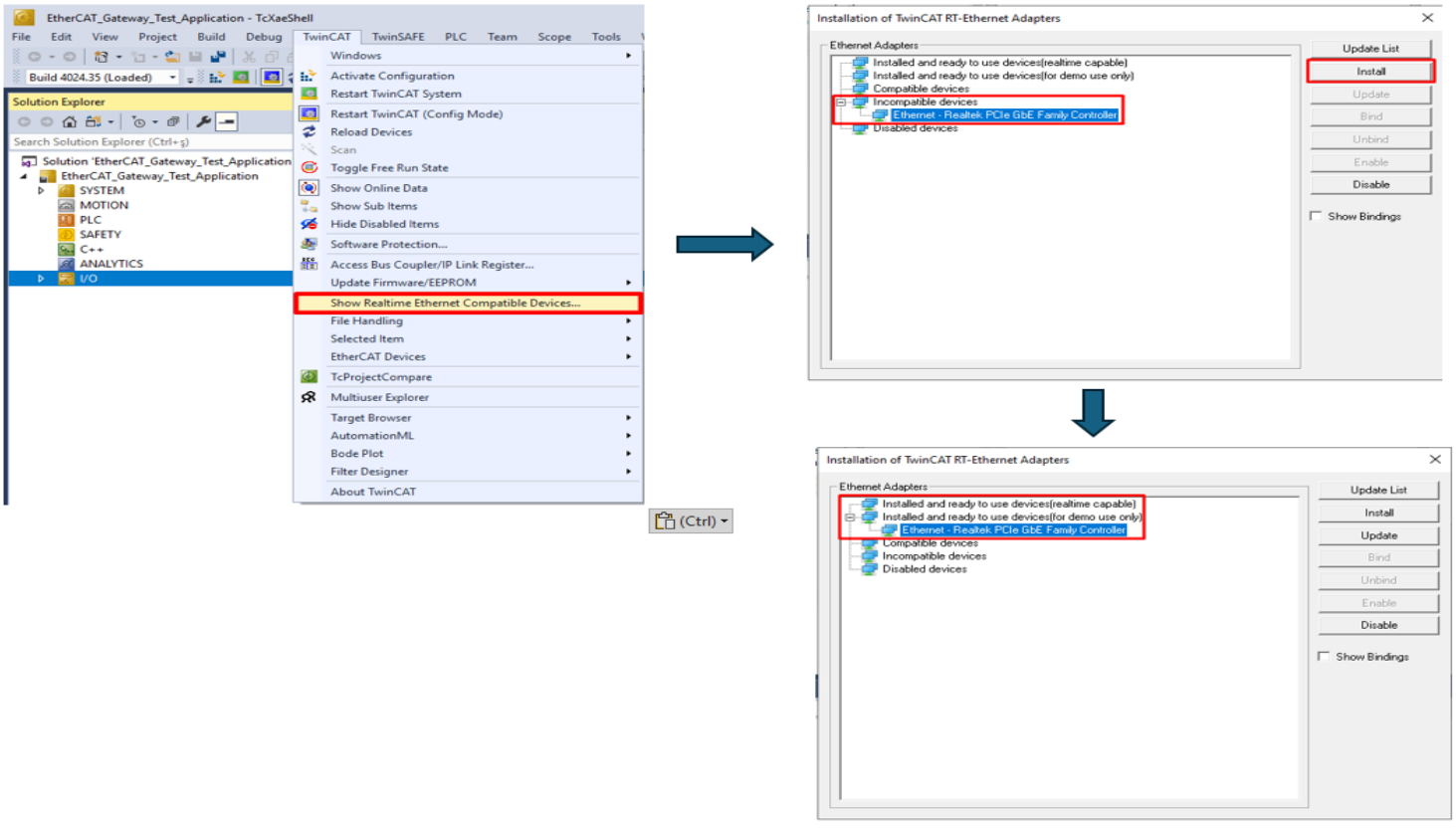
İşlem adımları tamamlandıktan sonra Şekil 30'da görülen TwinCAT proje arayüzü oluşmaktadır.



Şekil 30. TwinCAT proje arayüzü

Adım 2 : Ethernet Adaptör Ayarı

TwinCAT programını kişisel bilgisayarına ilk defa kuran kullanıcılar için Ethernet adaptör ayarlarının gerçekleştirilmesi gerekmektedir. Bu işlem için öncelikle **TwinCAT** menüsü altında **Show Realtime Ethernet Compatible Devices** sekmesine girilir. Karşımıza çıkan izin uyarısı onaylanarak işleme devam edilir. **Incompatible devives** başlığı altında listelenen adaptörler arasından kişisel bilgisayarınızın Ethernet adaptörü seçilip **Install** ile yükleme gerçekleştirilir. Yükleme tamamlandığında seçilen adaptör **Installed and ready to use devices** başlığı altında listelenir. İşlem adımları Şekil 31'de gösterilmektedir.



Şekil 31. TwinCAT Ethernet adaptör ayarları işlem adımları

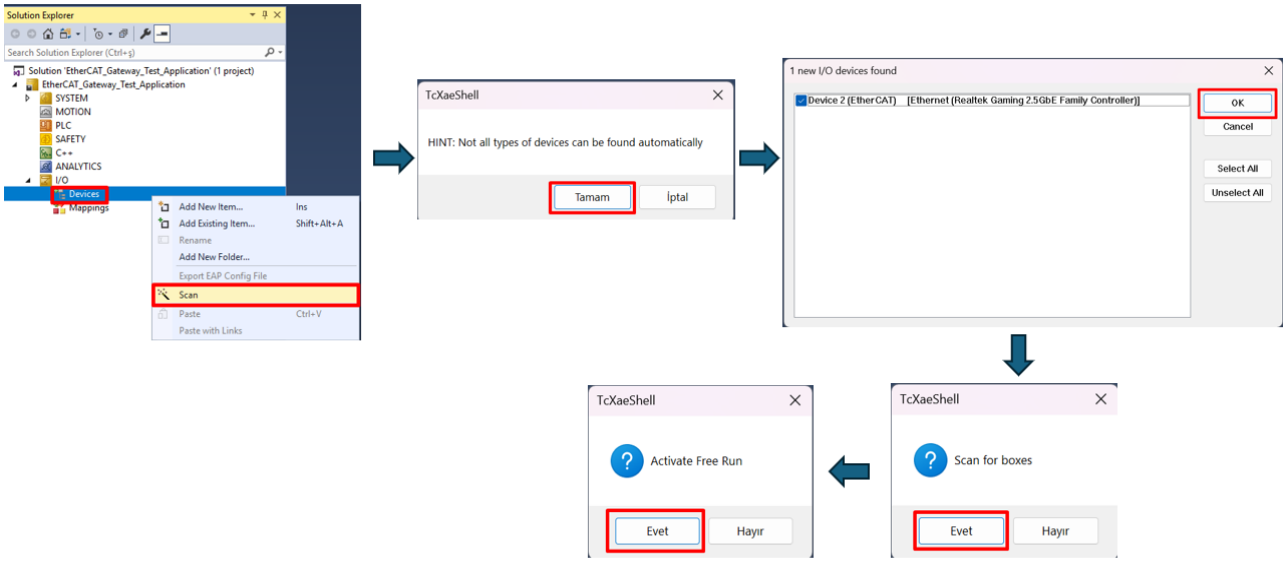
Adım 3 : EtherCAT Gateway Modülünün Projeye Eklenmesi ve Çalıştırılması
Ayar işleminin tamamlanmasının ardından EtherCAT Gateway modülünün EtherCAT slave olarak tanıtılarak projeye dahil edilmesi gerekmektedir.

Tanıtma işlemine başlamadan önce, Şekil 23'te gösterildiği gibi EtherCAT Gateway modülünün 24V DC besleme bağlantısının sağlanmış olması ve Ethernet kablosunun kişisel bilgisayarın Ethernet portu ile Bus Coupler üzerindeki "IN" olarak etiketlenmiş Ethernet soketi arasında bağlantı kuracak şekilde takılmış olması gerekmektedir.

EtherCAT Gateway modülünün tanıtılarak projeye dahil edilmesi, otomatik ve manuel olarak iki şekilde gerçekleştirilmektedir.

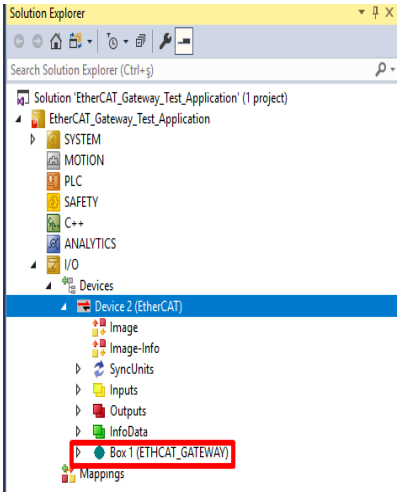
Otomatik Tanımlama:

Solution Explorer bölümünde IO sekmesi altında yer alan Devices'a sağ tıklanarak Scan sekmesi seçilir. Pop-up olarak açılan tüm cihazların otomatik olarak bulunacağına yönelik uyarıyı belirten ekranda OK seçeneği işaretlenerek devam edilir. Tarama işlemi sonucunda kullanılan Ethernet soketinde bir cihaz bulunduğuna yönelik uyarı ekranı oluşur. Burada da OK butonuna tıklanarak işleme devam edilir. Daha sonra Scan for boxes uyarısında da Evet seçeneği işaretlenir. Bu işlemin ardından Free Run aktive edilsin mi uyarı ekranında Evet seçeneği seçilerek işlem tamamlanır. İşlem adımları Şekil 32'de gösterilmektedir.



Şekil 32. EtherCAT Gateway modülünün otomatik tanımlama ile projeye dahil edilmesi işlem adımları

Tamamlamanın ardından Şekil 33'te görüldüğü üzere EtherCAT Gateway modülü projeye dahil edilmiş olur ve aktive edilerek herhangi bir hata olmaması durumunda Operation (OP) modunda çalışmaya başlar.

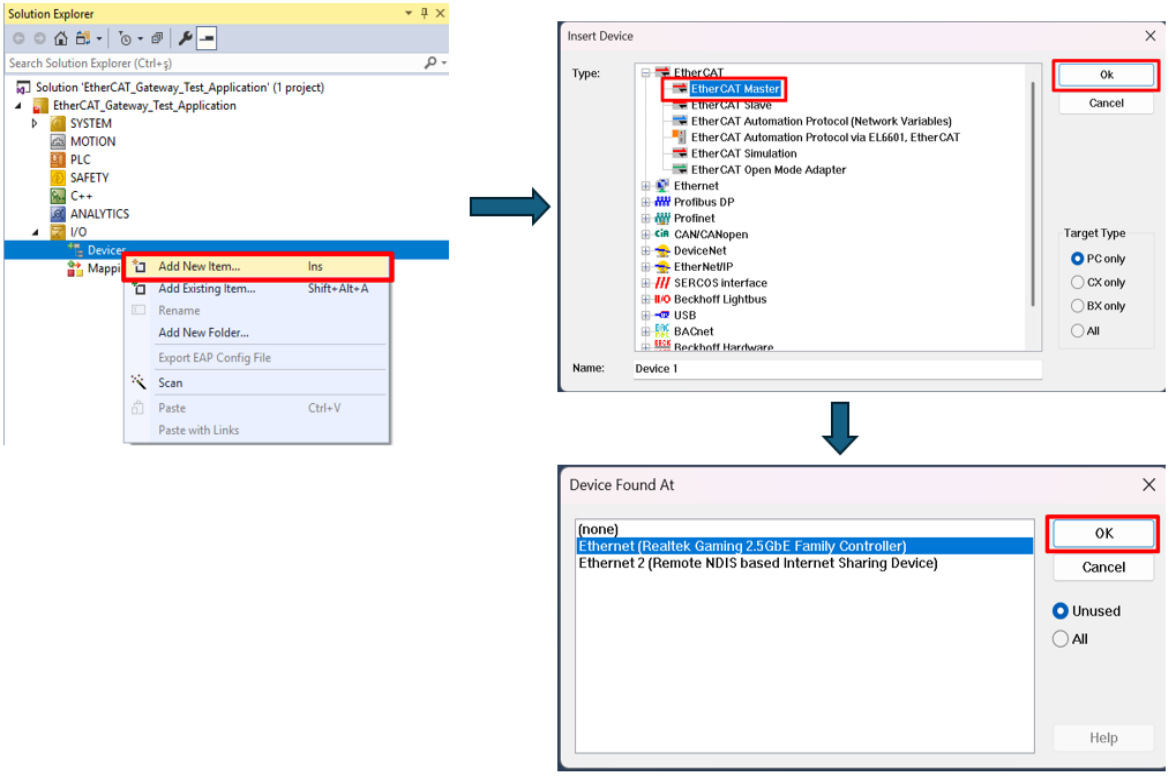


Şekil 33. Otomatik tanımlama ile EtherCAT Gateway modülünün projeye dahil edilmesi

Burada Device 2 (EtherCAT) EtherCAT master'ı, Box 1 (ETHCAT_GATEWAY) EtherCAT GATEWAY modülünü temsil etmektedir.

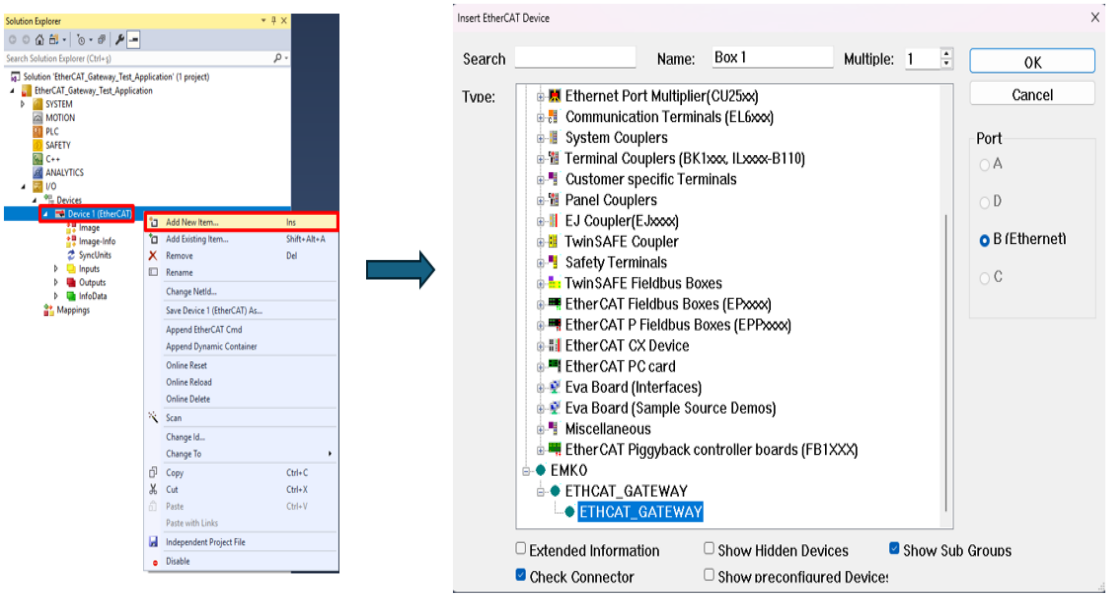
Manuel Tanımlama:

Solution Explorer bölümünde IO sekmesi altında yer alan Devices'a sağ tıklanarak Add New Item sekmesi seçilir. Ardından açılan Insert Device penceresinde EtherCAT başlığı altındaki EtherCAT Master işaretlenir ve OK tuşuna basılır. Daha sonra hangi Ethernet adaptörünün kullanılacağına yönelik seçim ekranı oluşur. Burada Coupler ile bağlantı sağlanan adaptör seçilir ve OK seçeneği ile ekleme işlemi tamamlanır. İşlem adımları Şekil 34'te gösterilmektedir.



Şekil 34. Manuel tanımlama işlem adımında EtherCAT Master'ın eklenmesi

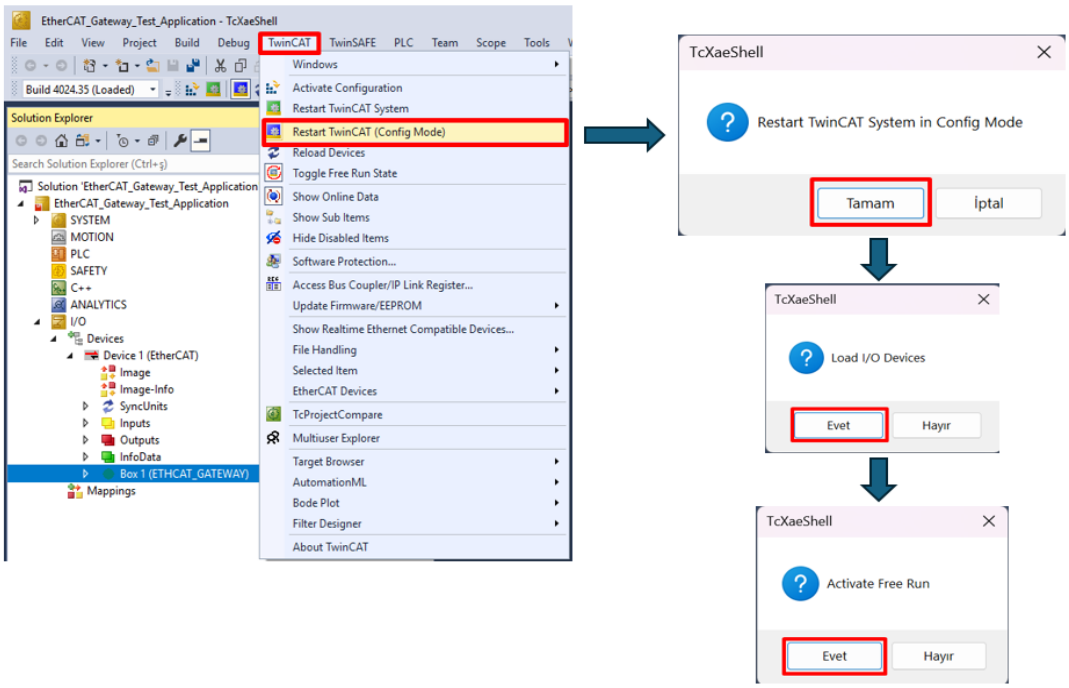
Master'ın eklenmesinin ardından EtherCAT Gateway modülünün eklenmesi gerekmektedir. Bu işlem için ilk olarak Devices'ın altında listenen Master'a (Bu örnekte Device 1 (EtherCAT)) sağ tıklanarak Add New Item sekmesi seçilir. Açılan Insert EtherCAT Device penceresinde EMKO başlığının altında yer alan ETHCAT_GATEWAY başlığının içerisindeki ETHCAT_GATEWAY seçilir ve OK tuşuna basılarak ekleme işlemi tamamlanır. İşlem adımları Şekil 35'te gösterilmektedir.



Şekil 35. Manuel tanımlama işlem adımında EtherCAT Gateway modülünün eklenmesi

İşlemin tamamlanmasının ardından EtherCAT Gateway modülü, Şekil 33'te otomatik tanımlama modunda olduğu gibi projeye eklenmiş olur. Burada manuel ekleme işlemi yapıldığından otomatik olarak çalışmaya başlamaz. Başlatma işlemi için TwinCAT menüsü altında veya Standard durum çubuğunda yer alan ya da yer Restart TwinCAT (Config Mode)

sekmesine tıklanır. Restart TwinCAT System in Config Mode uyarısı için açılan pop-up ekranında Tamam butonuna tıklanarak devam edilir. Load I/O Devices uyarısı için açılan pop-up ekranında Evet'e tıklanarak işleme devam edilir. Sonrasında Activate Free Run uyarısı için açılan pop-up ekranında Evet'e tıklanarak işlem EtherCAT Gateway modülü Operation mode'da çalışmaya başlatılır. İlgili işlem adımları Şekil 36'da gösterilmektedir.

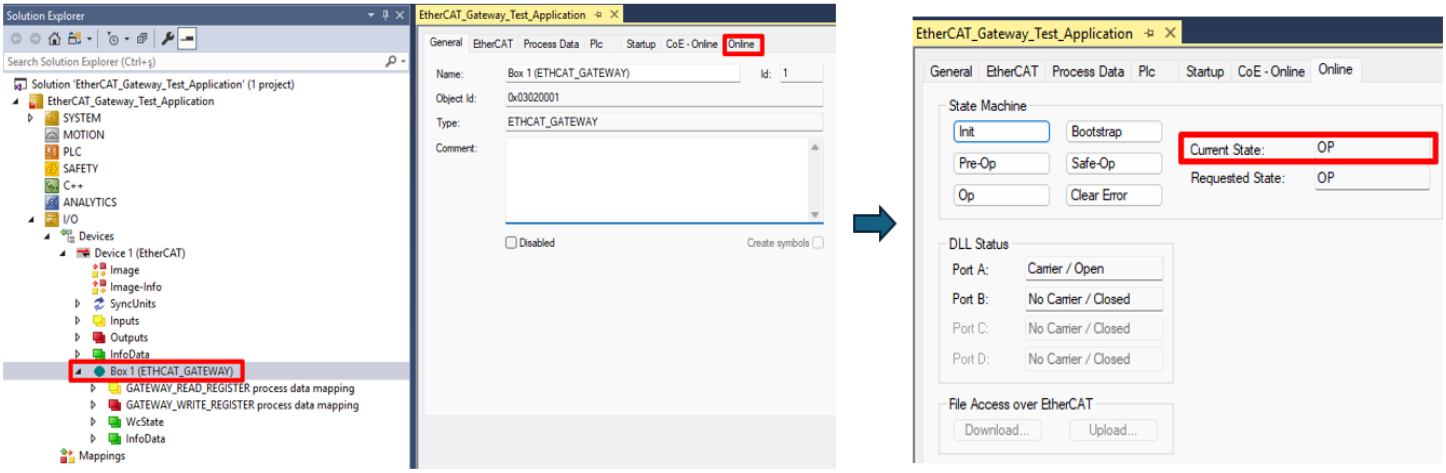


Şekil 36. EtherCAT Gateway modülünün çalıştırılma işlemi

Modülün Operation Mode'da çalışmaya başladığı modül üzerinden ve master yazılımdan olmak üzere iki şekilde teyit edilebilmektedir.

0- Modül üzerinde yer alan Status/Run LED Davranışı (Sarı LED) sürekli yanar. Ayrıca NS LED'i yeşil renkte sürekli olarak yanmaktadır.

1- TwinCAT arayüzünde projeye eklenen Box 1 (ETHCAT_GATEWAY)' e çift tıklanarak modül arayüzüne erişilir. Burada, Online sekmesine tıklanarak modülün çalışma durumu gözlemlenir. İşlem adımları Şekil 37'da gösterilmiştir.



Şekil 37. EtherCAT Gateway modülünün çalışma modunun kontrolü

Adım 6 : TxPDO ve RxPDO Verilerinin Okunup Yazılması

TwinCAT arayüzünde, *Device 1 (EtherCAT)* başlığı altında yer alan *Box 1 (ETHCAT_GATEWAY)* modülü seçildiğinde, gateway cihazına ait process data mapping alanları görüntülenebilmektedir. Bu alanlar:

- GATEWAY_READ_REGISTER (TxPDO, 0x1A00)
- GATEWAY_WRITE_REGISTER (RxPDO, 0x1600)

şeklindedir.

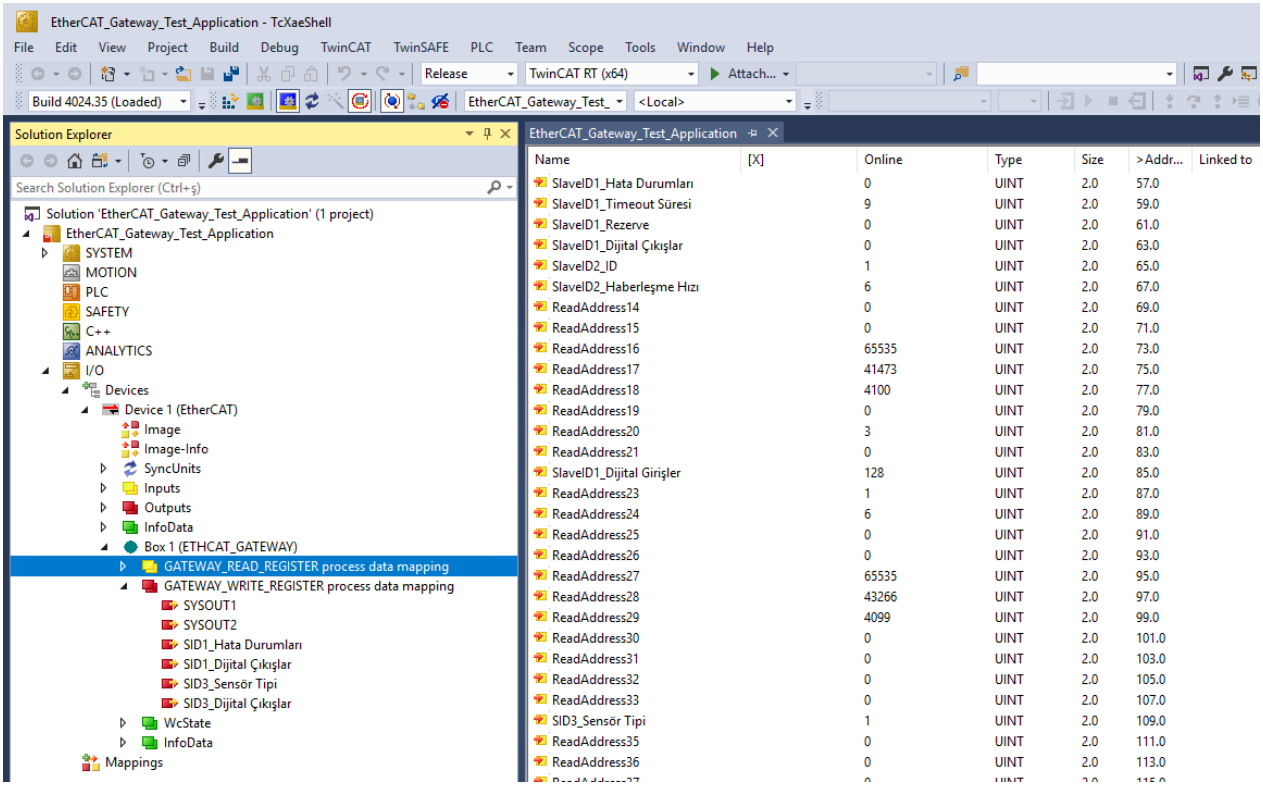
Bu PDO alanları, RS485 hattındaki genişleme modüllerinden okunan veya bu modüllere yazılacak verilerin EtherCAT sistemine yansıtıldığı register yapısını oluşturur.

Bu PDO başlıklarının altında, EMKO ETHERCAT GATEWAY CONFIGURATOR yazılımı ile yapılandırılan **okuma ve yazma mappinglerine** karşılık gelen register'lar yer almaktadır. Bu alandaki register isimleri varsayılan olarak ReadAddresX, WriteAddressX şeklinde sıralanır. Örneğin bu uygulama senaryosunda:

- ReadAddress1 – ReadAddress11 arası, Slave ID 1 adresli genişleme modülünün Modbus adresi 0'dan başlayan ve 11 uzunluğunda olan veri alanını temsil eder. Modbus F3 (0x03) fonksiyonu ile okunur
- ReadAddress12 – ReadAddress22 arası, Slave ID 2 üzerindeki Modbus adresi 0'dan başlayan 11 uzunluğundaki veri alanını yansıtır. Modbus F3 (0x03) fonksiyonu ile okunur
- ReadAddress23 – ReadAddress54 arası, Slave ID 3'teki Modbus adresi 0'dan başlayan ve 32 uzunluktaki veri alanını temsil eder. Modbus F3 (0x03) fonksiyonu ile okunur
- ReadAddress55, Slave ID 3'ün Modbus adresi 128'den başlayan ve uzunluğu 1 olan veri alanına karşılık gelir. Modbus F3 (0x03) fonksiyonu ile okunur
- WriteAddress1, Slave ID 1'in Modbus adresi 7'ye yazılacak tek bir veriyi temsil eder. Modbus F6 (0x06) fonksiyonu ile yazılır.
- WriteAddress2, Slave ID 1'in Modbus adresi 10'a yazılacak veriyi gösterir. Modbus F6 (0x06) fonksiyonu ile yazılır.
- WriteAddress3, Slave ID 3'ün Modbus adresi 11'e yazılacak tek bir register'ı temsil eder. Modbus F6 (0x06) fonksiyonu ile yazılır.
- WriteAddress4, Slave ID 3'teki Modbus adresi 128'e yazılacak tek bir veriyi temsil eder. Modbus F6 (0x06) fonksiyonu ile yazılır.

TwinCAT arayüzünde bu adresler, Input ve Output gruplarında sırasıyla **Input Process Data** ve **Output Process Data** olarak yer almakta olup, mapping sırasında belirlenen sıraya göre otomatik olarak oluşturulmuştur. Her bir adres, yapılandırma esnasında belirlenen modbus adreslerine, fonksiyon kodlarına, uzunluklara ve veri davranışına göre gateway cihazı tarafından yönetilmektedir.

ReadAddressX ve WriteAddressX alanları, varsayılan olarak oluşturulmuş isimlerle görünmektedir. Ancak kullanıcı arayüzünde daha anlamlı ve okunabilir bir yapı elde etmek amacıyla, bu alanlar isteğe bağlı olarak yeniden adlandırılabilir. Örneğin, ReadAddress22 alanı "SlaveID1_Dijital Girişler" gibi daha açıklayıcı bir isimle değiştirilerek ilgili analog kanaldan alınan ortalama değeri temsil edebilir. Böylece kullanıcı, sistemdeki her bir register'ın hangi fiziksel veriyle ilişkili olduğunu doğrudan anlayabilir. Bu işlem için Box1 altında yer alan PDO alanlarında ilgili register değerine sağ tıklanıp Rename yapılarak gerçekleştirilebilir. Bu tür yeniden adlandırmalara örnek bir yapı Şekil 38'de gösterilmiştir:

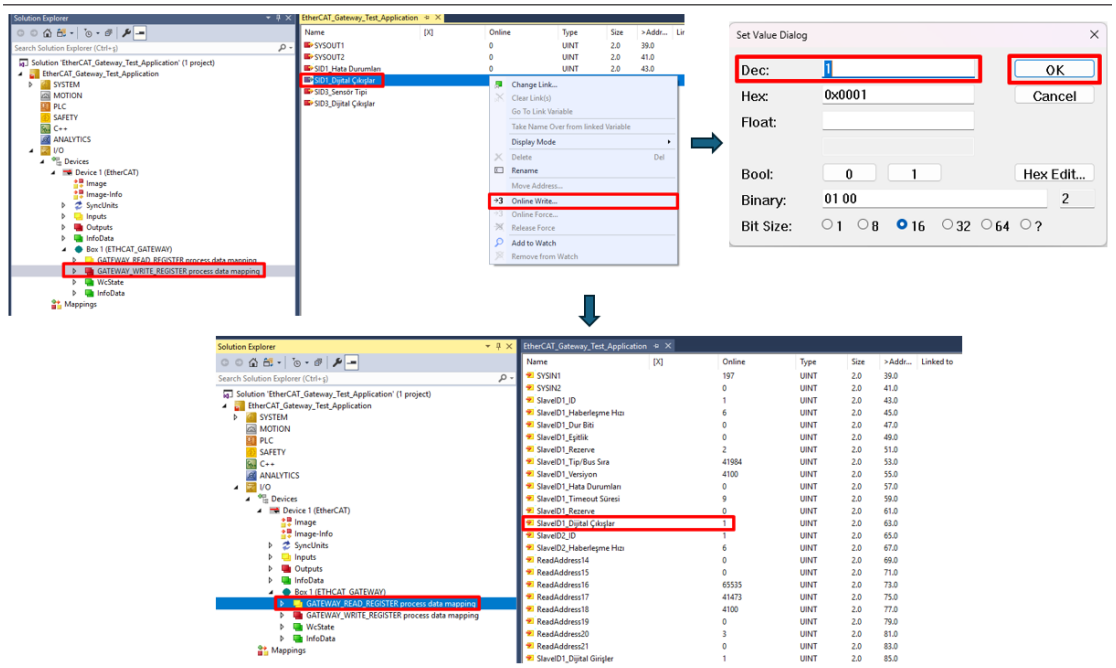


Şekil 38. Okuma ve Yazma Register'larının yeniden adlandırılmasına bir örnek

Bu senaryoda, Slave ID 1 olarak tanımlı olan **SMR1P1-00-00-01-00_1 (15xTO)** kodlu dijital çıkış modülündeki **Q0 çıkışı**, bir AC lambayı sürmek amacıyla kullanılmaktadır. Yapılan PDO mapping konfigürasyonuna göre, bu çıkışın kontrolü Modbus adresi 10 üzerinden yapılmakta ve bu alan konfigürasyon arayüzündeki yapılandırmaya göre **WriteAddress2** olarak tanımlanmıştır *(yeniden adlandırılmış haliyle: SID1_Dijital_Çıkışlar)*.
Q0 çıkışını aktif etmek için aşağıdaki adımlar izlenebilir:

1. **TwinCAT** arayüzünde, *Device 1 > Box 1 (ETHCAT_GATEWAY) > GATEWAY_WRITE_REGISTER process data mapping* altında bulunan ilgili PDO alanı içerisinden **WriteAddress2** (veya yeniden adlandırılmış haliyle SID1_Dijital_Çıkışlar) bulunur.
2. Bu alana sağ tıklanarak, açılan menüden **Online Write** seçeneği seçilir.
3. Açılan pencerede, ilgili alana değer olarak **1** yazılır. (Q0 çıkışı, ilgili adresin 0. bitinde yer aldığından, yalnızca bu biti aktif etmek için 0x0001 değeri yeterlidir.)
4. Bu işlem sonrasında, Q0 çıkışı aktif hâle gelir ve çıkış terminaline bağlı olan SSR (Solid State Röle) tetiklenir.
5. Pozitif tetikleme mantığıyla çalışan bu yapı sayesinde, SSR çıkışı üzerinden bağlı olan AC lamba çalıştırılır.

Çıkışın durum bilgisini okumak için ise, GATEWAY_READ_REGISTER PDO altında yer alan ReadAddress11 (yeniden adlandırılmış haliyle SID1_Dijital_Çıkışlar) adresi kullanılabilir. Bu alana incelendiğinde ilk çıkış aktif olduğundan (0.bit) 1 değeri gözlemlenir. İlgili işlemler Şekil 39'da gösterilmiştir.



Şekil 39. SMR1P1-00-00-01-00_1 (15xTO) kodlu genişleme modülünde dijital çıkışlardan Q0 çıkışının aktif edilmesi ve çıkış durumunun okunması

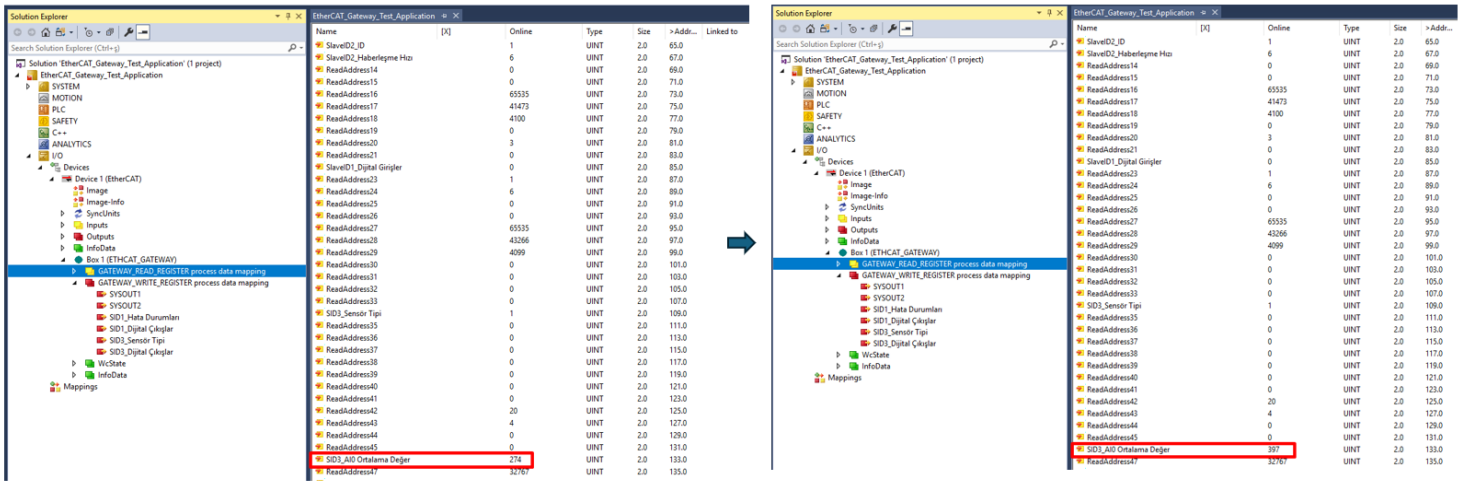
Q0 çıkışına karşılık gelen I0 dijital girişinin okunması:
Q0 çıkışı, ikinci genişleme modülünde (SMR1P1-01-00-00-00_1 (16xDI)) bulunan I0 dijital girişine bağlanmıştır. Yapılan PDO mapping konfigürasyonuna göre, bu girişin kontrolü Slave ID 2 Modbus adresi 10 üzerinden yapılmakta ve bu alan konfigürasyon arayüzündeki yapılandırmaya göre **ReadAddress22** olarak tanımlanmıştır *(yeniden adlandırılmış haliyle: SlaveID2_Dijital Girişler)*.

İlgili PDO adresi incelendiğinde (0. bit) aktif olduğundan **ReadAddress22** *(yeniden adlandırılmış haliyle: SlaveID2_Dijital Girişler)* değişkeni üzerinden 1 değeri gözlemlenir.

Analog giriş üzerinden sıcaklık değerinin okunması:
Üçüncü genişleme modülü olan SMR1P1-00-01-04-00_1 (4xAI-SU, 4xTO), AC lambaya temas ettirilen PT100 sıcaklık sensöründen veri okumaktadır. Yapılan PDO mapping konfigürasyonuna göre, bu girişin kontrolü Slave ID 3 Modbus adresi 23 üzerinden yapılmakta ve bu alan konfigürasyon arayüzündeki yapılandırmaya göre **ReadAddress46** olarak tanımlanmıştır (yeniden adlandırılmış haliyle: SID3_AI0 Ortalama Değer).

İlgili PDO adresi incelendiğinde **ReadAddress46** *(yeniden adlandırılmış haliyle:* SID3_AI0 Ortalama Değer*)* değişkeni üzerinden sıcaklık değeri gözlemlenir. Şekil 40'da lamba açılmadan önceki ve sonraki sıcaklık değerleri gösterilmektedir.

Not : Bu modülde analog sıcaklık değeri, ondalıklı (float) formatta değil, sabit bir ölçekleme faktörü ile çarpılarak tam sayı (integer) formatında iletilir. Ölçekleme katsayısı 10 olduğundan, örneğin 27.6 °C sıcaklık değeri PDO alanında 276 olarak görülür.

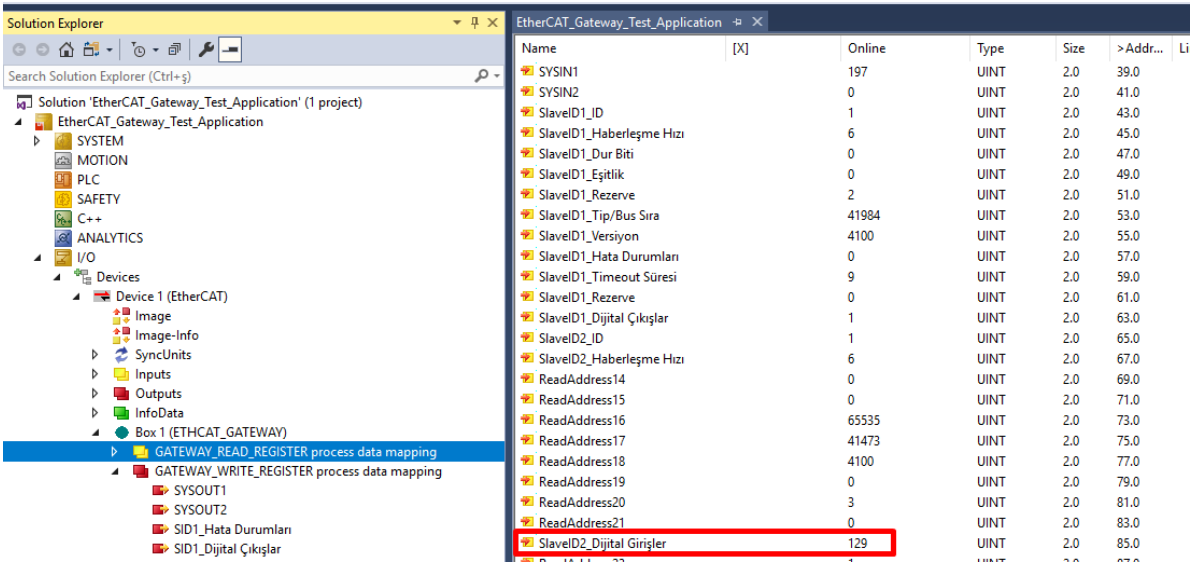


Şekil 40. SMR1P1-00-01-04-00_1 (4xAI-SU, 4xTO) kodlu genişleme modülündeki analog giriş değerinin okunması

Anahtar Durumuna Göre Dijital Giriş Okunması:
Uygulama senaryosu kapsamında, ikinci genişleme modülü olan SMR1P1-01-00-00-00_1 (16xDI) üzerinde yer alan I7 dijital girişine bir manuel anahtar bağlanmıştır. Anahtarın bir ucu +24 V hattına, diğer ucu ise doğrudan I7 girişine bağlanmıştır. Anahtar kapatıldığında, I7'ye (7.bit) lojik '1' uygulanarak giriş aktif hale getirilir. Ayrıca, hali hazıırda Q0 çıkışı ile I0 (0.bit) girişi birbirine bağlandığından ve bu çıkış da aktif olduğundan I0 da aktif durumdadır.

Yapılan PDO mapping konfigürasyonuna göre, bu girişin kontrolü Slave ID 2 Modbus adresi 10 üzerinden yapılmakta ve bu alan konfigürasyon arayüzündeki yapılandırmaya göre **ReadAddress22** olarak tanımlanmıştır (yeniden adlandırılmış haliyle: SID3_AI0 Ortalama Değer).

İlgili PDO adresi incelendiğinde **ReadAddress46** *(yeniden adlandırılmış haliyle:* SlaveID2_Dijital Girişler*)* değişkeni üzerinden girişlerin durumu gözlemlenir. Daha önce belirtildiği üzere QO çıkışı da I0 girişine bağlı olduğundan anahtar kapatıldığı durumda Şekil 41'de de görüldüğü üzere **ReadAddress46** *(yeniden adlandırılmış haliyle:* SlaveID2_Dijital Girişler*)* değişleninde bu iki bitin toplamı olan **129 ($2^7 + 2^0$)** değeri okunur.



Şekil 41. SMR1P1-01-00-00-00_1 (16xDI) kodlu genişleme modülünde anahtar kapatılınca okunan giriş değeri

Uygulama Notu 2 : Çoklu EtherCAT Gateway Kullanımı ve Explicit Cihaz ID ile Modül Tanımlama

Hangi Durumlarda Birden Fazla EtherCAT Gateway Kullanılır:

Gerçek uygulamalarda Modbus RTU cihazları genellikle farklı panolara, üretim hücrelerine veya uzak saha noktalarına dağılmıştır.
Tek bir Gateway kullanmak, uzun kablo mesafeleri ve karmaşık RS485 topolojileri nedeniyle zorlayıcı olabilir. Bu sebeple:

- Her saha bölgesine bir Gateway yerleştirilerek o bölgedeki Modbus slave cihazlarının yönetimi sağlanır.
- EtherCAT zinciri üzerinde tüm Gateway'ler tek bir master'a bağlanarak merkezi kontrol elde edilir.
- Bu yapı sayesinde:
    - RS485 kablolama mesafesi kısalır ve daha güvenilir haberleşme sağlanır.
    - Sistem modüler ve genişletilebilir olur.
    - Farklı bölgelerdeki Modbus cihazları tek noktadan erişilebilir hale gelir.
    - Arızalı bir Gateway kolayca değiştirilerek sistemin geri kalanı çalışmaya devam eder.

Daisy Chain Topolojisinde IN ve OUT Portları:

- **IN Portu (RJ45)**: Önceki EtherCAT cihazından gelen veriyi alır.
- **OUT Portu (RJ45)**: Veriyi bir sonraki EtherCAT cihazına iletir.

Bu yapı sayesinde EtherCAT ağı, bir zincir şeklinde uzatılabilir. Bu zincir boyunca tüm cihazlar sıralı olarak tanımlanır.

Çoklu EtherCAT Gateway Kullanım Senaryoları:

Aynı Mapping Yapısına Sahip Gateway'ler

- Tüm Gateway cihazları aynı Modbus mapping ayarlarını kullanıyorsa, EtherCAT master için tek bir **ESI dosyası** yeterlidir.
- Master, her Gateway'i fiziksel bağlantı sırasına veya Explicit Cihaz ID'lerine göre tanıyarak haberleşme gerçekleştirilir.

Farklı Mapping Yapılarına Sahip Gateway'ler

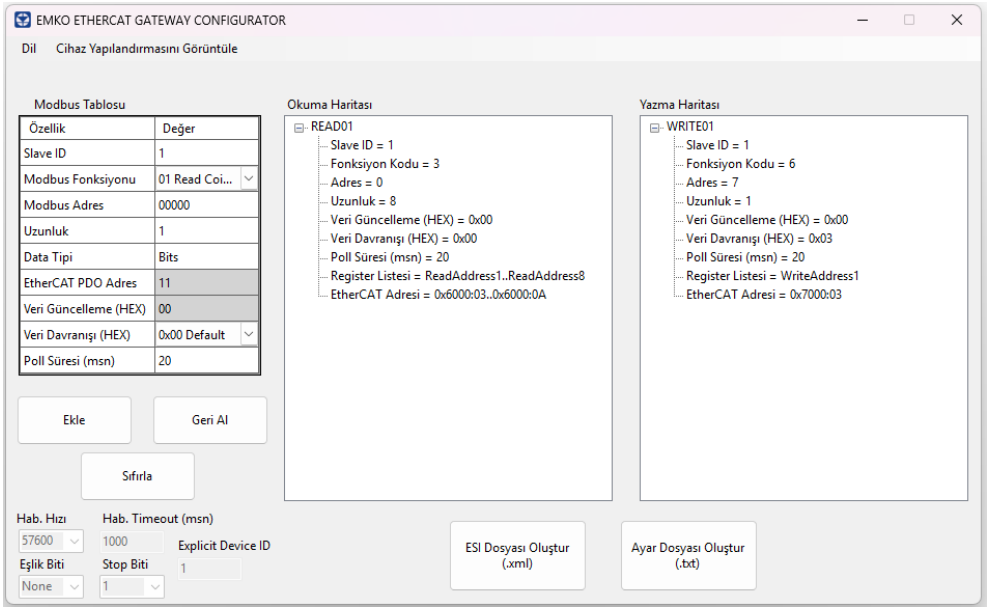
- Eğer Gateway cihazları farklı Modbus mapping yapılandırmalarıyla çalışıyorsa, her biri için farklı ESI dosyaları gerekir.
- Ancak tüm cihazların Vendor ID ve Product Code bilgileri aynı olduğundan EtherCAT master, her Gateway'e ait farklı ESI dosyalarını ayrıştıramaz. Master tarafında ayrım yapabilmek için **Revision Number** kullanılır.
- Kullanıcı, ESI dosyalarını manuel düzenleyerek bir Gateway'lerin **Revision Number** değerlerini değiştirebilir.
- EtherCAT master, modül ekleme aşamasında bu revision numaralarına göre doğru cihazı doğru ESI dosyasıyla eşleştirir.

Uygulama Senaryosu: Aynı Mapping Yapısına Sahip EtherCAT Gateway ile Dağıtık I/O Kontrolü:

Bu uygulama senaryosunda, EtherCAT haberleşmesi ile farklı fiziksel konumlarda yer alan aynı I/O modüllerin merkezi bir master üzerinden kontrol edilmesi ele alınmıştır. Senaryoda iki ayrı EtherCAT Gateway modülü kullanılmıştır.

Her iki EtherCAT Gateway modül ile SMR1P1-03-00-02-00_1 (8xDI, 8xTO) modülleri arasında RS485 bağlantı sağlanmıştır.

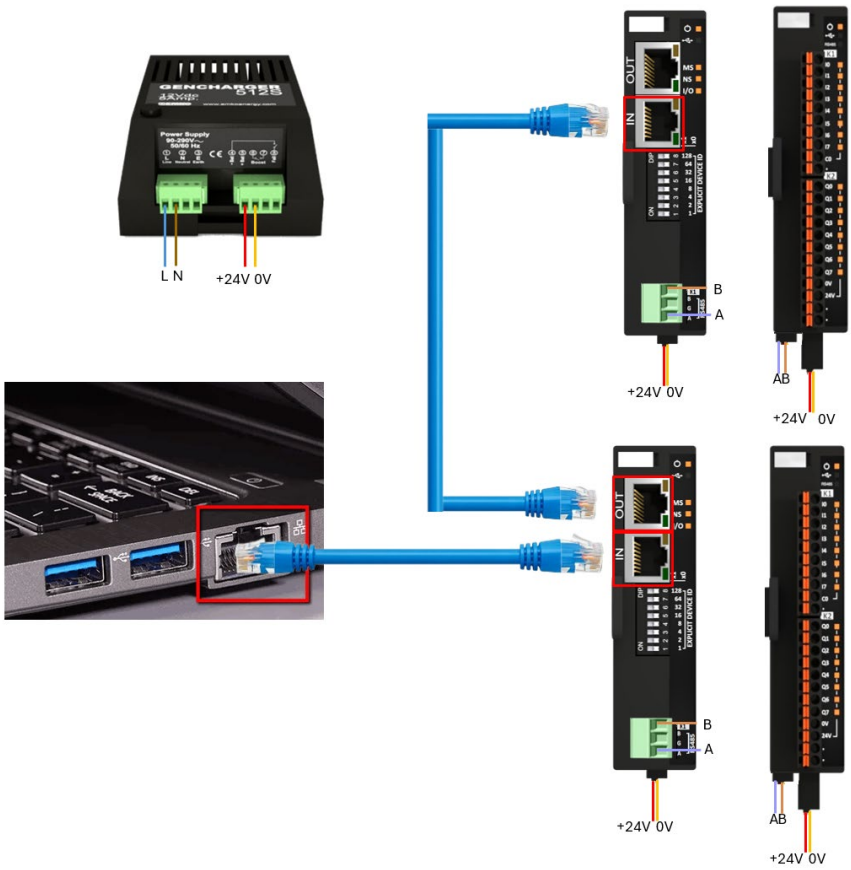
Gateway modüllerinde tanımlanmış olan Modbus mapping yapılandırması ve haberleşme parametreleri, **EMKO ETHERCAT GATEWAY CONFIGURATOR** yazılımında **Cihaz Yapılandırmasını Görüntüle** menüsü üzerinden Şekil 42'de gösterildiği gibi görüntülenmiştir.



Şekil 42. Gateway modüllerinde tanımlanmış olan Modbus mapping yapılandırması ve haberleşme parametreleri

Üretilen ESI dosyası **C:\TwinCAT\3.1\Config\Io\EtherCAT** klasörüne eklenir.

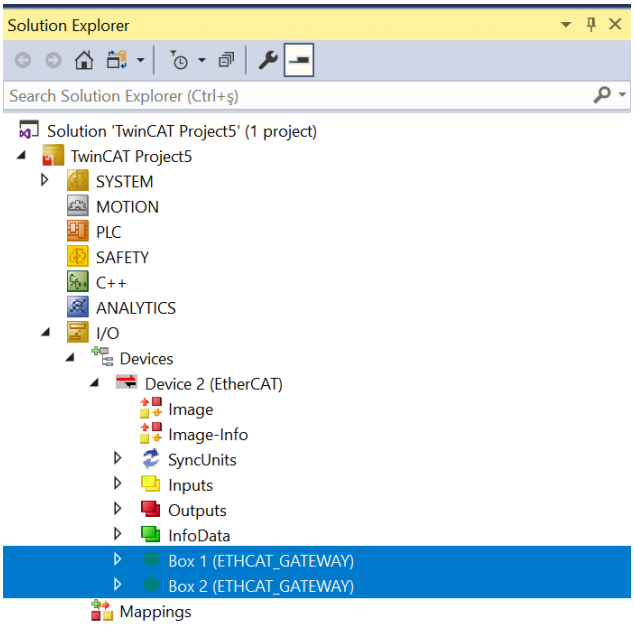
Bağlantı yapısı Şekil 43'te gösterildiği gibi EtherCAT'in daisy-chain topolojisine uygun olarak, birinci Gateway'in OUT portundan ikinci Gateway'in IN portuna Ethernet kablosu ile yapılmıştır.



Şekil 43. Aynı mapping yapılandırmasına sahip iki EtherCAT Gateway kullanılan senaryonun bağlantı diyagramı

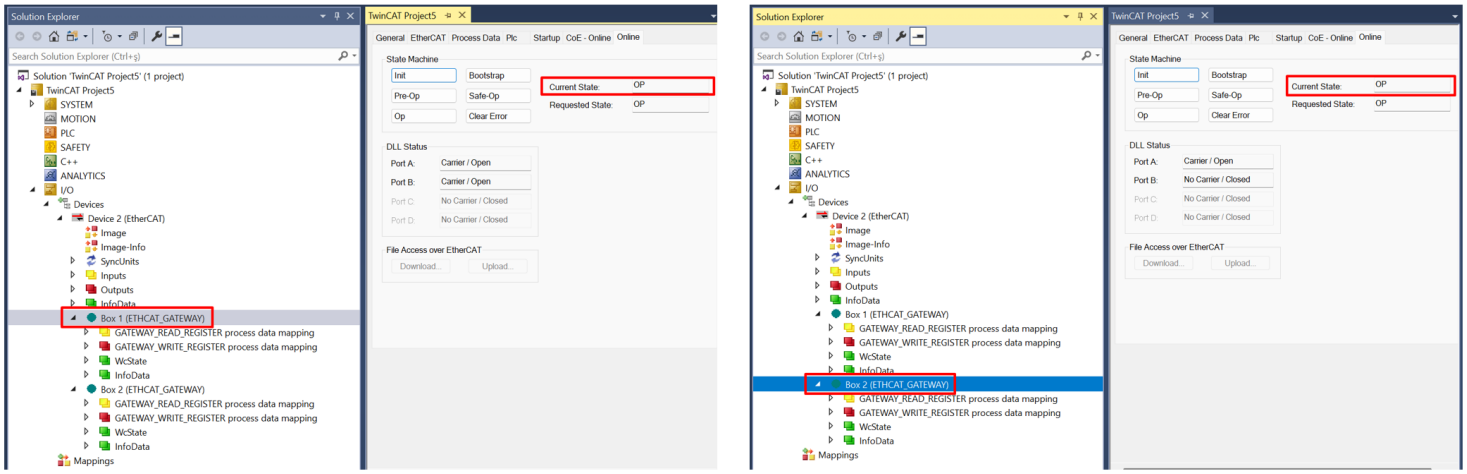
Modüllerin TwinCAT master'a tanıtımı için Uygulama Notu 1'de Adım 3'te açıklanan otomatik tarama yöntemi veya manuel yapılandırma yöntemlerinden biri kullanılabilir. Manuel yapılandırma tercih edildiğinde, her bir EtherCAT Gateway için TwinCAT projesine ayrı ayrı EtherCAT Device tanımlamaları yapılmalıdır.

Sistemdeki tüm EtherCAT cihazları fiziksel bağlantı sırasına göre tanınır ve Box numaraları bu sıraya göre atanır. Örneğin, master'a fiziksel olarak ilk bağlı olan Gateway "Box 1", ikinci sıradaki Gateway ise "Box 2" olarak tanımlanır. Tanılamadan sonra TwinCAT içerisinde modüllerin görünümü Şekil 44'te gösterilmiştir.



Şekil 44. Tanılama sonrası iki EtherCAT Gateway modülünün master arayüzünde görüntülenmesi

Tanılama sonrasında sistemin çalışır duruma geçtiği ve her iki Gateway modülün OP (Operational) moduna geçtiği Şekil 45'te görülmektedir.



Şekil 45. İki EtherCAT Gateway modülün OP mode'da çalışma durumunun görüntülenmesi

Son olarak, Uygulama Notu 1'de Adım 6'da anlatıldığı şekilde, tanımlanan modüllerin PDO üzerinden veri alışverişi ve haberleşme testleri gerçekleştirilebilir.

Explicit Device ID ve Hot Connect Kullanımı:

EtherCAT sistemlerinde cihazlar, ağ üzerindeki **fiziksel bağlantı sırasına göre** otomatik olarak adreslenir. Ancak bazı durumlarda, cihazların fiziksel bağlantı sırasından bağımsız olarak mantıksal bir sırayla (örneğin coğrafi konum, kontrol önceliği vb.) tanımlanması gerekebilir. Bu amaçla Coupler modüllere sabit bir kimlik atamak ve bu kimlik üzerinden adresleme yapmak gerekebilir. İşte bu gibi durumlar için **Explicit Device ID** ve **Hot Connect** mekanizması devreye girer.

Bu uygulama notunun ilk aşamasında, iki adet EtherCAT Gateway'in fiziksel olarak sıralı şekilde bağlandığı bir yapı üzerinden tanılama gerçekleştirilmiş ve TwinCAT tarafından bu cihazlar fiziksel bağlantı sırasına göre otomatik olarak sırasıyla Box 1 ve Box 2 olarak tanımlanmıştı. Fiziksel sıradan bağımsız tanımlamanın nasıl gerçekleştirilebileceğini ortaya koymak amacıyla, aşağıdaki senaryo uygulanmıştır.

Senaryo: Fiziksel Sıradan Bağımsız Tanımlama için Explicit Cihaz ID ve Hot Connect Kullanımı

Bu uygulama senaryosunda, fiziksel bağlantı sıralamasından **bağımsız olarak** EtherCAT Gateway'in mantıksal adreslerinin belirlenmesi amaçlanmıştır. Özellikle, fiziksel olarak **ikinci sırada bağlanan** bir Gateway'in, TwinCAT tarafında **ilk cihaz** olarak tanımlanması hedeflenmiştir. Bunun için Hot Connect ve **Explicit Cihaz ID** mekanizmaları kullanılacaktır.

Adım 1: Gateway'lere Explicit Cihaz ID Atanması

Her iki Coupler'a benzersiz bir cihaz kimliği atanır. Bu işlem, Gateway üzerindeki **DIP switch** üzerinden gerçekleştirilir[4]:

- İkinci sırada fiziksel olarak bağlı olan Gateway'in DIP switch değeri 1 olarak ayarlanır
- İlk sıradaki Gateway'in DIP switch değeri 2 olarak ayarlanır

Not 4: DIP switch değişikliklerinin geçerli olması için, Coupler'ların enerjisi kesilip yeniden verilmelidir

Adım 2: Eski Tanımlamaların Temizlenmesi

TwinCAT projesinde daha önce yapılmış fiziksel sıraya bağlı cihaz tanımlamalarının temizlenmesi gerekir. Bunun için:

- **Device 1 > Box 1** ve **Box 2** cihazları üzerine sağ tıklanarak Remove komutu ile sistemden tanımlamalar sistemden çıkarılır.

Adım 3: Yeniden Tanılama

Uygulama Notu 1 – Adım 3'te anlatıldığı gibi otomatik tarama veya manuel tanımlama yöntemiyle Coupler modüller tekrar master'a tanıtılır.

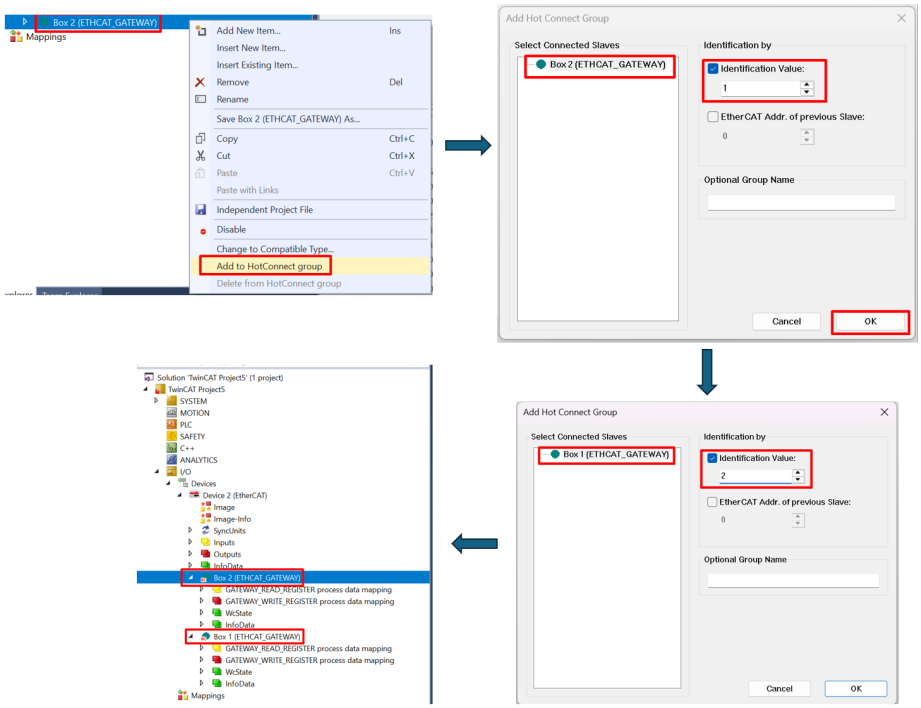
Adım 4: Hot Connect Yapılandırması

Tanılama işlemi sonrasında sistemdeki EtherCAT Gateway cihazları, **fiziksel bağlantı sırasına göre** otomatik olarak tanımlanır. Örneğin, fiziksel olarak ikinci sırada bağlanan Gateway, TwinCAT içinde yine **Box 2**, birinci sıradaki Gateway ise **Box 1** olarak yer alır.

Bu senaryoda, fiziksel olarak ikinci sırada bağlanan Gateway'in TwinCAT arayüzünde ilk sırada, fiziksel olarak ilk sırada bağlanan Gateway'in ise arayüzde ikinci sırada kontrol edilmesi sağlanacaktır.

Burada Hot Connect yapısı kurmak için, öncelikle TwinCAT arayüzünde fiziksel olarak ikinci sırada yer alan Box 2'ye sağ tıklanır ve **Add to HotConnect Group** seçeneği seçilir. Açılan pencerede, **Identification Value** alanına belirlenen Explicit Cihaz ID'nin girilmesi gerekir. Bu amaçla, ilgili alana ikinci Gateway için ayarlanmış olan 1 değeri girilir ve **OK** butonuna tıklanarak işlem tamamlanır. Aynı adımlar, fiziksel olarak birinci sırada bulunan Box 1 için de tekrarlanır; bu kez Identification Value olarak 2 girilir.

Yapılandırma işlemi tamamlandıktan sonra, sistemdeki Gateway cihazları fiziksel bağlantı sırasından bağımsız olarak, atanmış olan Explicit Cihaz ID değerlerine göre ve Hot Connect bağlantı simgeleri ile yeniden sıralanır. Son olarak, Restart Config Mode komutu uygulanarak haberleşme başlatılır. Ardından, Uygulama Notu 1 Adım 6'da belirtildiği gibi PDO alanlarına erişim sağlanarak sistemin çevrimsel veri alışverişi başlatılabilir. İşlem Adımları Şekil 46'da gösterilmektedir.



Şekil 46. Hot Connect Bağlantı İşlem Adımları

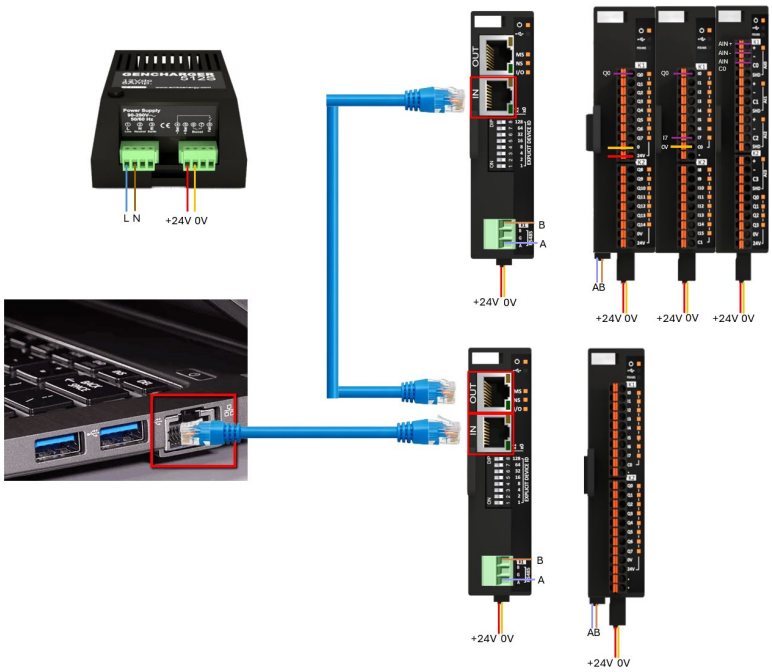
Uygulama Senaryosu: Farklı Mapping Yapısına Sahip EtherCAT Gateway ile Dağıtık I/O Kontrolü:

Bu senaryoda, bir önceki uygulama örneğinden farklı olarak farklı Modbus mapping yapılarına sahip olan ve buna bağlı olarak farklı ESI dosyaları gerektiren iki EtherCAT Gateway modülünün, daisy-chain topolojisi üzerinden tek bir EtherCAT Master tarafından kontrol edilmesi ele alınmaktadır.

Senaryoda iki gateway'e RS485 üzerinden farklı genişleme modülleri bağlanmış ve farklı modbus mapping yapılandırması uygulanmıştır. Kullanılan donanımlar:

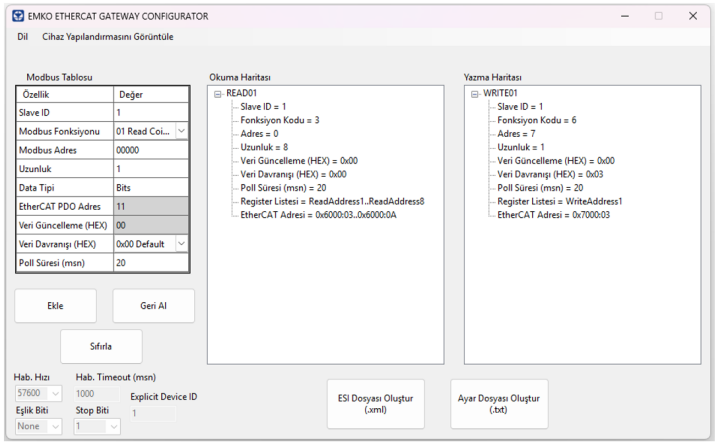
- Birinci Coupler (SGEAP1-00-00-00-00_1):,
    - Daha önceki uygulama senaryosunda yer alan genişleme modülü bu senaryoda da kullanılmıştır.
- İkinci Coupler (SGEAP1-00-00-00-00_1):
  Aşağıdaki iki genişleme modülünü içermektedir:
    - SMR1P1-00-00-01-00_1 (15xTO)
    - SMR1P1-01-00-00-00_1 (16xDI)
    - SMR1P1-00-01-04-00_1 (4xAI-SU, 4xTO)

Bağlantı yapısı, Şekil 47'de gösterildiği üzere, EtherCAT'in daisy-chain topolojisine uygun olacak şekilde gerçekleştirilmiştir. Bu yapıda, birinci Gateway'in OUT portu ile ikinci Gateway'in IN portu Ethernet kablosu üzerinden birbirine bağlanmıştır.

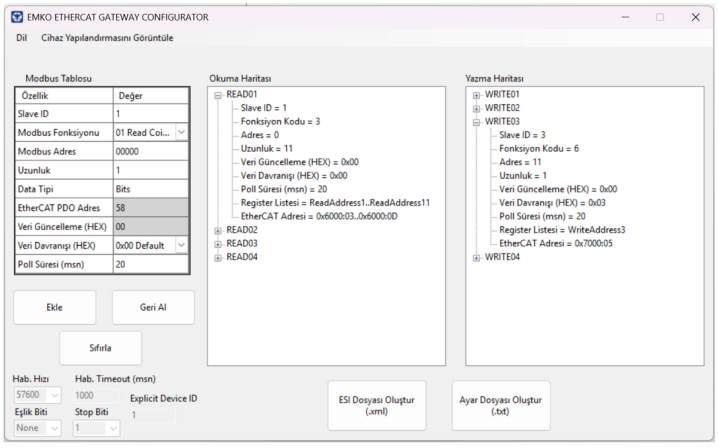


Şekil 47. Farklı mapping yapılandırmasına sahip iki EtherCAT Gateway kullanılan senaryonun bağlantı diyagramı

Gateway modüllerinde tanımlanmış olan Modbus mapping yapılandırması ve haberleşme parametreleri, EMKO ETHERCAT GATEWAY CONFIGURATOR yazılımında Cihaz Yapılandırmasını Görüntüle menüsü üzerinden Şekil 48'd8 gösterildiği gibi görüntülenmiştir.



1. sıradaki EtherCAT Gateway'in Modbus Mapping yapılandırması



2. sıradaki EtherCAT Gateway'in Modbus Mapping yapılandırması

Şekil 48. Gateway modüllerinde tanımlanmış olan Modbus mapping yapılandırması ve haberleşme parametreleri

Üretilen ESI dosyası **C:\TwinCAT\3.1\Config\Io\EtherCAT** klasörüne eklenir. Bu örnekte, ilk Gateway için oluşturulan ESI dosyası ETHCAT_GATEWAY1.xml, ikinci Gateway için oluşturulan dosya ise ETHCAT_GATEWAY2.xml adıyla yeniden adlandırılmıştır.

TwinCAT ve diğer EtherCAT master yazılımları, bağlı cihazları tanımlarken Vendor ID, Product Code ve Revision Number alanlarını esas alır. EMKO EtherCAT Gateway Configurator yazılımı tarafından oluşturulan ESI dosyalarında Vendor ID ve Product Code Revision no sabittir. Bu durumda, her iki Gateway cihazı için üretilen ESI dosyalarında bu alanlar aynı olup, master her iki cihazı da aynı modül olarak algılayacak ve yanlış PDO yapılandırmasıyla haberleşmeye çalışarak hata oluşmasına neden olacaktır.

Çözüm: Revision Number Düzenlemesi

Vendor ID ve Product Code alanları cihazın üretici kimliği ve ürün tanımı açısından sabittir. **Bu değerlerin değiştirilmesi cihaz güvenliği ve uyumluluğu açısından uygun değildir.**

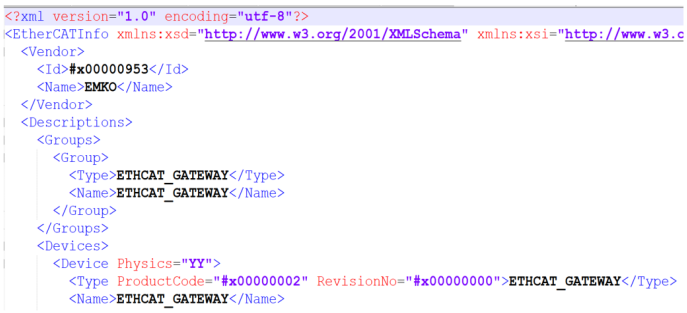
Bu nedenle, aynı Vendor ID ve Product Code'a sahip birden fazla Gateway aynı sistemde kullanılacaksa, farklı cihazlar olarak tanınabilmeleri için Revision Number değerleri ESI dosyalarında manuel olarak düzenlenmelidir.

Bunun için:

1. İlgili ESI dosyaları (ör. ETHCAT_GATEWAY1.xml, ETHCAT_GATEWAY2.xml) bir XML düzenleyici ile açılır.
2. <RevisionNo> alanı her dosya için farklı değerlerle güncellenir (örn. birincisi 0x00000001, ikincisi 0x00000002).
3. Dosyalar kaydedilir.

Not: Değişikliklerin geçerli olabilmesi için TwinCAT yazılımının yeniden başlatılması gerekmektedir.

Şekil 49'da XML dosyasında <RevisionNo> alanının güncelleme işlemi gösterilmiştir .



Varsayılan XML dosyası



1. Gateway XML dosyası



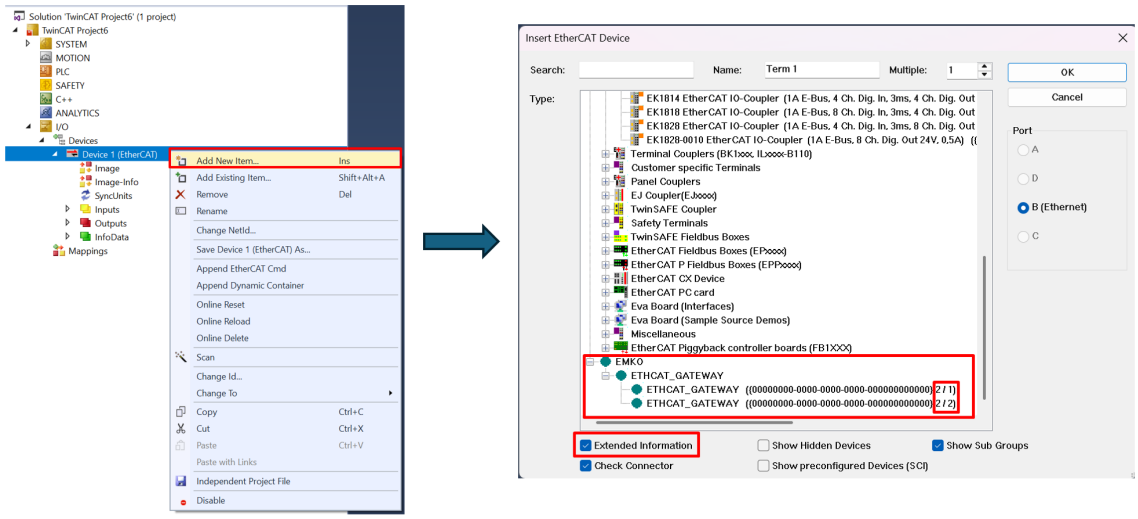
2. Gateway XML dosyası

Şekil 49. Varsayılan ve her iki EtherCAT Gateway için değiştirilen XML dosyaları

İşlemlerin gerçekleştirilmesinin ardından TwinCAT yazılımı açılarak Uygulama notu 1 Adım 1 uygulanarak proje oluşturulur. Daha sonra Adım 3 'te anlatılan manuel tanımlama yöntemi uygulanır. Burada öncelikle EtherCAT Master eklenir.

Master'ın eklenmesinin ardından EtherCAT Gateway modüllerinin eklenmesi gerekmektedir. Bu işlem için ilk olarak Devices'ın altında listenen Master'a (Bu örnekte Device 1 (EtherCAT)) sağ tıklanarak Add New Item sekmesi seçilir. Açılan Insert EtherCAT Device penceresinde **Extended Information** kutucuğu işaretlenerek cihazların ek özellikleri görüntülenir. Burada cihazlara ait product code/revision number bilgileri görüntülenir.

EMKO başlığının altında yer alan ETHCAT_GATEWAY başlığının içerisindeki iki adet ETHCAT_GATEWAY cihazı görüntülenir. Burada ilk sıradaki modülün revision numarası 1 olduğundan genişletilmiş bilgilerde ilgili alanın 1 olduğu dosya seçilir ve OK tuşuna basılarak ekleme işlemi tamamlanır. İşlem adımları Şekil 50'de gösterilmektedir.

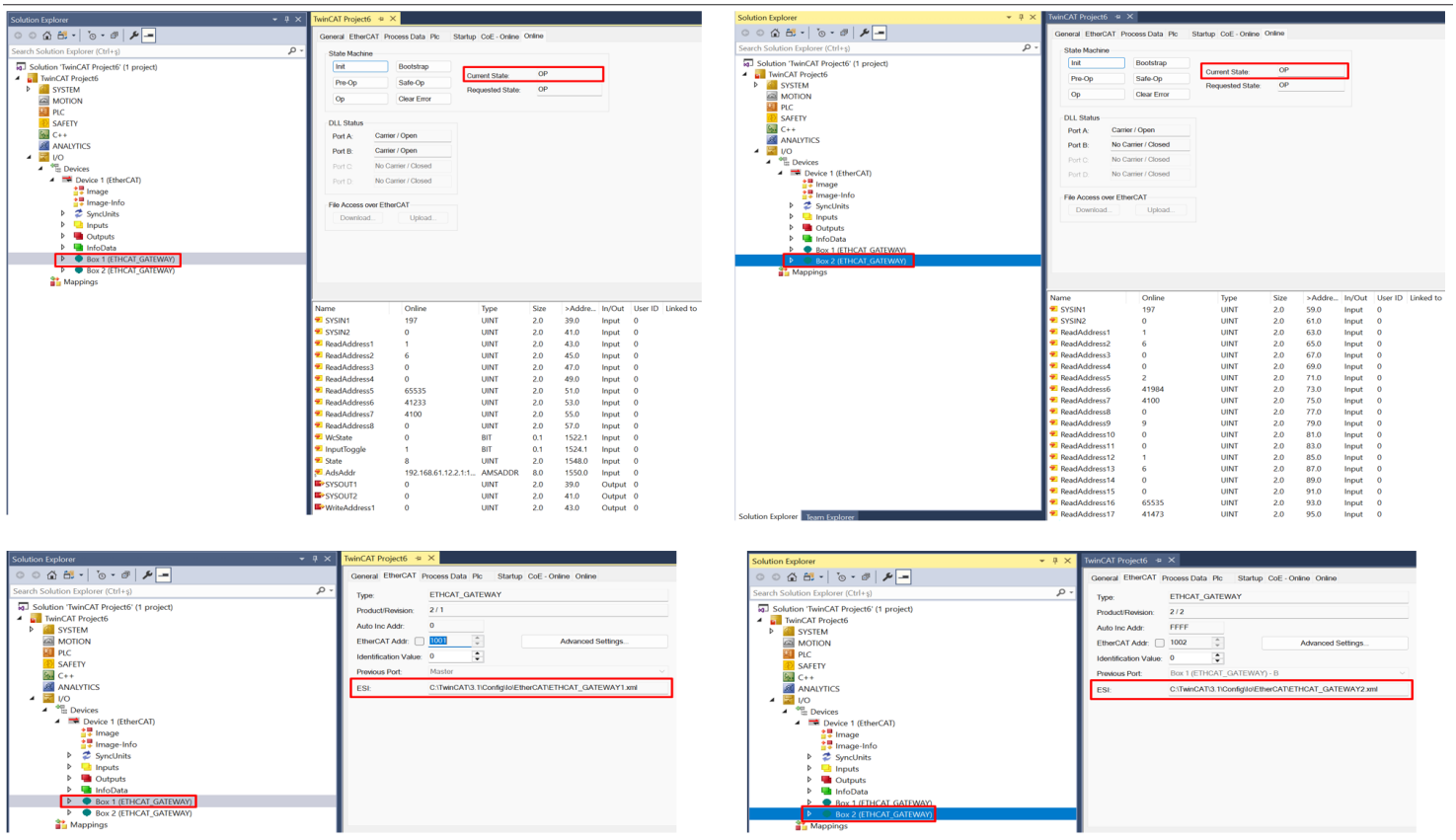


Şekil 50. 1. EtherCAT Gateway'in cihaz olarak eklenmesi işlem adımları

Burada 2/1'deki 2 product code'u 1 revision number'i temsil etmektedir. Aynı işlem 2. Gateway modülü için de uygulanır.

İşlemin tamamlanmasının ardından EtherCAT Gateway modülleri, projeye eklenmiş olur. Burada manuel ekleme işlemi yapıldığından otomatik olarak çalışmaya başlamaz. Başlatma işlemi için Uygulama Notu 1 Adım 3'te manuel tanımlama başlığı altında anlatıldığı TwinCAT menüsü altında veya Standard durum çubuğunda yer alan ya da yer Restart TwinCAT (Config Mode) sekmesine tıklanarak çalışma başlatılır.

Şekil 51'de, EtherCAT Gateway modüllerinin çalışma durumuna (OP moduna) geçtikleri, farklı PDO yapılarına sahip oldukları ve her bir modülün kendi ESI dosyası ile ilişkilendirildiği gösterilmektedir.



a) b)

Şekil 51. İki farklı EtherCAT Gateway modülünün TwinCAT yazılımında tanınması, çalışma durumları (OP) ve kullanılan ESI dosyalarının gösterimi (a) 1. Gateway, (b) 2. Gateway

Uygulama Notu 3 : CODESYS Üzerinden EtherCAT Gateway ile IPERTU Genişleme Modüllerinin Kontrolü

Bu uygulama notunda, CODESYS tabanlı EtherCAT master kullanılarak EtherCAT Gateway ve ona RS485 üzerinden bağlı genişleme modüllerinin yapılandırılması adım adım açıklanacaktır. Fiziksel donanım olarak, daha önce Uygulama Notu 1'de kullanılan Şekil 23'te gösterilen kurulum kullanılmıştır. ESI dosyası ve mapping yapılandırma için ise ve Şekil 26 ve 27'de üretilen dosyalar kullanılmıştır.

Projelerin geliştirilmesi, cihaz konfigürasyonu ve PLC programlaması için **CODESYS Geliştirme Ortamı (IDE)** kullanılır. Bu uygulama notunda **CODESYS Development System 3.5** kullanılmıştır. Yazılım aşağıdaki bağlantı üzerinden indirilebilir:

https://store.codesys.com/en/codesys.html

Geliştirilen programların PC üzerinde çalıştırılabilmesi için **CODESYS Control Win SL** adlı yazılım runtime motoru kurulmalıdır. Bu uygulamada, fiziksel bir PLC yerine bu Soft PLC kullanılmakta olup, CODESYS Control Win V3 - x64 kullanılmıştır. Yazılım aşağıdaki bağlantıdan indirilebilir[5]:

https://store.codesys.com/en/codesys-control-win-sl-1.html

Not 5: Yukarıdaki bağlantılardan dosyaları indirebilmek için CODESYS Store üzerinden hesap oluşturulması ve oturum açılması gereklidir.

CODESYS yazılımları, **bireysel kullanıcılar (private customer)** ya da **kurumsal müşteriler (corporate)** olarak indirilebilir.

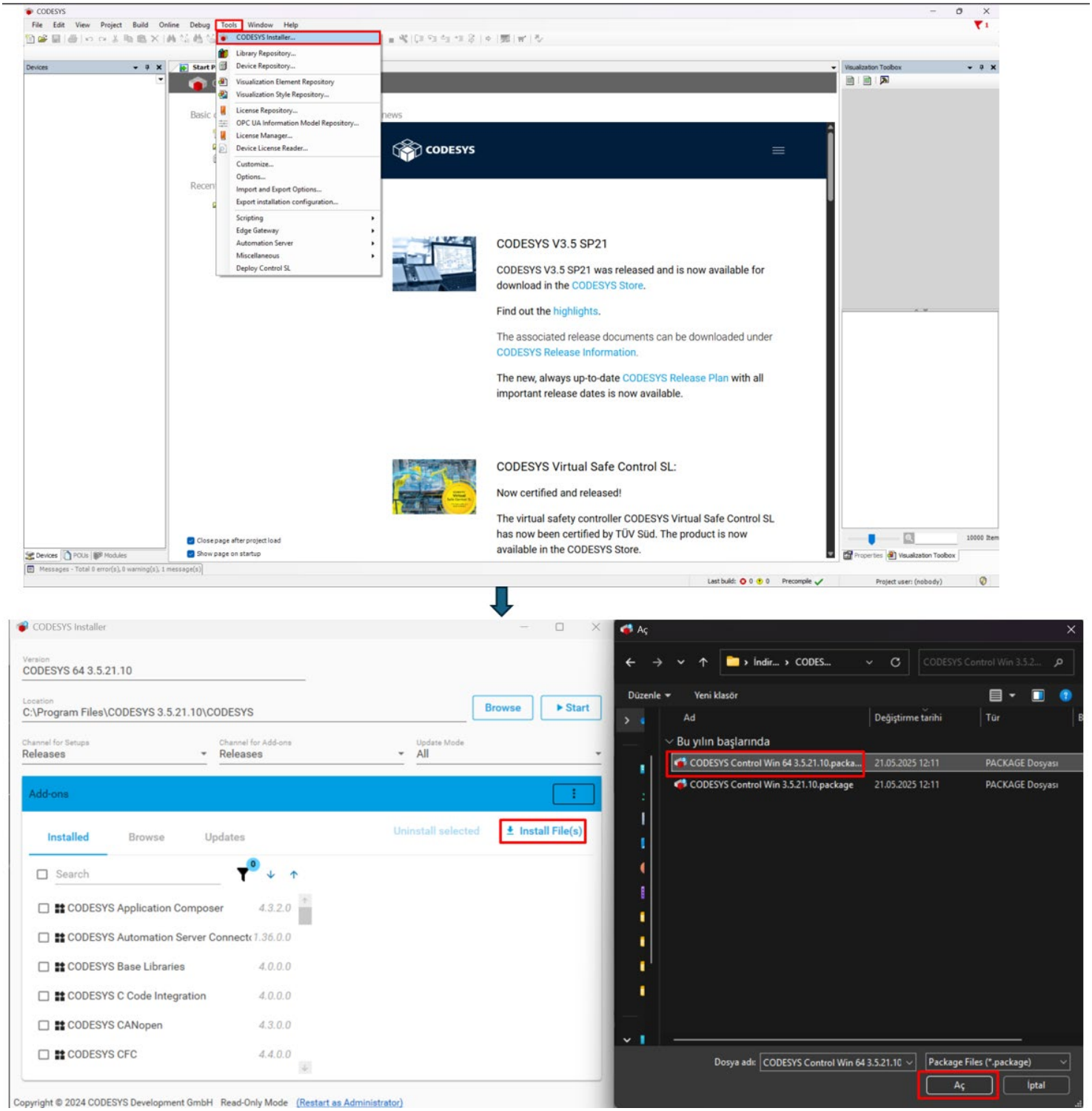
- Eğer sistem lisanslı bir firma altyapısı üzerinden kullanılacaksa, CODESYS Corporate hesabı tercih edilebilir.
- Bireysel kullanıcılar ise Private Customer seçeneğiyle ilerleyebilir.

İlk olarak CODESYS 64 3.5.21.10.exe ile IDE kurulumu gerçekleştirilir. Ardından **CODESYS Control Win 64 3.5.21.10.exe** çalıştırılarak Soft PLC kurulumu gerçekleştirilir. Ancak bu işlem, CODESYS Development System içerisine ilgili runtime bileşenlerini otomatik olarak eklemez. Bu nedenle ikinci adımda, CODESYS Installer aracı kullanılarak runtime paketinin IDE'ye entegre edilmesi gereklidir.

Bu amaçla:

1. CODESYS V3.5 SP21 Patch 1 IDE açılır.
2. Üst menüden **Tools > CODESYS Installer** sekmesine tıklanır.
3. Açılan Installer sayfasında, **Install File** butonuna tıklanır.
4. Daha önce indirilmiş olan CODESYS Control Win SL sıkıştırılmış klasörü içerisindeki CODESYS Control Win 64 3.5.21.10.package dosyası bulunarak seçilir.
5. Kurulum adımları tamamlanarak runtime motoru IDE'ye entegre edilir.

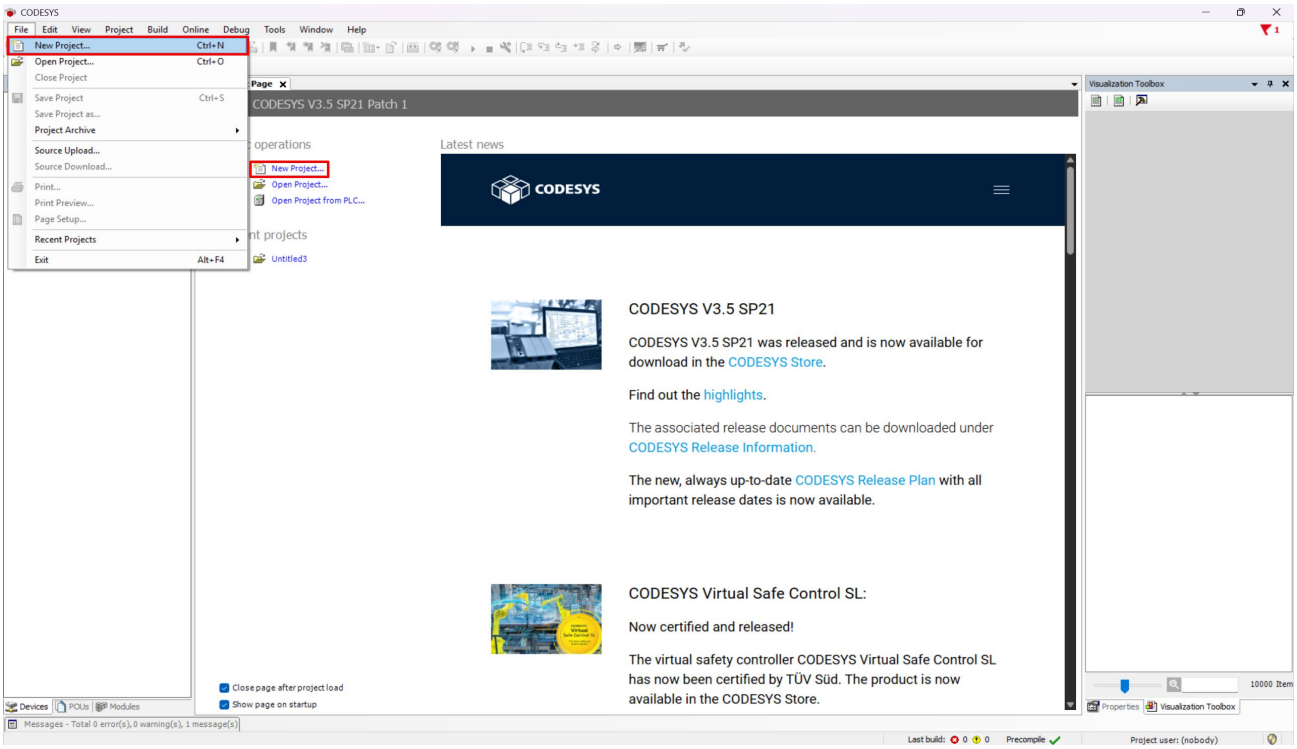
İşlem adımları Şekil 52'de gösterilmektedir. Bu işlem tamamlandıktan sonra CODESYS sistemi, geliştirilmiş PLC projelerini doğrudan PC üzerinde çalıştırabilecek hale gelir.



Şekil 52. CODESYS Control Win 64 paketinin yüklenmesi

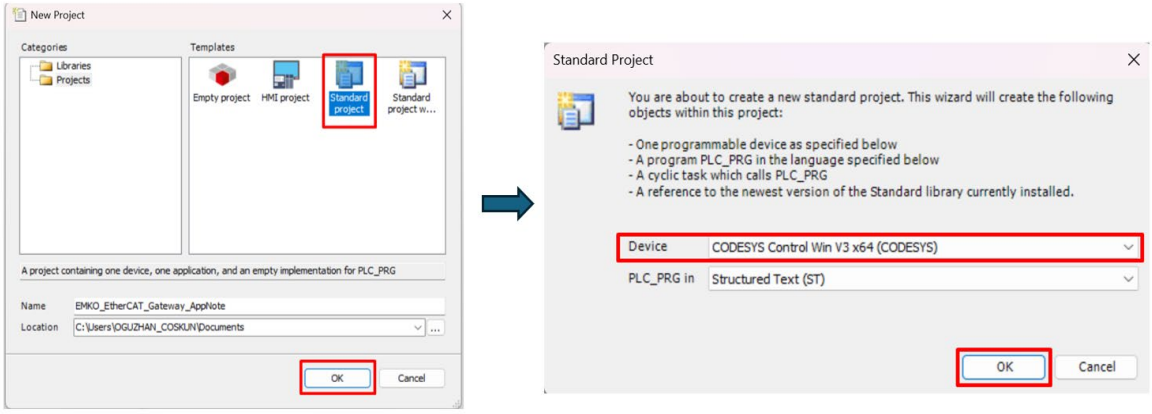
Adım 1 : Proje Oluşturma

CODESYS V3.5 SP21 Patch 1 IDE'de **File** menüsünden **New Project** yolunu izleyerek veya arayüzde kısayol olarak yer alan **New Project** sekmesinden proje oluşturulmalıdır. İlgili arayüz ve sekmeler Şekil 53'te gösterilmektedir.



Şekil 53. CODESYS V3.5 arayüzü ve proje oluşturma sekmeleri

Burada Standart Project seçilip isim ve dosya yolu ayarlanarak işleme devam edilir. Daha sonra karşımıza çıkan proje ayar sayfasında Device olarak yüklenilmiş olan CODESYS Control Win V3 x64 seçilmelidir. Dil olarak ise bu uygulamada Structered Test (ST) kullanılmıştır. İşlem adımları Şekil 54'te gösterilmiştir.



Şekil 54. Proje oluşturma ayarları

Adım 2 : Soft PLC ve Gateway Hizmet ve Yazılımlarının Başlatılması

EtherCAT sisteminin PC üzerinde çalıştırılabilmesi için, kurulan yazılım bileşenlerinin doğru şekilde başlatılması ve birbirleriyle haberleşebilecek durumda olması gerekir. Bu noktada üç temel bileşen aktif olmalıdır:

1. CODESYS Gateway V3

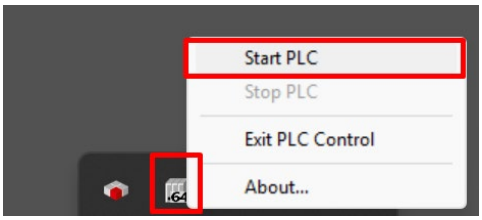
- **Rolü:** CODESYS IDE ile Soft PLC (Control Win) arasında bağlantı kurar.
- **Kurulumu:** IDE yüklenirken otomatik olarak kurulmaktadır.
- **Başlatılması:** Bir Windows uygulaması olarak çalışır. Eğer aktif değilse uygulamayı bularak çalıştırınız.

2. CODESYS Control Win V3 – x64 (Soft PLC)

- **Rolü:** Soft PLC
- **Başlatılması:**. CODESYS Control Win V3 - x64 uygulaması çalıştırılarak başlatılır.
- **Durum Kontrolü:** Başarılı şekilde başlatıldığında, konsol ekranında "CODESYS Control ready" ifadesi görünür ve çalıştığına dair aşağıdaki gibi bir çıktı alınır:
  Cmp=CM, Class=1, Error=0, Info=34, pszInfo= CODESYS Control ready

3. CODESYS Control Win SysTray Aracı

- **Rolü:** Soft PLC'nin çalıştırılması (Start), durdurulması (Stop) gibi işlemleri kontrol eder.
- **Kullanımı:** Sistem tepsisinde (sağ alt köşede saat yanındaki alan) yer alır. Eğer görünmüyorsa, SysTray uygulaması manuel olarak çalıştırılmalıdır (örneğin kurulum klasöründen).
- PLC **Start** edilmeden, Gateway ile bağlantı kurmak mümkün değildir çünkü Soft PLC dinleme moduna geçmemiş olur. Bu nedenle Şekil 55'te gösterildiği gibi sistem tepsisinde yer alan **CODESYS Control Win SysTray** aracından Start PLC işleminin gerçekleştirilmesi gerekmektedir.



Şekil 55. CODESYS Control Win SysTray'den Soft PLC'nin başlatılması

Adım 3 : CODESYS Device Bağlantı Ayarlarının Yapılması

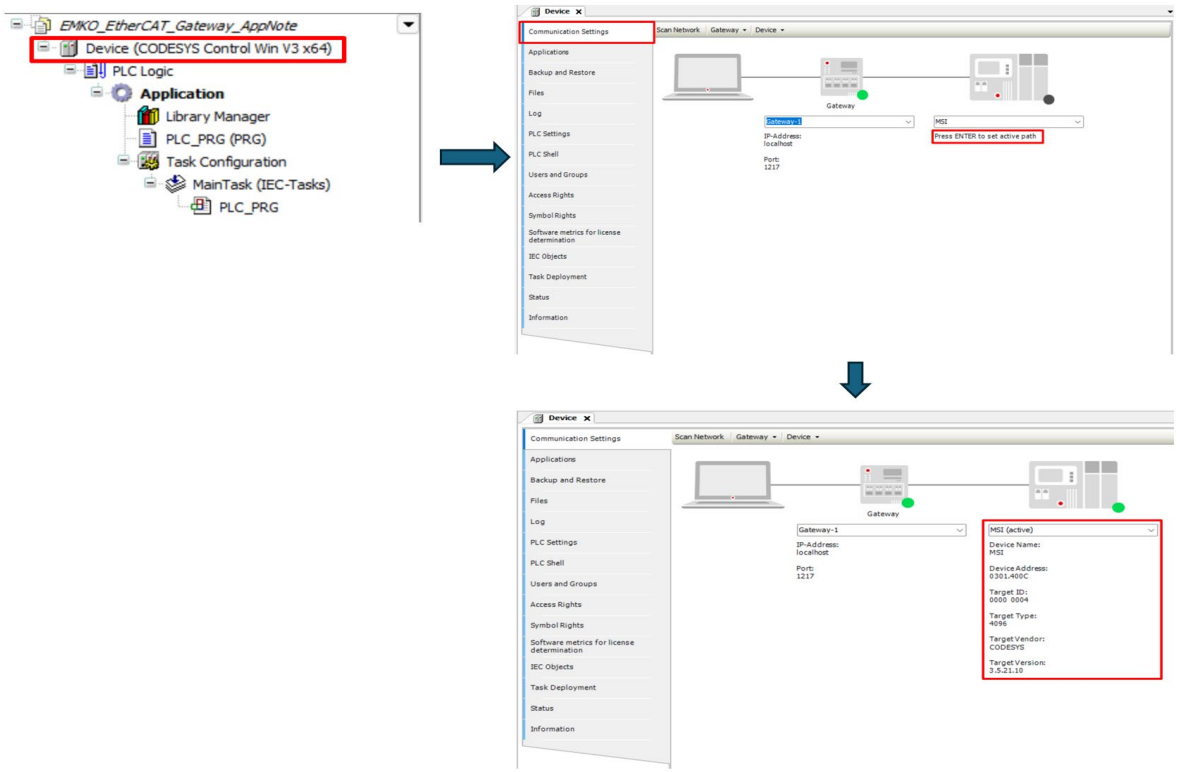
CODESYS IDE üzerinde oluşturulan projede, cihazın (Soft PLC) haberleşme ayarlarının yapılabilmesi için sol taraftaki Devices sekmesinden tanımlı olan Device (CODESYS Control Win V3 x64) üzerine çift tıklanır. Açılan pencerede Communication Settings bölümü aktif hale gelir.

Bu arayüzde, bilgisayar ile Soft PLC (CODESYS Control Win) arasında bağlantıyı sağlayan Gateway yapılandırması yer alır. Gateway, CODESYS IDE ile runtime (çalışma zamanı) ortamı arasında haberleşmeyi sağlayan ara katmandır. IDE kurulumu sırasında CODESYS Gateway V3 servisi otomatik olarak yüklenir, ancak çalışır durumda olması gerekir. Servis çalışmıyorsa Windows hizmetlerinden manuel olarak başlatılmalıdır.

Gateway seçimi çoğunlukla otomatik olarak yapılır ve bilgisayar adıyla birlikte görünür (örneğin: MSI, localhost). Ağ üzerindeki port varsayılan olarak 1217'dir. Bu ekranda:

- Sol kutucukta Gateway seçilir (örneğin: Gateway-1)
- Sağ kutucukta bilgisayar adı seçilir (örneğin: MSI)
- Seçimden sonra bağlantı yolunun aktif hale gelmesi için Enter tuşuna basılır.
- Eğer bağlantı başarılı ise, Gateway ile cihaz arasında yeşil bağlantı simgesi belirir.

İşlem adımları Şekil 56'da gösterilmiştir.



Şekil 56. IDE ile Soft PLC'nin gateway ara katmanı ile bağlantısının sağlanması

Adım 4 : EtherCAT Master'ın eklenmesi

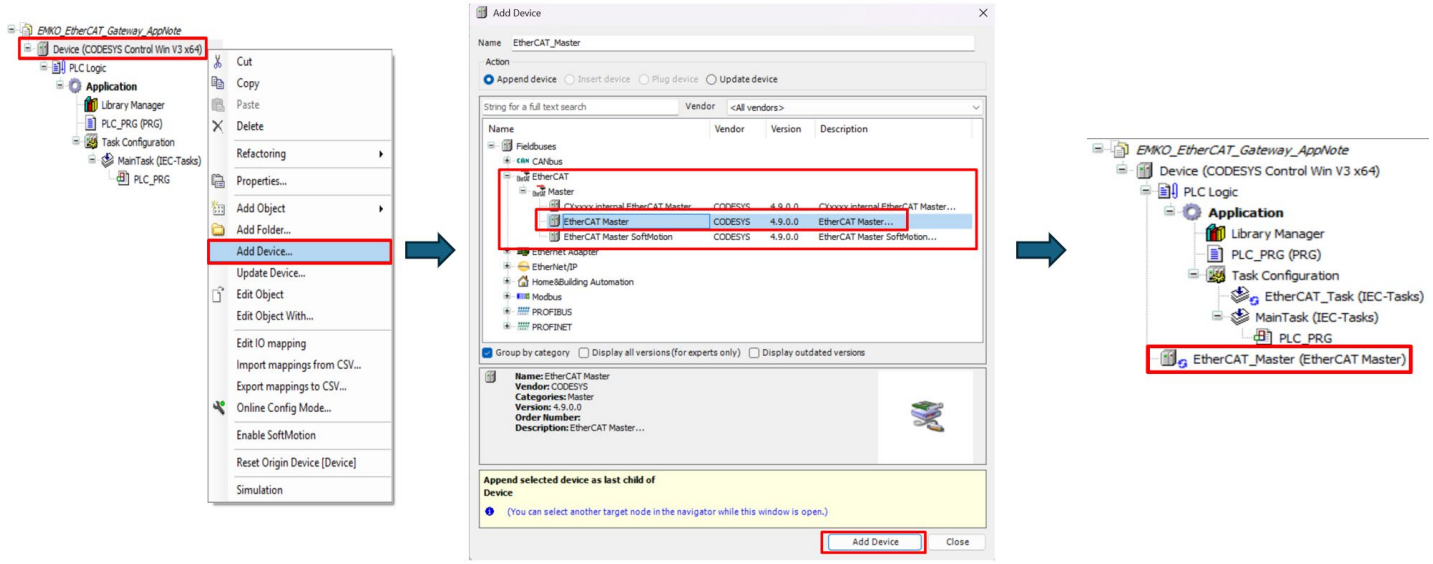
EtherCAT tabanlı bir sistemde, ağın yönetilebilmesi ve veri alışverişinin sağlıklı bir şekilde gerçekleştirilebilmesi için bir **EtherCAT Master** bileşeni gereklidir. Bu master cihaz, ağa bağlı tüm EtherCAT slave modülleriyle (örneğin Bus Coupler) haberleşmeyi yöneten kontrol birimidir.

Bu uygulamada kullanılacak olan **Bus Coupler** bir **EtherCAT slave** cihazıdır. Bu nedenle, slave cihazla haberleşebilecek bir EtherCAT Master modülünün projeye eklenmesi zorunludur. CODESYS üzerinde bu master görevini, projeye eklenecek bir **EtherCAT Master aygıtı (device)** üstlenir.

EtherCAT Master'ın projeye eklenmesi için aşağıdaki adımlar izlenir:

1. **Device (CODESYS Control Win V3 x64)** satırına **sağ tıklanır**.
2. Açılan menüden **"Add Device"** seçeneği seçilir.
3. Açılan cihaz seçim penceresinde, EtherCAT başlığı altındaki Master bölümünden **"EtherCAT Master (EtherCAT)"** cihazı seçilir
4. **"Add Device"** butonuna tıklanarak projeye eklenir.

Bu işlem tamamlandığında, proje ağacında kontrol cihazı altında EtherCAT Master bileşeni görüntülenir ve bu bileşen üzerinden slave cihazların (örneğin Bus Coupler) taranması ve yapılandırılması mümkün hale gelir. İşlem adımları Şekil 57'de gösterilmiştir.



Şekil 57. EtherCAT Master cihazının eklenmesi

Adım 5 : EtherCAT Master Network Adaptör Ayarı

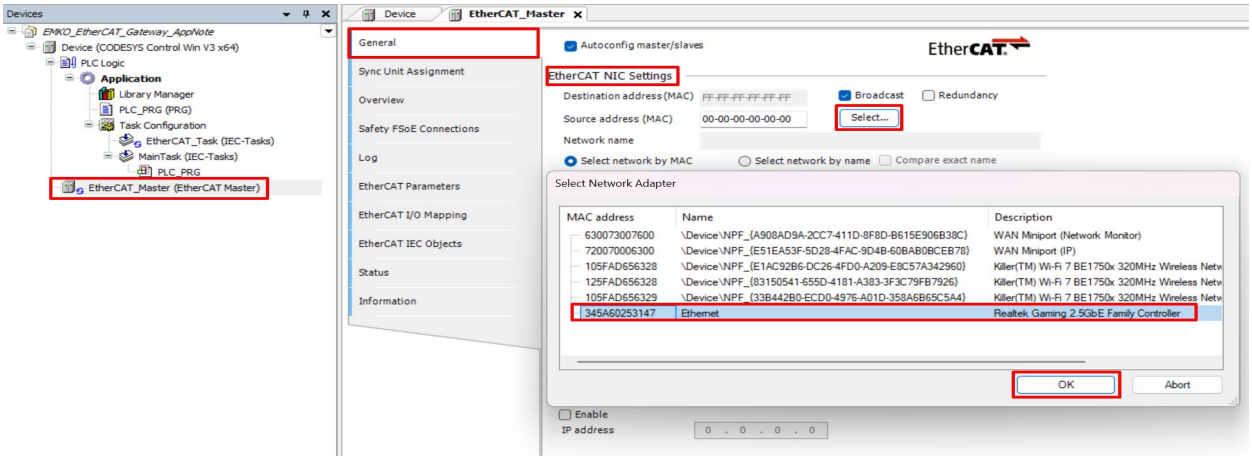
EtherCAT haberleşmesinin gerçekleştirilebilmesi için, EtherCAT Master cihazının hangi fiziksel ağ arabirimi (Ethernet adaptörü) üzerinden haberleşeceğinin belirlenmesi gerekir. Bu ayar, CODESYS üzerinden eklenen EtherCAT Master bileşenine özel olarak yapılır.

İlgili işlem için aşağıdaki adımlar izlenir:

1. Proje ağacında yer alan **EtherCAT_Master (EtherCAT Master)** cihazına çift tıklanır.
2. Açılan arayüzde sol menüden **"General"** sekmesi seçilir.
3. Ardından, **"EtherCAT NIC Settings"** başlığı altında bulunan **"Select..."** butonuna tıklanır.
4. Karşımıza çıkan adaptör seçim ekranında, bilgisayarda tanımlı tüm ağ arabirimleri listelenir[6].
5. Buradan EtherCAT ağına fiziksel olarak bağlı olan adaptör (örneğin, "Realtek Gaming 2.5GbE Family Controller") seçilir ve **OK** butonuna tıklanarak işlem tamamlanır.

Bu işlem adımları Şekil 58'de örnek bir adaptör seçimi ile görsel olarak sunulmuştur.

*Not 6: Soft PLC'nin bilgisayarınızda yüklü olan ağ arabirimlerini (Ethernet adaptörlerini) doğru şekilde tanıyabilmesi için **Npcap** yazılımının sistemde kurulu olması gerekmektedir.*
İndirme bağlantısı: https://npcap.com/#download



Şekil 58. EtherCAT Master Network Adaptör ayarı

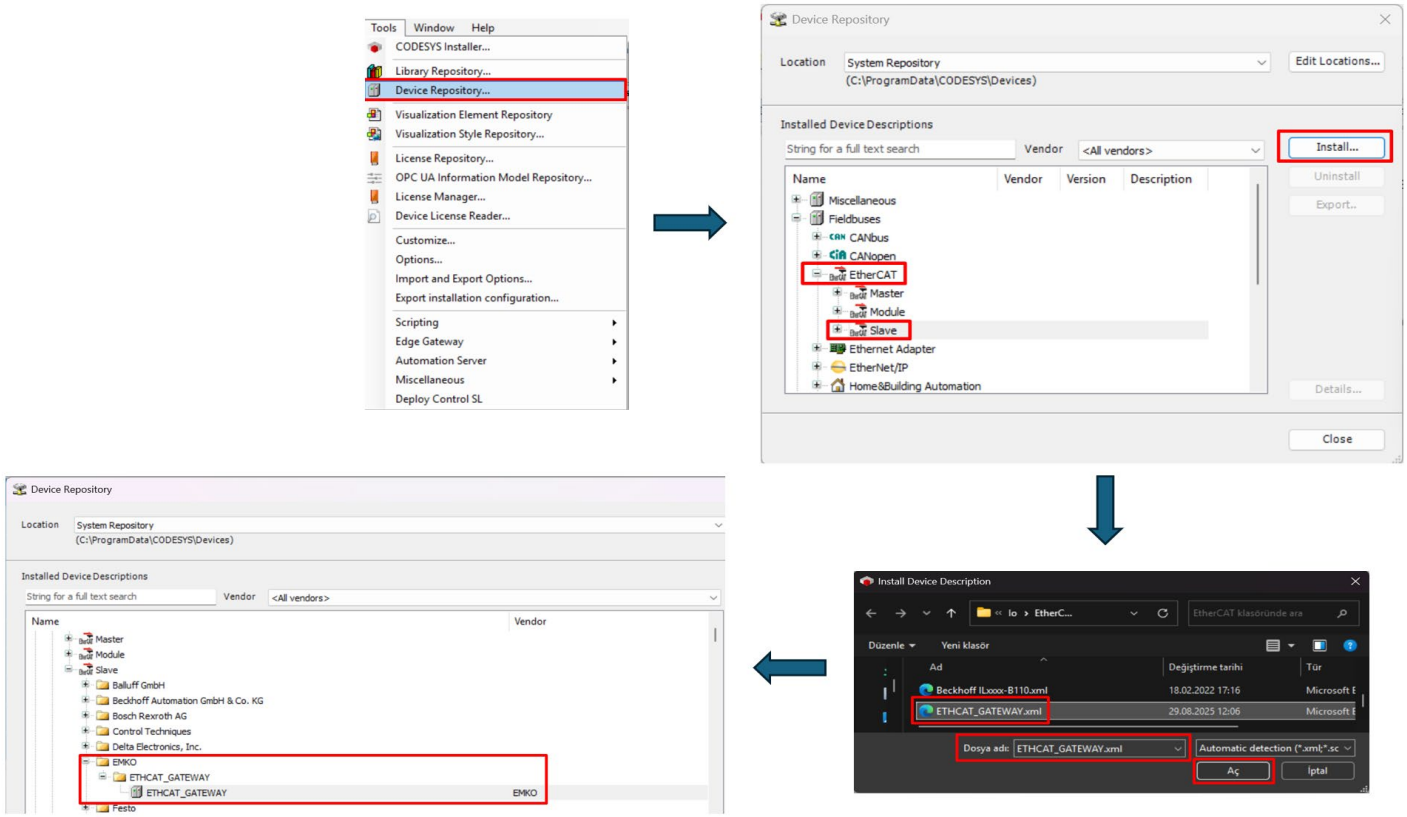
Adım 6 : IPERTU EtherCAT Gateway Modülünün Device Kütüphanesine Eklenmesi

EtherCAT haberleşmesinde her bir slave cihazın tanımlanabilmesi için, ilgili cihaza ait ESI (EtherCAT Slave Information) dosyasının geliştirme ortamına tanıtılması gerekir. Bu işlem sayesinde cihazın PDO/SDO yapısı, tanımlayıcı bilgileri ve konfigürasyon seçenekleri IDE tarafından tanınabilir hâle gelir.

Bu adımda, EMKO tarafından geliştirilen IPERTU EtherCAT Gateway modülünün CODESYS içerisine tanıtılması sağlanacaktır.
İşlem Adımları:

1. CODESYS üst menüsünden **Tools** sekmesi açılır.
2. Açılan menüden **Device Repository...** seçeneğine tıklanır.
3. Karşımıza gelen pencerede **EtherCAT** başlığı altında **Slave** sekmesi seçilir.
4. Sağda yer alan **Install...** butonuna tıklanır.
5. Açılan dosya seçim penceresinden, IPERTU EtherCAT Bus Coupler cihazına ait ESI dosyası olan **ETHCAT_GATEWAY.xml** seçilir ve **Aç** butonuna tıklanarak yükleme işlemi tamamlanır.

İşlem tamamlandığında, Slave cihaz listesi altında "EMKO" adlı bir üretici klasörü oluşur ve bu klasörün içinde IPERTU EtherCAT Bus Coupler modülü yer alır. işlem adımları Şekil 59'da gösterilmiştir.



Şekil 59. IPERTU EtherCAT Gateway ESI dosyası yükleme adımları

Adım 7 : EtherCAT Gateway Modülünün Projeye Eklenmesi ve Çalıştırılması

Ekleme işlemine başlamadan önce, Şekil 23'te gösterildiği gibi EtherCAT Bus Coupler modülünün 24V DC besleme bağlantısının sağlanmış olması ve Ethernet kablosunun kişisel bilgisayarın Ethernet portu ile EtherCAT Gateway üzerindeki "IN" olarak etiketlenmiş Ethernet soketi arasında bağlantı kuracak şekilde takılmış olması gerekmektedir.

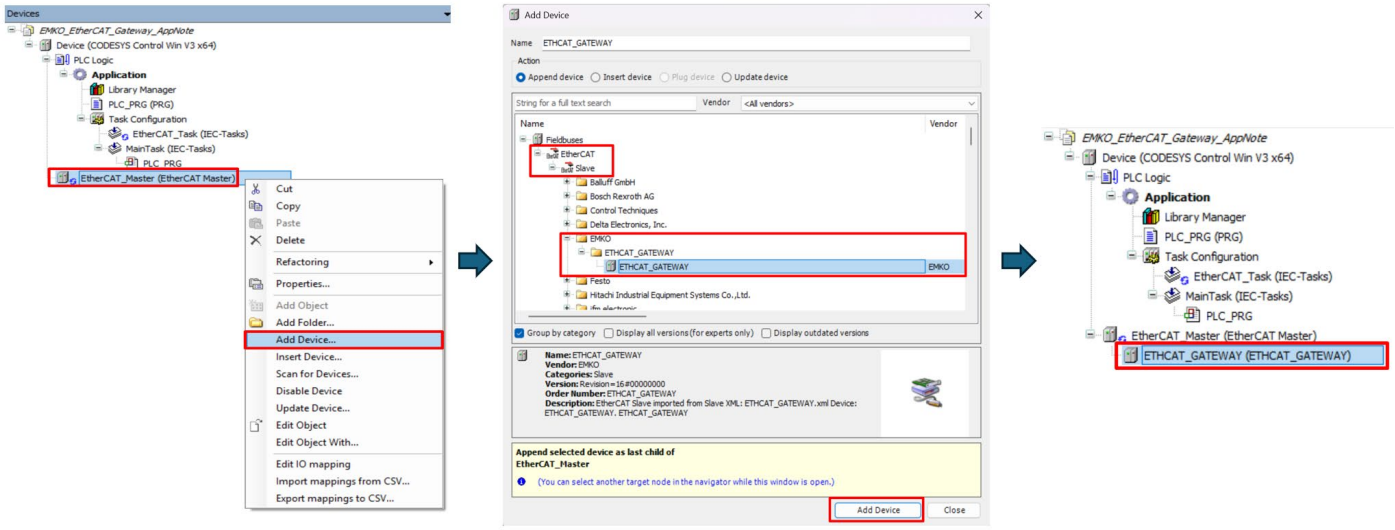
EtherCAT Gateway modülünün tanıtılarak projeye dahil edilmesi, manuel ve otomatik olarak iki şekilde gerçekleştirilmektedir.

Manuel Tanımlama

EtherCAT Gateway modülü, projeye manuel olarak aşağıdaki adımlar takip edilerek eklenebilir:

1. Daha önce projeye eklenmiş olan **EtherCAT Master** cihazı üzerinde **sağ tıklanır**.
2. Açılan menüden **Add Device...** seçeneği seçilir.
3. Açılan pencerede **Slave cihazlar** listesinden, daha önce Device Repository'ye yüklenmiş olan **EMKO > ETHCAT_GATEWAY > ETHCAT_GATEWAY** cihazı seçilir.
4. Sağ alttaki **Add Device** butonuna tıklanarak modül projeye eklenir.

Bu işlem sonucunda, EtherCAT Master'ın altında **ETHCAT_GATEWAY** eklenmiş olur. İşlem adımları Şekil 60'da gösterilmiştir.



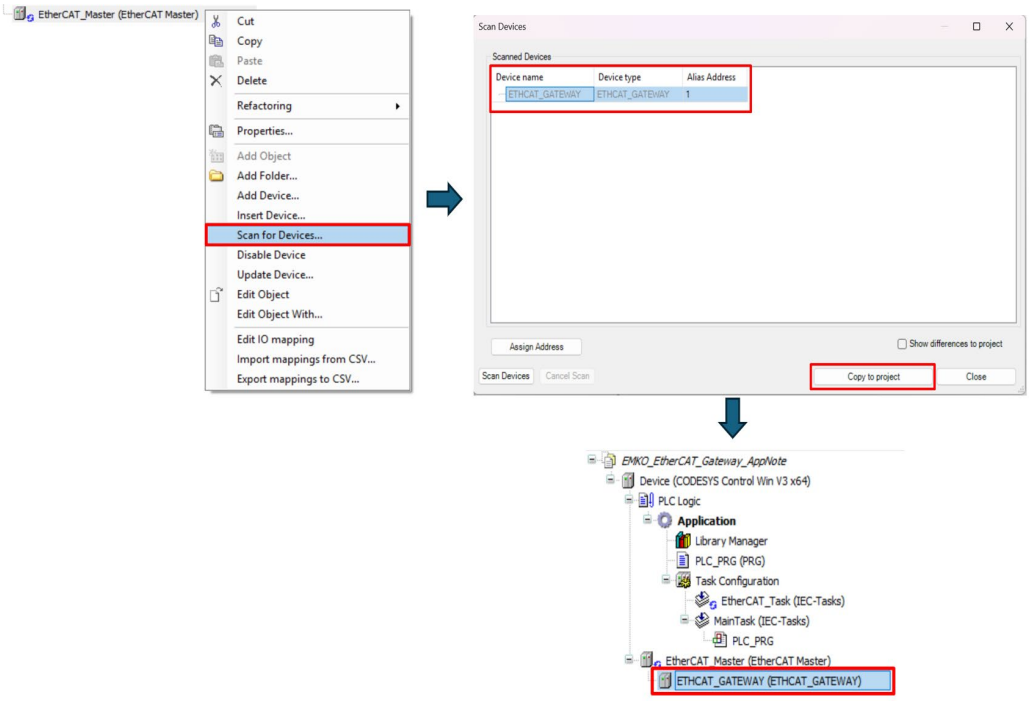
Şekil 60. Manuel tanımlama ile EtherCAT Gateway modülünün eklenmesi

Otomatik Tanımlama

EtherCAT Bus Coupler modülü, projeye otomatik olarak da eklenebilir. Bu yöntem, sistemde aktif olarak bağlı olan slave cihazların taranarak otomatik şekilde projeye eklenmesini sağlar. İşlem adımları şu şekildedir:

1. Projeye daha önce eklenmiş olan **EtherCAT Master** cihazı üzerinde **sağ tıklanır**.
2. Açılan menüden **Scan for Devices...** seçeneği seçilir.
3. Açılan tarama penceresinde bağlı olan cihazlar taranır ve sonuçlar listelenir.
4. Tarama sonucunda **ETHCAT_GATEWAY** cihazı algılanır ve listede görüntülenir.
5. Bu cihaz seçili iken, sağ alt köşedeki **Copy to project** butonuna tıklanır.

Bu işlemin ardından, EtherCAT Master altında otomatik olarak EtherCAT Gateway cihazı oluşturulur. İlgili işlem adımları Şekil 61 üzerinde görsel olarak gösterilmiştir.



Şekil 61. Otomatik tanımlama ile EtherCAT Gateway modülünün eklenmesi

Adım 8 : Genişleme Modülleri Verilerini Okuma

Cihaz yapılandırmasının ardın sistem, çalışma (RUN) moduna geçirilmelidir. Bu işlem, **Project** menüsü altındaki **Online Config Mode** seçeneği ile gerçekleştirilir. Sistem yapılandırması doğruysa — yani:

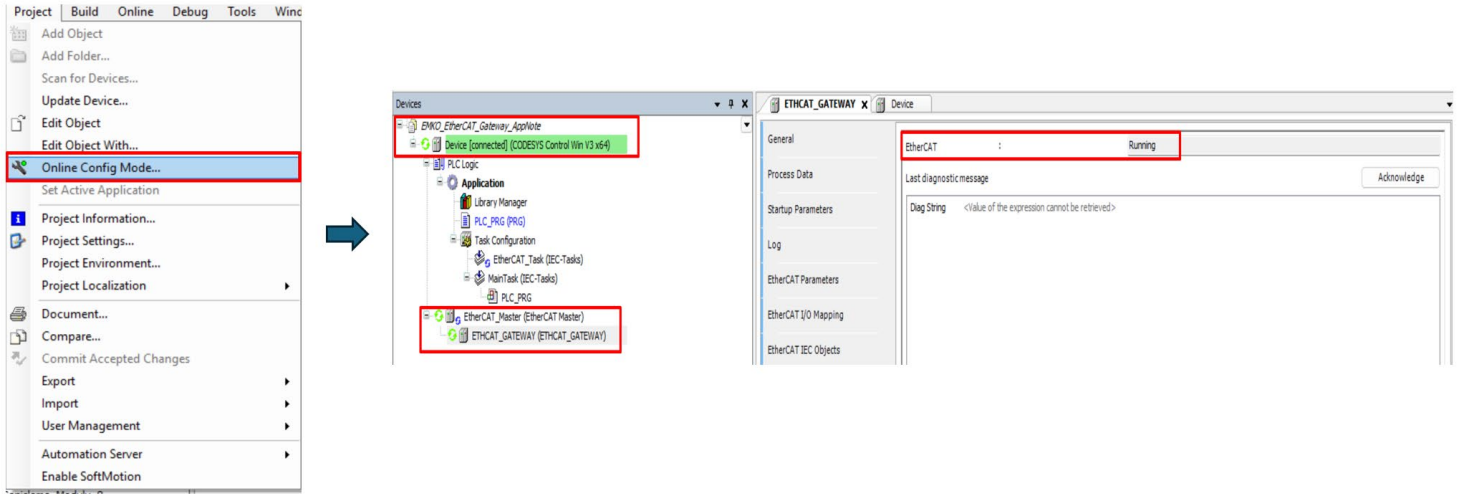
- Genişleme modüllerine doğru slotlar atanmışsa,
- EtherCAT bağlantılarında fiziksel sorun (kablo, port vs.) yoksa,
- Master-Slave haberleşmesi uyumluysa,

bu durumda sistem **çalışma moduna başarıyla geçer.**

Cihaz RUN durumundayken:

- Proje ağacında **Device** başlığı altında "connected (CODESYS Control Win V3 x64)" ibaresi görünür ve simge **yeşil** olur.
- EtherCAT Master ve bağlı EtherCAT Gateway modül **yeşil döngü** simgesi ile gösterilir.
- Sağ panelde yer alan **Status > EtherCAT** sekmesinde sistem durumu **"Running"** olarak görünür.
- Donanım üzerinde de **bus coupler üzerindeki RJ45 port LED'leri** sürekli sarı ve **NS LED'i** sürekli yeşil yanar. Bu da fiziksel bağlantının ve veri alışverişinin sorunsuz olduğunu gösterir.

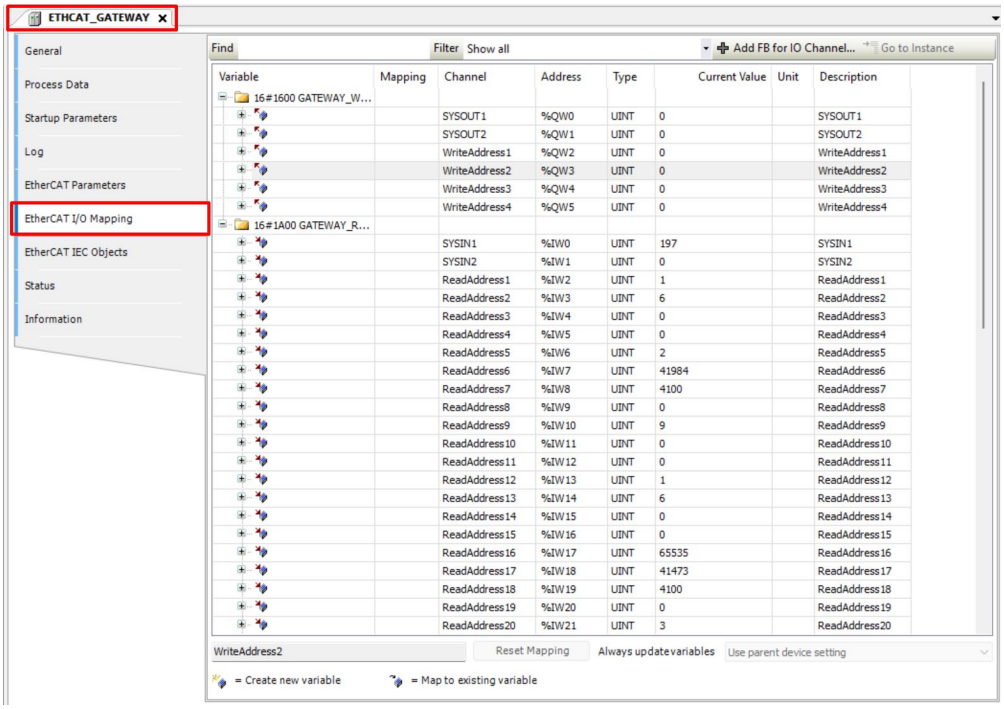
Bu durum, yapılandırmanın tamamlandığını ve genişleme modülleriyle birlikte sistemin çevrimsel haberleşmeye hazır hale geldiğini gösterir. Bahsedilen işlemler Şekil 62'de gösterilmiştir.



Şekil 62. EtherCAT Gateway modülünün çalıştırılarak haberleşmenin başlaması işlem adımları

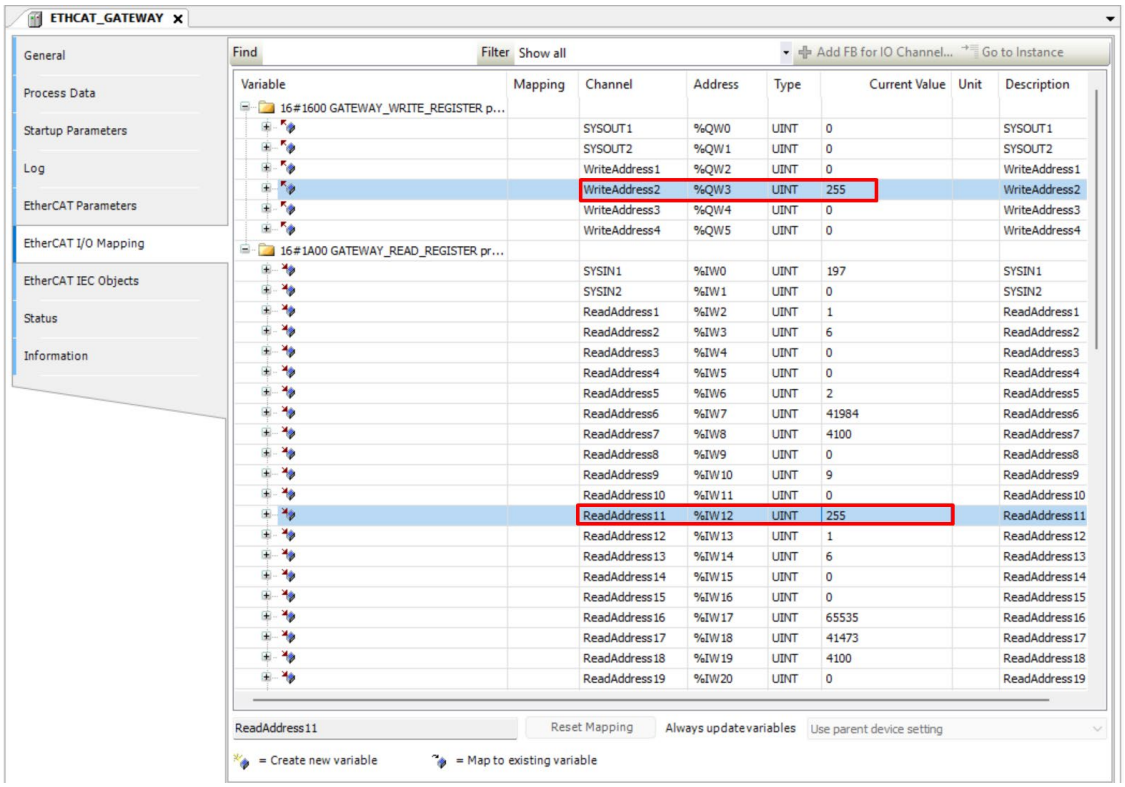
I/O Mapping Alanı Üzerinden Genişleme Modül Verilerinin Kontrolü

PDO alanları üzerinden dijital/analog giriş ve çıkışlara ait çevrimsel veriler okunabilir ve yazılabilir. CODESYS içerisinde bu alanlar I/O Mapping sekmesiyle gösterilmektedir. Şekil 63'te 0x1600 adresli RxPDO (Modbus yazma paketleri) ve 0x1A00 adresli TxPDO (Modbus okuma paketleri) alanlarının EtherCAT Master tarafından nasıl listelendiği gösterilmektedir.



Şekil 63. CODESYS yazılımında EtherCAT Gateway'e ait PDO alanlarının I/O Mapping sekmesi üzerinden gösterimi

Örneğin, SMR1P1-00-00-01-00_1 (15xTO) kodlu genişleme modülündeki dijital çıkışların kontrolü için Modbus mapping'de 06 fonksiyonu kullanılmış ve veri WriteAddress2 PDO adresine eşlenmiştir. Tüm çıkışların aktif edilmesi için bu adrese 255 değeri yazıldığında, karşılık gelen çıkışlar devreye girer. Aynı modülde durum bilgilerinin okunması için ise Modbus mapping'de 03 fonksiyonu kullanılmıştır. Dijital, çıkış bilgisi ReadAddress11 PDO adresine denk gelmekte olup, ilgili adres izlendiğinde dijital çıkış değerinin 255 olduğu görülür. Yazma ve okuma işlemlerine ait bu örnek Şekil 64'te gösterilmektedir.



Şekil 64. SMR1P1-00-00-01-00_1 (15xTO) modülü için WriteAddress2 ve ReadAddress11 üzerinden çıkışların kontrolü ve geri besleme bilgisinin gösterimi