
GAZİ ÜNİVERSİTESİ
Gazi
1926

ADAPTİF KONTROLLÜ BİR CNC PROTOTİPİNİN GELİŞTİRİLMESİ

Esin POLAT TEKTAŞ

YÜKSEK LİSANS TEZİ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANA BİLİM DALI

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ARALIK 2019

Esin POLAT TEKTAŞ tarafından hazırlanan "ADAPTİF KONTROLLÜ BİR CNC PROTOTİPİNİN GELİŞTİRİLMESİ" adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Gazi Üniversitesi Elektrik Elektronik Mühendisliği Ana Bilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Danışman Prof.Dr. Mahir DURSUN

Elektrik Elektronik Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi ……..……..

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

Başkan: Prof.Dr. Mustafa ALKAN

Elektrik Elektronik Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi ...……

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

Üye: Prof.Dr. Bilgin KAFTANOĞLU

İmalat Mühendisliği Ana Bilim Dalı, Atılım Üniversitesi ...……

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

Tez Savunma Tarihi: 17/12/2019

Jüri tarafından kabul edilen bu çalışmanın Yüksek Lisans Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum

……..…….

Prof. Dr. Sena YAŞYERLİ

Fen Bilimleri Enstitüsü Müdürü

ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

……..
Esin POLAT TEKTAŞ
17/12/2019

ADAPTİF KONTROLLÜ BİR CNC PROTOTİPİNİN GELİŞTİRİLMESİ

(Yüksek Lisans Tezi)

Esin POLAT TEKTAŞ

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Aralık 2019

ÖZET

Bu çalışmada adaptif kontrollü 3 eksenli CNC tezgah prototipi tasarlanmıştır. Çalışmada, önce 3 adet servo motorun eşzamanlı olarak kontrol programı hazırlanmıştır. Kontrol programında Schineider SoMachine, SoMove, Kepserver program arayüzleri kullanılmıştır. Adaptif kontrollü CNC makine prototipini oluşturmak için; Schineider marka 3 servo motor (1,3 kw), 3 motor sürücüsü (lexium 32), 3 adet harici konum algılayıcı (Osicoder), güç kaynağı , dokunmatik panel , Motion Controller PLC (Modicon LMC078), haberleşme kabloları, güç kaynağı gibi yardımcı aparatlar, 3 eksenli CNC makinesine montajı yapılmıştır. Harici konum algılayıcılar ile sistemin kontrolü geri besleme sağlanarak olası hatalara karşı arındırılmıştır. Sistemin güvenlik kontrolü ve PLC ile veri alışverişi sağlanabilmesi amacıyla KEPServerEX OPC programı kullanılmıştır. KEPServerEX OPC programı vasıtasıyla alınan veriler Visual Studio programı ve C# programlama dili kullanılarak işlenmiştir. Adaptif kontrolün sağlanabilmesi için motor hız ve kp değerleri akım değerleri baz alınarak düzenlenmesi sağlanmıştır.

Bilim Kodu	:	90527
Anahtar Kelimeler	:	Adaptif, CNC, SoMachine, Sercos
Sayfa Adedi	:	135
Danışman	:	Prof.Dr. Mahir DURSUN

A NEW PROTOYPE DESIGNING OF CNC BY USING ADAPTIVE CONTROL

(M. Sc. Thesis)

Esin POLAT TEKTAŞ

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

December 2019

ABSTRACT

In this study, 3 axis CNC machine prototype with adaptive control is designed. In this study, the control program for 3 servo motors was prepared simultaneously. Schineider SoMachine, SoMove, KEPserverEX program interfaces were used in the control program. To create a prototype CNC machine with adaptive control; Schineider 3 servo motors (1,3 kw), 3 motor drives (Lexium 32), 3 external encoders (Osicoder), power supply, touch panel, Motion Controller PLC (Modicon LMC078), communication cables, auxiliary devices such as power supply , 3 axis CNC machine have been assembled. Control of the system with external encoders is provided by feedback to ensure that possible errors are avoided. KEPServerEX OPC program is used for security control of the system and data exchange with PLC. The data obtained through KEPServerEX OPC program was processed using Visual Studio program and C # programming language. In order to provide adaptive control, motor speed and kp values are arranged based on current values.

Science Code	:	90527
Key Words	:	Adaptive, CNC, SoMachine, Sercos
Page Number	:	135
Supervisor	:	Prof.Dr. Mahir DURSUN

TEŞEKKÜR

Bu Tez BAP Biriminin 07/2017-06 kodu ile desteklediği proje üzerine yazılmıştır.

Çalışmalarım boyunca desteğini, ilgisini ve yardımlarını esirgemeyen değerli danışmanım Prof.Dr. Mahir DURSUN'a, saygıdeğer hocalarım Doç. Dr. Ali SAYGIN ve Dr. Semih ÖZDEN'e, SİMKOL Elektrik ailesinden Selçuk BİRTANE'ye, BAP birimi çalışanlarına, maddi ve manevi desteklerinden dolayı eşime, beni sonsuz sevgiyle bekleyen kızım Asya'ya, anne ve babama, çalışmalarım esnasında gösterdikleri sabırdan dolayı mesai arkadaşlarıma teşekkürü bir borç bilirim.

İÇİNDEKİLER

Sayfa

ÖZET iv

ABSTRACT v

TEŞEKKÜR vi

İÇİNDEKİLER vi

ÇİZELGELERİN LİSTESİ x

ŞEKİLLERİN LİSTESİ xi

RESİMLERİN LİSTESİ xiii

SİMGELER VE KISALTMALAR xix

1. GİRİŞ 1

2. ADAPTİF KONTROLLÜ CNC'NİN MEKANİK TASARIMI 9

3. CNC'NİN DENETLEME PROGRAMLAMALARI 19

3.1. SoMove Programı ile Sürücü Programlama 19

3.1.1. SoMove programında cihaz eklenmesi 19

3.1.2. SoMove programında cihaz parametreleri optimizasyonu 21

3.2. SoMachine Programının Kullanılması 22

3.2.1. SoMachine programında cihazların eklenmesi 22

3.2.2. SoMachine'de programlama 30

3.2.3. Vijeo programının kullanılması 55

3.2.4. Kepserver programının kullanılması 60

4. ADAPTİF KONTROL 73

4.1. Klasik Kontrol Sistemleri 73

4.2.Kontrol Sistemi Tanımı 73

Sayfa

4.2.1. Açık döngü kontrol sistemi 73

4.2.2. Kapalı döngü kontrol sistemi 74

4.3.Denetleyici Etkileri 74

4.3.1. Aç-kapa kontrol 75

4.3.2. Oransal kontrol 75

4.3.3. Diferansiyel (Türev) kontrol 75

4.3.4. İntegral kontrol 76

4.3.5. Oransal + integral + türevsel kontrol (PID) 76

4.4. Adaptif Kontrolün CNC'ye Uyarlanması 78

4.4.1. Tanımlama fonksiyonu 80

4.4.2. Karar fonksiyonu 80

4.4.3. Düzenleme fonksiyonu 81

4.5. Adaptif Kontrol Sınıflandırmaları 82

4.5.1. Basit Adaptif Kontrol Sistemleri 82

4.6. Adaptif Kontrol Yöntemleri 83

4.6.1. Optimizasyonlu adaptif kontrol (OAK) 83

4.6.2. Geometrik adaptif kontrol (GAK) 84

4.6.3. Kısıtlı adaptif kontrol (KAK) 84

4.7. Takım Tezgâhlarında İzleme Yöntemleri 89

4.8. Adaptif Kontrollü Talaş Kaldırma İşlemi 90

4.8.1. Giriş değişkenleri 91

4.8.2. Çıkış değişkenleri 91

4.8.3. Adaptif kontrolün talaş kaldırma işleminde gerekliliğinin nedenleri 91

4.8.4. Talaş kaldırmada adaptif kontrolün faydaları 92

Sayfa

4.9. Talaş Kaldırma İşlemlerinde Adaptif Kontrol Uygulaması 92

4.10 Harici Konum Algılayıcı İle Yapılan Adaptif Kontrol 93

4.11 Hız Regülatörü P Faktörü İle Yapılan Adaptif Kontrol 98

4.11.1. SoMachine içerisinde kullanılan fonksiyon blokları 98

4.11.2 Sabit hızda P faktörünün akım değerine etkisi 102

4.11.3 Değişken hızlarda P faktörünün akım değerine etkisi 109

5. SONUÇ VE ÖNERİLER 115

KAYNAKLAR 117

EKLER 119

EK-1. SoMove programının kurulması 120

EK-2. SoMachine programının kurulumu 123

EK-3. Kepserver Programının Kurulması 129

EK-4. G-CODE ve M-CODE ile programlama 131

ÖZGEÇMİŞ 135

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

Çizelge 4.1. Kp, Ki, Kd Katsayılarının çıkışa etkisi. 78

Çizelge 4.1 G-Code Dizilimi 133

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

Şekil 1.1. Program işlem basamakları 8

Şekil 4.1. Açık döngü kontrol sistemi 74

Şekil 4.2. Kapalı döngü kontrol sistemi. 74

Şekil 4.3. Farklı KP değerlerinin sistem cevabına olan etkisi 77

Şekil 4.4. PID kontrol blok şeması 77

Şekil 4.5. Adaptif kontrol 79

Şekil 4.6.Adaptif Kontolün Temel Konfigürasyonu 82

Şekil 4.7. Optimizasyonlu adaptif kontrol 84

Şekil 4.8. Kısıtlı adaptif kontrol 85

Şekil 4.9. İntegral kısıtlı adaptif kontrol 86

Şekil 4.10. K.A. Adaptif kontrol blok şeması 87

Şekil 4.11. M.R. adaptif kontrol blok şeması 88

Şekil 4.12. Tornalamada değişken yapılı kontrol. 89

Şekil 4.13. Sabit hızda ve boşta 3 kez artan P faktörü 103

Şekil 4.14. Sabit hızda ve boşta 3 kez azalan P faktörü 103

Şekil 4.15. Sabit hızda ve yükte 3 kez azalan P faktörü 104

Şekil 4.16. Sabit hız, yükte 3 kez azalan P faktörü 104

Şekil 4.17. Sabit hızda, 100 P faktörü, boşta ve yükte akım 105

Şekil 4.18. Sabit hızda, 335 P faktörü, boşta ve yükte akım 105

Şekil 4.19. Sabit hız, boşta 2 kez azalan, 2 kez artan P faktörü 106

Şekil 4.20. Sabit hız, yükte 3 kez azalan P faktörü 107

Şekil 4.21. Sabit hız, yükte artan P faktörü 108

Şekil **Sayfa**

Şekil 4.22. Sabit hız, yükte 3 kez artan P faktörü 110

Şekil 4.23. Sabit hız, yükte, sabit P faktörü..... 110

Şekil 4.24. Sabit hız, yükte P faktörü 111

Şekil 4.25. Sabit hız, yükte P faktörü 112

RESİMLERİN LİSTESİ

Resim **Sayfa**

Resim 2.1 3 eksenli CNC prototipi karşıdan görünüşü 9

Resim 2.2. 3 eksenli CNC prototipi yandan görünüşü 10

Resim 2.3. Servo motor kontrol arayüzü 11

Resim 2.4.(a) Servo-motor (b) External konum algılayıcı 12

Resim 2.5 (a)Vidalı mil (b)Mil tutucu (c)Kaplin (d)Rulman (e)Bir eksen için kızak (f)Alt destekli mil sac (g) Lineer rulman (h) Sigma profil 12

Resim 2.6. CNC HMI panel görüntüsü 13

Resim 2.8. CNC Y ekseni 14

Resim 2.9. CNC Z ekseni 15

Resim 2.10. CNC kontrol üniteleri 15

Resim 2.11. 3 Eksenli CNC bağlantı şeması 16

Resim 2.12. PLC, konum algılayıcı, bilgisayar, ekran bağlantısı 17

Resim 2.13. Servo motor ve sürücü bağlantısı 18

Resim 3.1 Sürücü seçimi 19

Resim 3.2 Sürücü ve motor seçimi 20

Resim 3.3 Sürücüye adres atama 20

Resim 3.4. Sürücü ile haberleşme 21

Resim 3.5. Motor paremetre optimizasyon 22

Resim 3.6 SoMachine programlama-1 23

Resim 3.7.(a) SoMachine programlama-2 (b) SoMachine programlama-3 23

Resim 3.8 SoMachine programlama-4 24

Resim 3.9. PLC ekleme 24

Resim 3.10 Sürücü ekleme 25

Resim **Sayfa**

Resim 3.11 Sürücü tanıtma 26

Resim 3.12 Servo motor ayarları 27

Resim 3.13 CANbus modülünün eklenmesi 27

Resim 3.14.(a)CANbus kurulum-1(b) CANbus kurulum-2 28

Resim 3.15. CANopen manager eklenmesi 28

Resim 3.16. Konum algılayıcı eklenmesi 29

Resim 3.17.(a)Konum algılayıcı Adres Atama-1(b) Konum algılayıcı Adres Atama-2 29

Resim 3.18. Panelin eklenmesi 30

Resim 3.19.(a) PLC'nin programlanması-1(b) PLC'nin programlanması-2 31

Resim 3.20.(a) PLC'nin programlanması-3 (b) PLC'nin programlanması-4 31

Resim 3.21. GVL kütüphanesinin eklenmesi 32

Resim 3.22.(a) SERVO POU programı-1 (b) SERVO POU programı-2 32

Resim 3.23 GVL Kütüphanesi-1 33

Resim 3.24. MC_POWER komutu 34

Resim 3.25. MC_ SetPosition komutu 35

Resim 3.26. MC_ MC_ReadActualPosition komutu 35

Resim 3.27. MC_STOP komutu 36

Resim 3.28. MC_Reset komutu 36

Resim 3.29. MoveAbsolute komutu 37

Resim 3.30. MoveAbsolute komutu 38

Resim 3.31. Pozisyon/hız bilgisi çevrimi 39

Resim 3.32. X ekseni için hazırlanan program 40

Resim 3.33. Y ekseni için hazırlanan program 40

Resim 3.34. Z ekseni için hazırlanan program 41

Resim **Sayfa**

Resim 3.35. X-Y-Z ekseni için hazırlanan program-1 42

Resim 3.36. X-Y-Z eksenleri için hazırlanan program-2 43

Resim 3.37. SMC_ReadNCFile komutu 44

Resim 3.38. Prg_2 programı ve SMC_ReadNCFile komutunun uygulaması 44

Resim 3.39. SMC_NCDecoder komutu 45

Resim 3.40. SMC_RoundPath Komutu 46

Resim 3.41. SMC_ToolCorr komutu 46

Resim 3.42. SMC_AvoidLoop komutu 47

Resim 3.43. SMC_LimitCircularVelocity komutu 47

Resim 3.44. CheckVelocity komutu 48

Resim 3.45. Prg_3'ün ekran görüntüsü 49

Resim 3.46 SMC_RestoreQueue komutu 50

Resim 3.47. SMC_Interpolator komutu 51

Resim 3.48. SMC_ControlAxisByPos 51

Resim 3.49. SMC_TRAFO_Gantry3 komutu 52

Resim 3.50 SMC_TRAFO_Gantry3 komutu 52

Resim 3.51. İnterpolasyon programı 53

Resim 3.52. Konum algılayıcı veri haritası 54

Resim 3.53. ENC program görüntüsü 55

Resim 3.54. Vijeo haberleşme ayarları-1 55

Resim 3.55. Vijeo haberleşme ayarları-2 56

Resim 3.56. Vijeo haberleşme ayarları-3 56

Resim 3.57. Değişken seçimi-1 57

Resim **Sayfa**

Resim 3.58.(a) Değişken seçimi-2 (b) Değişken seçimi-3 57
Resim 3.59. Değişken seçimi-4 58
Resim 3.60. Paneli veri atama 58
Resim 3.61 HMI panel görüntüsü 59
Resim 3.62 HMI panel görüntüsü 59
Resim 3.63 Kepserver kanal ekleme-1 61
Resim 3.64. Kepserver kanal ekleme-2 61
Resim 3.65. Kepserver kanal ekleme-3 62
Resim 3.66. Kepserver kanal ekleme-4 62
Resim 3.67. Kepserver kanal ekleme-5 63
Resim 3.68 Kepserver cihaz ekleme-1 63
Resim 3.68 Kepserver cihaz ekleme-2 64
Resim 3.69. Kepserver cihaz ekleme-3 64
Resim 3.70. Kepserver cihaz ekleme-4 65
Resim 3.71. Kepserver cihaz ekleme-5 65
Resim 3.72. Kepserver cihaz ekleme-6 66
Resim 3.73. Kepserver cihaz ekleme-7 66
Resim 3.74. Kepserver cihaz ekleme-8 67
Resim 3.75. Kepserver cihaz ekleme-9 67
Resim 3.76. Kepserver cihaz ekleme-10 68
Resim 3.77. Kepserver cihaz ekleme-11 68
Resim 3.78 Kepserver cihaz ekleme-12 69
Resim 3.79 Kepserver cihaz ekleme-13 69
Resim 3.80 İzlenecek adres ekleme-1 70

Resim **Sayfa**

Resim 3.81. İzlenecek adres ekleme-2 71

Resim 4.1. VisualStudio adaptif kontrol ekranı-1..... 93

Resim 4.2. VisualStudio adaptif kontrol ekranı-2..... 94

Resim 4.3. VisualStudio adaptif kontrol ekranı-3..... 94

Resim 4.4. Harici konum algılayıcı ile adaptif programlama-1 96

Resim 4.5. Harici konum algılayıcı ile adaptif programlama-2 97

Resim 4.7 Frekans üreteci 99

Resim 4.8. Sercos akım okuma..... 99

Resim 4.9. Sercos P faktörü okuma 100

Resim 4.10. Sercos P faktörü yazma..... 101

Resim 4.11. P Faktörü program ara yüzü 102

Resim 4.12. X ekseninde değişken P faktörü ile doğrusal işleme 106

Resim 4.13. X-Y ekseninde azalan P faktörü ile doğrusal işleme 107

Resim 4.14. X-Y ekseninde artan P faktörü ile doğrusal işleme 108

Resim 4.15. Hız değişimi programlanması 109

Resim 4.16. Düşük Hız Doğrusal İşleme 111

Resim 4.17. Yüksek hız dairesel işleme-1..... 112

Resim 4.18. Düşük hız dairesel işleme-1 113

Resim 4.19. Yüksek hız dairesel işleme-1..... 113

Resim 4.20. Düşük hız dairesel işleme-2 113

Resim 1.1.(a) SoMove yükleme ikonu (b) SoMove dil seçimi 120

Resim 1.2.(a) SoMove yükleme ekranı-1(b) SoMove yükleme ekranı-2 120

Resim 1.3.(a) SoMove yükleme ekranı-3 (b) SoMove yükleme ekranı-4 121

Resim 1.4.(a) SoMove yükleme ekranı-5 (b) SoMove yükleme ekranı-6 121

Resim **Sayfa**

Resim 1.5.(a) SoMove yükleme ekranı-7 (b) SoMove yükleme ekranı-8 122

Resim 2.1.(a) Sanal disk kurulumu (b) SoMachine kurulum ekranı-1 123

Resim 2.2.(a) SoMachine kurulum ekranı-2 (b) SoMachine kurulum ekranı-3..... 123

Resim 2.3.(a) SoMachine kurulum ekranı-4 (b) SoMachine Kur kurulum ekranı-5..... 124

Resim 2.4.(a) SoMachine kurulum ekranı-6 (b) SoMachine kurulum ekranı-7..... 124

Resim 2.5. SoMachine kurulum ekranı-8..... 125

Resim 2.6. SoMachine lisans alma-1 125

Resim 2.7 SoMachine lisans alma-2 126

Resim 2.8. SoMachine lisans alma-3 126

Resim 2.9. SoMachine lisans alma-4 127

Resim 2.10. SoMachine lisans alma-5 127

Resim 2.11. SoMachine lisans alma-6 128

Resim 2.12.(a) SoMachine lisans alma-7 (b) SoMachine lisans alma-8..... 128

Resim 3.1. Kepserver kurulumu-1 129

Resim 3.2. Resim Kepserver kurulumu-2 129

Resim 3.3. Kepserver kurulumu-3 130

Resim 3.4. Kepserver kurulumu-4 130

Resim 4.1 Örnek G-Code 134

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	**Açıklamalar**
"	İnç
A	Amper
kW	Kilowatt
MM	Milimetre
V	Volt

Kısaltmalar	**Açıklamalar**
AC	Alternatif akım
CAN	Denetleyici alan ağı
CANBUS	Denetleyici alan ağı veri yolu
CNC	Bilgisayar nümerik denetleyici
FB	Fonksiyon bloğu
I/O	Giriş/çıkış
OPC	Planlama ve kontrol işlemleri
PC	Kişisel bilgisayar
PLC	Programlanabilir mantık denetleyici
POU	Program organizasyon birimi
PWM	Darbe genişlik modülasyonu

1. GİRİŞ

CNC makinalar endüstrinin hemen hemen bütün alanlarında ve imalat sektöründe yaygın olarak kullanılmaktadırlar. Ancak her ne kadar makinalar üzerinde çalışmalar devam etse de yay ve konik gibi nonlineer işlemelerde hala istenilen kalitede düzgün ürün üretilememektedir. Bu problemin başlıca nedeni özellikle yay veya konik gibi dereceli eğik geometrik şekilli çizimlerin imalatında her eksendeki tahrik motorunun kontrol sistemindeki mikroişlemcilerin çalışma ve pozisyon sezimi algılanması ile haberleşmenin eş zamanlı olarak gerçekleştirilememesidir. Ayrıca bu makinaların üzerinde yeni eğitim görenlerin kolaylıkla problemleri anlayabilmesi de yeni eğitim teknolojilerini de gerektirmektedir.

Genel olarak 3 ekseni kontrol eden 3 farklı motor, aynı anda farklı hız ve ivme ile çalıştırılır ve daha yumuşak ve pürüzsüz bir işleme interpolarasyon ve nonlineer interpolarasyon ile sağlanır. Bütün bu işlemlere ek olarak; motorların akım değerleri anlık olarak okunmuş, elde edilen akım verileri kullanılarak, en uygun akım değerinde çalıştırılması için anlık p faktörü değişimi ve en uygun sürede çalışabilmesi için anlık hız değişimi sağlanmıştır. Mil boşluğundan kaynaklanan farklar tespit edilerek kontur hatalarının düzeltilmesi sağlanmıştır. CNC'de yer alan GTO touch HMI panel,12.1" panel vasıtasıyla sistemin el ile kontrolü sağlanmıştır.

Mevcut halinde mekanik kısmı bulunan 3 eksenli CNC prototipine, Schnider marka PLC, sürücüler, servo motorlar, konum algılayıcılar takılmıştır. Güç ve haberleşme kablo bağlantıları gerçekleştirilmiştir. Donanımların birbiri ile düzgün çalışabilmesi için gerekli haberleşme ayarları ve aygıt tanımlama ayarları yapılmıştır. Servo motorların ve konum algılayıcıların istenilen şekilde çalışabilmesi için program blokları kullanılmış ve gerekli programlamaları tamamlanmıştır. Bilgisayar ile PLC arasındaki haberleşmeler ilave yazılım programları aracılığı ile dinlenmiş, C# ile hazırlanan programlama ile adaptifliği sağlamak amacıyla komuta edilmiştir.

Tasarlanan 3 eksenli CNC ile yay ve konik dereceli köşe şekillendirmede pürüzlülük ve merdiven basamakları en aza indirgenmesi hedeflenmiştir. Ayrıca CNC'nin kullanıcı hatalarından, mekanik hatalardan ve haberleşme hatalarından arındırılması güvenli hale

getirilmesi yönünde çalışmalar yapılmıştır. Uyarlanabilirliğin (adaptifliğin) sağlanması ile fire miktarının, güç sarfiyatının azaltılarak maliyet etkinlik amaçlanmıştır.

Araştırmacıların kavisli geometrik şekilli çizimlerin üretimini çözmek için literatürde birçok çalışma yaptıkları görülmektedir. Ayrıca benzer çalışmalarda programlama, tasarım, hız ve hızlanmanın CNC makinelerin işlem kalitesine etkisi ile ilgili işlem çalışmaların bazıları aşağıda özetlenmiştir.

Beşyaprak tarafından 2016 yılında Matlab programı kullanılarak PLC'ye bağlı servo motorda pozisyon ve hız kontrolü gerçekleştirilmiştir. Schneider Motion Controller LMC058 PLC Lexium servo motor sürücüsü kullanılmıştır. Projede motor kontrolü yapılmış, KEPServerEX OPC programı kullanılarak veri alışverişi sağlanmıştır. KEPServerEX OPC arayüzünde değişkenler Matlab OPC araçları ile yazılım içerisinde işlenmiş, gerçek konumlar okunmuş ve istenen yeni konum bilgileri gönderilmiştir [1].

Zhang, Sun, Gao ve diğerleri 2011 yılında yaptığı çalışmasında, işleme hassasiyetini sağlarken NC makinenin maksimum hızlanma yeteneklerini kullanan doğrusal hızlanma ve yavaşlama modunu kullanan bağlantı noktalarındaki ilerleme hızını iyileştirmek için çok dönemli bir tornalama yöntemi önermektedir. Global işleme hızını arttırmak için yeni ve daha verimli ileriye dönük bir yöntem ve ilerleme hızı geçersiz kılma yöntemi de önerilmektedir. Ayrıca haz ırlanan algoritmanın gerçek zamanlı enterpolasyon ihtiyacını da karşıladığı anlaşılmıştır [2].
Jee ve Koren tarafından 2004 yılında; hassas kontur işleme için uyarlanabilir bulanık mantık denetleyicisini (AFLC) açıklayan, denetleyicinin performansının ölçülmesi ile tahmini bozulma değerlerini içeren ve üyelik fonksiyonlarının ayarlanması için kararlılık analizi içeren bir çalışma yayınlanmıştır. Çalışmada hem simülasyonlar hem de deneyler, diğer iki denetleyiciye kıyasla kontur izleme doğruluğu açısından performansın, özellikle sürtünmenin besleme tahriklerinin ciddi bir sorun olduğu makine sistemlerinde AFLC'nin üstünlük sağlayabildiği gösterilmiştir [3].

Ambrizal ve arkadaşları tarafından 2017 yılında yayınlanan makalede; bir freze bıçağındaki tek bir dişte aşınmanın nasıl tespit edildiğini ve geliştirilen yöntemin tekrarlanabilirliğini araştırmıştır. Çalışmanın sonucunda takım aşınması ile ilişkili çeşitli sinyal özelliklerinin ilk birkaç salınımdan elde edilebileceği elde edilmiştir. Konum kodlayıcılardan gelen cevaplar

özellikle kısa vadede iyi tekrarlanabilirlik sergilerken, uzun vadeli tekrarlanabilirliğin güvenilir olmadığı anlatılmaktadır [4].

2012 yılında Pejryd, Repo ve Beno tarafından yapılan yayında, CNC kullanarak pozisyonlama sonucunda oluşabilecek makinedeki aşınma, kesme kenarlarının geometrisini değiştirir, bu da kesme kuvvet bileşenlerinin yönünü ve genliğini ve işleme prosesindeki dinamikleri etkilemektedir. Kuvvetlerdeki ve dinamiklerdeki bu değişiklikler toplanıp işlenmesi anlatılmaktadır [5].

2005 yılında Büyükşah tarafından 3 eksenli bir CNC makine tasarımı yapılmıştır Visual Nastran programı kullanılarak gerilme ve sehim analizleri yapılmıs, sayısal hesaplamalarla desteklenmistir. Her 3 eksen için de gerekli olan vidalı millerin modelleri saptanması, servo motor güçleri hesaplanması ve maliyetleri hakkında çalışma yapılmıştır [6].

Kayalık tarafından 2009 yılında sunulduğu çalışmada; Siemens'in Simotion D425 adlı servo motor kontrol ünitesi, bu üniteye bağlı iki adet CUA31 servo güç yükselticisi ve iki adet servo motor kullanılmıştır. Bu ürünün dik işlem tezgâhı seklinde çalışabilmesi için, motorları kontrol etmekte kullanılan Siemens Scout adlı programın içerisinde yer alan Structured Text adlı programlama dilinde, temel hareket fonksiyonları tanımlanmıştır. Visual Basic programlama dilinde geliştirilen bir yazılımla, G kodu tarzında verilen şekil komutları, temel hareket fonksiyonlarını çağıran Structured text programlarına dönüştürülmüştür. Bu üretilen Structured Text programlarının Siemens Scout içerisinde çalıştırılmasıyla, istenilen şekillerin işletilmesi sağlanmıştır [7].

Turgut tarafından 2004 yılında yapılan çalışmada CNC makinesi için diyalog yöntemini kullanarak kodları nasıl üretebileceğini göstermektedir. Takım yolunun oluşturulmasında, geometrik tanımlama teknikleri kullanılarak, FANUC formatına uygun CNC parça programı türetilmiştir. Geliştirilen bilgisayar programı VISUAL BASIC 6.0 programlama dili ile yazılmıştır. Program ACCESS 7.0'da hazırlanan malzeme ve kesici takım veri tabanları ile desteklenmiştir [8].

Lin ve arkadaşları tarafından 2006 yılında yayınlanan CNC makineye uygulanan uyarlamalı bir tekrarlayan sinir ağı (ARNN) hareket kontrolünün modern kontrol yöntemlerini içeren bir çalışma vardır [9].

Mo ve arkadaşları, Sarma makinasında PLC tabanlı kontrol sistemi uygulamalarında PLC ile AC servo motor hız kontrolü gerçekleştirmişlerdir [10].

Birok 2013 yılında yapmış olduğu yüksek lisans tezinde, S7-300/400 PLC tipleri için sayısal benzetim yöntemi kullanarak PID (proportional integral derivative) denetleyici tasarımı yapılmasını gerçekleşmiştir. Böylece gerçek sisteme ihtiyaç olmadan PID katsayılarını kolay ve güvenilir bir şekilde belirlemiştir. Elde edilen sonuçların oluşturduğu temel doğrultusunda PID katsayılarını otomatik ayarlayabilen (self tuning) sistem üzerinde çalışmalar yapmanın mevcut çalışmayı daha ileriye taşıyabileceğini belirtmiştir [11].

Bayrak ve arkadaşları, IEC 61131-3 protokolüne sahip farklı PLC'lerde CFC (sürekli işlem diyagramı) programlama diline dönüştürülmüş Matlab/Simülink modellerin karşılaştırmalarını yapmışlardır [12].

Bedir yüksek lisans tezinde, Günümüzde PLC ile çalışan tüm otomasyon sistemlerinin SCADA sistemleri ile kontrol edilmesi birçok yönden avantaj sağlayacağını belirtmiştir. Ayrıca manyetik alan kuvveti ile ileri-geri doğrusal hareket elde edilirken geri getiren yay kullanmak yerine, doğal mıknatısla elektro mıknatısı bir arada kullanarak PLC ve SCADA sistemi ile sürekli hale getirilmesinin endüstriyel uygulaması gerçekleştirmiştir [13].

Akçura tez çalışmasında, Analog sistemde günün ve çağın değişen şartlarına uyum sağlamak mümkün olmazken, PLC'nin kullanıldığı sistemler her türlü soruna optimal çözüm getirebildiğini, ek yatırım, kablo bağlantı değişikliği v.b. gibi birçok ayrıntıya gerek kalmadan ihtiyaçların zamanında karşılanabildiğini savunmuştur. PLC'nin programlanması için Gemo Ladder Editör programını kullanmıştır. Programlanmanın anlaşılır olması ve takibinin kolaylığı için LAD programlama metodu kullanmıştır. Uygulamada kullanılan prototip sistemin gerçek bir endüstriyel sistemle benzerlik taşıdığını savunmuştur [14].

Cordes ve arkadaşları, 2019 yılında yapmış olduğu bir çalışmada büyük havacılık parçalarının üretiminde kullanılan Endüstriyel robotların frezelemedeki tırlama (chatter) davranışları incelenmişlerdir. Bir mafsal üzerine eklenen bir takım aleti ile modelleme yapılmıştır. Deneysel olarak yapılan tahmin kararlılık çizelgeleri alüminyum ve titanyum parçalarının üzerinde onaylanmıştır. Endüstriyel robotlar, düşük frekanslı yapısal modlara sahiptir. Poz-bağımlı bütün düşük frekanslı yapısal moda sahip endüstriyel robotların poz

bağımlı olanları, yüksek hızdaki iş milindeki işlenme prosesi tarafından sönümlenirler. Sadece poz bağımsız iş mili modları, yüksek hızlı frezelemede tırlamaya neden olur, bu nedenle yüksek malzeme kaldırma oranları, analitik olarak öngörülen sabit kesim derinliği ve mil hızları seçilerek elde edilebilir [15].

Namazi, Altintas ve arkadaşları 2007 yılında, takım tutucu-iş mili ara yüz dinamiğinin modellenmesi ve tanımlanması konusunda araştırma yapmışlardır. Takım tutucusunda ve iş mili arayüzünde temas sertliği ve sönümleme modellemesi üzerinde çalışmışlardır. Yüksek hızlı frezelemedeki tırlama (chatter) titreşimlerinin çoğu, takım tutucu milindeki esnek bağlantılar ve takım tutucu nedeniyle ortaya çıktığını belirtmişlerdir. Timoshenko ışın elemanları tarafından modellenmiştir. Tutucu mili konik teması, düzgün dağılmış translasyon ve rotasyon yayları ile modellenmiştir. Mil düzeneğinin deneysel olarak ölçülen ve tahmin edilen frekans tepkisi arasındaki hatayı yaylar ile minimize etmek amaçlanır [16].

Comak, Altintas 2017 yılında yaptığı çalışmasında, Torna tezgâhları karmaşık kinematik ve takım-iş parçası etkileşimleri nedeniyle, dönüş frezeleme işlemleri temel olarak pahalı işleme denemelerine ve deneyimlerine dayanarak gerçekleştirilir. Tornalama işlemlerinde gerekli olan kesme kuvvetlerini, tork ve gücü tahmin etmek üzerine çalışma yapılmıştır. Frezeleme işlemi, makinenin üç lineer (x, y, z) ve iki döner tahrik motorunu içerir. Elde edilen besleme vektörü, sürücülerin doğrusal hızlarının ve açısal hızlarının bir fonksiyonu olarak modellenmiştir. Üretken takım ve iş parçası iş mili hızlarının tanımlanması, takımların talaş yükü sınırını ve dönüş frezeleme takım tezgâhlarının tork gücü sınırlarını kullanarak gösterilmiştir [17].

Aslan, Altintas 2018 yılında yaptığı çalışmasında, iş mili hız kontrolünü sağlayan cihazın transfer fonksiyonu kontrol yasası parametrelerini okuyarak ve harici bir bilgisayarı kullanarak Frekans Tepkisi İşlevini (FRF) CNC ile gerçek zamanlı olarak ölçerek yapar. Ölçülen FRF'den sert gövde bazlı FRF'nin, esneklikleri içeren hız kontrol cihazı, iş mili tahrikinin yapısal modları tanımlanır. Zorlanmış titreşim bileşenlerini çıkardıktan sonra, verilen mevcut komutların Fourier Spektrumu tırlama (chatter) frekansı, CNC tarafından gerçek zamanlı olarak tespit edilmektedir ve freze testlerinde doğrulanmıştır [18].

Yang, Aslan, Altintas 2018 yılında, yapmış olduğu çalışmasında; beslemeyi takım yolları boyunca programlayarak kesme yükünün neden olduğu takım ucu konumunu ve beş eksenli

CNC tezgahlarını takım oryantasyon hatalarını istenen tolerans limiti dahilinde tutmak için bir model önermektedir. Önerilen algoritma beş eksenli ticari CNC tezgahında kavisli bir takım yolunun frezelenmesi ile deneysel olarak onaylanmış ve işleme süresinin arttırmadan parça işlemenin doğruluğunu önemli ölçüde artırabileceği gösterilmiştir [19].

Sistemin mekanik kısmında temel olarak100x100x100 mm boyutlarında 5mm/tur adımlı 3 adet vida, kaplin, vidalı mil, somun, indiksiyonlu mil, doğrusal rulman kullanılmıştır. Böylece vida boyları büyütüldüğü takdirde gerçek uygulamalarda da direk kullanılabilecek 3 eksenli bir CNC prototipi oluşturulmuştur.

Kullanmış olduğumuz Schneider marka 1,3 kW, BMH 100mm key IP54 st 16 angular servo motorların içinde yer alan enkoder ile hız, moment ve pozisyon kontrolü yapılabilmektedir. Bu kontroller için gerekli sinyaller konum algılayıcısının çözünürlüğü ile orantılı olarak değişmektedir. PLC tarafından sürücüye gönderilen komutlar dâhilinde hareket gerçekleştirilmektedir. Sistemde Schneider marka Lexium 32, Sercos sürücü ve SERCOS haberleşme protokolü kullanılmaktadır.

Sürücü kontrolleri için hazırlanan SoMove program arayüzü ile dijital/analog giriş ve çıkış adresleri, cihaz kimliği tanımları ve motor parametreleri optimizasyonları yapılmıştır.

Schneider marka Motion Controller Modicon LMC078 PLC üzerinden motorların kontrolü sağlanabilmesi için SoMachine program arayüzü kullanılmıştır. Sistemde yer alan tüm ekipmanlar bu program vasıtası ile kontrol edilmiştir.

Servo motor içerinde yer alan dahili konum algılayıcıdan gelen bilgilerin doğruluğunu kontrol edebilmek ve sistemi adaptif hale getirebilmek için, mil uçlarına takılan harici olarak Absolute Konum algılayıcı Multi Turn XCC3510PS84CB konum algılayıcı kullanılmıştır. Harici konum algılayıcıların PLC ile haberleşmesinde CanOpen haberleşme protokolü kullanılmıştır. CanOpen haberleşme protokolü, SERCOS haberleşme protokolünden bağımsız olarak bilgi akışı sağladığından, sistemi haberleşme hatalarına karşı daha güvenli hale getirilmiştir [20].

Sisteme ilave olarak takılan GTO touch HMI panel,12.1" panel kullanılarak el ile kontrol de gerçekleştirilebilmiştir. Panel programlaması için SoMachine ek yazılımı olan Vijeo

programı kullanılmıştır. PLC ve çevre ekipmanlarının çalıştırılabilmesi için Schneider marka güç kaynağı ve uyumlu bağlantı kabloları kullanılmıştır. 3 eksenli CNC endüstriyel amaçlar için kullanılmasına rağmen, 3 eksen mümkün olduğunca eksene kolayca dönüştürülebilir.

Her 3 ekseni kontrol eden 3 farklı motor, aynı anda farklı hız ve ivme ile çalıştırılır ve daha yumuşak ve pürüzsüz bir işleme lineer ve nonlineer interpolarasyon ile sağlanır. Bu şekilde, yay ve konik dereceli köşe şekillendirmede pürüzlülük ve merdiven basamakları en aza indirgenmiştir. Bütün bu işlemlere ek olarak; motorların akım değerleri anlık olarak okunmuş, elde edilen akım verileri kullanılarak, en uygun akım değerinde çalıştırılması için anlık p faktörü değişimi ve en uygun sürede çalışabilmesi için anlık hız değişimi sağlanmıştır.

Bu çalışmadaki CNC makinasının kontrolünde, harici CAN BUS protokolü vasıtasıyla haberleşebilen harici konum algılayıcı ve servo motora akuple edilmiş haldeki SERCOS protokolü ile haberleşen dâhili enkoder de aynı anda senkronize haberleştirilmiştir.

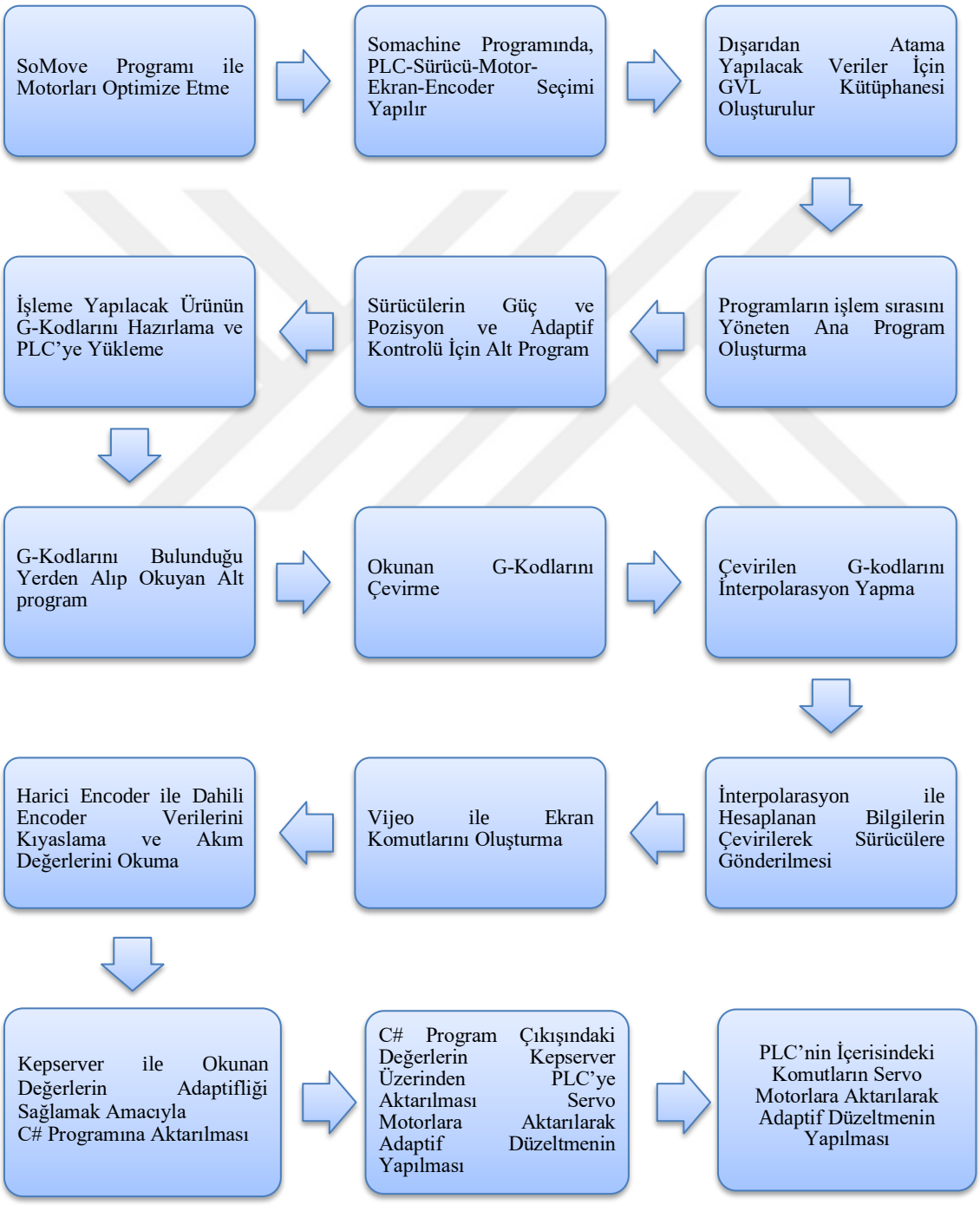
Haberleşmede SoMachine programı ile SERCOS haberleşme protokolü ve CAN BUS haberleşme protokolü birlikte kullanılmıştır. Böylece her iki protokol ile aynı anda haberleşme sağlanmış ve hatalara karşı daha güvenli ve dayanıklı hale getirilmiş ve CNC'nin çalışmasında hata azaltılmıştır. Ayrıca konum algılayıcılar arasındaki veri kıyaslaması neticesinde, mil boşluğundan kaynaklanan farkın tespit edilmesi halinde, oluşan fark kadar eksenin ilerlemesi sağlanarak, oluşabilecek boşluğun telafisi sağlanmıştır.

Sistemin güvenliğinin daha çok arttırılması, PLC üzerinden sürücü ve diğer ekipmanlarla haberleşme sağlanabilmesi, bilgi okuyup komut gönderebilmek için KEPServerEX OPC programı kullanılarak veri alışverişi sağlanmıştır. OPC (planlama ve kontrol işlemleri) komutları ile haberleşme ve hareket kontrolü gerçekleştirilebilmektedir. OPC gerçek zamanlı verinin iletişimi için oluşturulmuş bir standart olup bilgisayar ile harici cihazlar arasında haberleşme için oluşturulmuş protokoldür [1].

Kontrol programı 64 ekseni destekleyecek şekilde hazırlanmış olmasına rağmen 3 temel eksendeki kontrolün eğitim için yeterli olduğu düşünülerek çalışmalar 3 eksende

denenmiştir. Üstelik bir servo motorun adaptif kontrolü gelecekteki profesyonel CNC makinalarının tasarımına alt yapı sağlanabilecektir.

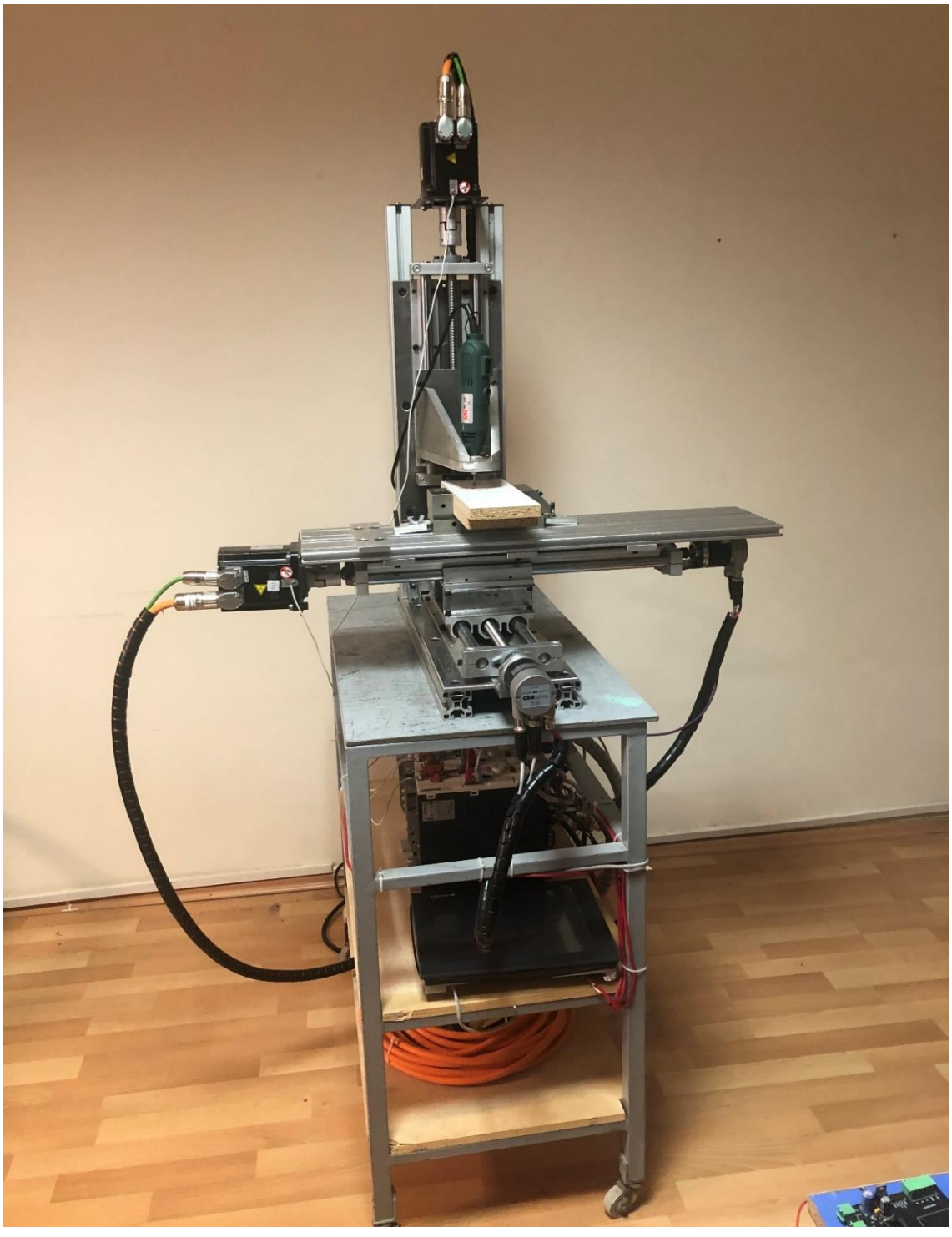
Üç eksenli CNC'nin tasarımında ilk olarak donanımı tamamlanmış ve daha sonra programlama aşamasına geçilmiştir. Programlama aşamasında izlenen adımlar Şekil 1'de verilmiştir.



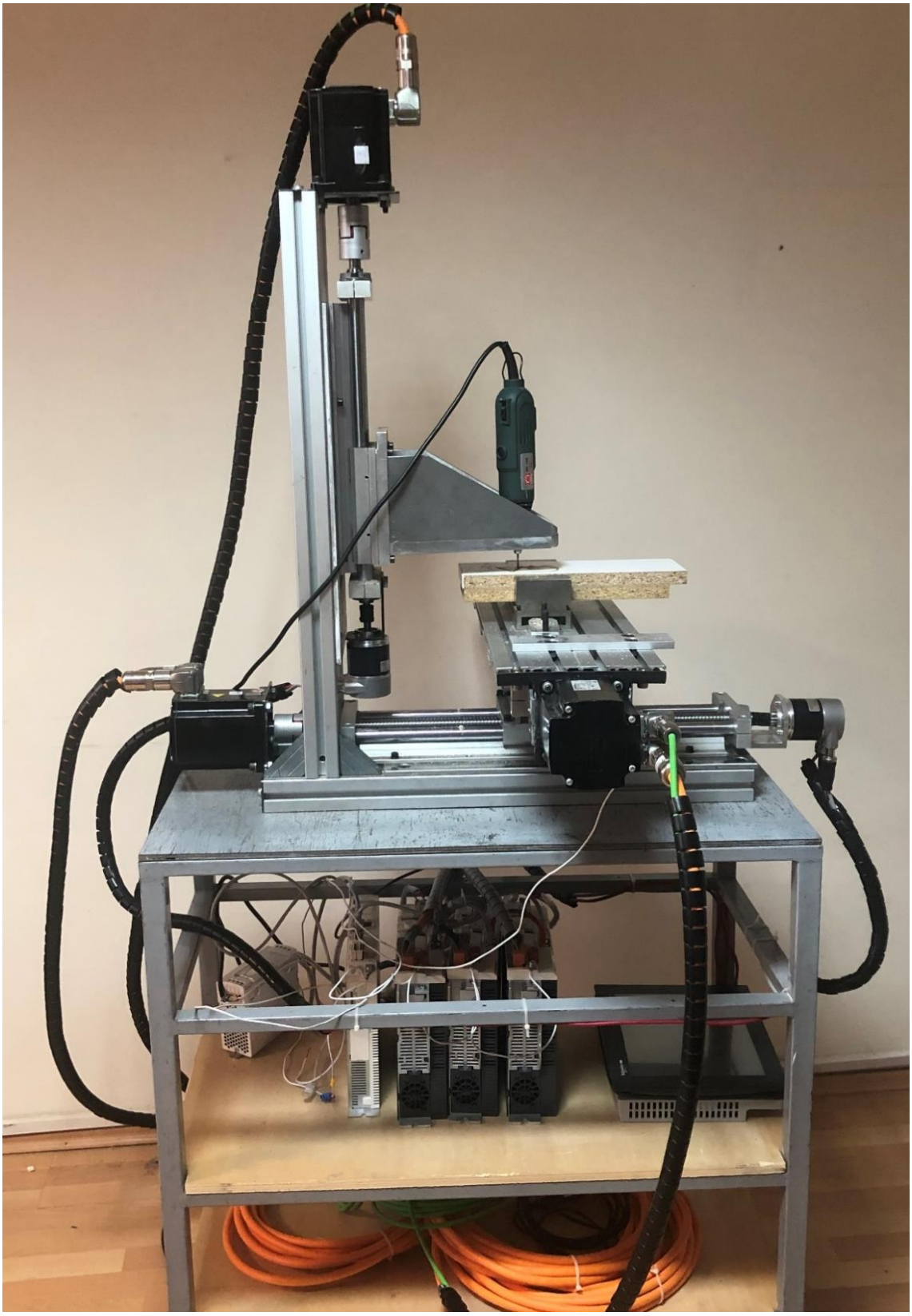
Şekil 1.1. Program işlem basamakları

2. ADAPTİF KONTROLLÜ CNC'NİN MEKANİK TASARIMI

Hazırlanan adaptif kontrollü 3 eksenli bir CNC makinesinin prototipinin tasarımında, Schneider marka servo motorlar akuple edilmiştir. Oluşturulan 3 eksen CNC makinasının görüntüsü Resim 2.1'de ve Resim 2.2 'de verilmiştir.

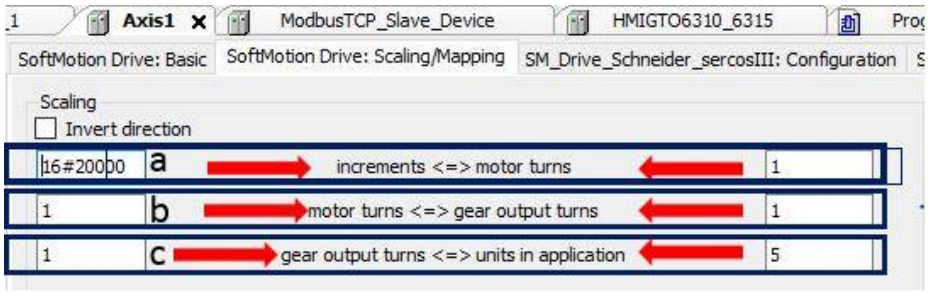


Resim 2.1 3 eksenli CNC prototipi karşıdan görünüşü



Resim 2.2. 3 eksenli CNC prototipi yandan görünüşü

Prototipte 100x100x100 mm boyutlarında 5mm/tur adımlı 3 adet vida kullanılmıştır. Bu vidalara 3 adet 480V, 1.3 kW, 4000 rpm, 3.4 Nm etiket değerlerine sahip bu servo motorlar akuple edilerek, yine 3 adet Schineider marka Lexium sürücü ve bir adet LMC78 PLC ile sürülmüştür. Motor mil uçlarında Multi Turn XCC3510PS84CB konum algılayıcı ile pozisyon bilgisi ölçülmüştür. Resim 2.3'de pozisyon algılamada kullanılan program arayüzü görülmektedir.



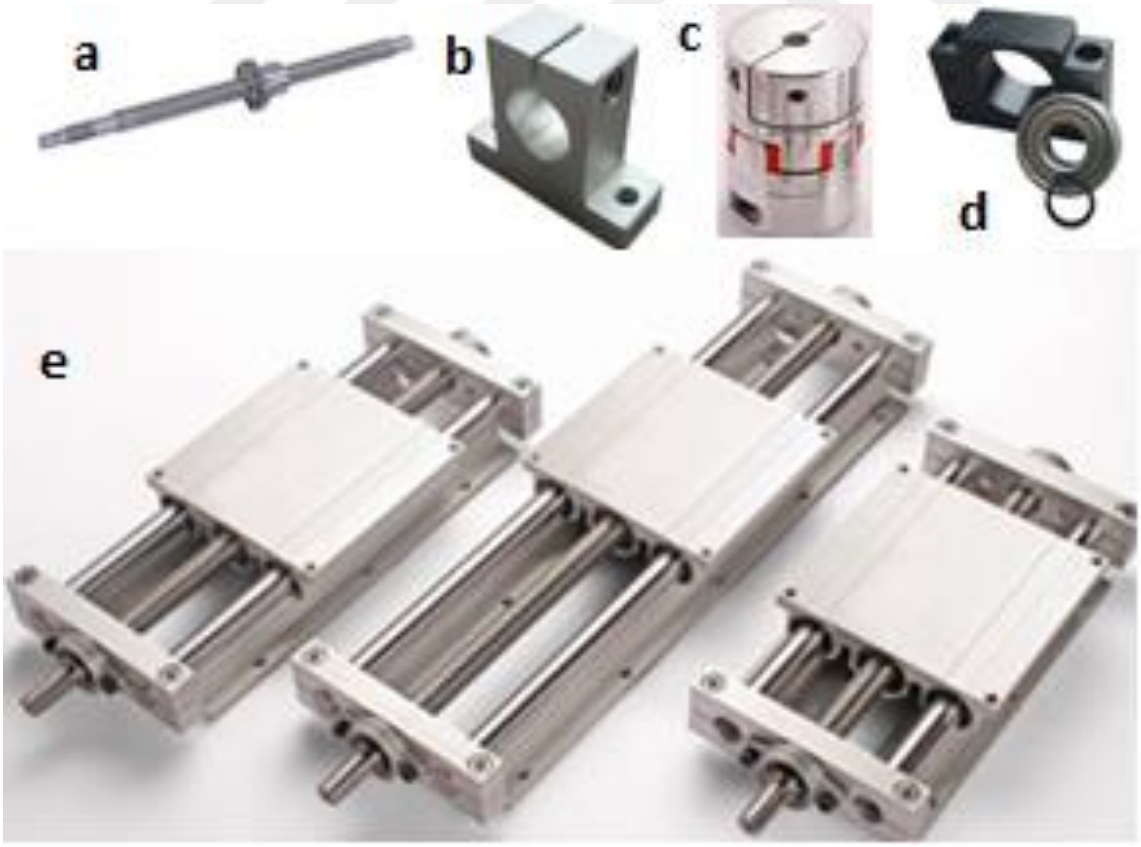
Resim 2.3. Servo motor kontrol arayüzü

Resim 2.3'de; a ile gösterilen 16# işareti dahili konum algılayıcıdan bir turda ölçülecek pozisyon sayısının 16 bit olduğunu, sonraki gelen 20000 rakamı ise bu sayının 20000 ile sınırlandırıldığını ifade etmektedir. Bunun yanında 16#Sistemde ve programda servo motorun kendi içerisinde yer alan konum algılayıcıdan gelen bilgiler ile yine PLC ve motor sürücüsünün haberleşme protokolü olan SERCOS'dan gelen pozisyon bilgileri ayrı ayrı ölçülmektedir. Aynı şeklin karşısındaki 1 rakamı bu sayının bir turda gerçekleştirileceğini belirtmektedir. Resim 2.3'de b ile gösterilen kutucuktaki 1 motor turu ile vidanın dönme sayısı oluşacağı belirlenmektedir. Oluşturulan sistemde ayrıca bir dişli kutusu kullanılmadığı için karşısındaki kutucuktaki sayı da 1 olarak verilmiştir. Aynı şekildeki c ile gösterilen kutucuktaki 1 rakamı ise 1 vida turunda kaç mm hareket oluştuğu belirlenmektedir. Tasarlanan sistemde vida adımı 5 mm olduğundan sağ taraftaki rakam da 5 olarak girilmiştir.

Resim 2.4'de projede kullanılan servo motor ve konum algılayıcı görseli bulunmaktadır. Resim 2.5'de kullanılan mekanik parçalar gösterilmiştir. Böylece vida boyları büyütüldüğü takdirde gerçek uygulamalarda da direk kullanılabilecek 3 eksenli bir CNC prototipi oluşturulmuştur.



Resim 2.4.(a) Servo-motor (b) External konum algılayıcı



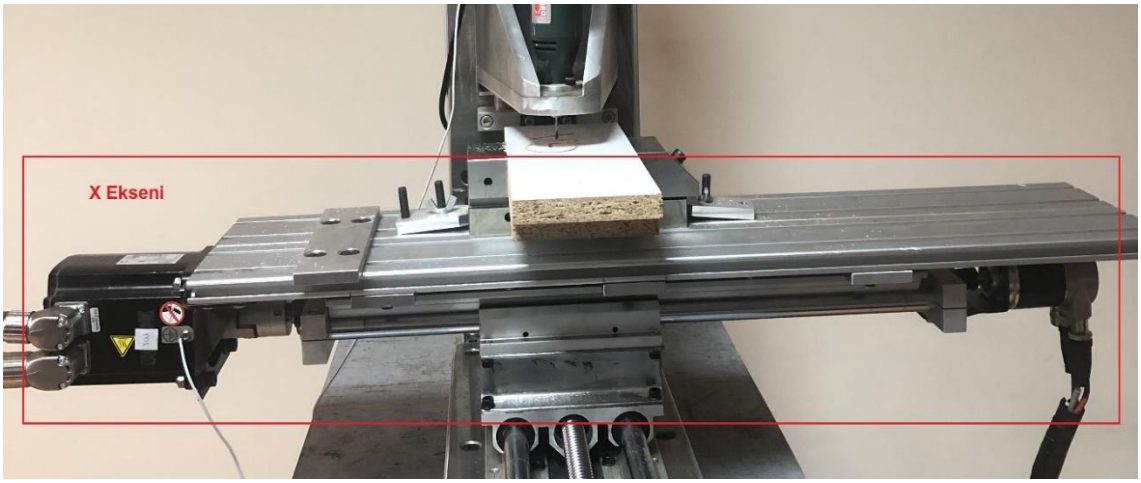
Resim 2.5 (a)Vidalı mil (b)Mil tutucu (c)Kaplin (d)Rulman (e)Bir eksen için kızak (f)Alt destekli mil sac (g) Lineer rulman (h) Sigma profil

Resim-2.6'da bilgisayardan bağımsız çalışma yapabilmek için kullanılan HMI panel yer almaktadır.

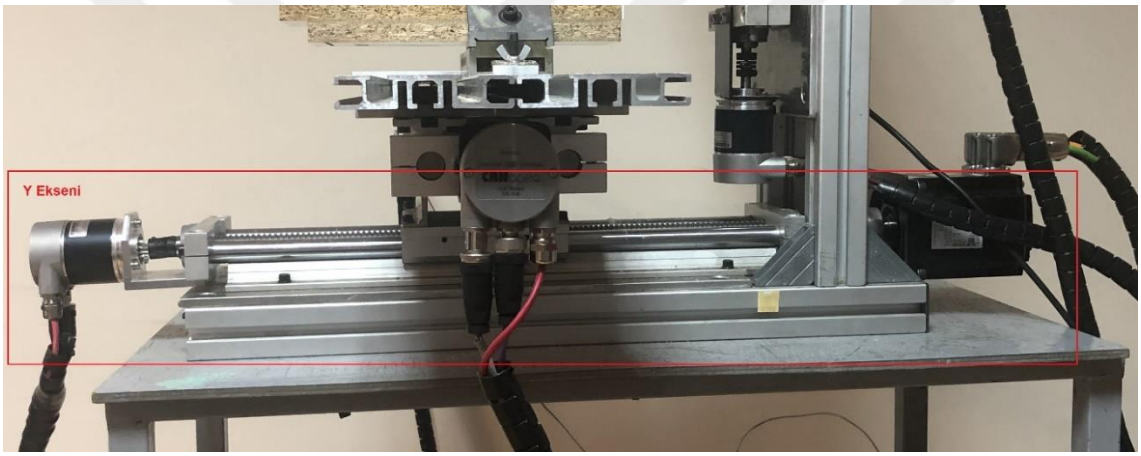


Resim 2.6. CNC HMI panel görüntüsü

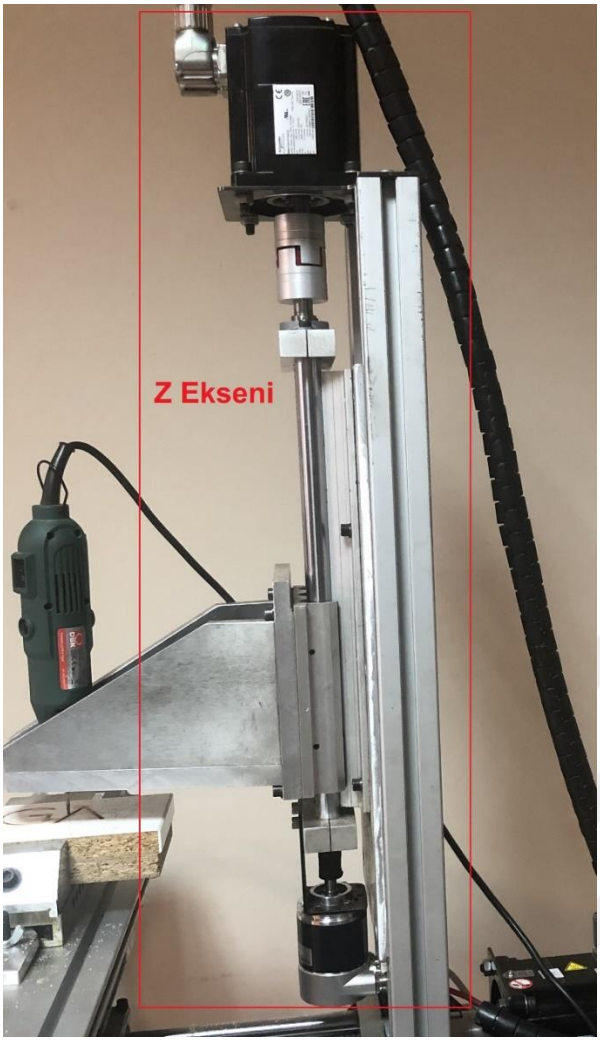
Resim 2.7.'de se CNC ye ait X ekseni, Resim 2.8.'de se CNC ye ait Y ekseni Resim 2.9.'da se CNC ye ait Z ekseni ve Resim 2.10.'de İse CNC ye ait kontrol ünitesi bulunmaktadır. Kontrol ünitesi LMC078 PLC, 3 adet LXM 32 sürücü ve güç kaynağından oluşmaktadır.



Resim 2.7. CNC X ekseni



Resim 2.8. CNC Y ekseni

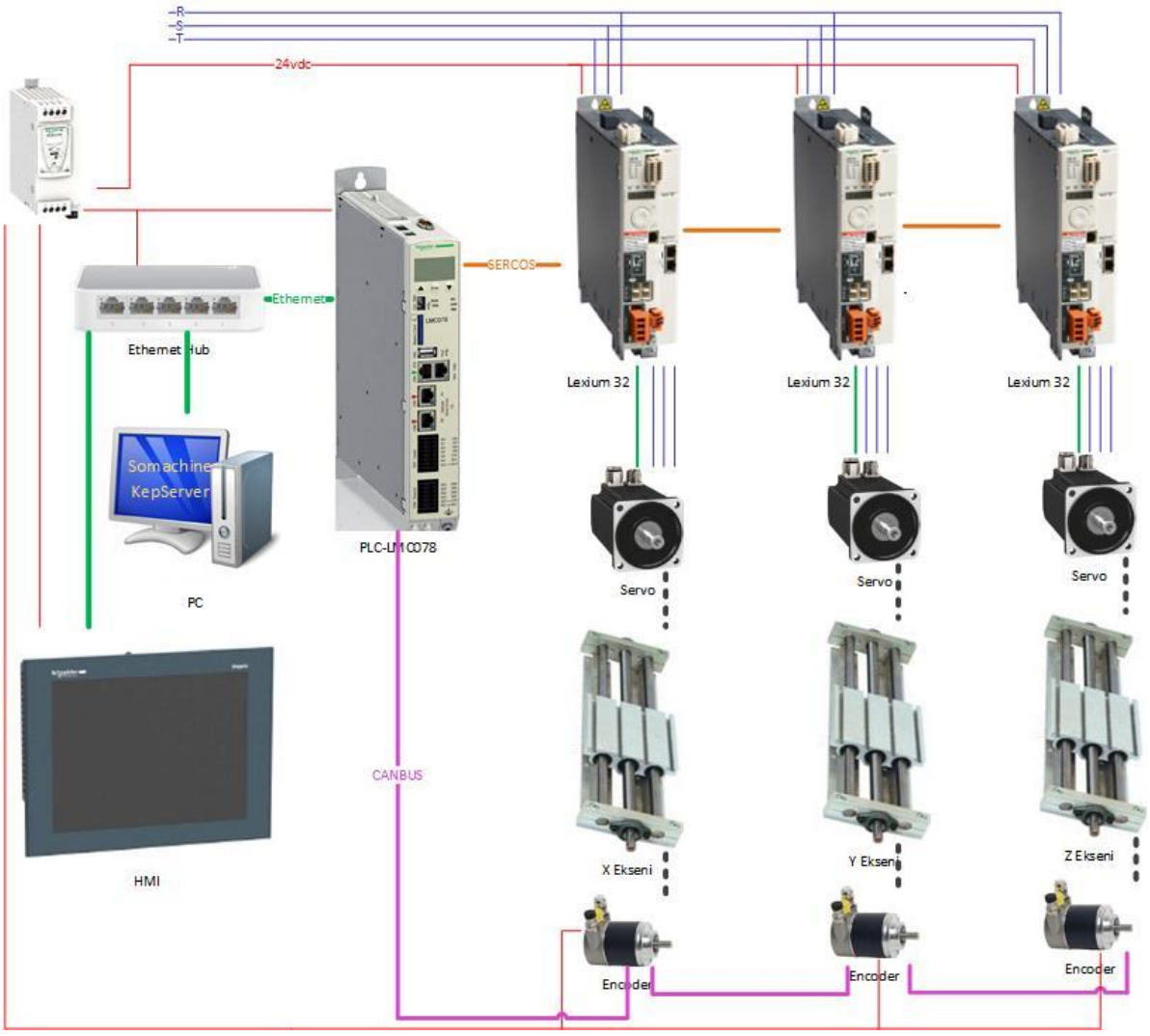


Resim 2.9. CNC Z ekseni



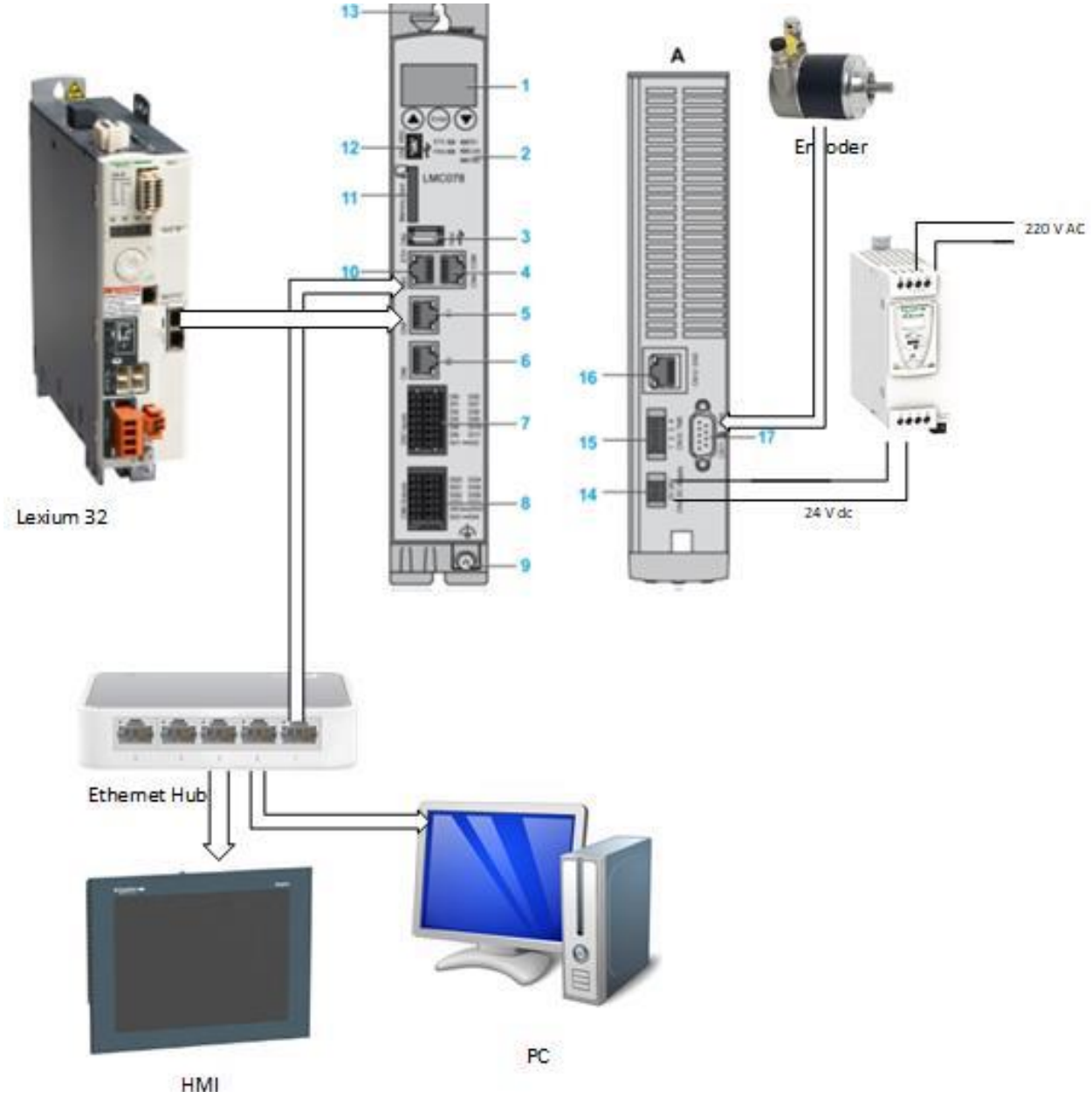
Resim 2.10. CNC kontrol üniteleri

Hazırlanan 3 eksen CNC donanımında yer alan 1 adet PLC, 3 adet sürücü, 3 adet servo motor, 3 adet konum algılayıcı, HMI ekran, PC, Ethernet anahtar, güç kaynağı bağlantı şekilleri Resim 2.11'de sunulmuştur.



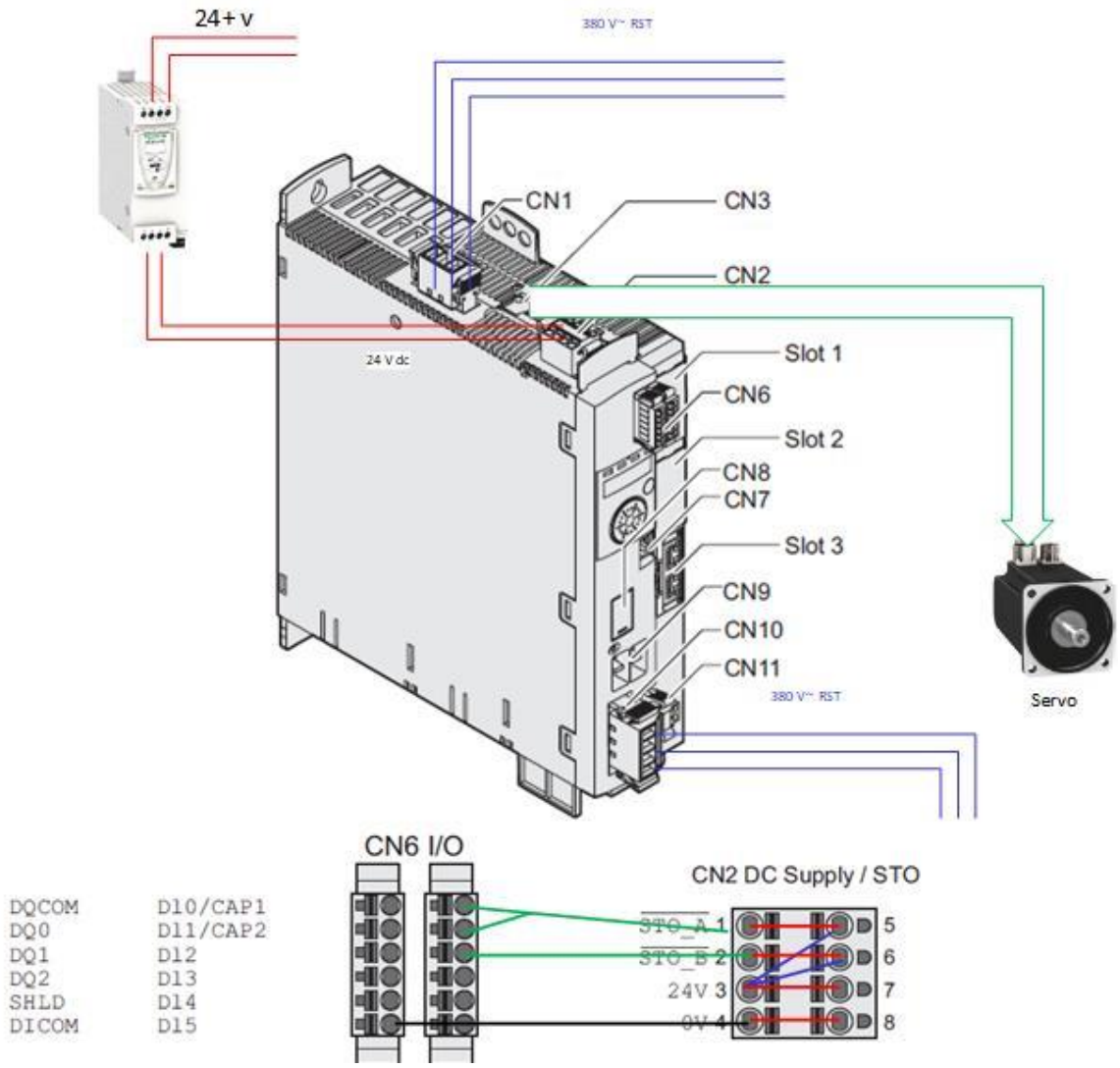
Resim 2.11. 3 Eksenli CNC bağlantı şeması

PLC üzerinde yapılan detaylı kablola bağlantısının gösterimi Resim 2.12'de yer almaktadır. Sürücü üzerinde yapılan detaylı kablola bağlantısının gösterimi Resim 2.13'de yer almaktadır.



Resim 2.12. PLC, konum algılayıcı, bilgisayar, ekran bağlantısı

Resim 2.12'de gösterilen detaylı kablo bağlantısına göre LMC078 PLC'nin SECOS 5 numaralı girişinden Ethernet kablosu ile Lexium 32 sürücüsünün SERCOS girişine bağlantı yapılır. Lexium 32 sürücüsünün diğer SERCOS yuvasından diğer Lexium 32 sürücüsünün SECOS yuvasına bağlantı yapılır, en sondaki sürücünün açıkta kalan SERCOS yuvasından PLC üzerindeki 2. SERCOS yuvasına bağlantı yapılır. PLC'nin 10 numaralı Ethernet yuvasından HUB'a bağlantı yapılır. HUB'dan ise HMI ve PCnn Ethernet yuvalarına giriş yapılır. Ayrıca LMC078'in arka yüzeyinde bulunan CanBus girişine 1. encoder doğrudan bağlanır.



Resim 2.13. Servo motor ve sürücü bağlantısı

3. CNC'NİN DENETLEME PROGRAMLAMALARI

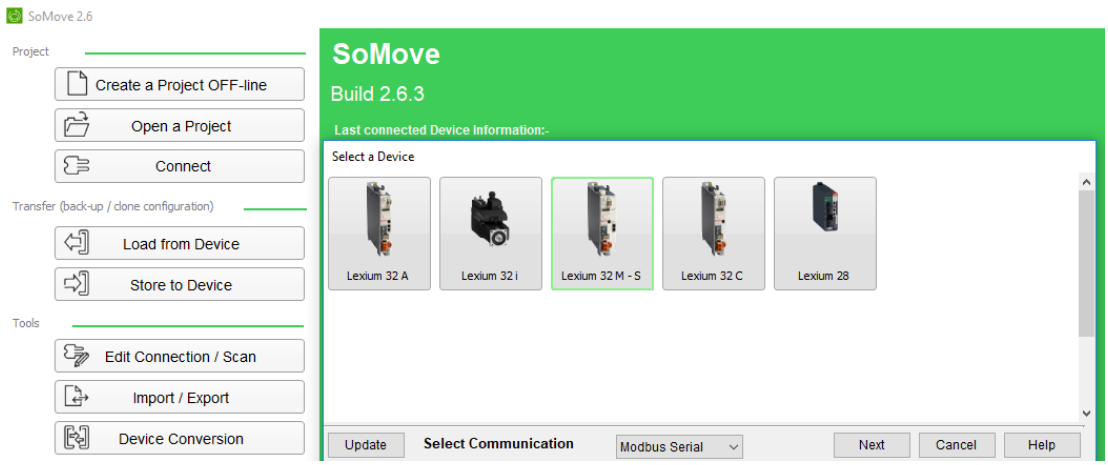
Bu bölümde projede kullanılan programların kuruluşu, program içerisinde yapılan programlamaya, programlama komutlarına, haberleşme ayarlarına yer verilmiştir. SoMove programının kurulumuna ve programlanmasına, ikinci olarak So-Machine programının kurulmasına, yapılan programlama, komutlar, cihaz tanımla ve haberleşme ayarlarına, dördüncü olarak Vijeo programında ekran ayarlarının yapılmasına ve son olarak KEPServerEX OPC programının kurulum ve kullanım bilgilerine yer verilmiştir.

3.1. SoMove Programı ile Sürücü Programlama

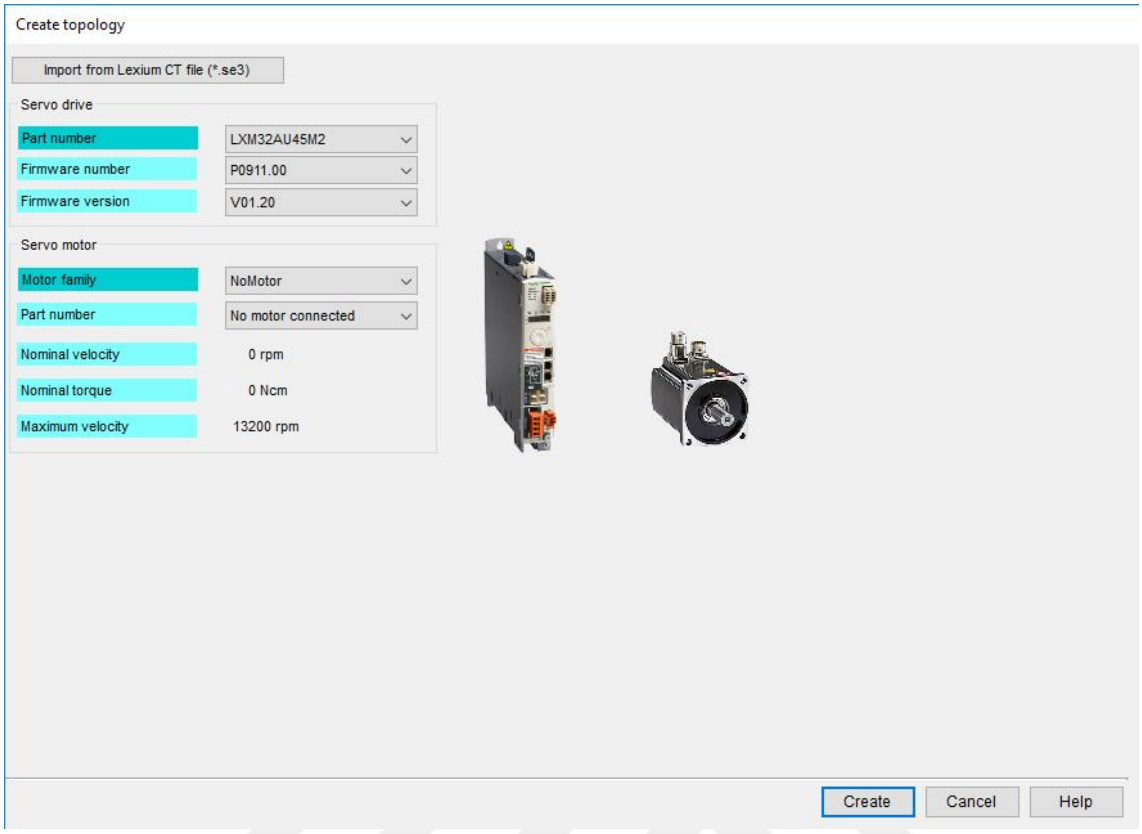
Sürücülerin ve motor parametrelerinin optimizasyonu ve haberleşme ayarları için SoMove programı kullanılmaktadır. SoMove programının kurulum aşamaları içeren bilgiler EK-1'de sunulmuştur. Aşağıda sıralanan işlemler sırası ile her üç sürücü için farklı adresler atanarak

3.1.1. SoMove programında cihaz eklenmesi

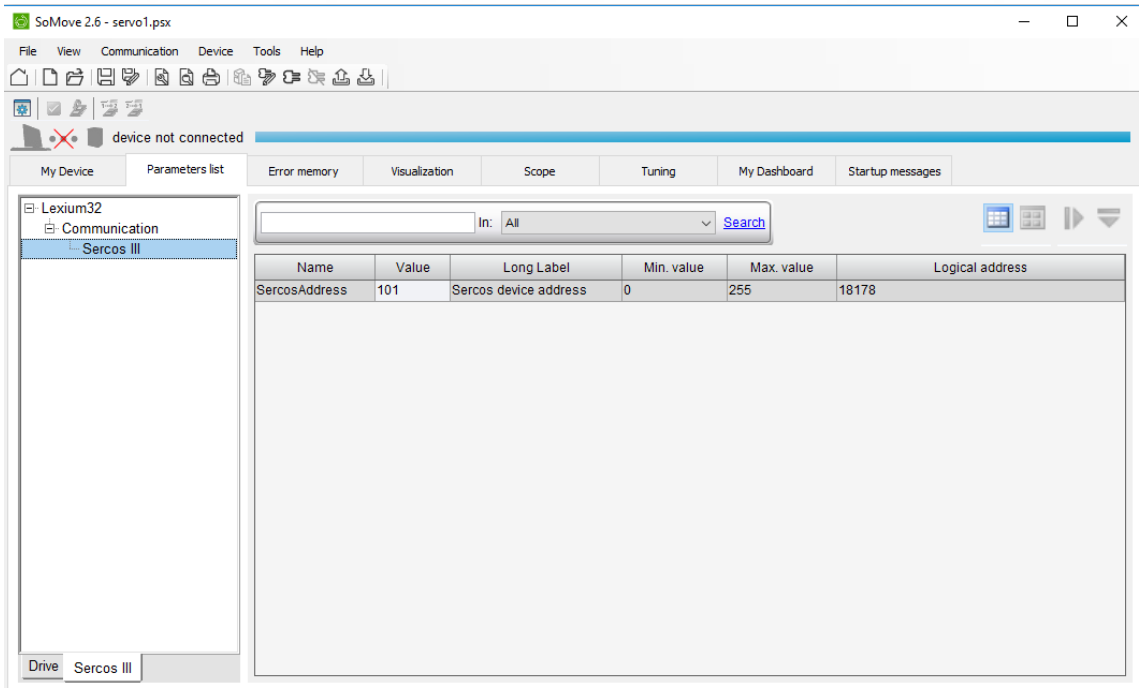
SoMove program başlatma ikonuna tıklanarak program açılır. Resim 3.1'de bulunan sayfada kullanmış olduğumuz sürücü modeli ve haberleşme protokolü seçilerek next butonuna basılır. Daha sonra Resim 3.2'de görülen ekranda sürücünün ve kullanmış olduğumuz servo motorun seri numaraları dikkate alınarak seçimleri yapılır.



Resim 3.1 Sürücü seçimi



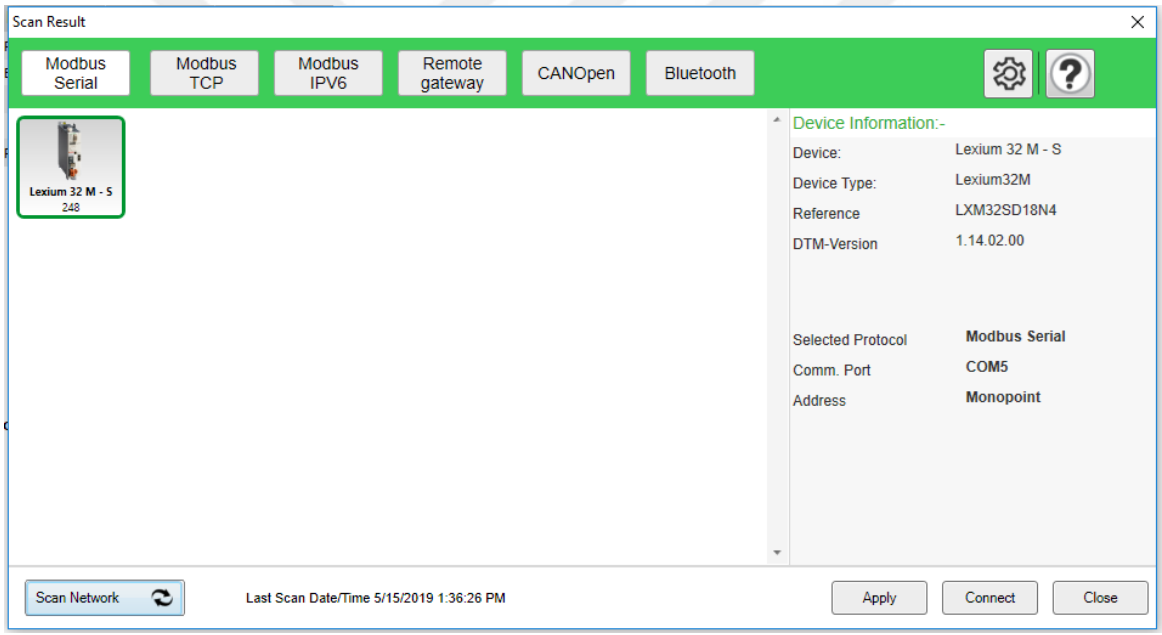
Resim 3.2 Sürücü ve motor seçimi



Resim 3.3 Sürücüye adres atama

Resim 3.3'de görülen ekranda Parameter List sekmesi ve aşağıda yer alan Sercos III sekmesi tıklanır ve sürücünün adreslemesi 0-255 arasında bir sayı girilerek yapılır. Bu projede 1. sürücü için adresleme 101 olarak yapılmıştır.

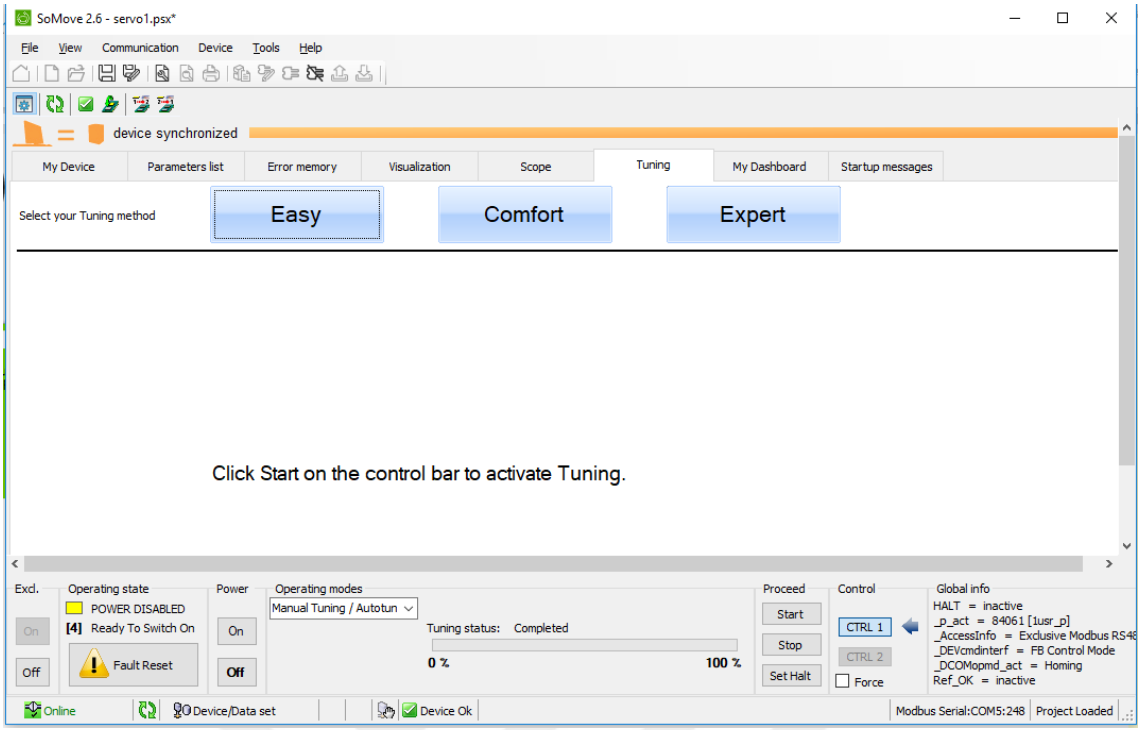
Sürücüye bu bilgilerin yüklenebilmesi için bağlantı yapılması gerekmektedir. Bunun için Schneider TCSMCNAM3M002P haberleşme kablosu kullanılmıştır. Kablonun USB ucu bilgisayara Ethernet ucu ise sürücü üzerinde yer alan CN7 MODBUS girişine takılmıştır. Daha sonra Resim 3.3'de edit connection ikonu tıklanmıştır. Resim 3.4'de gösterilen ekranda Modbus Serial ve ardından scan network butonu tıklanır ve burada kullanmış olduğumuz sürücünün görülmesi gereklidir. Sürücü seçilerek connect butonuna basılır. Çıkan uyarı ekranında Alt+F tuşuna basılarak bağlantı gerçekleştirilmiştir.



Resim 3.4. Sürücü ile haberleşme

3.1.2. SoMove programında cihaz parametreleri optimizasyonu

Resim 3.5'de gösterilen ekranda Tuning sekmesi ve ardından easy seçeneği tıklanarak motor paremetrelerinin optizasyonu sağlanır. Burada on/off butonu kullanılarak motorlara enerji verilmesi sağlanmıştır. Operating modes seçeneğinden homing ayarları, Jog, Manual tuning yapılabilmektedir.



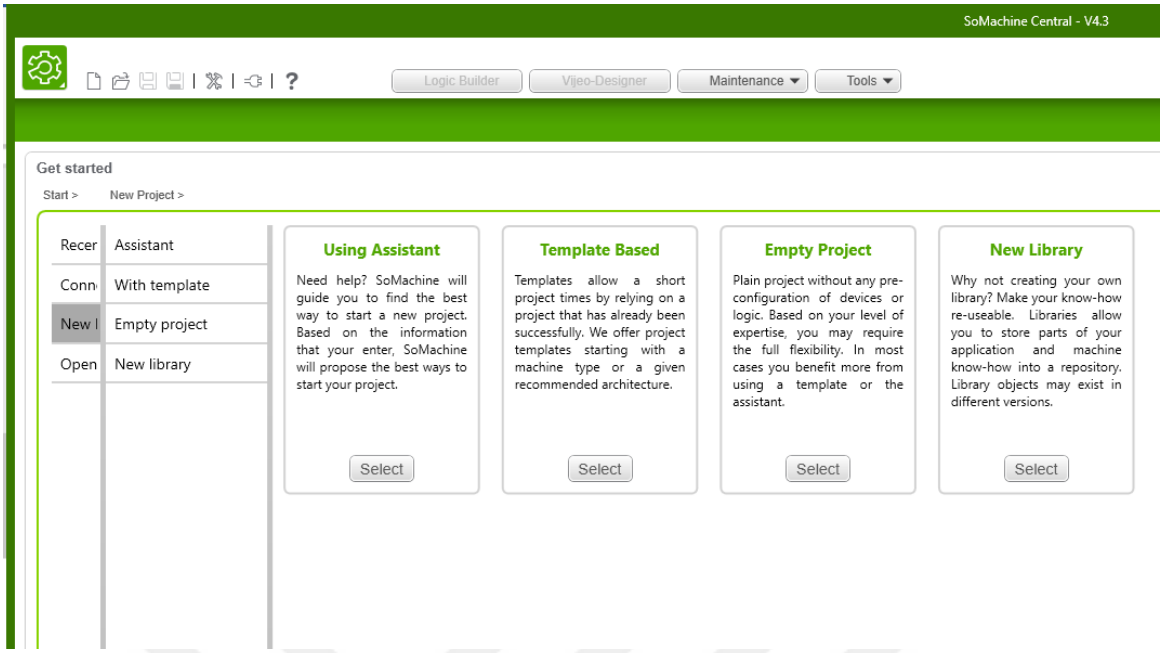
Resim 3.5. Motor paremetre optimizasyon

3.2. SoMachine Programının Kullanılması

Projede yer alan çevre ekipmanların LMC078 PLC üzerinden kontrolü sağlanabilmesi için SoMachine program arayüzü kullanılmış, sistemde yer alan tüm ekipmanlar bu program vasıtası ile tanınmış ve kontrol edilmiştir. SoMachine programının kurulum aşamasını içeren bilgiler Ek-2'de sunulmuştur.

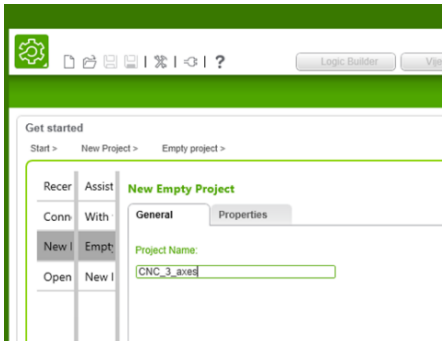
3.2.1. SoMachine programında cihazların eklenmesi

SoMachine V4.3 ikonuna tıklayarak programı açıyoruz. Resim 3.6'da görüldüğü üzere New Project ardından Empty Project seçenekleri tıklanır.

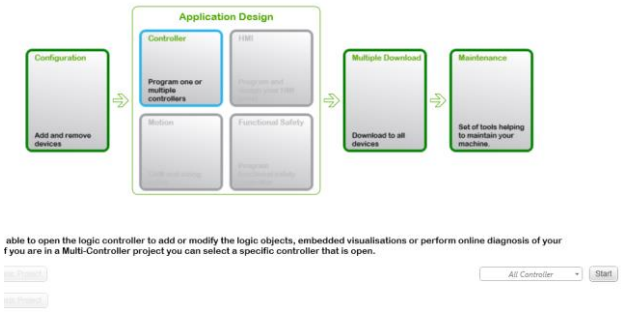


Resim 3.6 SoMachine programlama-1

Resim 3.7.a'da görüldüğü proje adı yazılıp start butonuna basılır. Ardından Resim 3.7.b'deki gibi çıkan ekranda Program one or multiple controllers seçeneği tıklanır daha sonra All controllers seçilerek start butonuna basılır.



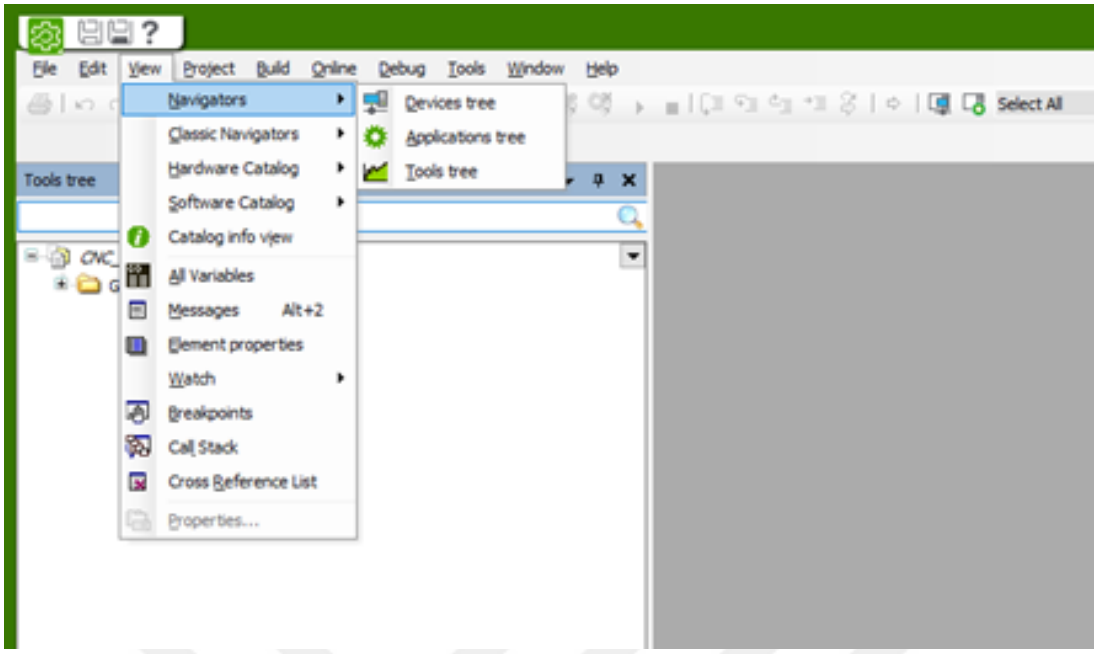
(a)



(b)

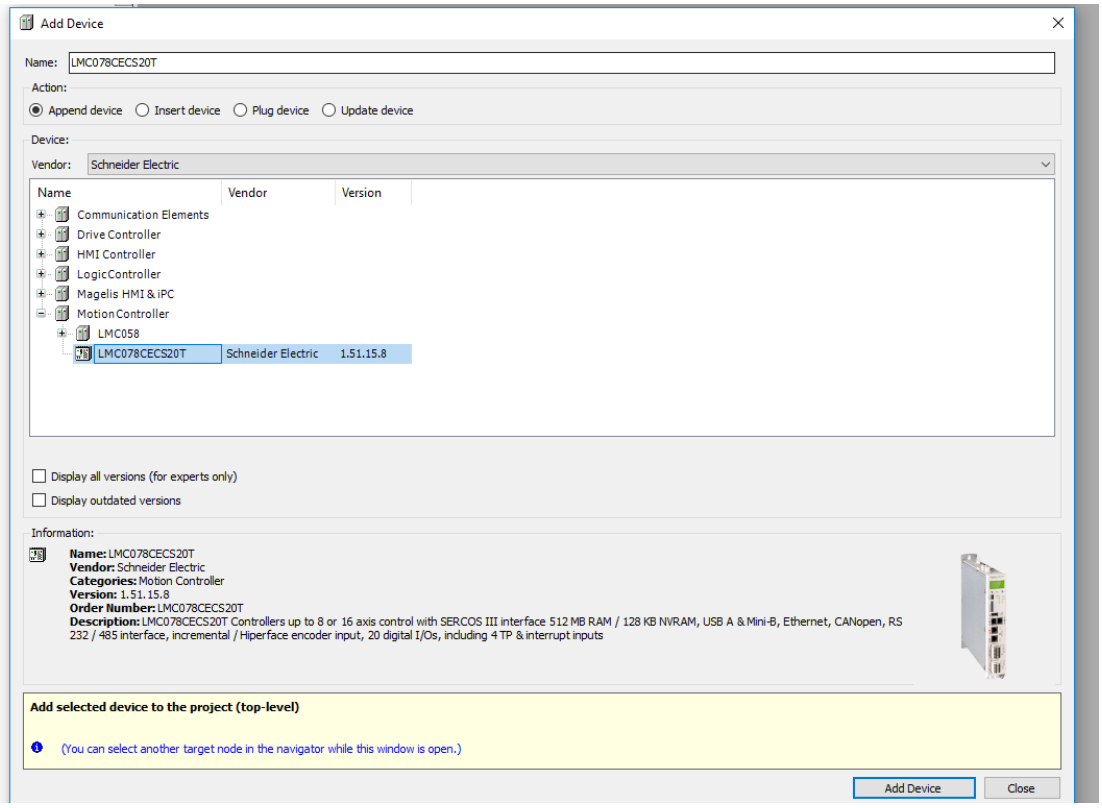
Resim 3.7.(a) SoMachine programlama-2 (b) SoMachine programlama-3

Program sayfası açıldıktan sonra çıkan ekranda kullanım kolaylık sağlaması açısından Resim 3.8'de gösterildiği üzere View// Navigator menüsünden Device Tree, Aplication Tree, Tools Tree menüleri aktif hale getirilir.



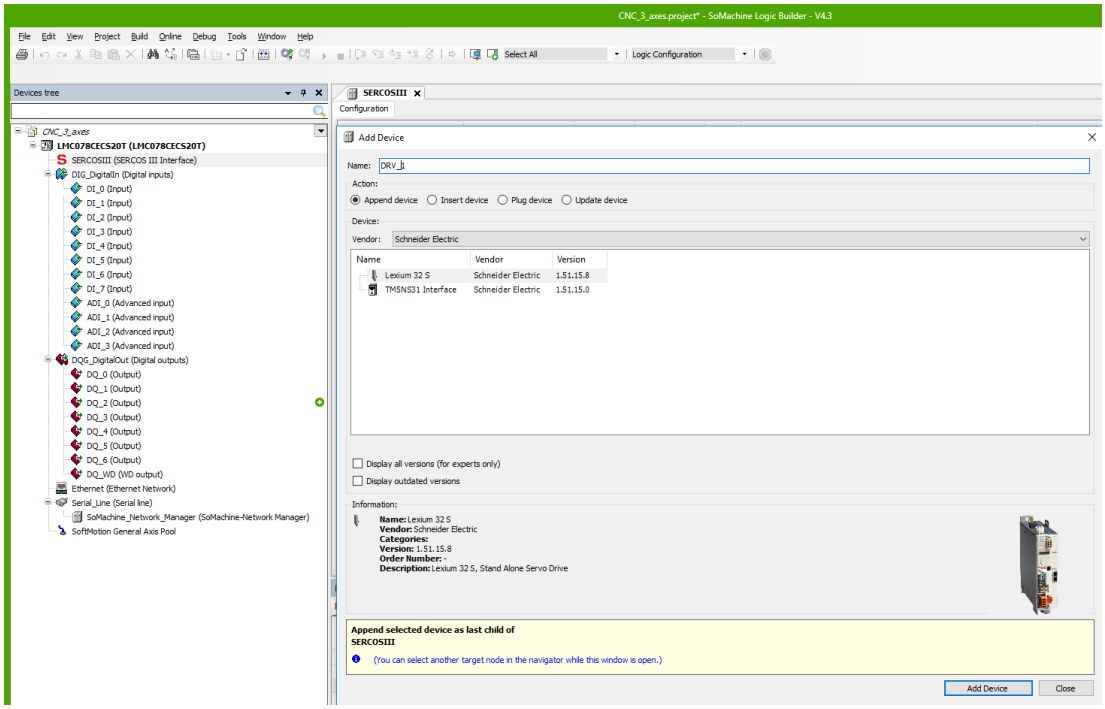
Resim 3.8 SoMachine programlama-4

Resim 3.9'da görüldüğü üzere Device Tree menüsü üzerinde mause sağ tıklandığında add device menüsü seçilir. Add device penceresinden kullanmış olduğumuz PLC seçimi yapılır. Burada projede kullanmış olduğumuz Motion Controller LMC078 seçilmiştir.



Resim 3.9. PLC ekleme

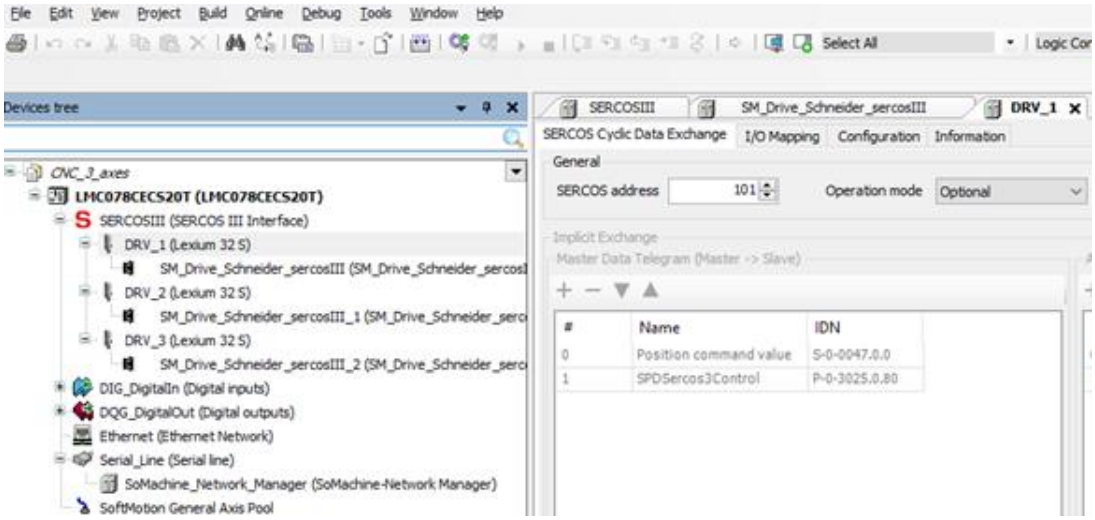
Resim 3.10'da Device Tree Menüsünde eklemiş olduğumuz PLC görülmektedir. Menü içerisinde LMC078CECS20T cihazının alt kısmında SERCOS III seçeneğinin yanında yer alan artı işareti tıklanır ve buradan sürücü seçimi yapılır. Burada LXM32s sürücüsü seçilir. Name kısmına sürücü adı yazılır ve add device butonuna basılarak eklenmiş olur. Sistemde üç adet sürücü olduğu için bu işlem basamağı üç kez tekrarlanmış olup sürücülere DRV_1, DRV_2, DRV_3 isimleri verilmiştir.



Resim 3.10 Sürücü ekleme

Resim 3.11'de görüldüğü üzere, DRV_1 sürücüsünün üzeri tıklandığında SERCOS Cyclic Data Exchange çıkmaktadır. Buraya daha önce SoMove programında belirlediğimiz sürücü adresi tanımlanmalıdır. Daha önce 101 olarak belirlediğimiz adres buraya girilir. Diğer sürücüler için ise 102 ve 103 adres tanımlamaları kullanılmıştır.

Daha sonra sürücülerin altında ekli olarak gelen servo motorlara üzerine tıklanarak isimler verilir. Burada Servo_1, Servo_2, Servo_3 isimleri kullanılmıştır.

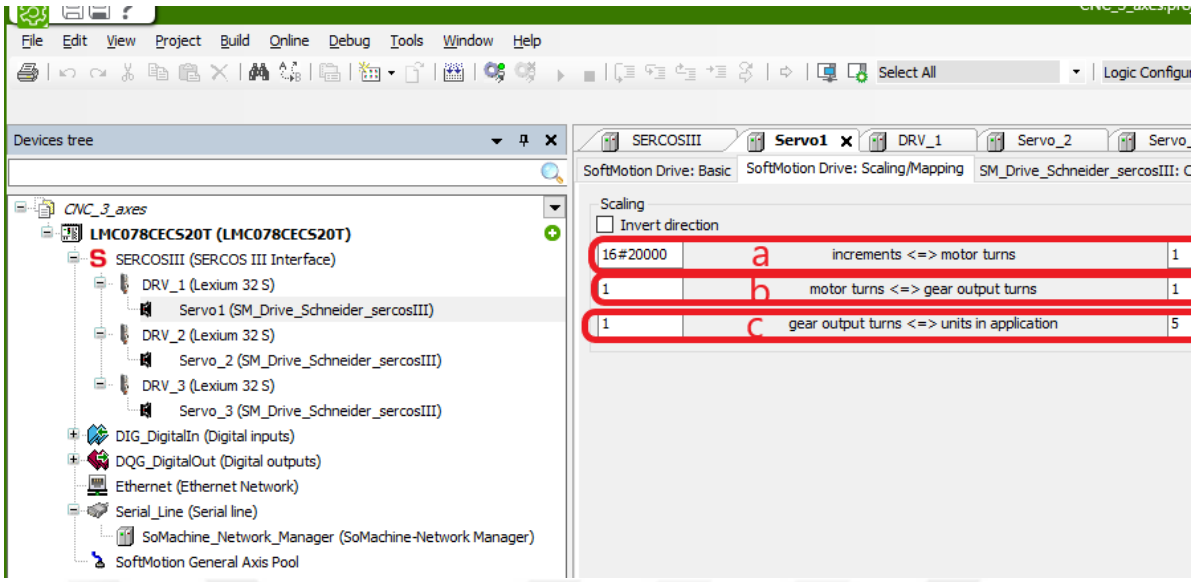


Resim 3.11 Sürücü tanıtma

Resim 3.12'de Her bir motorun tur sayısı, adım sayısı ayarları yapılmıştır. Söz konusu resimde a ile gösterilen 16# işareti dahili konum algılayıcıdan bir turda ölçülecek pozisyon sayısının 16 bit olduğunu sonraki gelen 20000 rakamı ise bu sayının 20000 ile sınırlandırıldığını ifade etmektedir. Bunun yanında 16#Sistemde ve programda servo motorun kendi içerisinde yer alan konum algılayıcıdan gelen bilgiler ile de yine PLC ve motor sürücüsünün haberleşme protokolü olan SERCOS'dan gelen pozisyon bilgileri ayrı ayrı ölçülmektedir.

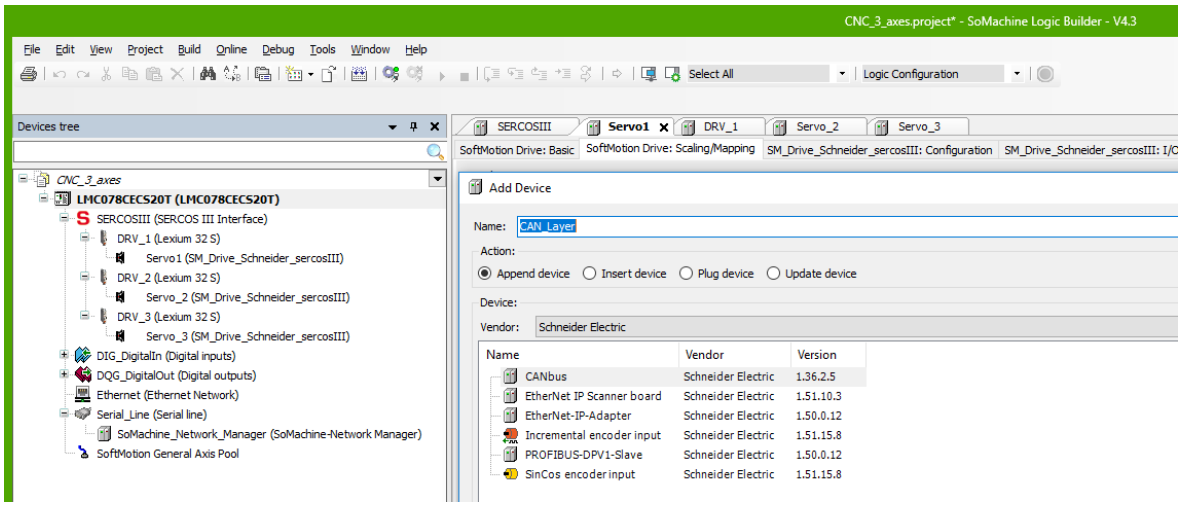
Aynı kutucuğun karşısındaki 1 rakamı bu sayının bir turda gerçekleştirileceğini belirtmektedir. Resim 3.12'de b ile gösterilen kutucuktaki 1 motor turu ile vidanın dönme sayısı oluşacağı belirlenmektedir. Oluşturulan sistemde ayrıca bir dişli kutusu kullanılmadığı için karşısındaki kutucuktaki sayı da 1 olarak verilmiştir.

Aynı şekildeki c ile gösterilen kutucuktaki 1 rakamı ise 1 vida turunda kaç mm hareket oluştuğu belirlenmektedir. Tasarlanan sistemde vida adımı 5 mm olduğundan sağ taraftaki rakam da 5 olarak verilmiştir.



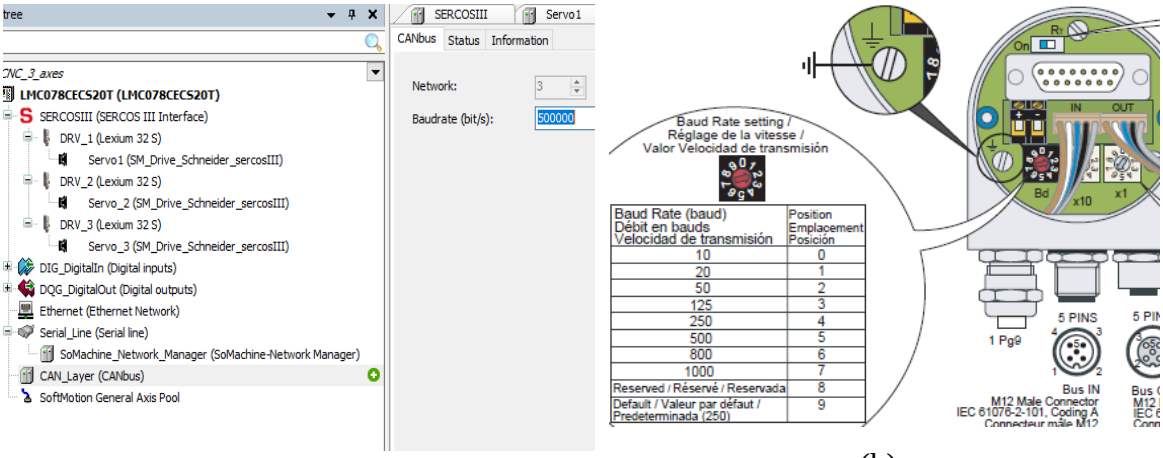
Resim 3.12 Servo motor ayarları

Cihaz ekleme işlemlerine Resim 3.13'de görüldüğü üzere, harici konum algılayıcı haberleşmesi için gerekli olan CANbus Modülünün aktifleştirilmesi ile devam ediyoruz. Bunun için Device Tree üzerine tıklanıp, Add Device tıklanır ve burada CANbus basamağı seçilerek Add Device'a tıklanır.



Resim 3.13 CANbus modülünün eklenmesi

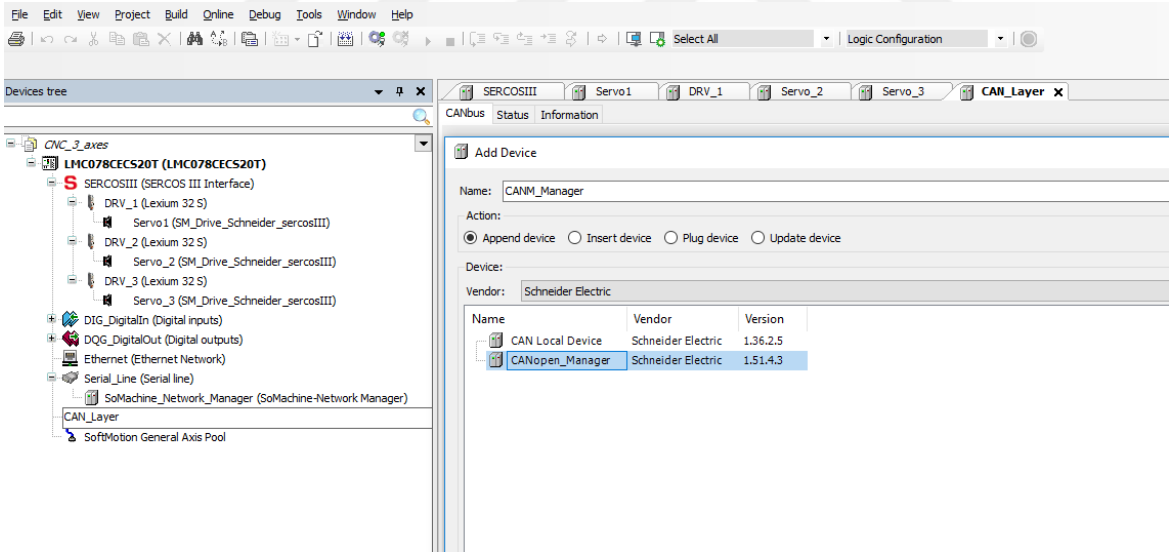
Daha sonra device tree içerisinde eklenmiş olan CANbus üzerine tıklanır ve ekranın sağ tarafında haberleşme ayarları Resim 3.14.a'daki gibi yapılır. Bunun için kullanmış olduğumuz konum algılayıcıda tercih edilen ve konum algılayıcı içerisinde Resim 3.14.b'de ayarlanan 500000 boundrate seçilmiştir.



(a) (b)

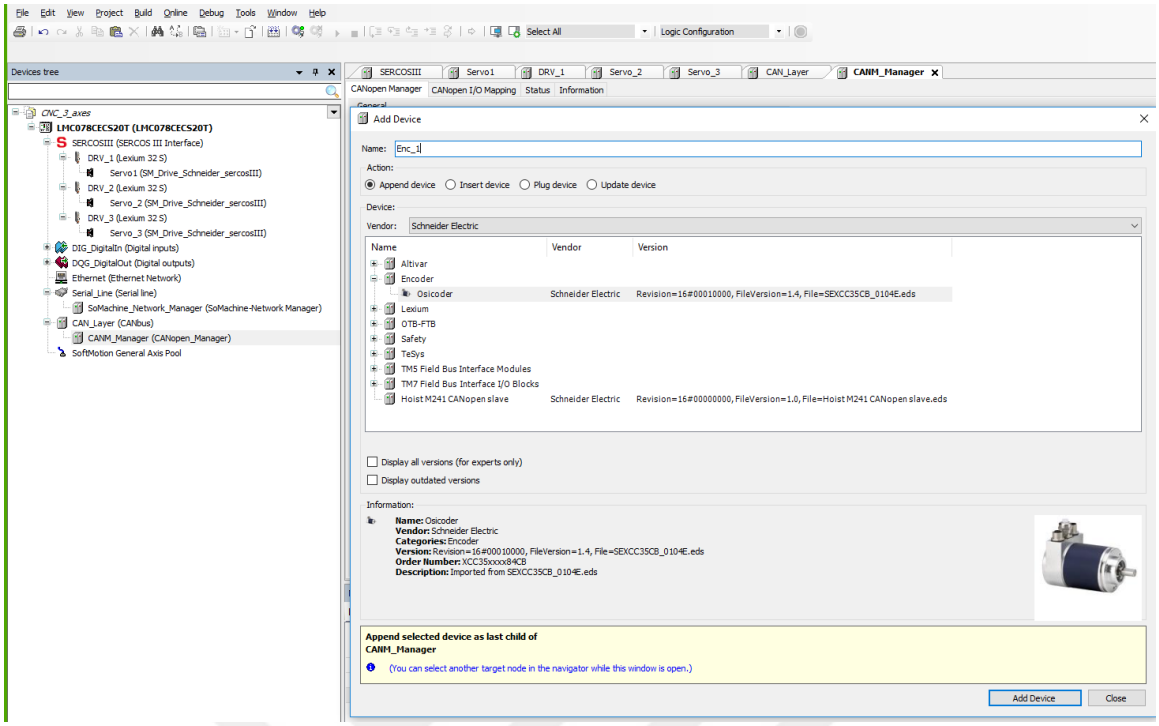
Resim 3.14.(a)CANbus kurulum-1(b) CANbus kurulum-2

CAN bus modülü aktifleştirildikten sonra, device tree içerinde CANbus üzeri tıklanarak Device Tree tıklanır. Resim 3.15'de görüldüğü üzere CANopen Manager tıklanır ve Add Device butonuna basılır.



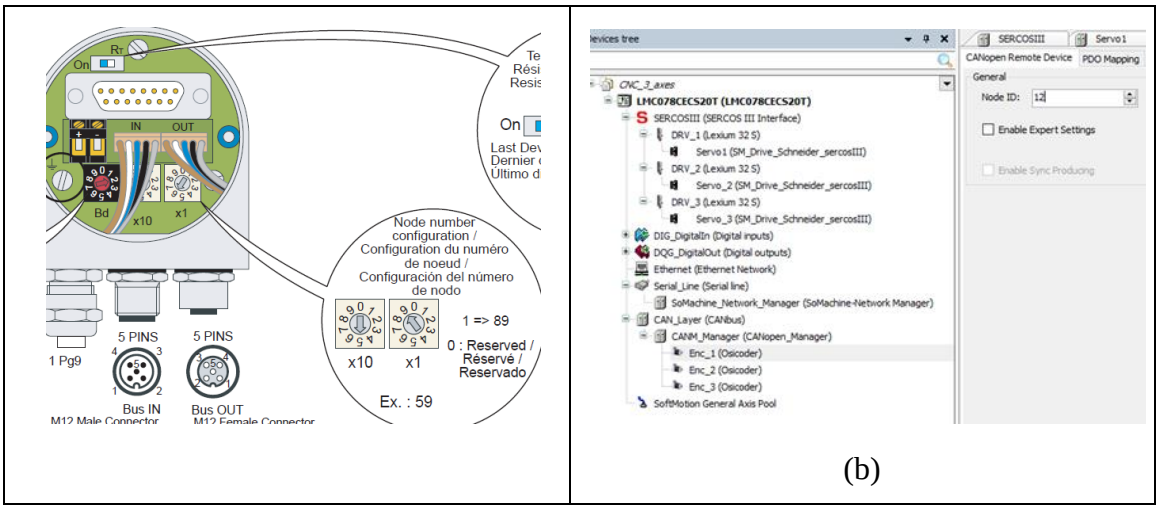
Resim 3.15. CANopen manager eklenmesi

Device Tree içerisinde eklenen CANM_Manager seçilerek add device seçeneği tıklanır ve daha sonra Resim 3.16'da görüldüğü üzere konum algılayıcı ve onun altında bulunan Osicoder seçilir. Daha sonra konum algılayıcı isimlendirmesi için Enc_1, Enc_2, Enc_3 isimleri verilerek Add Device butonuna basılır.



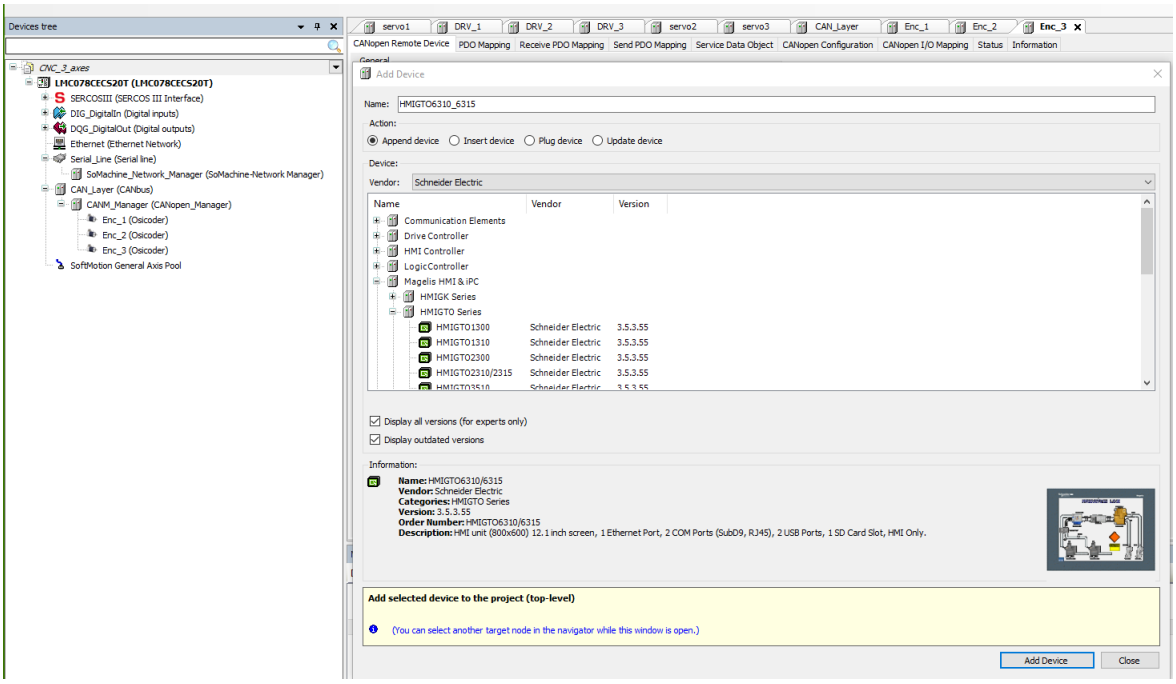
Resim 3.16. Konum algılayıcı eklenmesi

Eklenen konum algılayıcıların sisteme adres tanımlanması yapılabilmesi için, daha önce konum algılayıcı içerisinde belirlediğimiz 1. Konum algılayıcı için 12, 2. Konum algılayıcı için 14, 3. Konum algılayıcı için 16 adresleri mekanik olarak Resim 3.31.a'da görüldüğü gibi ayarlanır. Belirlenen bu adresler Resim 3.17.b'deki gibi ENC_1 üzerine tıklanarak Node ID kısmında 12 olarak, diğer konum algılayıcılar için 14 ve 16 olarak yazılır.



Resim 3.17.(a)Konum algılayıcı Adres Atama-1(b) Konum algılayıcı Adres Atama-2

Sistemi kullanacak son kullanıcı için HMI panel eklemesine ihtiyaç duyulmuştur. Resim 3.18'de görüldüğü gibi HMI panel eklenebilmesi için device tree alanında proje adına tıklanır ve add device sekmesinden Magelis HMIveIPC seçilerek kullandığımız panelin modeli seçilerek add device butonuna basılır. Böylelikle kullanılan bütün donanımlar program içine yüklenmiştir.

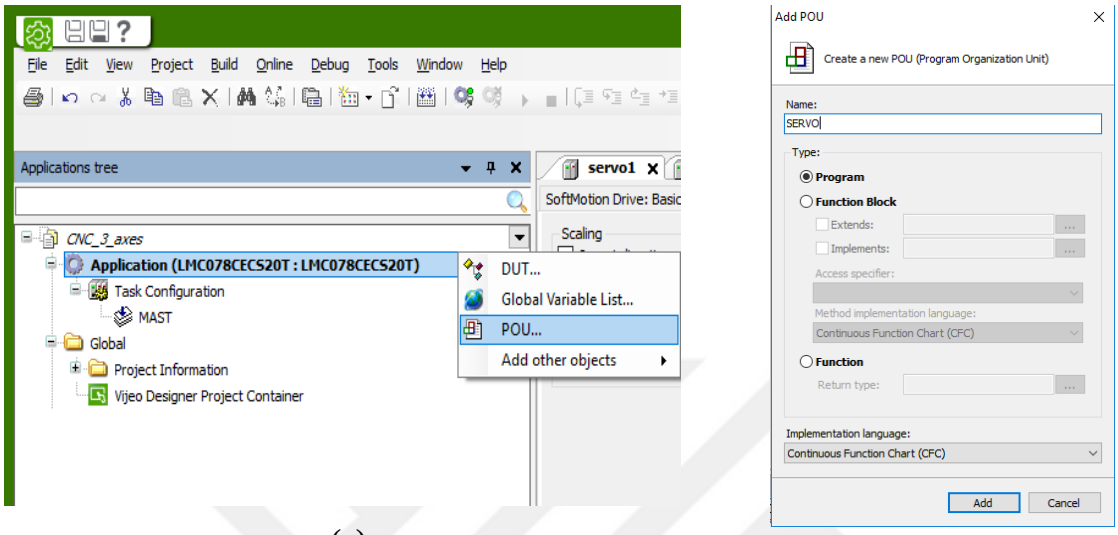


Resim 3.18. Panelin eklenmesi

3.2.2. SoMachine'de programlama

Proje içerisindeki iki ana başlık halinde programlama yapılmıştır. Bunlardan biri harici konum algılayıcı programlamak için ENC adında, diğeri ise servo motorların kontrolü için SERVO adında tasarlanmıştır. Servo motor kontrolü için yapılan ana program, 4 alt programı içermektedir. Genel olarak bunlardan 1.si Prg_1; her üç eksen için motorun serbest kontrolünü sağlar. İkinci program olan Prg_2 ise hazırlanan G-Codeların bulunduğu yerden okunmasını sağlar, üçüncü program Prg_3 okunan G-code komutlarının dönüştürülmesini, Prg_4 ismindeki son program ise servo motorların interpolarasyon yapılarak çalıştırılmasını sağlar. Bütün programlama işlemleri application tree alanında yapılmaktadır. Programın sol-alt kısmında yer alan applications tree tıklanır.

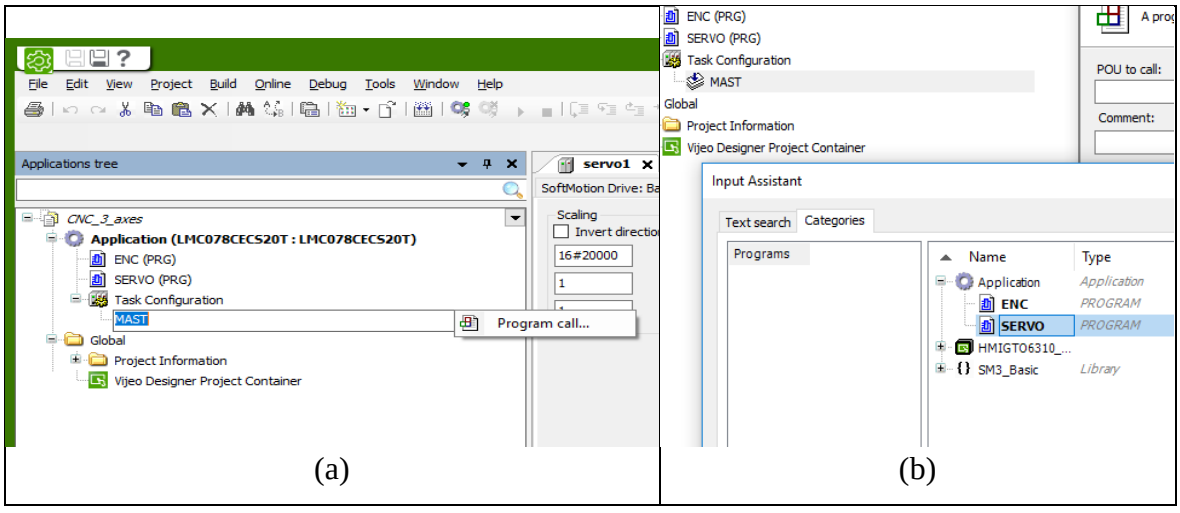
Öncelikle Resim 3.19.a'da görüldüğü üzere PLC isminin yer aldığı alan tıklanarak 2 ana POU oluşturulur, Resim 3.19.b'deki gibi program ve CFC seçenekleri seçilir, programın adı yazılır. Ardından aynı işlem ENC programı için de yapılır.



(a) (b)

Resim 3.19.(a) PLC'nin programlanması-1(b) PLC'nin programlanması-2

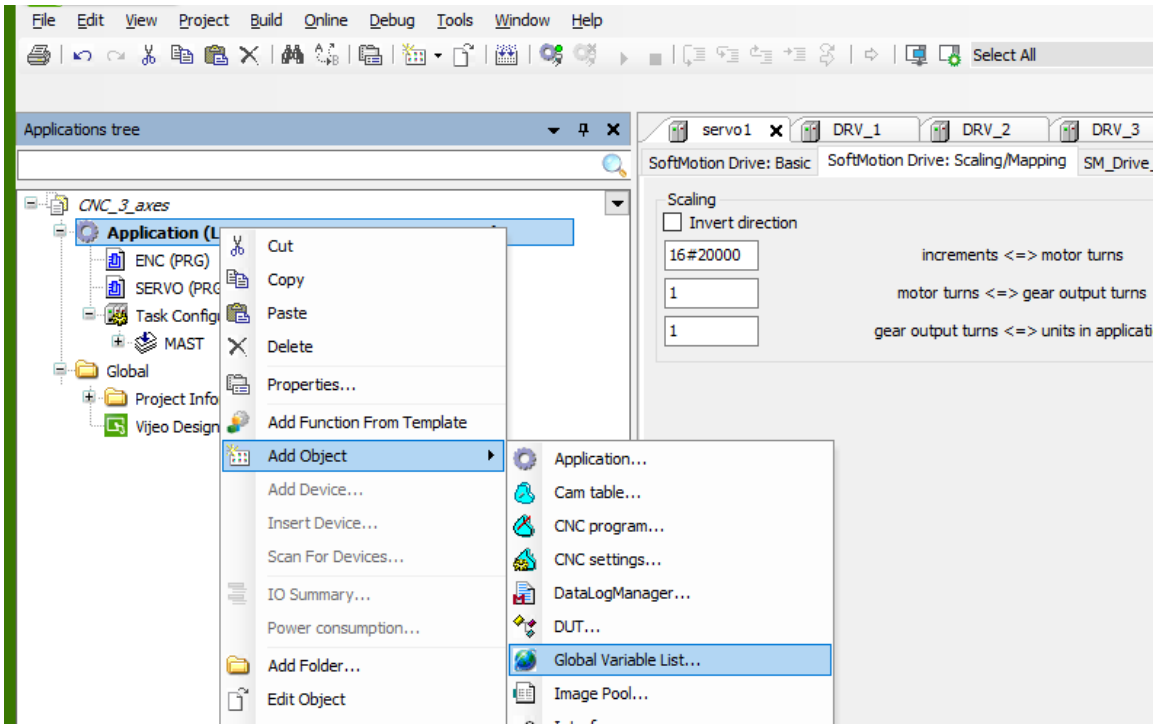
Task configuration alanında iki adet ana programın sırayla Mast içinde çağrılması gereklidir. TaskConfiguration üzerinden Resim 3.20.a'daki gibi Mast tıklanarak Program call seçeneği tıklanmalıdır. Çıkan Program call ekranında Resim 3.20.b'deki gibi SERVO ve ENC programları sırayla çağrılır.



(a) (b)

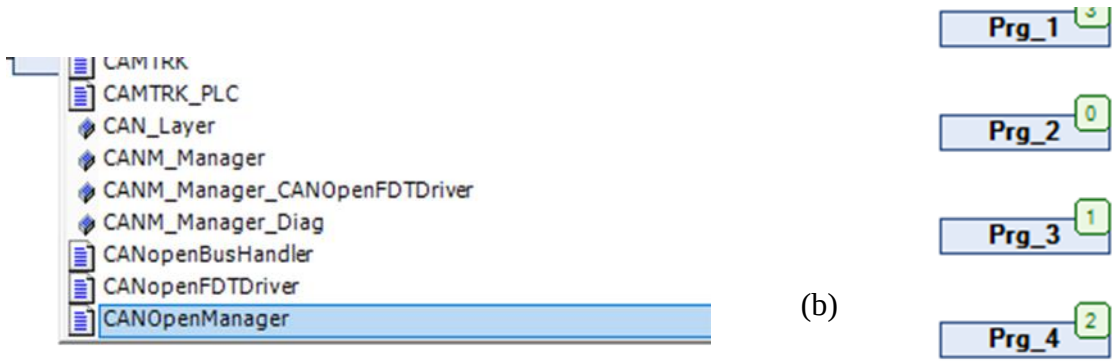
Resim 3.20.(a) PLC'nin programlanması-3 (b) PLC'nin programlanması-4

SERVO (PRG) üzerindeki artıya tıklanarak CFC türünde 4 adet action eklenir. Bu eklenen Action dosyalarına Prg_1, Prg_2, Prg_3, Prg_4 isimleri verilir. Daha sonra dışarıdan komut almayı sağlayabilmek için Resim 3.21'deki gibi GVL kütüphanesi Application//Add Object//Global Variable List tıklanarak eklenir.



Resim 3.21. GVL kütüphanesinin eklenmesi

Programlama için SERVO programı tıklanır ve Tools kısmından 4 adet box eklenmektedir. Bu box üzerinde Resim 3.22.a'da görüldüğü gibi canopenmanager eklenir. Daha sonra sırasıyla Resim 3.22.b'deki gibi Prg_1, Prg_2, Prg_3, Prg_4 programlarına link verilmelidir.



Resim 3.22.(a) SERVO POU programı-1 (b) SERVO POU programı-2

Motorlarda temel hareketin sağlanması

Prg_1 POU alt programda her bir eksen için temel olarak 9 adet FUNCTION BLOCK (FB) kullanılmıştır. Motorun çalıştırılması, durdurulması, hata kodlarının okunması, pozisyon bilgisi atanması, pozisyon bilgilerinin okunması, hız kontrolünün sağlanması, reset yapılması, serbest çalıştırılması amaçlanmıştır. *MC_Power, MC_Setposition, MC_Moveabsolute, MC_stop, MC_Jog, MC_Reset, MC_Stop, Lreal_To_Real, Real_To_Lreal, Move, EQ* program blokları kullanılmıştır. Seçilen FB'lere POWER_X, SET_POS_X, READ_POS_X, RESET_X, MOVE_X, JOG_X, STOP_X isimleri verilerek iç değişken atamaları yapılmıştır. Ayrıca dışarıdan açma kapama işlemlerini yapabilmek için GVL kütüphanesine değişkenler atanmıştır. Söz konusu değişkenler x-y-z eksenleri için atanmış olup, ekran görüntüsü Resim 3.23'de yer almaktadır.

```
GLOBAL
gvl_power_x AT %MX0.0: BOOL;
gvl_power_y AT %MX2.0: BOOL;
gvl_power_z AT %MX4.0: BOOL;

gvl_stop_x AT %MX1.0:BOOL;
gvl_stop_y AT %MX3.0:BOOL;
gvl_stop_z AT %MX5.0:BOOL;

gvl_repos_x AT %MD120:REAL;
gvl_repos_y AT %MD140:REAL;
gvl_repos_z AT %MD160:REAL;

adp_move_x AT %MD610:LREAL;
adp_move_y AT %MD620:LREAL;
adp_move_z AT %MD640:LREAL;

adp_move2_x AT %MD710:REAL;
adp_move2_y AT %MD720:REAL;
adp_move2_z AT %MD730:REAL;

adp_vel_x AT %MD660:LREAL;
adp_vel_y AT %MD680:LREAL;
adp_vel_z AT %MD700:LREAL;

adp_vel2_x AT %MD740:REAL;
adp_vel2_y AT %MD760:REAL;
adp_vel2_z AT %MD780:REAL;

adp_exe_x AT %MX10.0:BOOL;
adp_exe_y AT %MX12.0:BOOL;
adp_exe_z AT %MX14.0:BOOL;

gvl_RESET_X AT %MX16.0: BOOL;
gvl_RESET_Y AT %MX18.0: BOOL;
gvl_RESET_Z AT %MX20.0: BOOL;

gvl_JOG_VEL_X AT %MD800: LREAL;
gvl_JOG_VEL_y AT %MD820: LREAL;
gvl_JOG_VEL_z AT %MD840: LREAL;

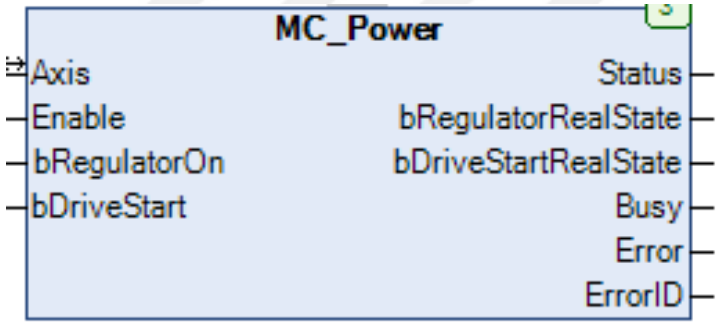
gvl_JOG_FW_X AT %MX21.0: BOOL;
gvl_JOG_FW_y AT %MX22.0: BOOL;
gvl_JOG_FW_z AT %MX23.0: BOOL;

gvl_JOG_BW_X AT %MX24.0: BOOL;
gvl_JOG_BW_y AT %MX25.0: BOOL;
gvl_JOG_BW_z AT %MX26.0: BOOL;
```

Resim 3.23 GVL Kütüphanesi-1

MC_Power komutu

Servo motora güç aktarımını yapacak olan komuttur. “Axis” ile ağda tanımlı daha önce eklediğimiz servo1 bağlanmalıdır. Servo motorun sürekli enerji halinde tutulması için “Enable” girişi aktif “True” olarak işaretlenir. Aynı şekilde drive start girişi de true olarak işaretlenir. Bunun için programın tools menüsünden input ikonu sürüklenerek ekrana bırakılır ve üzerine true yazılır. Enable ve bDriveStart girişleri true’ya bağlanır. Bu komut hareket ve durma konumlarında motorun enerjili olmasını sağlar. Sürücüye ait güç kontrolünün GVL kütüphanesinden yapılabilmesi için, tools kısmından bir adet box seçilir ve üzerine or yazılarak or kapısı haline dönüştürülür. İki adet pin uçlarının birisine daha önce GVL kütüphanesinde oluşturduğumuz gvl_power_x değişkeni atanır, diğer ucuna ise Powerx adında yeni bir iç değişken atanır ve or kapısının çıkışı bRegulateon girişine bağlanır. “MC_Power” komutunda çalışma durum bilgisi “Status” adresinden takip edilir. Durum bilgisi bit adres olarak iç rölelere veya çıkış rölelerine yönlendirilebilir. Hata (Error) oluşması durumunda ise servo motorun çalışması sonlanmaktadır. Resim 3.24’de MC_POWER komutunun ekran görüntüsü ve giriş-çıkış uçları görülmektedir.

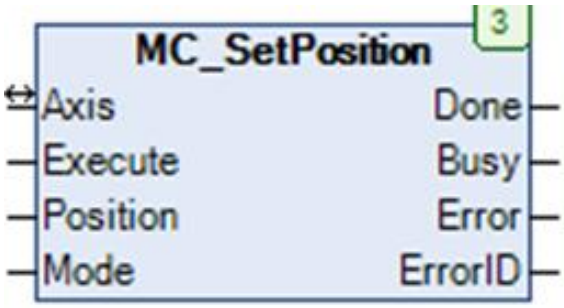


Resim 3.24. MC_POWER komutu

MC_SETPOSITION komutu

MC_SetPosition komutu ise eksenin bulunduğu konuma yer bilgisi atamak için tasarlanmıştır. Bu komut vasıtası ile gerçek konumunu herhangi bir harekete neden olmadan istenilen değerle değiştirir. Eksenin bulunduğu konumu referans almak için kullanılabilir. Bu programda konum bilgisini 0 olarak atamıştır. Enable girişini ise true’ya bağlanmıştır. MC_power’ın çıkışından MC_Setposition ‘ın execute girişine bağlantı yapılmıştır.

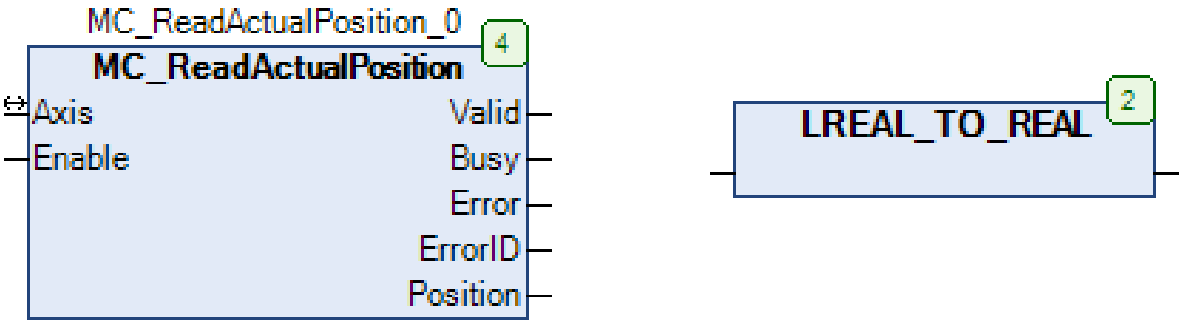
Motor aktif ise status true olacağından pozisyon istenilen değere set edilebilir. Axis girişi ise 1. Servo motor olan Servo1'e bağlanır. Resim 3.25'de MC_ SetPosition komutunun ekran görüntüsü ve giriş-çıkış uçları görülmektedir.



Resim 3.25. MC_ SetPosition komutu

MC_ReadActualPosition komutu

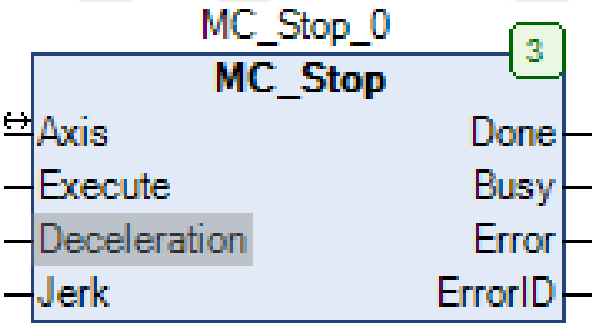
Servo motorun bulunduğu anlık gerçek konum bilgisini okumayı sağlamaktadır. Gerçek konum bilgisi motor dururken veya hareketliyken anlık olarak gerçekleştirilir. Komutun çalışması için "Enable" komutunun aktif edilmesi gerekir ve bunun için true olarak işaretlenir. Axis girişi ise 1. Servo motor olan Servo1'e bağlanır. Okunan konum bilgisi Longreal bir değer olduğundan ve HMI gibi dış ekipmanların real değer okuyabildiğinden tools menüsünden seçilen box ekrana sürüklenir ve LREAL_TO_REAL komutu yazılır. Valid değeri LREAL_TO_REAL girişine bağlanır ve elde edilen real değer GVL kütüphanesinde daha önce oluşturmuş olduğumuz gvl-repos_x değerine atanır. Resim 3.26'da MC_ReadActualPosition komutunun ekran görüntüsü ve giriş-çıkış uçları görülmektedir.



Resim 3.26. MC_ MC_ReadActualPosition komutu

MC_Stop komutu

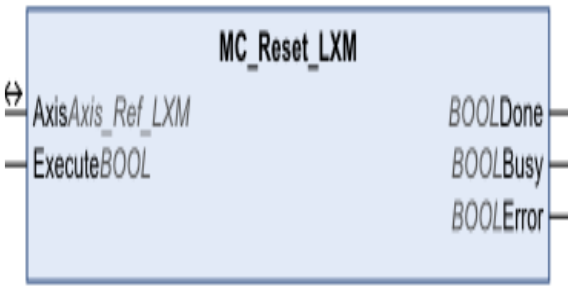
MC_Stop komutu servo motorun çalışması sırasında işlemin iptal edilip, motorun kontrollü olarak durdurulması için kullanılır. “Axis” ile ağda tanımlı daha önce eklediğimiz servo1 bağlanmalıdır. Servo motorun durdurulabilmesi için execute girişinin true olması gerekir. Sürücüye ait kapatma işleminin GVL kütüphanesinden yapılabilmesi için, OR function block kullanılmalıdır. İki adet pin uçlarının birisine daha önce GVL kütüphanesinde oluşturduğumuz gvl_stop_x değişkeni atanır, diğer ucuna ise stopx adında yeni bir iç değişken atanır ve or kapısının çıkışı execute girişine bağlanır. Resim 3.27’de MC_Stop komutunun ekran görüntüsü ve giriş-çıkış uçları görülmektedir.



Resim 3.27. MC_STOP komutu

MC_Reset komutu

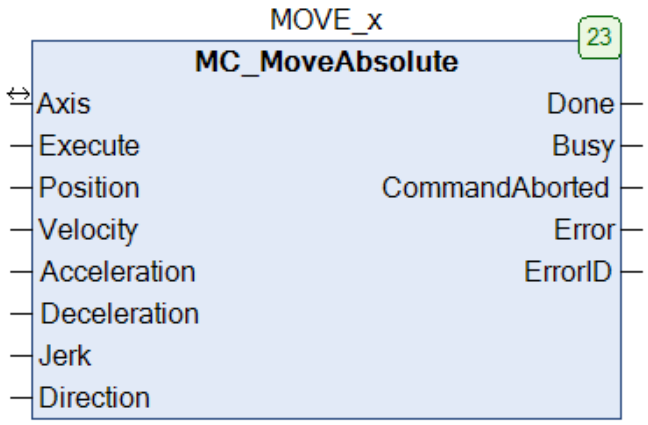
Servo motorun hataları sıfırlamak amacıyla “MC_Reset_LXM” komutu kullanılmaktadır. Hataların düzeltildikten sonra servo motorun yeniden çalıştırılası için bu komut kullanılması gerekir. “Axis” ile ağda tanımlı daha önce eklediğimiz servo1 bağlanmalıdır. Servo motorun durdurulabilmesi için execute girişinin true olması gerekir. Resim 3.28’de MC_Reset komutunun ekran görüntüsü ve giriş-çıkış uçları görülmektedir.



Resim 3.28. MC_Reset komutu

MC_MoveAbsolute Komutu

Servo motorların istenilen hızda, istenilen yönde ve istenilen konuma gitmesi için kullanılmaktadır. “Axis” ile ağda tanımlı daha önce eklediğimiz servo1 bağlanır. “Execute” komutu hareketin başlamasını kontrol etmektedir. Servo motorun hangi konuma gideceği “Position” adresinde tanımlı noktaya gitmesini sağlar. “Velocity” girişine hız bilgisi atanır. Acceleration, Deceleration girişlerine istenilen hıza artış ve eksilişi esnasında istenilen ivme bilgisi, direction girişine ise yön bilgisi girilir. Resim 3.29’da MC_MoveAbsolute komutunun ekran görüntüsü ve giriş-çıkış uçları görülmektedir. MC_MoveAbsolute komutu normalde hazırladığımız program içerisinde kullanılmamakta ancak adaptif uygulama amacıyla yeni hız ve pozisyon bilgisi ile servo motor kontrolü yapılabilmesi amacıyla kullanılır. Burada Execute girişine dışarıdan komut verebilmek amacıyla adp_exe değişkeni, Velocity girişine adp_vel_x, Position girişine adp_move_x değişkenleri gvl kütüphanesinden atanır.

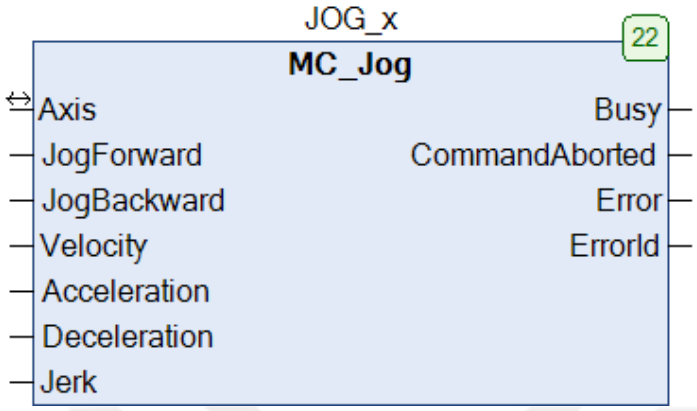


Resim 3.29. MoveAbsolute komutu

MC_Jog Komutu

Servo motorların istenilen hızda, istenilen yönde serbest çalıştırılmasını sağlamaktadır. “Axis” ile ağda tanımlı daha önce eklediğimiz servo1 bağlanır. Jogforward girişine ileri yönde hareket sağlanması istenilen süre boyunca true bilgisi, Jogbackward girişine ileri yönde hareket sağlanması istenilen süre boyunca true bilgisi verilir. “Velocity” girişine hız bilgisi atanır. Acceleration, Deceleration girişlerine istenilen hıza artış ve eksilişi esnasında istenilen ivme bilgisi girilir. Resim 3.41’de MC_MoveAbsolute komutunun ekran görüntüsü ve giriş-çıkış uçları görülmektedir. Burada dışarıdan komut verebilmek amacıyla Jogforward girişine gvl_JOG_FW_X, Jogbackward girişine gvl_JOG_BW_X, Velocity

girişine gvl_JOG_VEL_X,, değişkenleri gvl kütüphanesinden atanır. Resim 3.30'da MC_JOG komutunun ekran görüntüsü ve giriş-çıkış uçları görülmektedir.

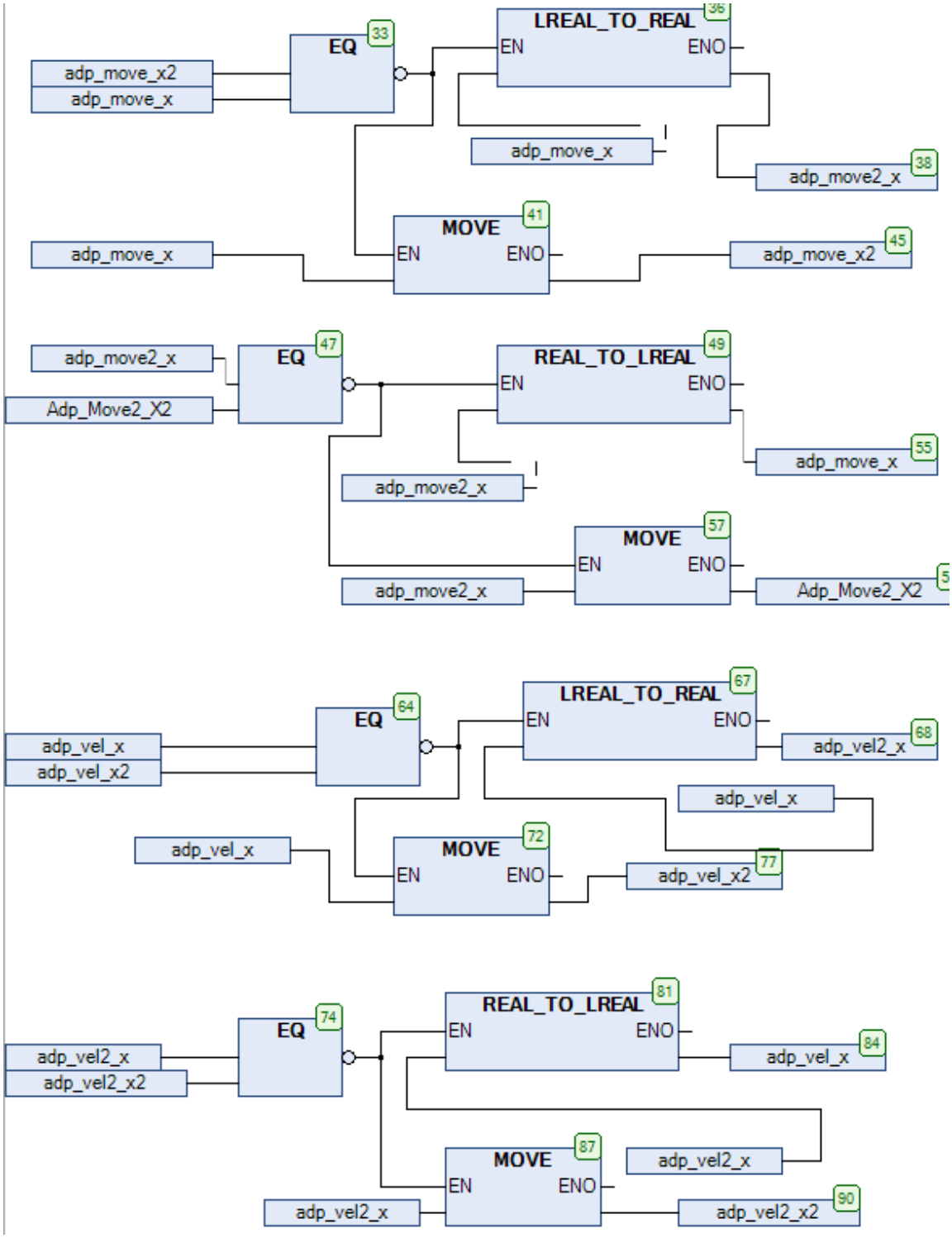


Resim 3.30. MoveAbsolute komutu

LREAL_TO_REAL, REAL_TO_LREAL, Move, EQ komutları

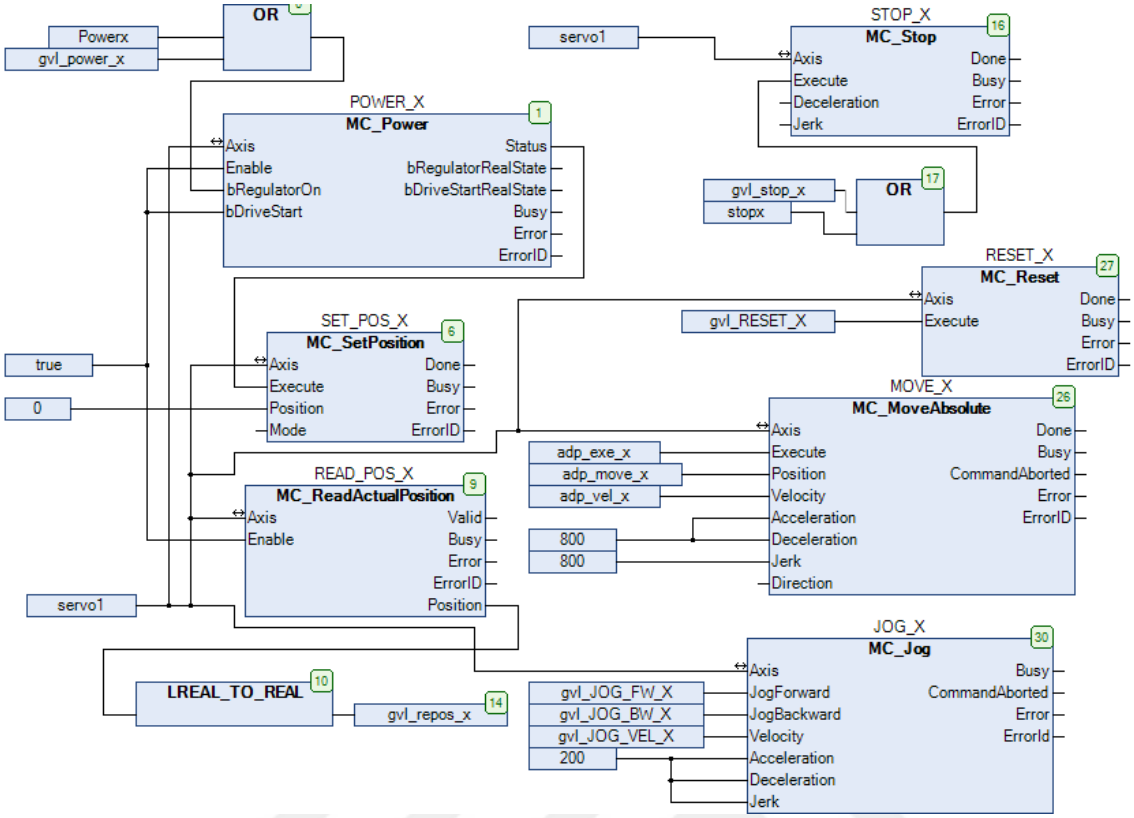
Hız ve pozisyon bilgileri program bloğunda LREAL değişken ile tanımlanmaktadır. Kepserver ile real bilgilerin okunabilmesi LREAL bilgilerin hatalı okunması sebebiyle LREAL olan bilgilerin REAL bilgilere çevrilmesine ihtiyaç duyulmaktadır.

Benzer biçimde kepserver üzerinden REAL bilgilerin program bloğuna gönderilmesi için LREAL'a çevrilmesi gereklidir. LREAL_TO_REAL, LREAL_TO_REAL, MOVE, EQ, FB kullanılarak LREAL haldeki eski hız/pozisyon bilgisi ile yeni hız/pozisyon bilgisi kıyaslanmış, yeni hız/pozisyon bilgisinin atanması halinde REAL'dan LREAL'a çevrilerek FB içerisine atanmıştır. Fonksiyon Blokları ile hazırlanan program Resim 3.31'de sunulmuştur.

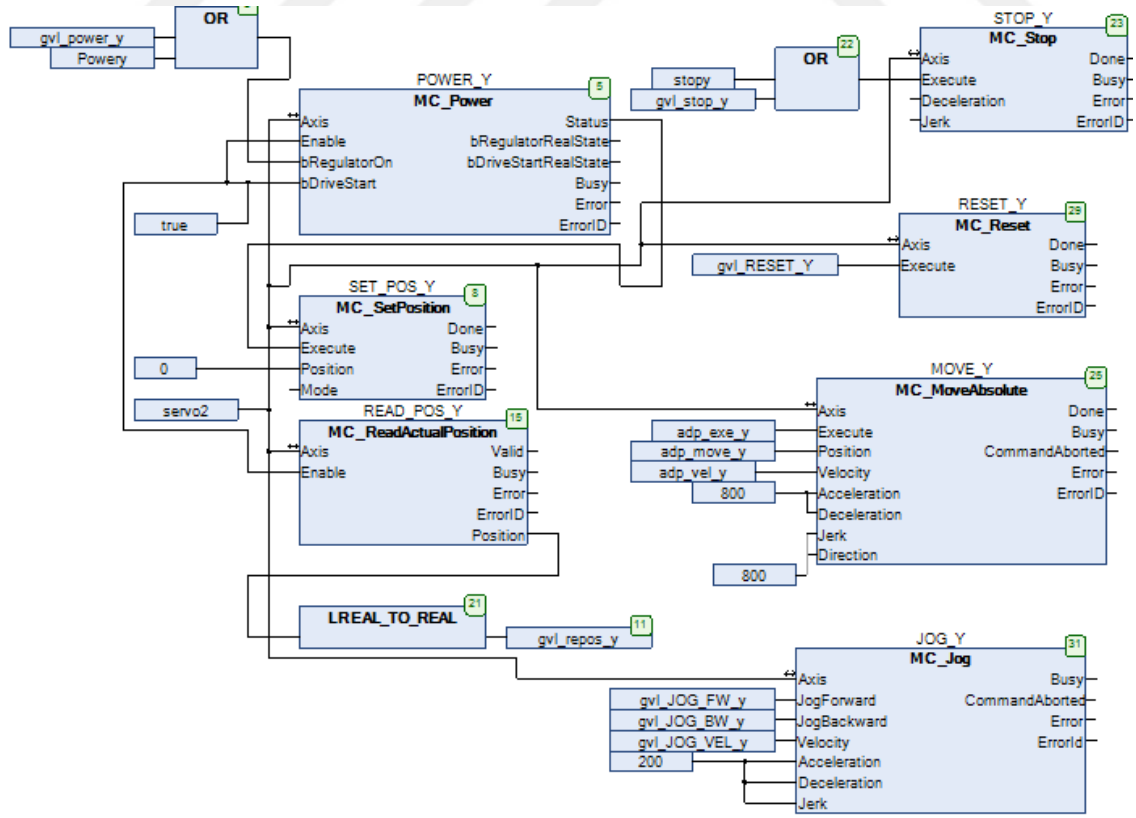


Resim 3.31. Pozisyon/hız bilgisi çevrimi

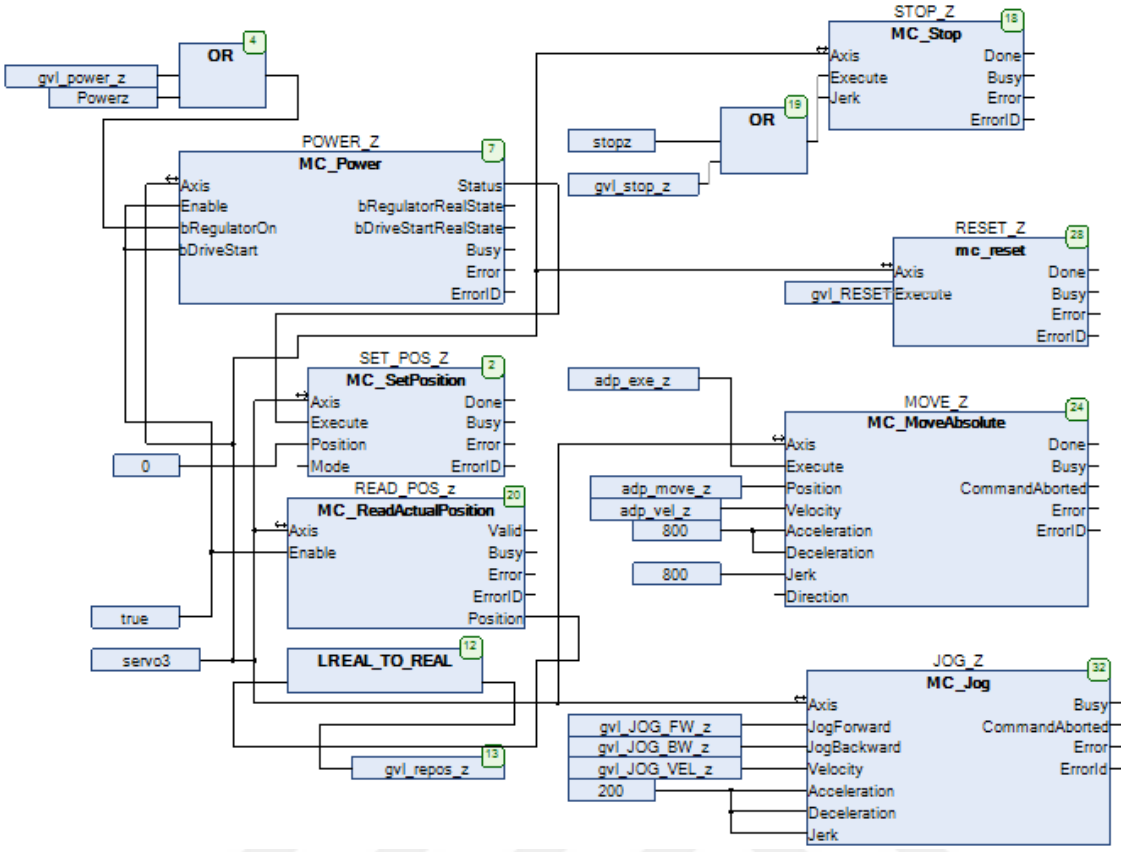
Resim 3.32'de X ekseni için, Resim 3.33'de Y ekseni için, Resim 3.34'de Z ekseni için hazırlanan programın ekran görüntüsü ve giriş-çıkış uçları görülmektedir. Resim 3.35'de ve Resim 3.36'da X-Y-Z eksenleri için Pozisyon/hız bilgisi çevrimi programı yer almaktadır.



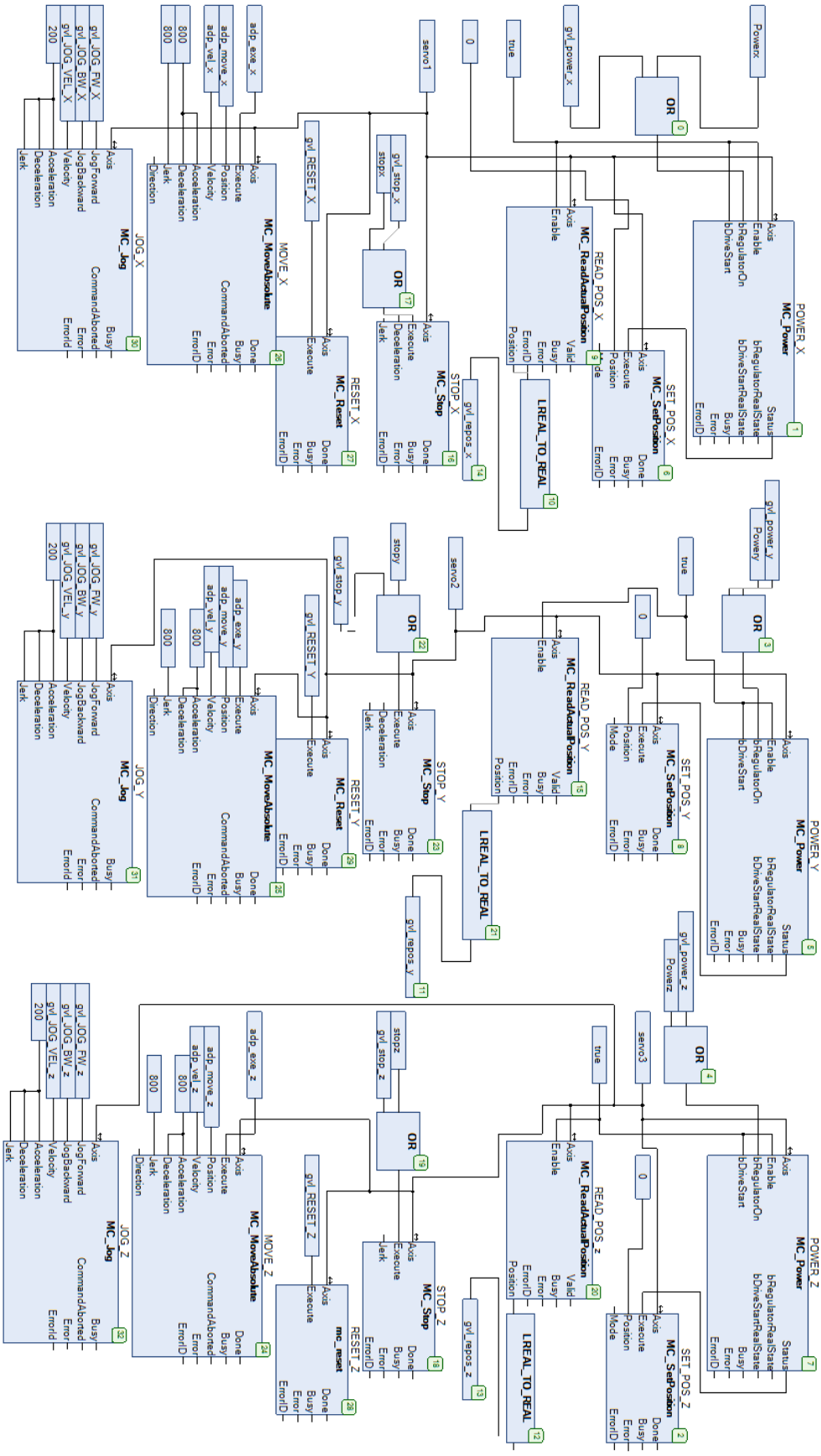
Resim 3.32. X ekseni için hazırlanan program



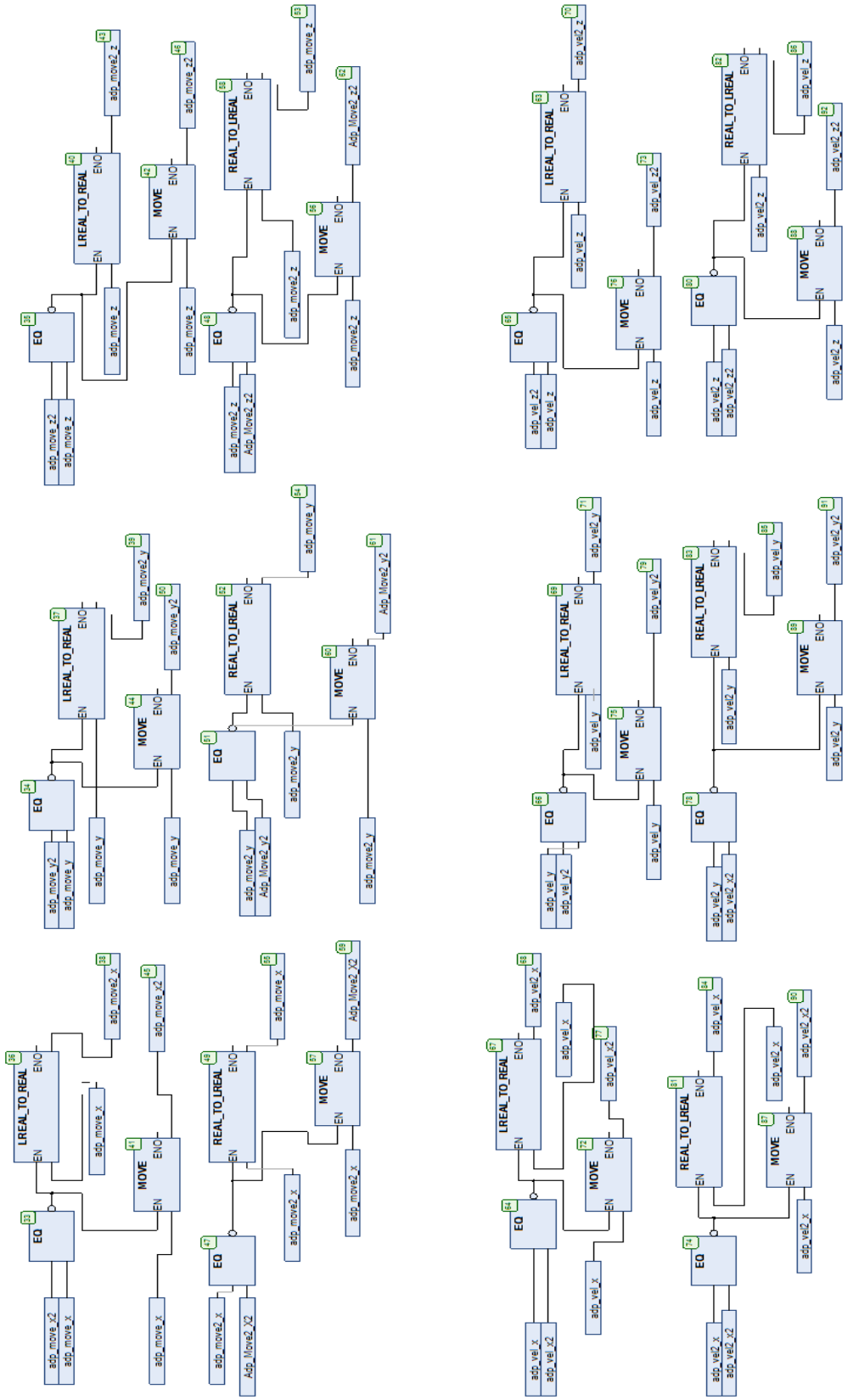
Resim 3.33. Y ekseni için hazırlanan program



Resim 3.34. Z ekseni için hazırlanan program



Resim 3.35. X-Y-Z ekseni için hazırlanan program-1

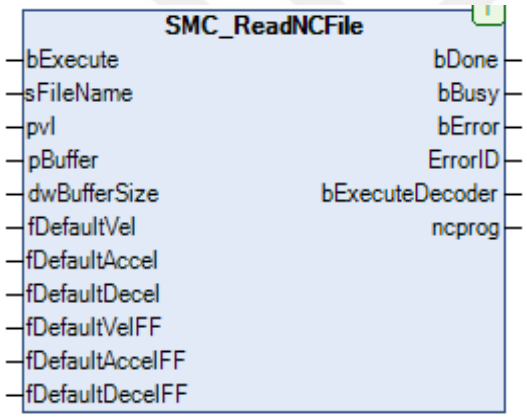


Resim 3.36. X-Y-Z eksenleri için hazırlanan program-2

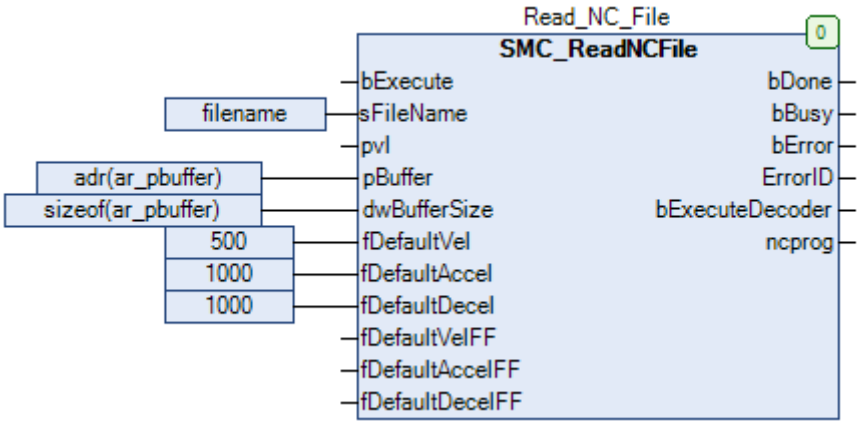
G-kodunun 0kunması

SMC_ReadNCFile komutu

SMC_ReadNCFile komutu denetleyicinin içerine aktarılan bir NC-ASCII dosyasını okur ve SMC_NCDecoder için kullanılabilir hale getirir. Program çalışırken bir NC programı okunabilir ve gerçekleştirilebilir. CNC dosyasında hız belirtilmemişse, artış hızı 1000, azalma hızı 1000 ve sabit hızı 500 olarak belirlenmiştir. Ayrıca pbuffer ve dwBuffersize değerlerine atanan ar_pbuffer1: ARRAY [0..10000] OF SMC_GCODE_WORD; belirlenmiştir. SMC_ ReadNCFile fonksiyon bloğunun görüntüsü Resim 3.37'de, program içerisindeki uygulaması Resim 3.38'de sunulmuştur.



Resim 3.37. SMC_ReadNCFile komutu

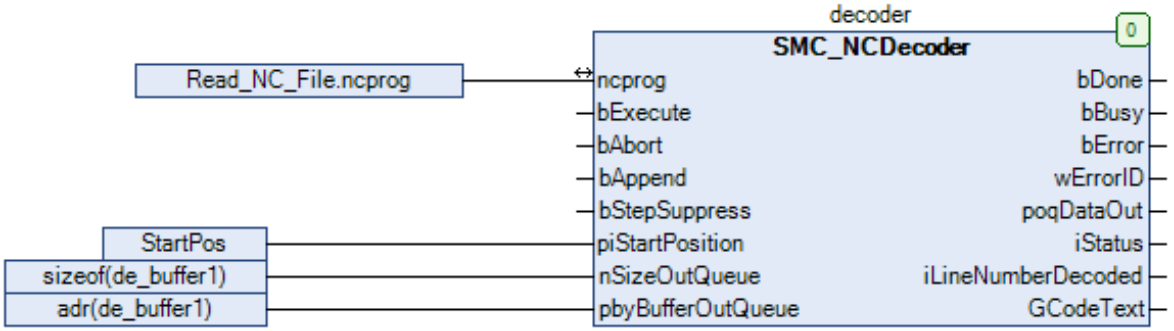


Resim 3.38. Prg_2 programı ve SMC_ReadNCFile komutunun uygulaması

G-kodunun çözümlenmesi

SMC_ NCDecoder komutu

Yazılan program satırlarını çözer ve CNC editöründe oluşturulan bir CNC programını SoftMotion GEOINFO haline dönüştürür. SMC_ReadNCFile FB'sinin çıkışı olan ncprog ise bu komutun ncprog girişine bağlanmalıdır. bExecute girişi SMC_ReadNCFile FB'sinin çıkışı olan bExecutedecoder ile bağlanmalıdır. Pistartposition girişi daha önce dx,dy,dz şeklinde atanan startpos bilgileri bağlanmıştır. Ayrıca nSizeOutQueue ve pbyBufferQueue değerlerine atanan de_buffer1: ARRAY [0..300] OF SMC_GEoinfo; olarak belirlenmiştir. SMC_ NCDecoder fonksiyon bloğunun görüntüsü Resim 3.39'da sunulmuştur.

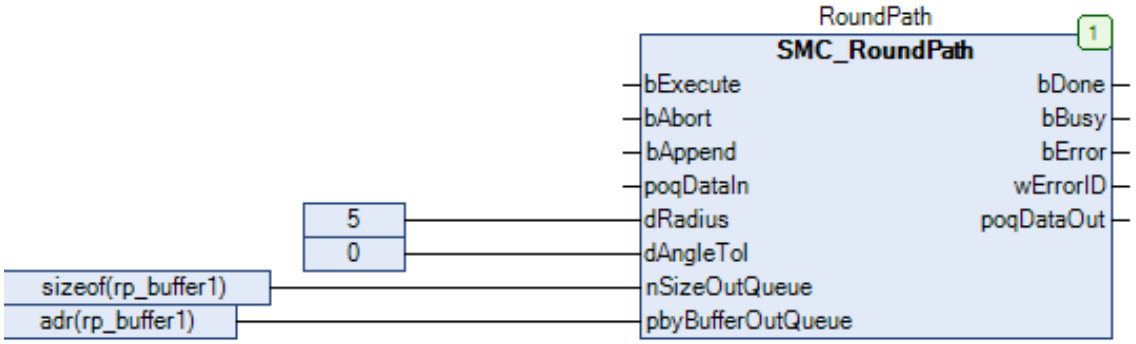


Resim 3.39. SMC_NCDecoder komutu

SMC_ RoundPath komutu

SMC_RoundPath fonksiyonu, İki nesnenin birleşiminde ortaya çıkan kenarları dairesel yaylarla yuvarlar. CNC programında yuvarlama işlemi için, başlangıç (G52), son (G50) ve yuvarlama yarıçapı (D <Radius>) ayarlanmalıdır.

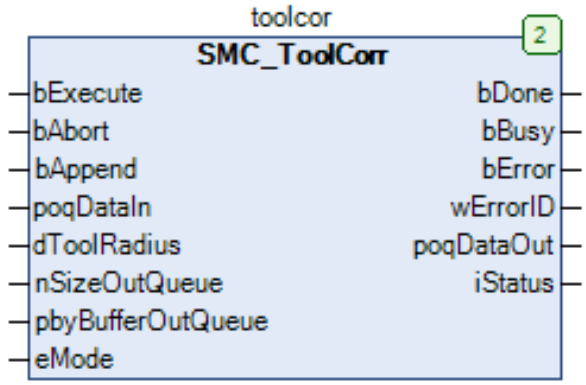
FB üzerindeki bExecute girişi SMC_ReadNCFile FB'sinin çıkışı olan bExecutedecoder ile bağlanmalıdır. G-code içerisinde herhangi bir Radius değeri girilmemişse dRadius 5ve dAngelTol 0 olarak belirlenmiştir. Ayrıca nSizeOutQueue ve pbyBufferQueue değerlerine atanan rp_buffer1: ARRAY [0..300] OF SMC_GEoinfo; olarak belirlenmiştir. SMC_RoundPath fonksiyon bloğunun görüntüsü Resim 3.40'da sunulmuştur.



Resim 3.40. SMC_RoundPath Komutu

SMC_ToolCorr komutu

SMC_ToolCorr komutu ile Takım yarıçapının veya uç yarıçapının telafisi (G41 / G42) komutlarının girilmesi ile mümkün olmakta ve iptali (G40) ile yapılmaktadır. Takım yarıçapı D <angle>ile tanımlanmalıdır. FB üzerindeki bExecute girişi SMC_ReadNCFile FB'sinin çıkışı olan bExecutedecoder ile bağlanmalıdır. G-code içerisinde herhangi bir alet yarıçapı değeri girilmemişse dToolRadius 5 olarak belirlenmiştir. Ayrıca nSizeOutQueue ve pbyBufferQueue değerlerine atanan tc_buffer1: ARRAY [0..300] OF SMC_GEoinfo; olarak belirlenmiştir. SMC_RoundPath komutunun poqDataOut çıkışı bu komutun poqDataIn girişine bağlanmıştır. Bu fonksiyon bloğun görüntüsü Resim 3.41'de sunulmuştur.

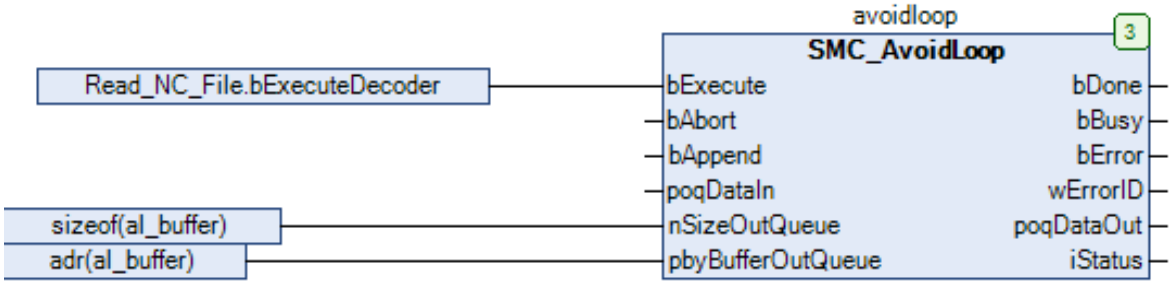


Resim 3.41. SMC_ToolCorr komutu

SMC_ AvoidLoop komutu

Döngüler arasında geçiş yaparken atlama yapmayı sağlamaktadır. CNC programında, düzeltmenin yapılması gereken yol bölümünün başlangıcı (G61) ve sonu (G60) tanımlanmalıdır. FB üzerindeki bExecute girişi SMC_ReadNCFile FB'sinin çıkışı olan

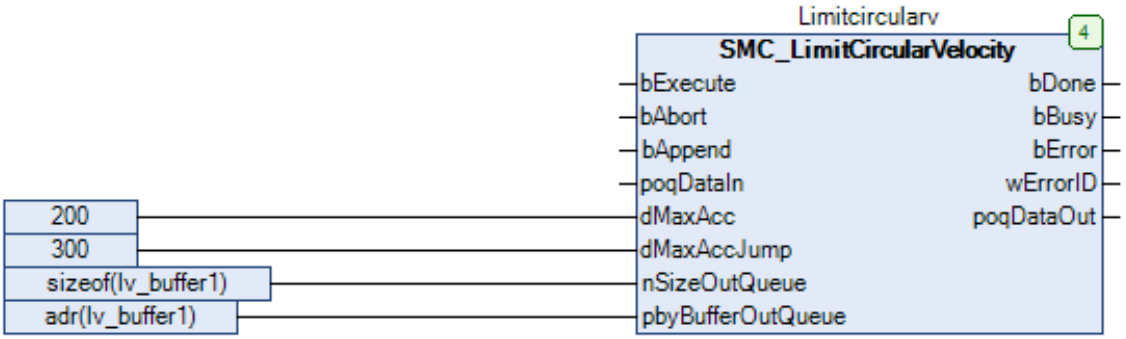
bExecutedecoder ile bağlanmalıdır Ayrıca nSizeOutQueue ve pbyBufferQueue değerlerine atanan al_buffer: ARRAY [0..300] OF SMC_GEoinfo; olarak belirlenmiştir. SMC_ToolCorr komutunun poqDataOut çıkışı bu komutun poqDataIn girişine bağlanmıştır. SMC_AvoidLoop fonksiyon bloğunun görüntüsü Resim 3.42'de sunulmuştur.



Resim 3.42. SMC_AvoidLoop komutu

SMC_LimitCircularVelocity komutu

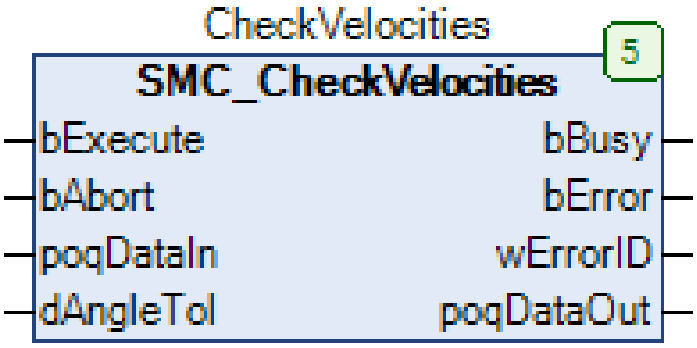
Bu program organizasyonu birimi, iki elemanın geçişini kontrol eder. Birinci elemanın son hızını, hızlanma atlayıcısının dMaxAccJump değerini aşmayacağı şekilde ayarlar. Ayrıca, dairelerin yol hızını buna göre azaltarak, dairelerdeki yol ivmesini dMaxAcc değerine sınırlandırır. Hazırlamış olduğumuz bu programda bu değerler dMaxAcc200 ve dMaxAccJump 300 olarak belirlenmiştir. FB üzerindeki bExecute girişi SMC_ReadNCFile FB'sinin çıkışı olan bExecutedecoder ile bağlanmalıdır. SMC_AvoidLoop komutunun poqDataOut çıkışı bu komutun poqDataIn girişine bağlanmıştır. SMC_LimitCircularVelocity komutunun program içerisindeki görüntüsü Resim 3.43'de sunulmuştur.



Resim 3.43. SMC_LimitCircularVelocity komutu

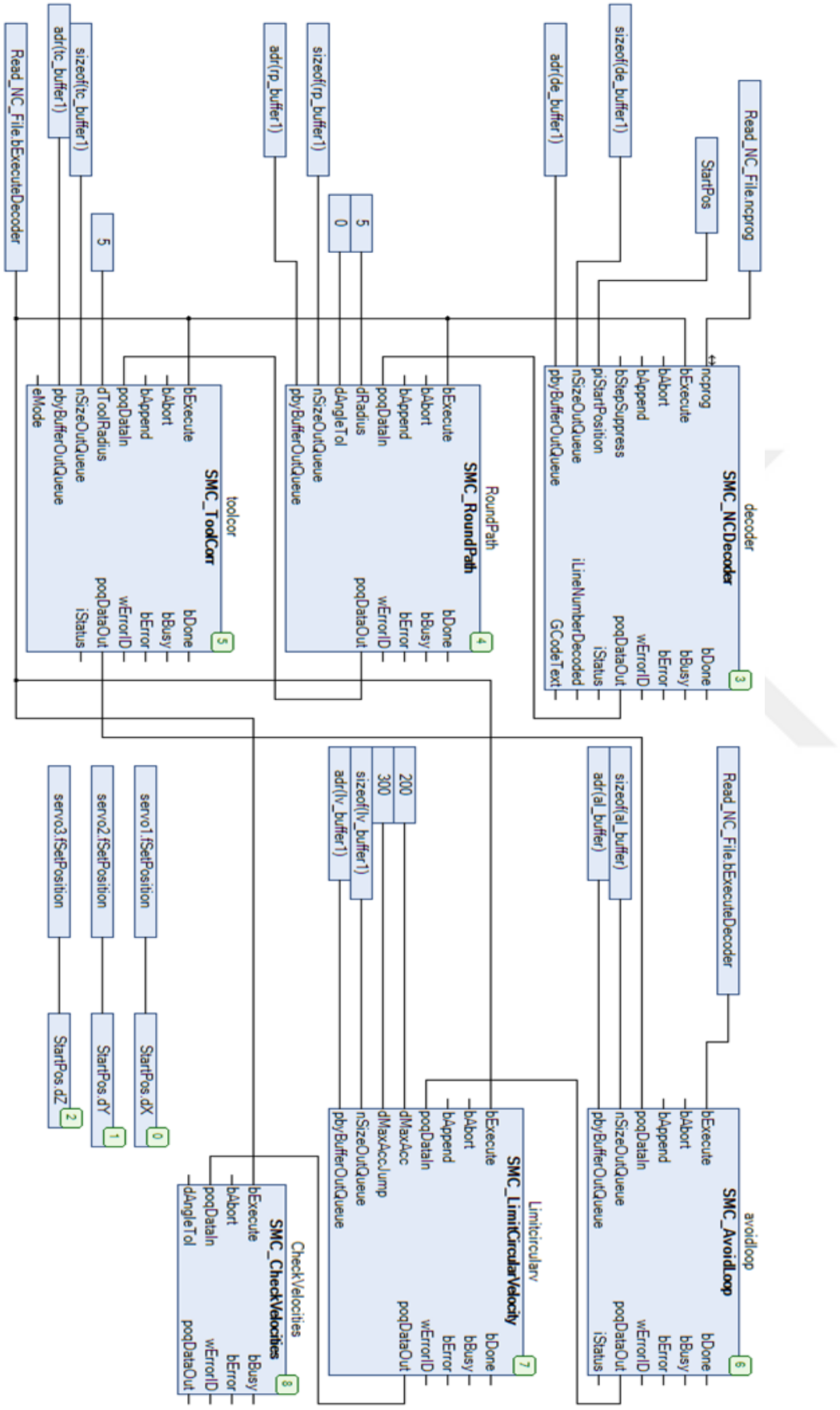
SMC_CheckVelocity komutu

Bu işlevin ana görevi, keskin virajların yolunu kontrol etmek ve bu virajlarda hızı sıfıra indirmektir. bExecute girişi SMC_ReadNCFile FB'sinin çıkışı olan bExecutedecoder ile bağlanmalıdır. SMC_LimitCircularVelocity komutunun poqDataOut çıkışı bu komutun poqDataIn girişine bağlanmıştır. Komutun program içerisindeki görüntüsü Resim 3.44'de sunulmuştur.



Resim 3.44. CheckVelocity komutu

Bu program içerisindeki bütün komutların uygulanmış hali Resim-3.45'de sunulmuştur.



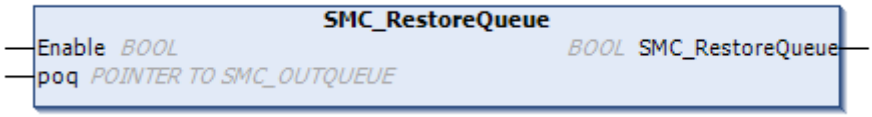
Resim 3.45. Prg_3'ün ekran görüntüsü

İnterpolarasyon programı

Motor hareketleri sabit hızla sağlanmışken, ilave yazılım ve programlama ile interpolarasyon özelliği kullanılmış, motorların gidilecek konuma göre hızlarını hesaplaması, birbiri ile koordineli olarak eşzamanlı olacak şekilde değişken ivme ve farklı hızlarla hareket etmesi sağlanmıştır. İnterpolarasyon amacıyla SMC_Restore Queue, SMC_Interpolator, SMC_ControlAxisByPos SMC_TRAFOF_Gantry, SMC_TRAFO_Gantry komutları kullanılmıştır.

SMC_RestoreQueue komutu

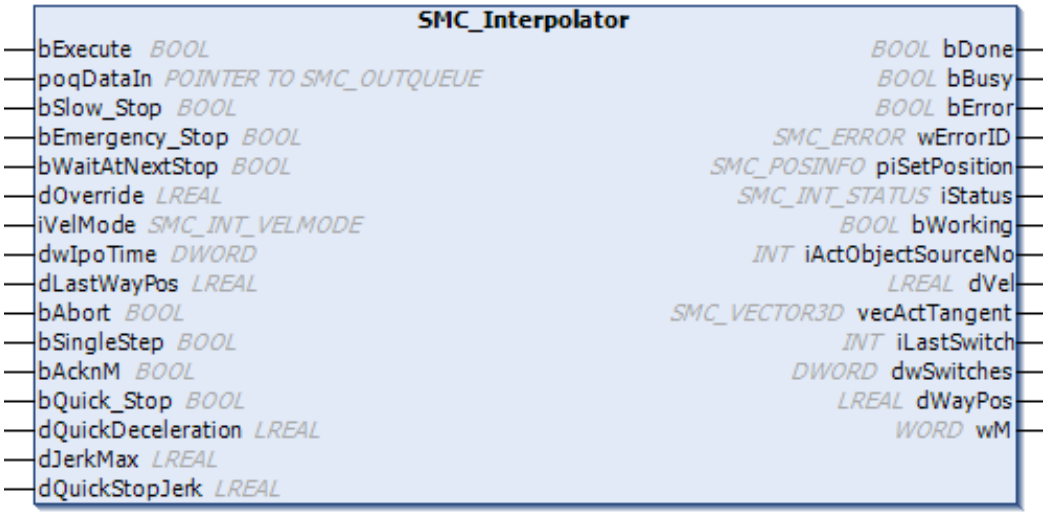
SMC_RestoreQueue İşlev, enterpolasyonlu bir yanıtı geri yükler. İşlenmiş yapının tüm yol tariflerini verilen sırayla saklanabilmesi ile mümkündür. Komutun program içerisindeki görüntüsü Resim 3.46'da sunulmuştur.



Resim 3.46 SMC_RestoreQueue komutu

SMC_Interpolator Komutu

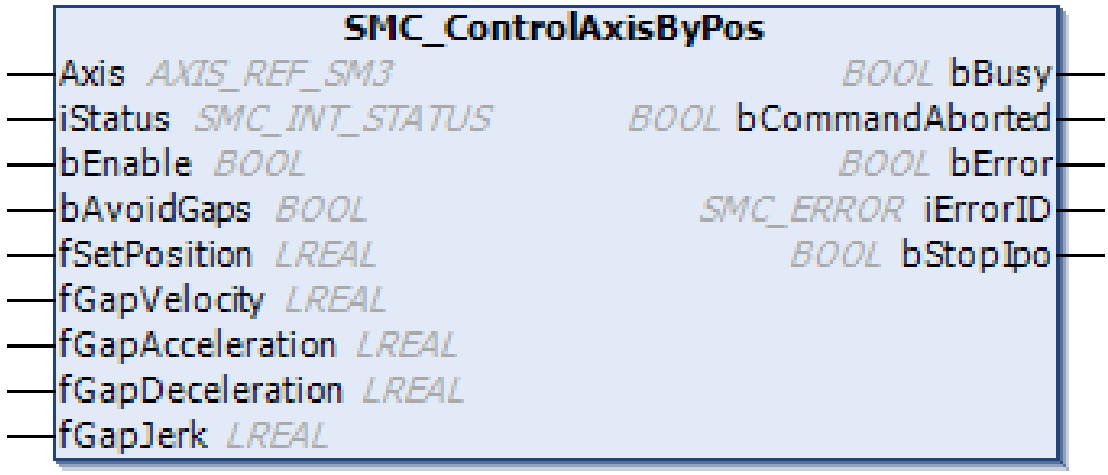
SMC_Interpolator GEOINFO nesnelerinin tanımladığı mevcut bir sürekli yolu, tanımlanan hız profili ve zaman kalıbı dikkate alınarak pozisyon noktalarına dönüştürülür. Bu konum noktaları daha sonra IEC programı vasıtası ile sürücü ekseni konumlarına dönüştürülür ve sürücü arayüzü üzerinden gönderilir. Komutun program içerisindeki görüntüsü Resim 3.47'de sunulmuştur.



Resim 3.47. SMC_Interpolator komutu

SMC_ControlAxisByPos Komutu

SMC_ControlAxisByPos;sürücülere hedef pozisyonları yazmak ve test etmek için kullanılabilir. Genellikle CNC ve SMC_Interpolator bağlamında kullanılır. Komutun program içerisindeki görüntüsü Resim 3.48'de sunulmuştur.



Resim 3.48. SMC_ControlAxisByPos

SMC_TRAFO_Gantry3 Komutu

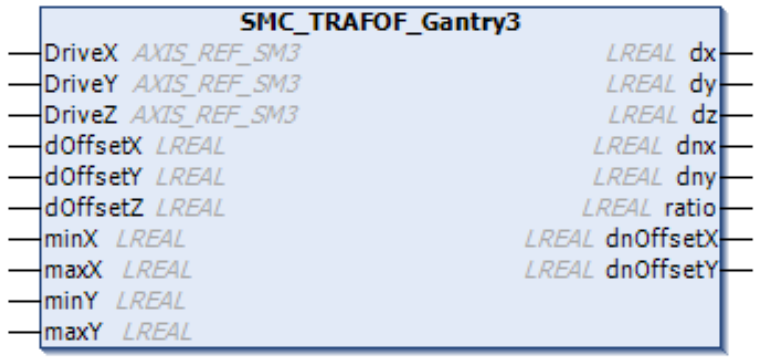
SMC_TRAFO_Gantry3; Dönüşüm modülür. Geriye doğru bir hesaplama yaparak, modül sadece x, y ve z eksenlerine bir kayma ekler ve üç boyutlu portal sistemlerini kullanmayı sağlar. Komutun program içerisindeki görüntüsü Resim 3.49'da sunulmuştur.



Resim 3.49. SMC_TRAFO_Gantry3 komutu

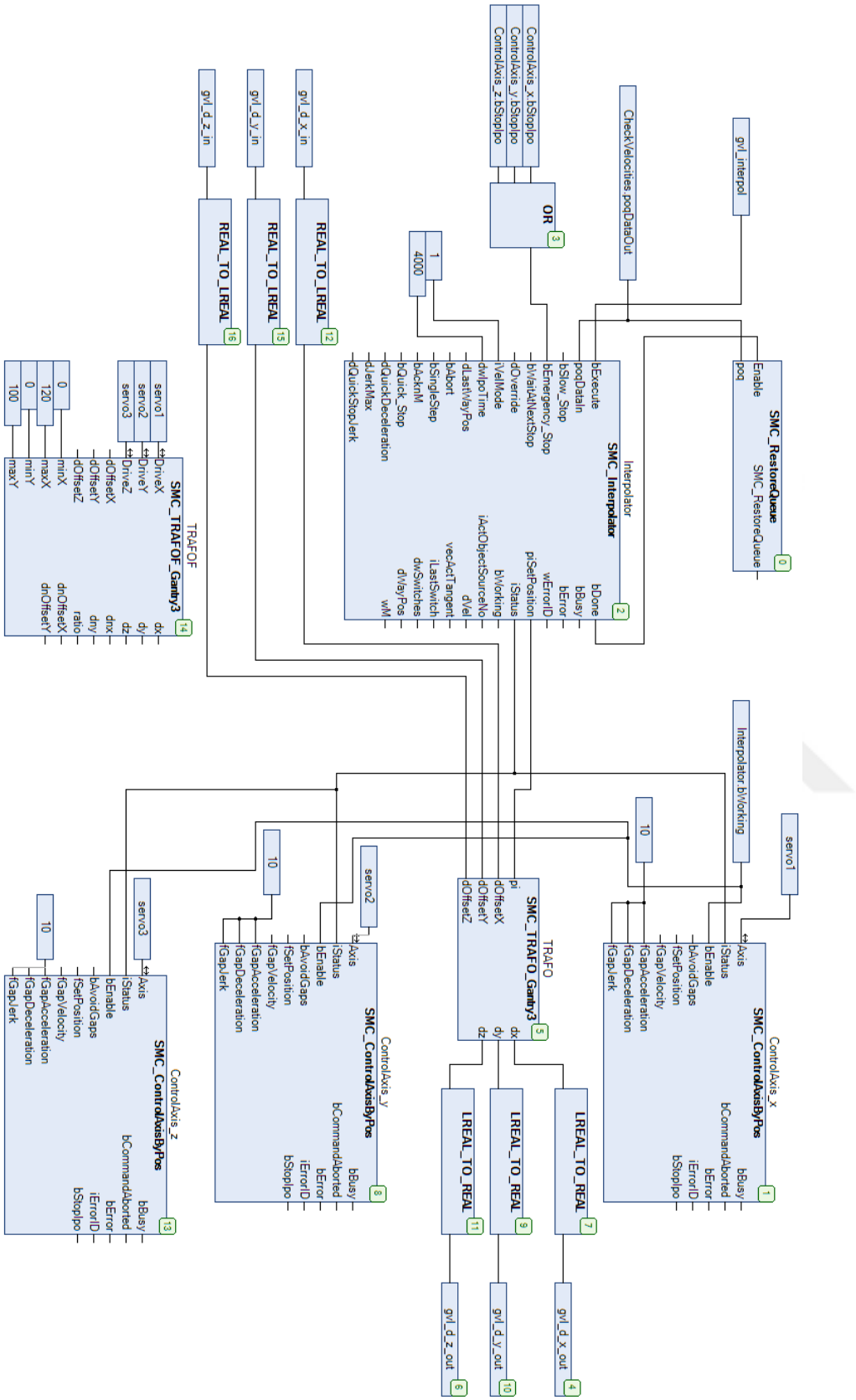
SMC_TRAFOF_Gantry3 komutu

Dönüşümün yapılmadığı üç boyutlu portal sistemleri kullanır. Bu modülünün her örneğini görselleştirme şablonuna bağlamak mümkündür. Komutun program içerisindeki görüntüsü Resim 3.50'de sunulmuştur.



Resim 3.50 SMC_TRAFO_Gantry3 komutu

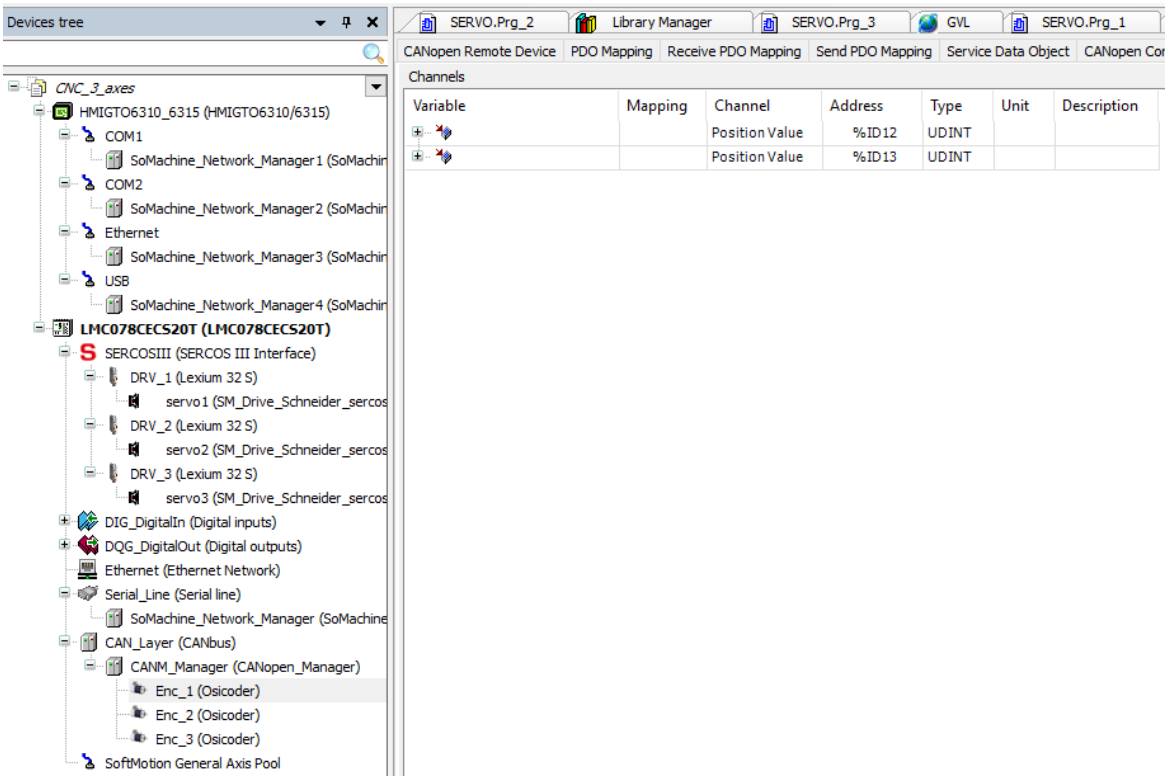
İnterpolarasyon komutlarının programda uygulanmış hali Resim 3.51'de sunulmuştur.



Resim 3.51. İnterpolasyon programı

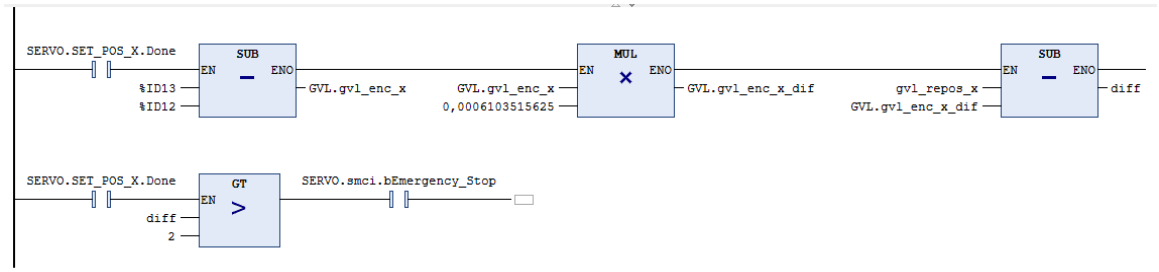
Harici konum algılayıcı'ın programlanması

CNC'nin adaptif hale gelebilmesi için mil uçlarına eklenen konum algılayıcı'lardan bilgi okuyup aktarılabilmesi için ENC(PRG) adında yeni bir POU eklenmiştir. Konum algılayıcı içerisindeki verilerin Resim 3.52'de %12 ID de ilk konum, %13'te son konum olarak saklanmış olduğu görülmüştür.



Resim 3.52. Konum algılayıcı veri haritası

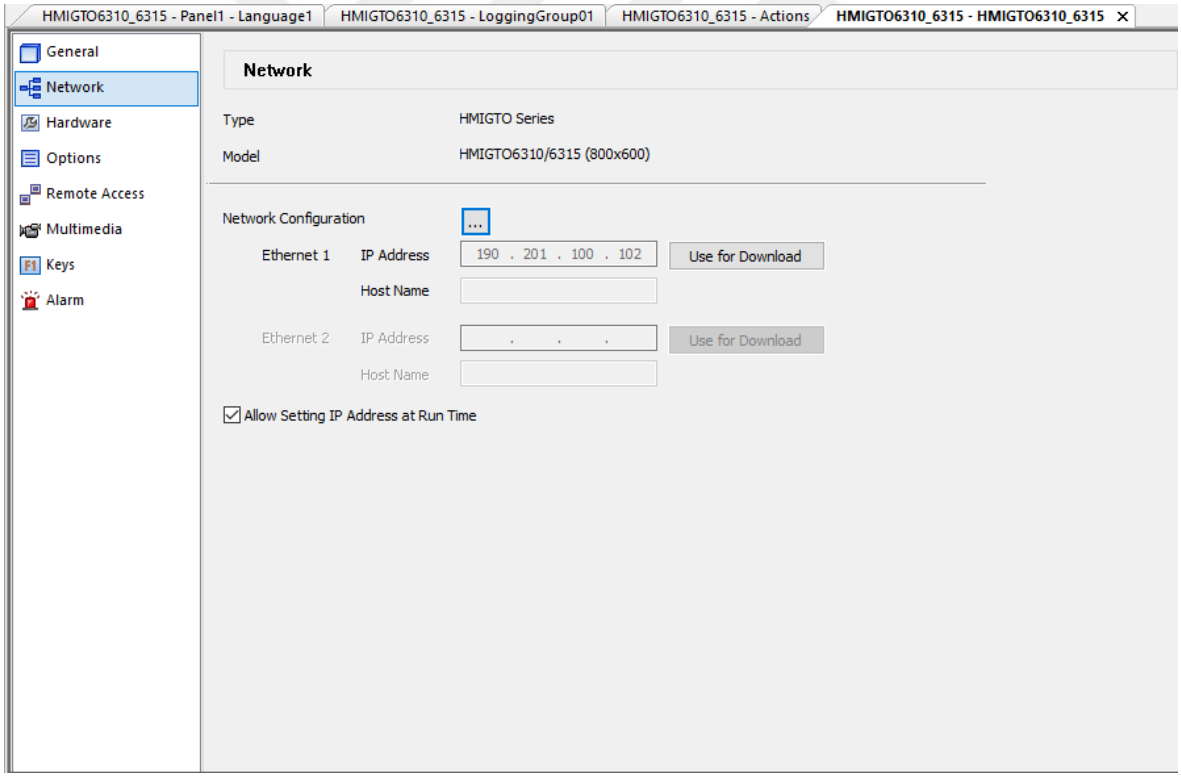
Resim 3.53'de ise ENC programı görülmektedir. Program içerisinde okunabilmesi için son konum bilgileri (%ID13, %ID15, %ID17) GVL kütüphanesine MOVE komutu ile taşınmıştır. Daha sonra ise eski konum ile yeni konum bilgisi SUB komutu ile birbirinden çıkartılmıştır. Konum algılayıcının 1 tam tur dönmesi ile 8.192 puls ürettiği tespit edilmiştir. Bir tam tur dönmesi ile 5 mm ilerlediği bilindiğinden 1mm hareket edebilmesi için gerekli puls sayısı 5/8192= 0,0006103515625 olarak hesaplanmıştır. Buna göre konum algılayıcıdaki tur sayısı farkı 0,0006103515625 sayısı ile MUL komutu ile çarpılarak eksendeki hareket miktarı mm cinsinden hesaplanmıştır. Hesaplanan bu değer dışarıdan okunabilmesi için GVL kütüphanesindeki, gvl_enc_diff_x, gvl_enc_diff_y , gvl_enc_diff_z değişkenlerine atanmıştır.



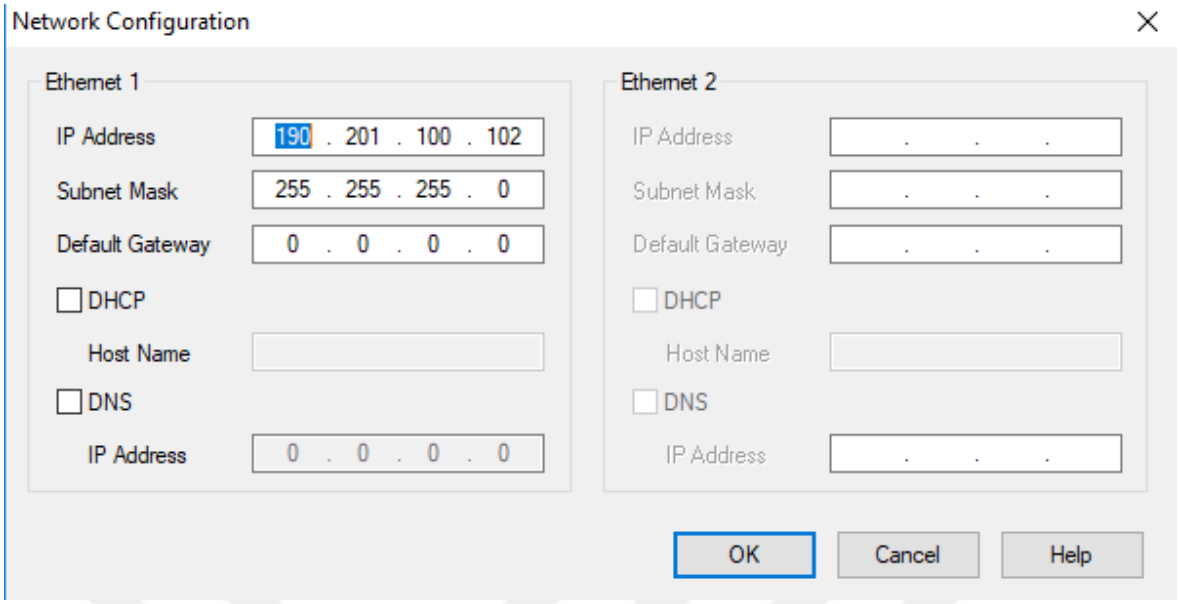
Resim 3.53. ENC program görüntüsü

3.2.3. Vijeo programının kullanılması

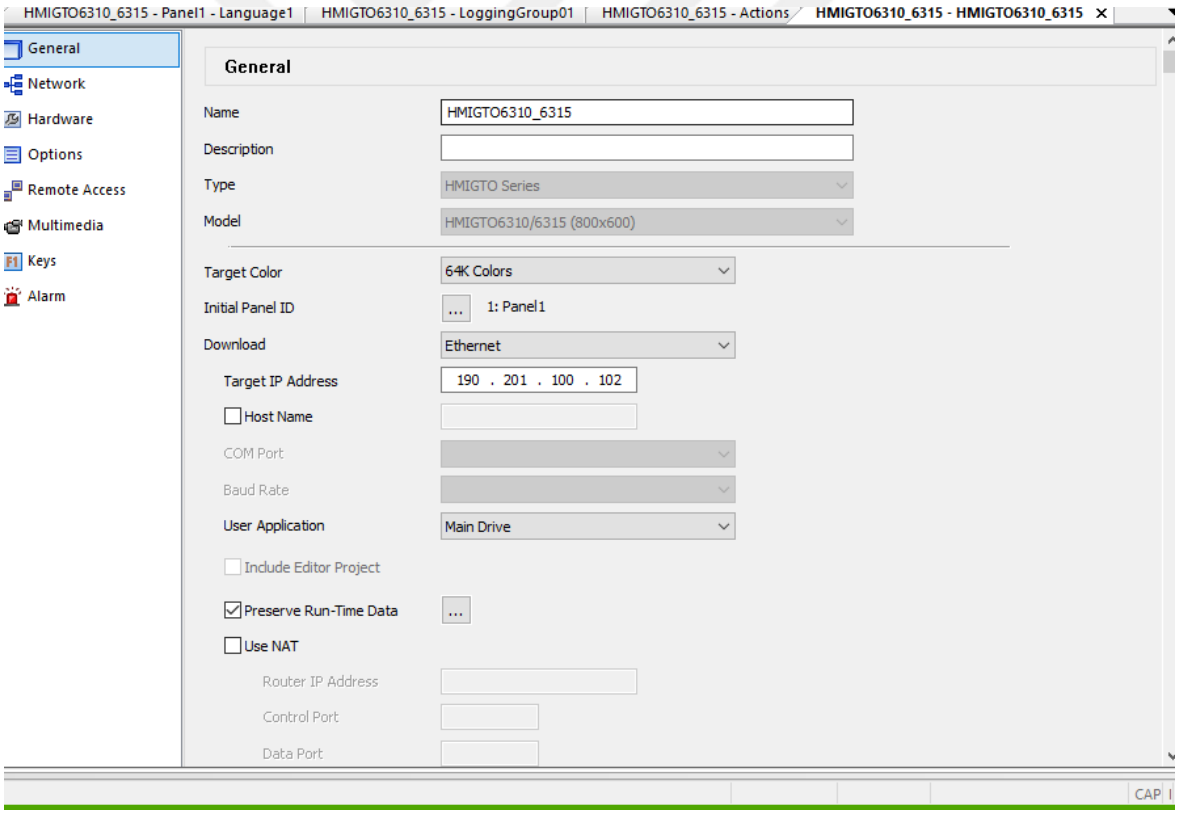
Bu bölümde HMI panel programlamanın nasıl yapıldığı anlatılmıştır. Vijeo-designer So-Machine içerisinde yer almaktadır üzerine tıklandığı takdirde program açılacaktır. Program içerisindeki haberleşme ayarları Resim 3.54, 3.55, 3.56'de gösterilmiştir.



Resim 3.54. Vijeo haberleşme ayarları-1

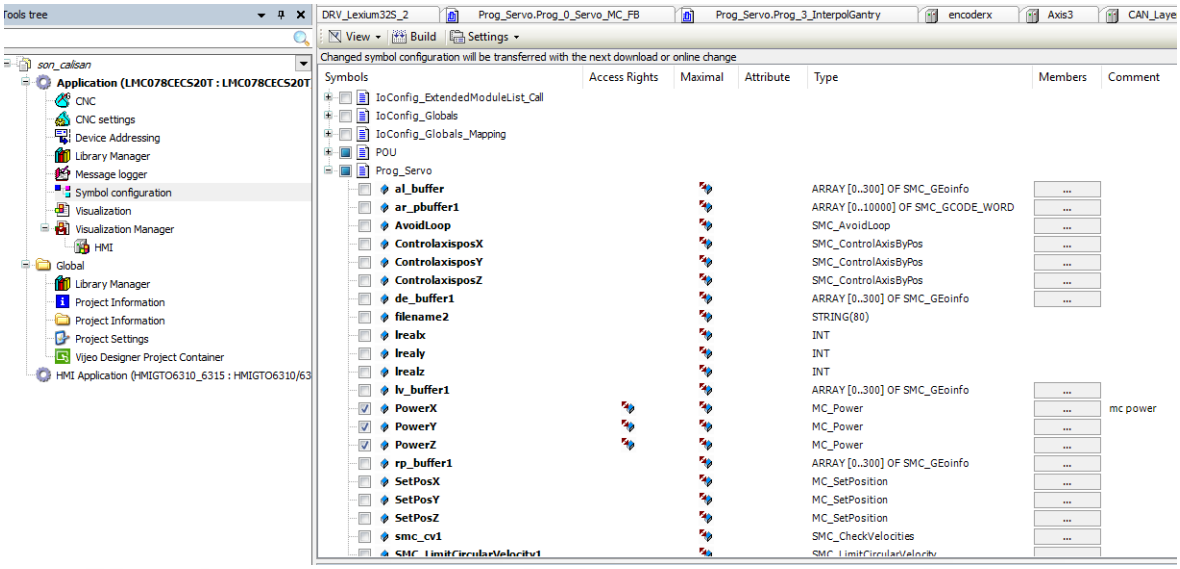


Resim 3.55. Vijeo haberleşme ayarları-2



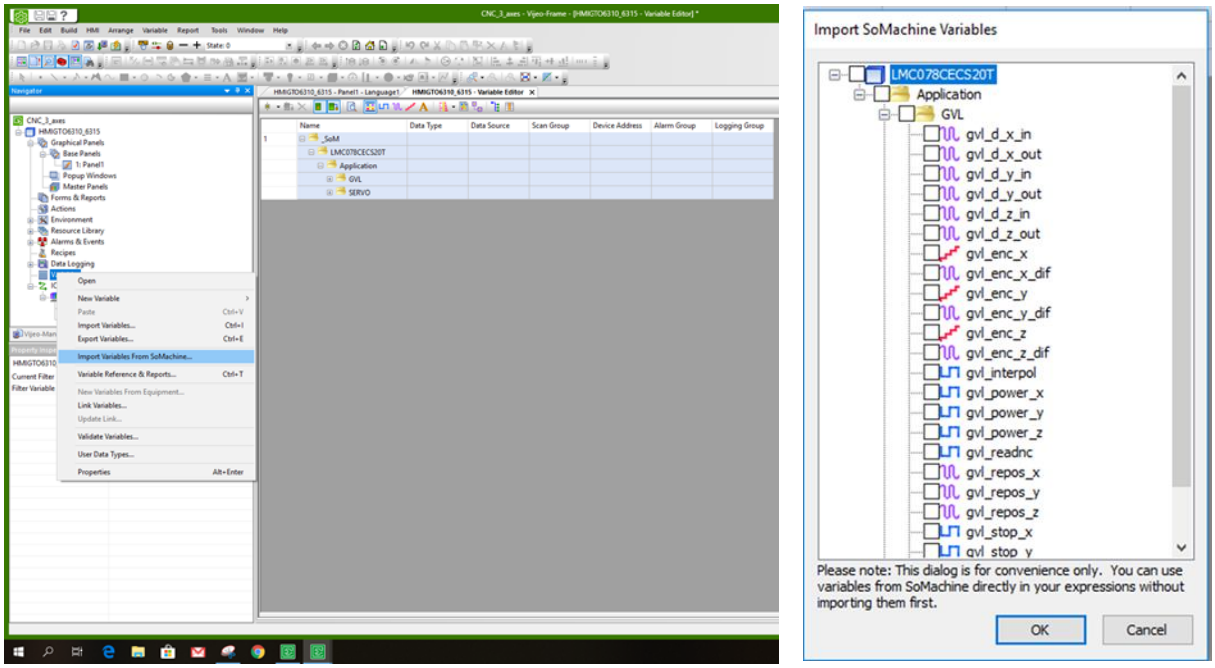
Resim 3.56. Vijeo haberleşme ayarları-3

So-machine içerisinde Symbol configuration tıklanırsa Resim 3.57'de yer alan ekran karşımıza çıkacaktır. HMI'da kullanılacak olan komutlar seçilerek yukarıda yer alan build tuşuna basılır.



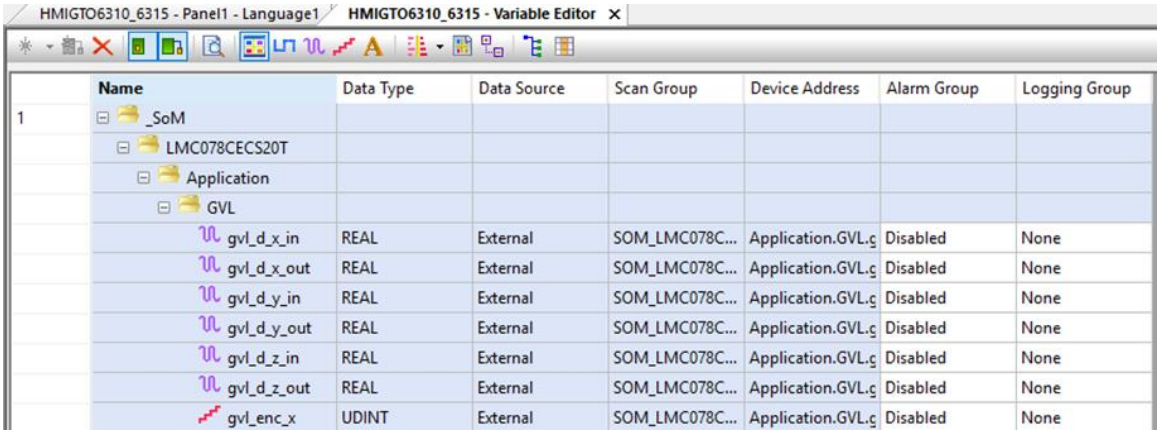
Resim 3.57. Değişken seçimi-1

Vijeo designer programı içerisinde Resim 3.58.a'daki gibi variable üzerinde tıklanarak import so machine üzerine basılır ve Resim-3.58.b'deki gibi çıkan ekranda istenilen komutlar yanına tik atılır. Seçilen değişkenler Resim 3.59'daki gibi görmek mümkün olacaktır.



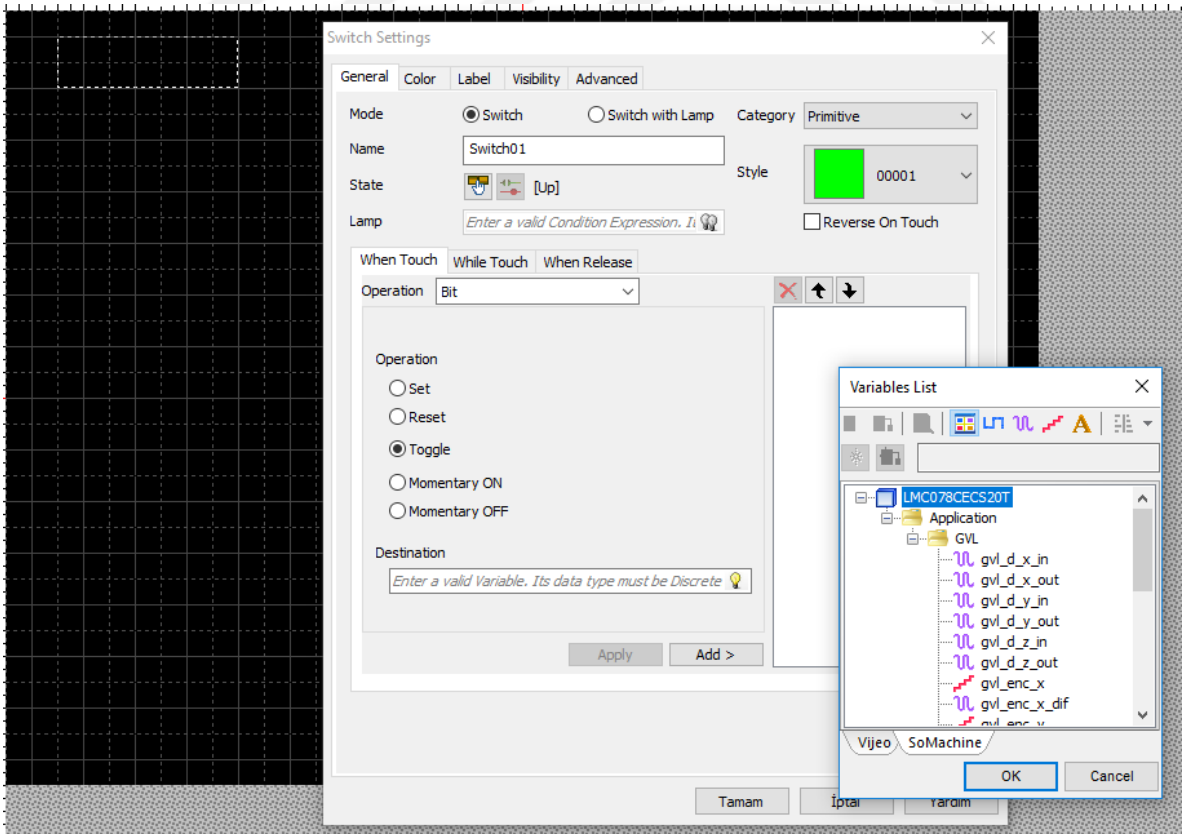
(a) (b)

Resim 3.58.(a) Değişken seçimi-2 (b) Değişken seçimi-3



Resim 3.59. Değişken seçimi-4

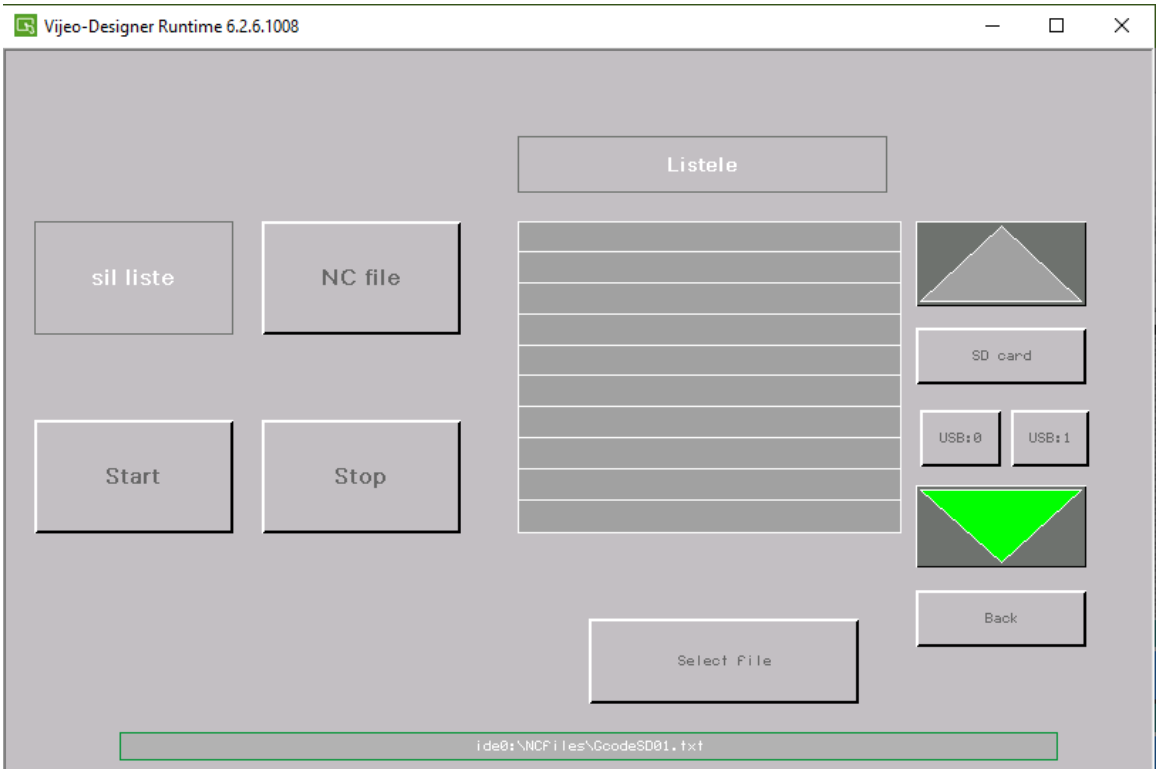
Eklemek istediğimiz verileri içeren butonlar ve sayılar Resim 3.60'daki gibi oluşturularak yukarıdaki build tuşuna basılır. Ardından ekran görüntüsü kontrol edebilmek için Simulate tuşuna basılarak simulasyonu yapılır. Yüklemek istenen panel download target'a basılarak ekrana yüklenir. Projede oluşturulan nihai ekran görüntüsü Resim 3.61'de ve Resim 3.62'de eklenmiştir.



Resim 3.60. Paneli veri atama



Resim 3.61 HMI panel görüntüsü



Resim 3.62 HMI panel görüntüsü

3.2.4. Kepserver programının kullanılması

KEPServerEX, kullanılan sistemin tipinden bağımsız olarak, tek bir sunucu arayüzü kullanarak proje geliştirmeyi kolaylaştırmaktadır. KepserverEX bir uygulamaya birden fazla cihaz sürücüsünün takılı olmasına, iletişimin merkezileştirilmesine ve kullanıcı yükünün azaltılmasına olanak sağlar. KEPServerEX kullanıcıya, tekli yapılandırma metodu ve güçlü program geliştirme araçları (tool) sunar. Kepware'in görevi, benzersiz uyumluluk ve performans sağlayan bir OPC server sunmaktır.

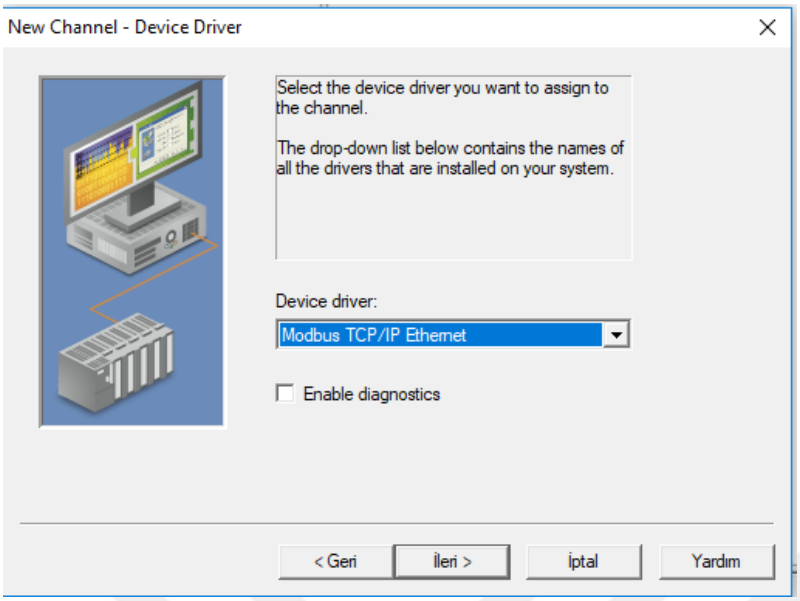
KEPServerEX, Server 32/64 bit windows uygulaması olup çok çeşitli endüstriyel cihaz ve sistemlerden veri ve bilgileri bilgisayarınızdaki client uygulamasına taşımayı sağlar. KEPserverEX'te haberleşme yapılacak OPC tipi seçilir. Seçilen haberleşme protokülüne cihaz eklenir. Cihazda okuma yazma yapılacak değişkenler ayarlanır.

Değişkenlerin işaretsiz veya işaretli olması önemli olup gönderilen veya alınan dataların doğruluğunu sağlamaktadır. Kepserver de OPC projesi oluşturmak için genel olarak gerekli adımlar; Sunucuyu başlattıktan sonra, Kanal oluşturmak, Cihaz oluşturmak, Tag'ları oluştumak şeklinde sıralanabilir [1].

KepserverEX programı çalışma yapılacak bilgisayara kurulduktan sonra çalıştırıldığında PLC ile bilgisayar arasında veri aktarımını çift yönlü olarak sağlayacaktır. PLC'den alınan veya gönderilen veriler OPC araçları bulunan Matlab, LabView gibi yazılımlara aktarılmaktadır [1].

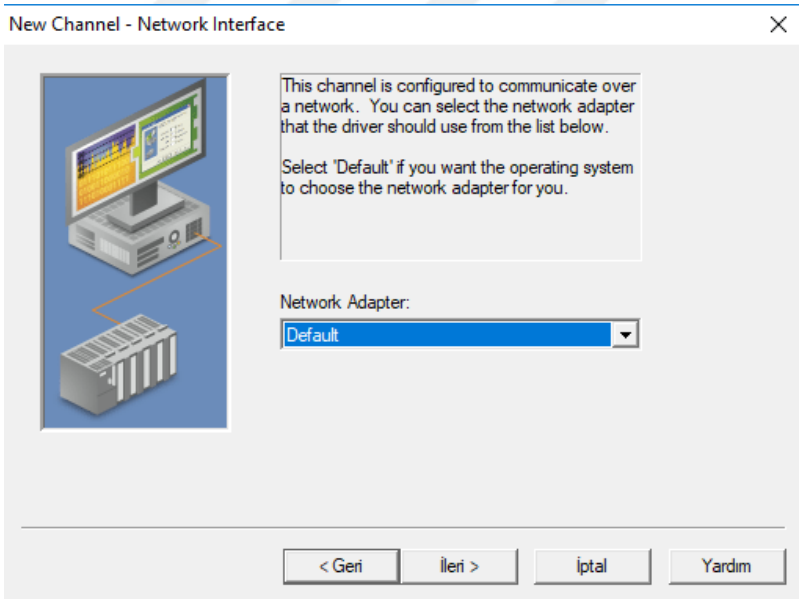
<u>Kepserver Programında Kanal Ekleme</u>

Yeni kanal eklemek için gerekli ayarlamalar Resim 3.63, 3.64, 3.65, 3.66, 3.67 arasında gösterilmiştir. Resim 3.63'de PLC ile PC arasındaki haberleşme protokolü olan MODBUS TCP/IP seçilir. Daha sonra ileri tuşuna basılır ve Resim 3.66'deki ekran karşımıza çıkacaktır.



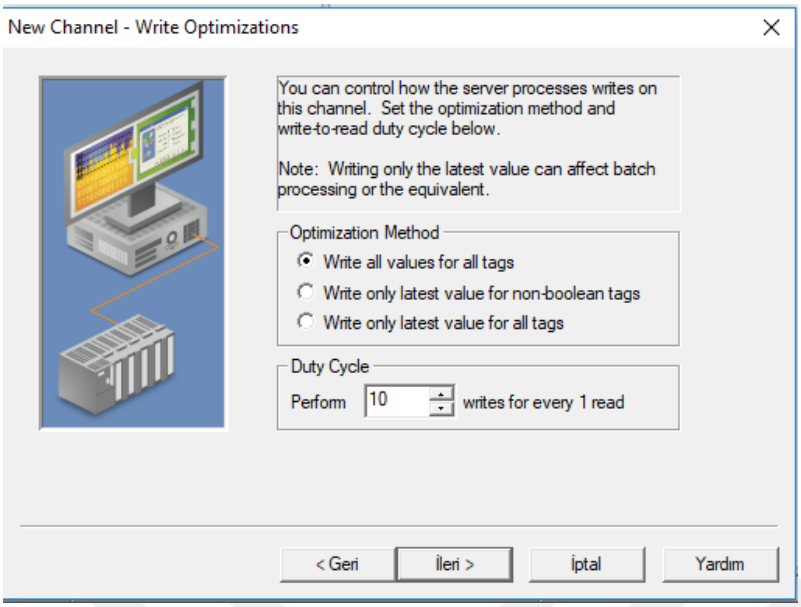
Resim 3.63 Kepserver kanal ekleme-1

Resim 3.64'de Ağ haberleşmesinde tanımlı Network adaptör seçimi yapılır. Daha sonra ileri tuşuna basılır ve Resim 3.65'deki ekran karşımıza çıkacaktır.



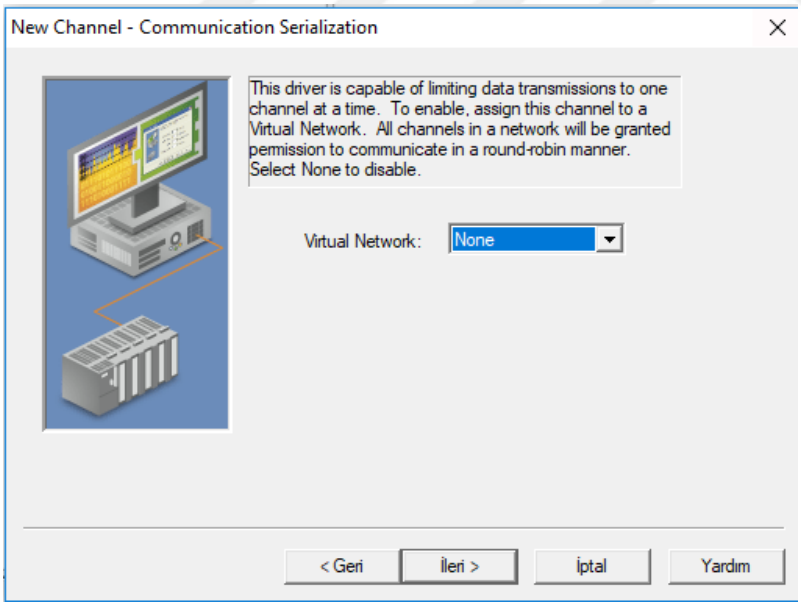
Resim 3.64. Kepserver kanal ekleme-2

Resim 3.65'de Haberleşme ağında verilerin hangi yapıda aktarılıp yazılması gerektiği sorgulanır. Daha sonra ileri tuşuna basılır ve Resim 3.66'daki ekran karşımıza çıkacaktır.



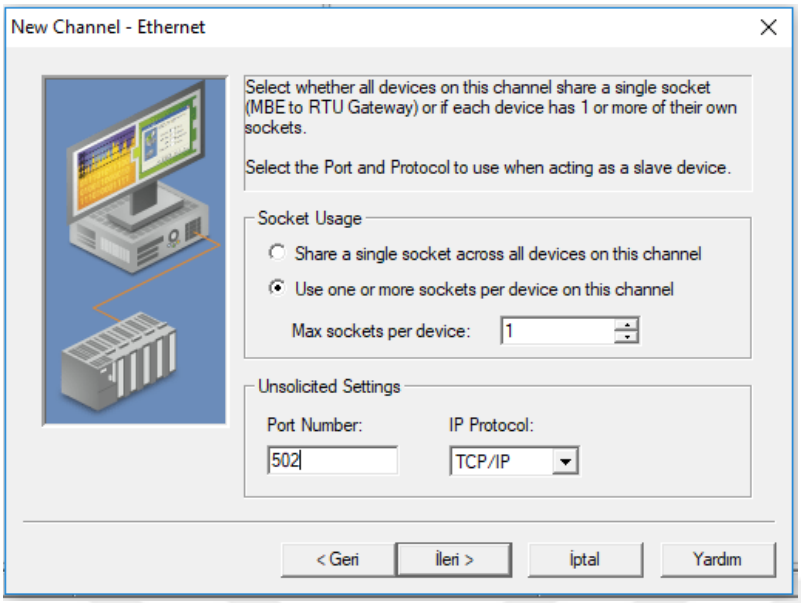
Resim 3.65. Kepserver kanal ekleme-3

Verilerin sınırlandırılması istenildiği takdirde Resim 3.66'daki ekranda none seçimi değiştirilmelidir. İleri tuşuna basılır ve Resim 3.67'deki ekran karşımıza çıkacaktır.



Resim 3.66. Kepserver kanal ekleme-4

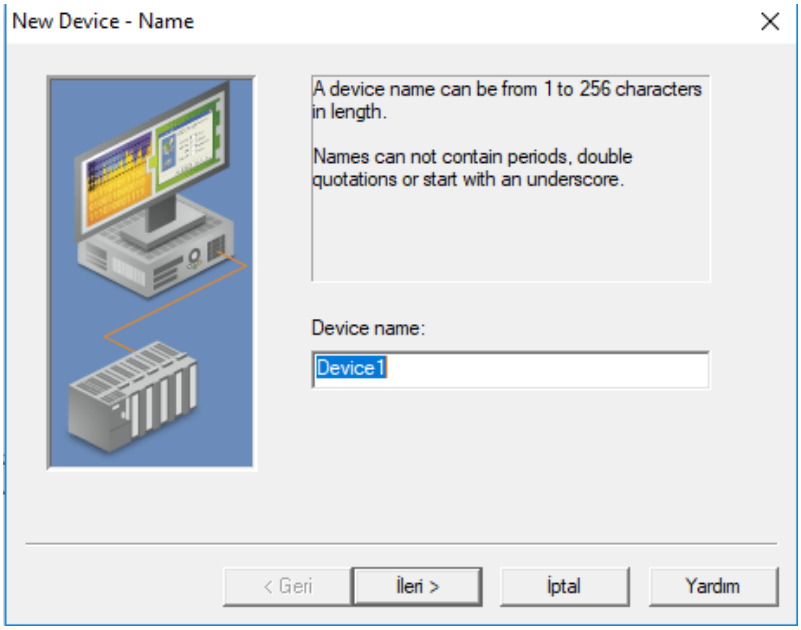
Resim 3.67'de Port numarası seçimi ve soket sayısı seçimi yapılmalıdır. Burada MODBUS protokolü için 502 seçilir. Daha sonra ileri tuşuna basılır ve Resim 3.68'deki ekran karşımıza çıkacaktır.



Resim 3.67. Kepserver kanal ekleme-5

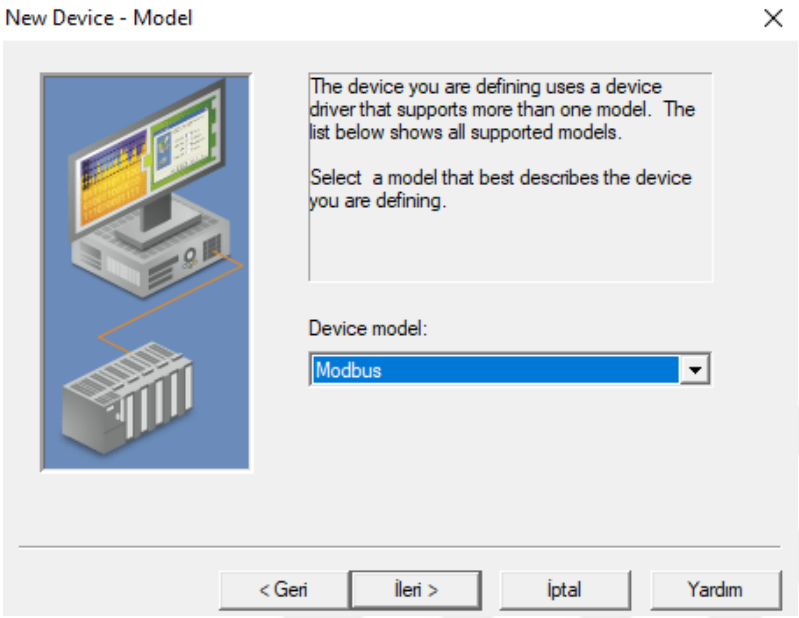
Kepserver programında cihaz ekleme

Eklenen kanala cihaz eklemek için yapılan ayarlar Resim 3.68,3.69, 6 3.70, 3.71, 3.72, 3.73, 3.74, 3.75, 3.76, 3.77, 3.78, 3.79, 3.80, 3.81, 3.82, 3.83'de sunulmuştur. Resim 3.68'de eklenecek cihaz ismi yazılır. Daha sonra ileri tuşuna basılır ve Resim 3.67'deki ekran karşımıza çıkacaktır.



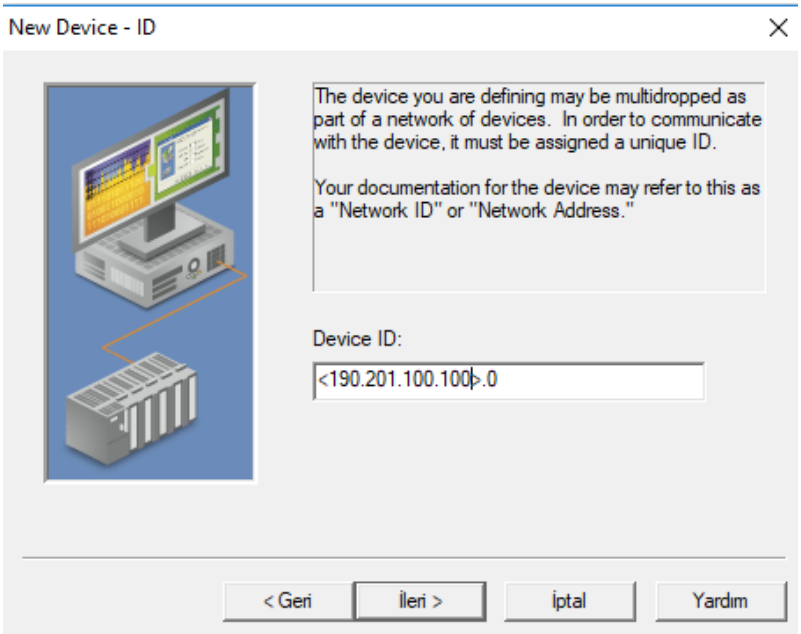
Resim 3.68 Kepserver cihaz ekleme-1

Resim 3.69'da cihaz modeli seçimi yapılır. Burada MODBUS seçilmiştir. Daha sonra ileri tuşuna basılır ve Resim 3.70'deki ekran karşımıza çıkacaktır.



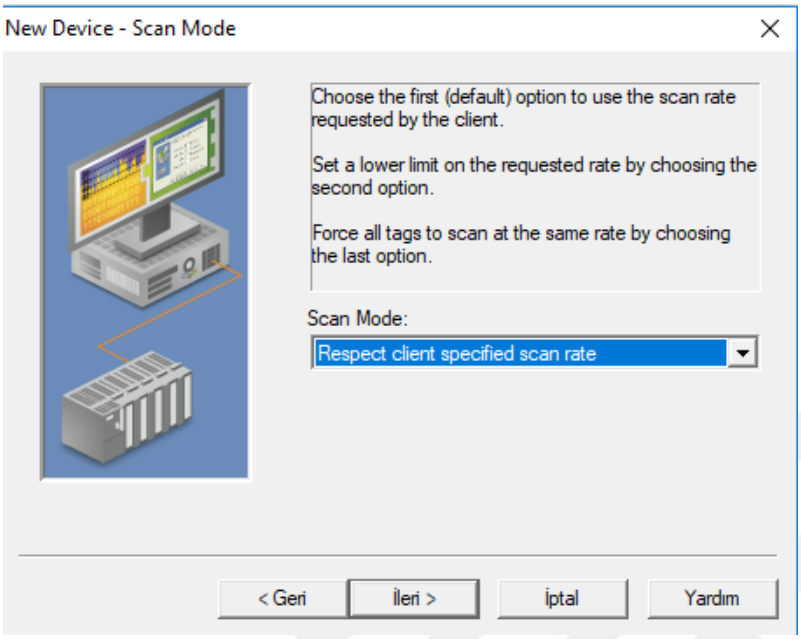
Resim 3.68 Kepserver cihaz ekleme-2

Resim 3.69'da Device ID kısmına PLC'nin IP numarası girilmiştir. Daha sonra ileri tuşuna basılır ve Resim 3.70'deki ekran karşımıza çıkacaktır.



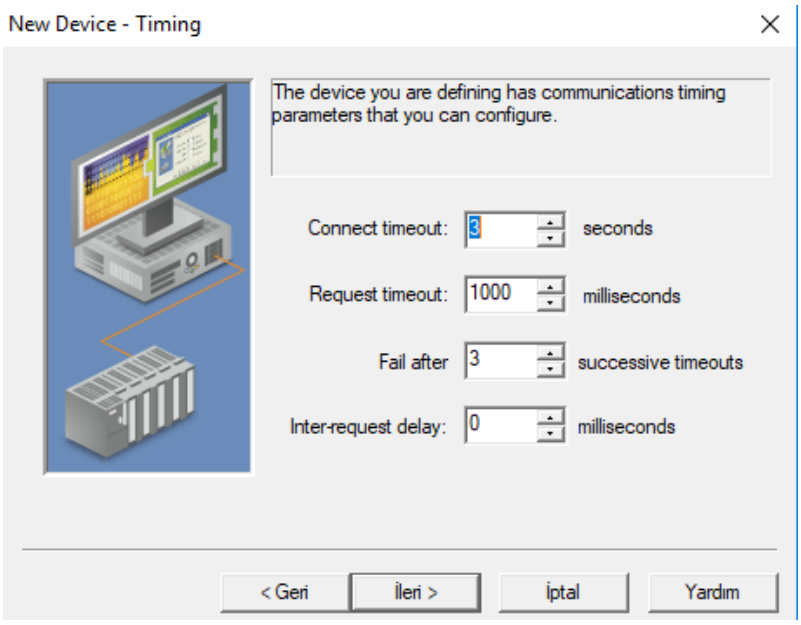
Resim 3.69. Kepserver cihaz ekleme-3

Resim 3.70'de veri yenileme hızı seçimi yapılmalıdır. Daha sonra ileri tuşuna basılır ve Resim 3.71'deki ekran karşımıza çıkacaktır.



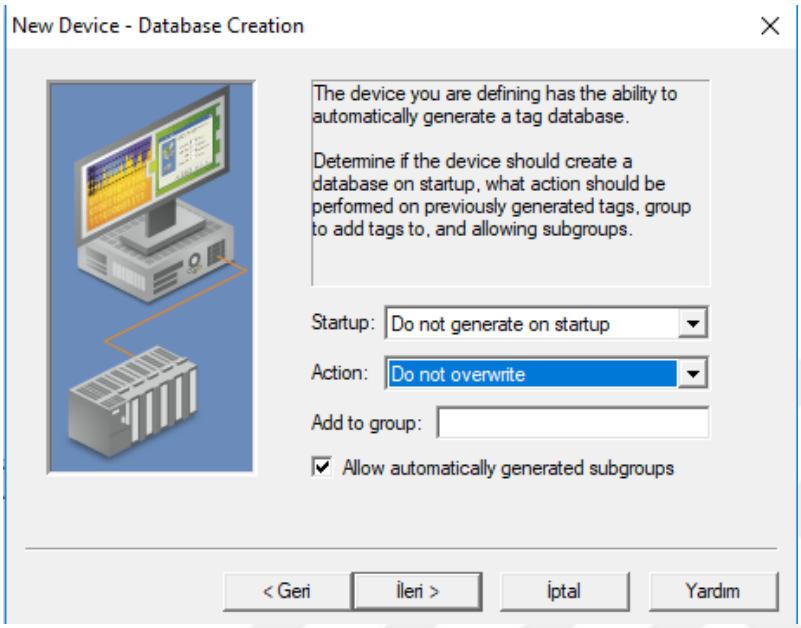
Resim 3.70. Kepserver cihaz ekleme-4

Haberleşme süresi ayarları Resim 3.71'de yapılmıştır. Daha sonra ileri tuşuna basılır ve Resim 3.72'deki ekran karşımıza çıkacaktır.



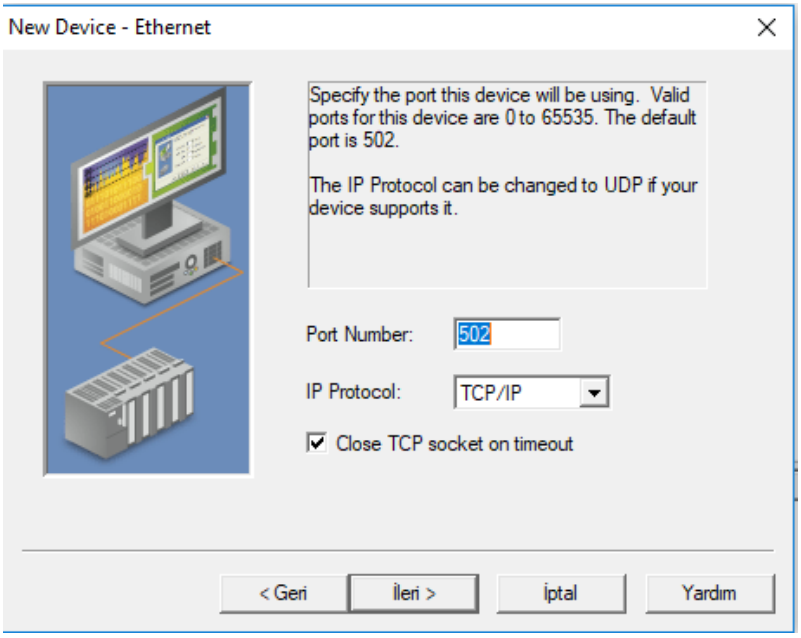
Resim 3.71. Kepserver cihaz ekleme-5

Haberleşme ağında PLC'den okunan verilerin değerlerinin veri tabanında otomatik yenilenmesi ve açılış esnasında göstermesi istenilen davranış biçimi Resim 3.72'de seçilir. Daha sonra ileri tuşuna basılır ve Resim 3.73'deki ekran karşımıza çıkacaktır.



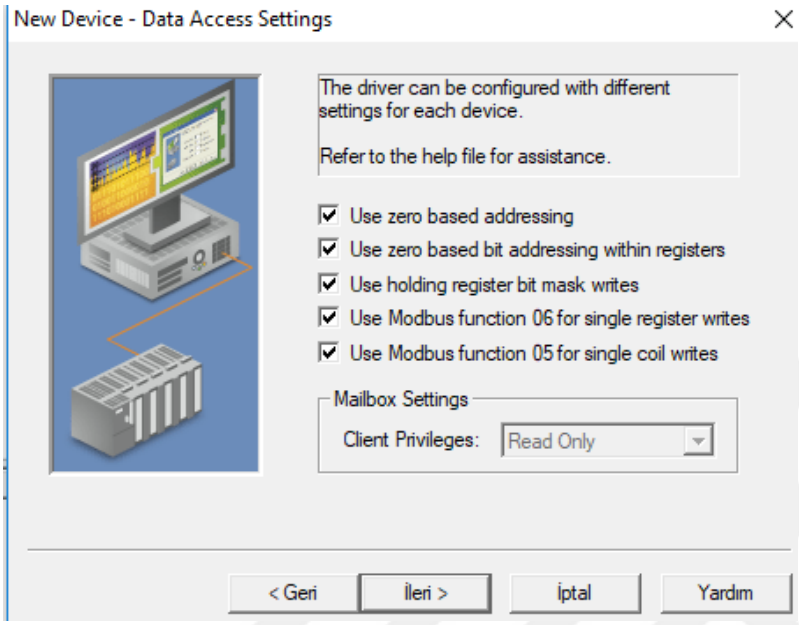
Resim 3.72. Kepserver cihaz ekleme-6

Ethernet ağ ayarlarında kullanılan protokol ve port numarası seçimi Resim 3.73'deki gibi yapılmıştır. Daha sonra ileri tuşuna basılır ve Resim 3.74'deki ekran karşımıza çıkacaktır.



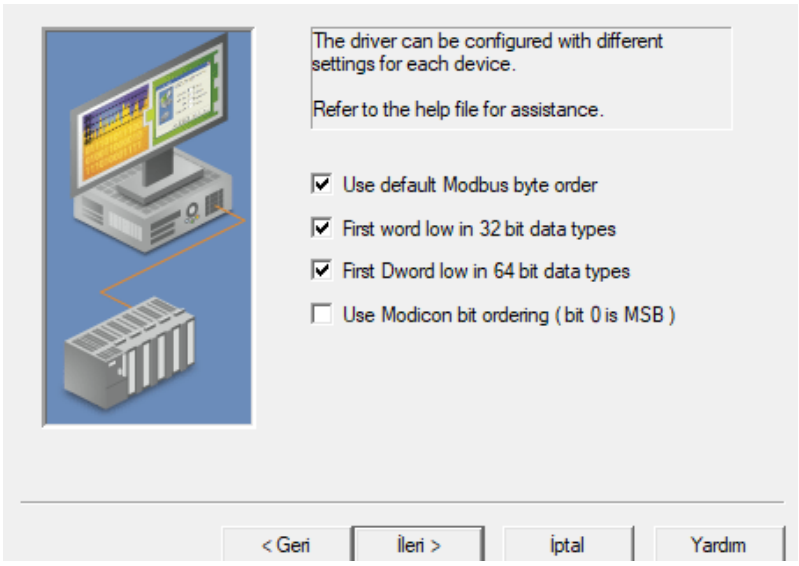
Resim 3.73. Kepserver cihaz ekleme-7

Resim 3.74'de olduğu gibi veri erişim ayarlarını eklenen her cihaz için farklı konfigürasyonda yapmak mümkündür. Daha sonra ileri tuşuna basılır ve Resim 3.75'deki ekran karşımıza çıkacaktır.



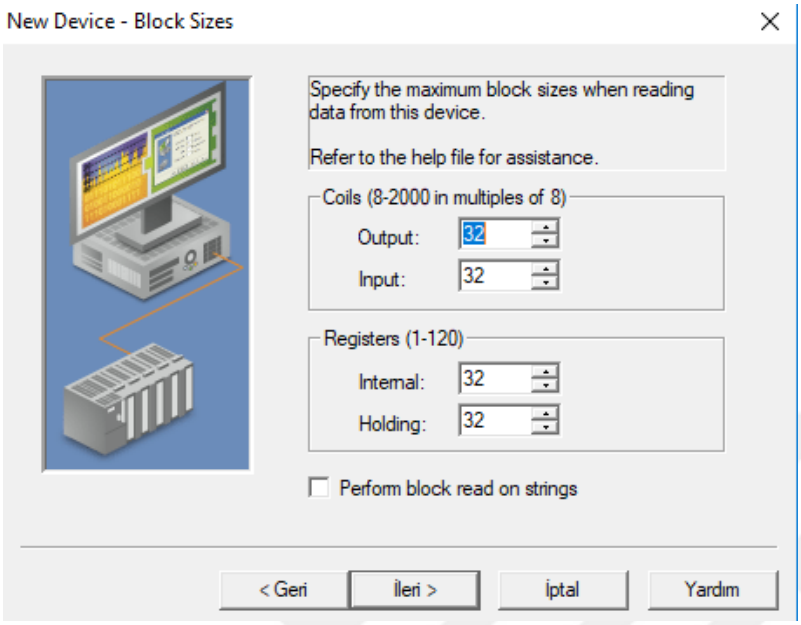
Resim 3.74. Kepserver cihaz ekleme-8

Haberleşmede kullanılan veri tipleri seçimi Resim 3.75'de yer alan ekranda yapılır. Daha sonra ileri tuşuna basılır ve Resim 3.76'daki ekran karşımıza çıkacaktır.



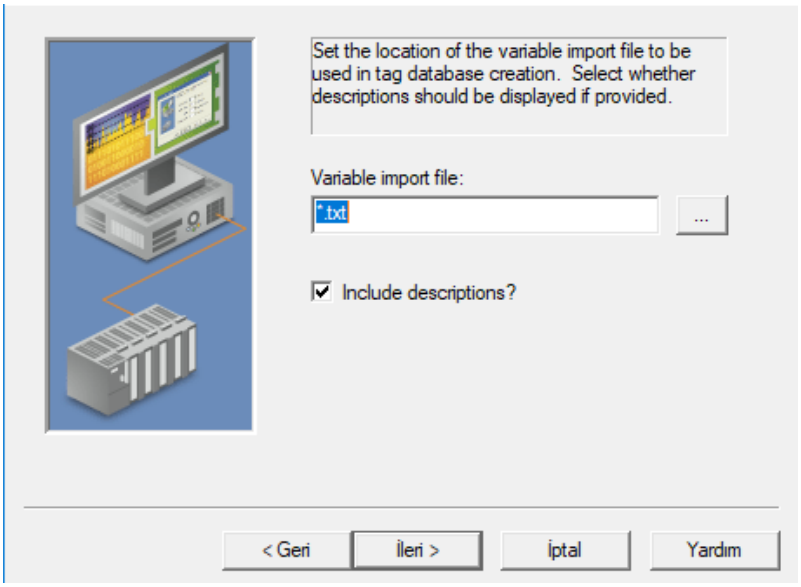
Resim 3.75. Kepserver cihaz ekleme-9

Haberleşme yapılırken veri paketinin kaç byte'a kadar olacağı Resim 3.76'de gösterilen ekranda belirlenir.



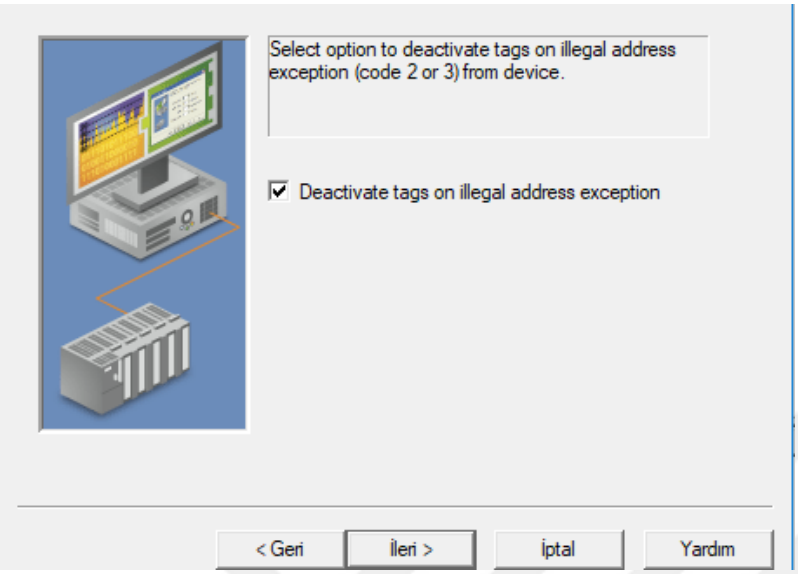
Resim 3.76. Kepserver cihaz ekleme-10

Yapılan projede atanan değişkenler liste halinde txt formatında saklamak mümkündür. Buna ilişkin görsel Resim 3.77'de sunulmuştur. İleri tuşuna basıldığında Resim 3.78'deki ekran karşımıza çıkacaktır.



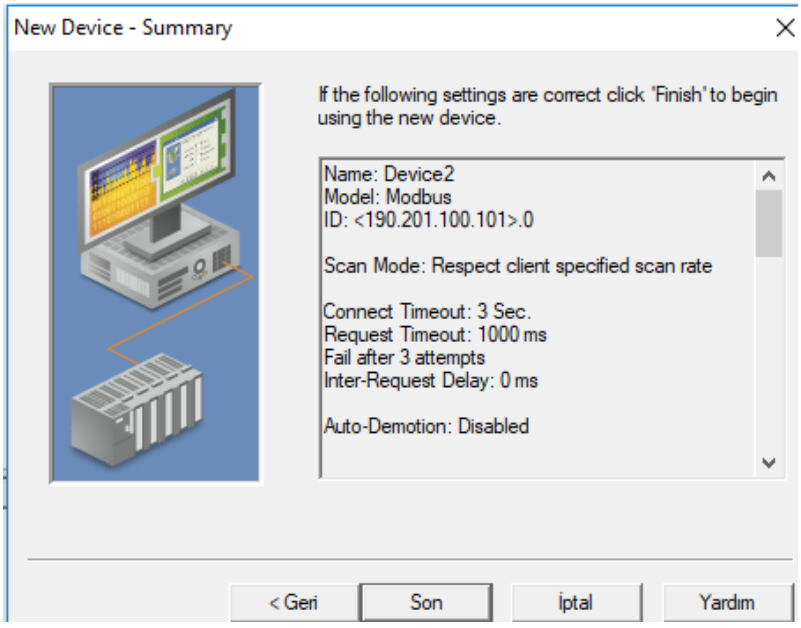
Resim 3.77. Kepserver cihaz ekleme-11

Haberleşme esnasında illegal adreslerden okuma yapılırken değişkenlerin aktif olup olmaması seçeneği Resim 3.78'de belirlenmiştir.



Resim 3.78 Kepserver cihaz ekleme-12

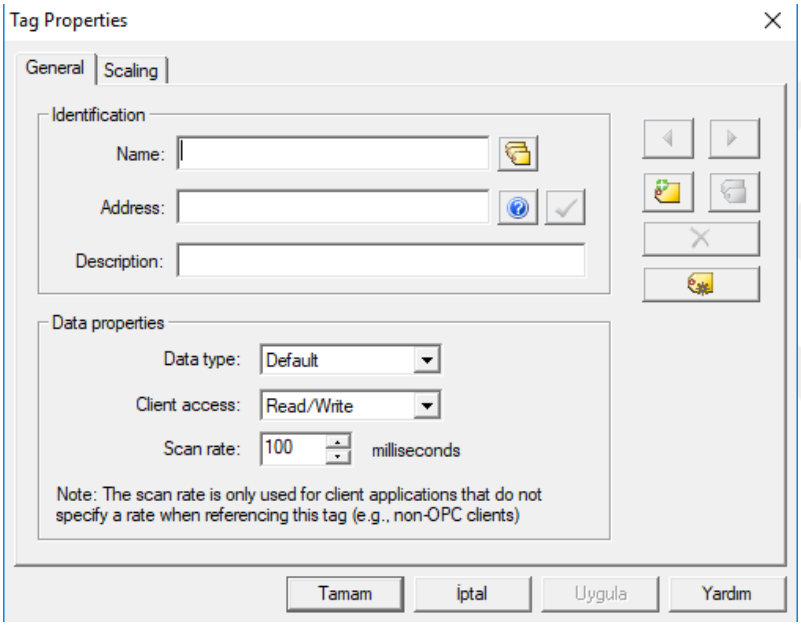
Yapılan haberleşme ayarlarının kontrolü Resim 3.79'de yapılarak son butonuna basılarak, cihaz ekleme işlemi tamamlanır. Daha sonra ileri tuşuna basılır ve Resim 3.80'deki ekran karşımıza çıkacaktır.



Resim 3.79 Kepserver cihaz ekleme-13

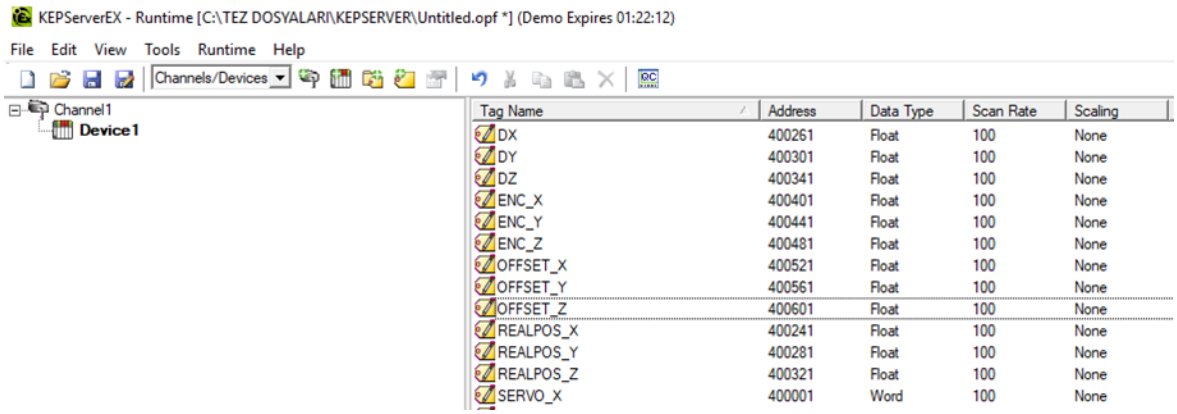
Kepserver programında adres ekleme

Kanal üzerinde cihaz eklenip tanımlandıktan sonra yukarıda bulunan Tag ikonuna tıklanır ve PLC'den çekilmesi istenen veriler burada eklenir. Resim 3.80'de sunulan ekran görüntüsünde; data properties kısmında eklenecek data'nın özellikleri girilir. İzlenecek olan değişkenin adı name kısmına yazılır, Data tipi seçimi yapılmalıdır, adres kısmında ise 40001 adresine, PLC'de yer alan değişken adresi iki ile çarpılır ve eklenir. Daha sonra ileri tuşuna basılır ve Resim 3.81'deki ekran karşımıza çıkacaktır.



Resim 3.80 İzlenecek adres ekleme-1

Kepserver arayüzünde değişkenler eklendikten sonraki hali Resim 3.81'de gösterilmiştir. Runtime üzerine tıklanarak sistemin çalışması başlatılır ve daha sonra OPC seçeneğinin aktifleştiği görülür, daha sonra OPC ikonuna tıklanır.



Resim 3.81. İzlenecek adres ekleme-2

4. ADAPTİF KONTROL

4.1. Klasik Kontrol Sistemleri

CNC tezgâhlarında kontrol devreleri açık devre ve kapalı devre olmak üzere iki şekilde gerçekleştirilir. Açık kontrol devresinde motora verilen bilgiler, motoru ve buna bağlı olan vidalı mili harekete geçirir ve kızağı istenilen konuma getirir.

Kapalı kontrol devresinde, kontrol sisteminden çıkan bilgiler, konum algılayıcı dan alınan bilgiler doğrultusunda motora verilir, motor ve iletim sistemi harekete geçer ve vidalı mili istenilen konuma getirilir. Bu sistemde vidalı milin konumu bir ölçme elemanı(enkoder) tarafından kontrol edilir.

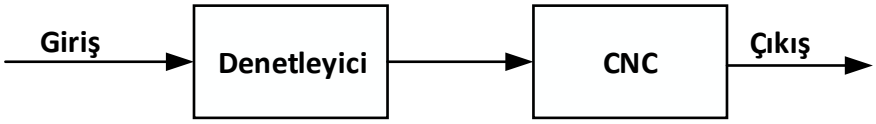
Konum algılayıcı vidalı milin gerçek konumunu sürekli olarak ölçer ve program bloğuna gönderir; burada değeri istenilen konum ile karşılaştırır ve fark olduğu durumda bu fark bir servo mekanizmanın yardımıyla giderilir.

4.2.Kontrol Sistemi Tanımı

Bir sistemi kumanda etmek üzere birleştirilen kontrol elemanlarıdır. Kontrol sistemi, insan faktörü en az olacak şekilde istenen performansı sağlamak üzere bir araya getirilen makine, süreç ve diğer donanımlarının otomatik çalışmasıdır. Denetim sistemleri, denetlenen niceliklerin değerlerini sabit tutar ya da bu değerleri, önceden belirlenmiş biçimde değişmesini sağlar.

4.2.1. Açık döngü kontrol sistemi

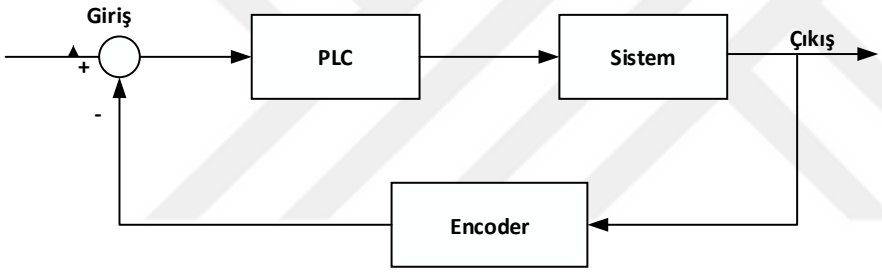
Kurulan sistemin denetim sonuçlarından bağımsız olduğu sistemdir. Açık-döngü denetimde istenen çıkışın denetlenen çıkış değişkeni karşılaştırması bulunmamaktadır. Şekil 4.1’de Açık Döngülü Kontrol Sistemi blok şeması verilmiştir.



Şekil 4.1. Açık döngü kontrol sistemi

4.2.2. Kapalı döngü kontrol sistemi

Kurulan sistemin denetim sonuçlarına bağlı olduğu sistemdir. Denetleme sonucundaki etkisi ise negatif geri besleme ve pozitif geri besleme olarak iki çeşittir. Negatif geri beslemede çıkış girişe ters yönde etki etmekte ve pozitif geri beslemede çıkış girişe aynı yönde etkilemektedir. Şekil 4.2'de Kapalı Döngü Kontrol Sistemi verilmiştir.



Şekil 4.2. Kapalı döngü kontrol sistemi.

4.3.Denetleyici Etkileri

Kapalı döngü kontrol sistemi içinde denetleyicinin görevi, ölçme elemanı üzerinden geri beslenen çıkış büyüklüğünü, giriş büyüklüğü ile karşılaştırmak ve karşılaştırmadan ortaya çıkabilecek hata değerinin yapısına ve kendi kontrol etkisine bağlı olarak uygun bir kumanda veya kontrol sinyali üretmektir [21].

Kullanılan başlıca 4 temel denetleyici etkisi şunlardır:

1. Aç-kapa kontrol etkisi
2. Orantılı kontrol etkisi (P etkisi)
3. Integral kontrol etkisi (I etkisi)
4. Türev kontrol etkisi (D etkisi)

4.3.1. Aç-kapa kontrol

Aç- Kapa kontrol cihazı set değeri üstünde veya altında ayar değişkenini açar veya kapar. Kontrol cihazının çıkışı iki konumda bulunur ya tamamen açık, ya da tamamen kapalıdır.

4.3.2. Oransal kontrol

Denetleyici çıkışı bir oransal sabit yoluyla denetleyici girişine oranlanır. K kazanç yani orantı sabitidir. Oransal etkilerde herhangi bir anda hata ne kadar büyükse, düzeltici sinyal o oranda büyük olur. Denetleyici çıkışı m(t) ile hata sinyali e(t) arasındaki bağlantı aşağıda sunulmuştur.

$$m(t) = Kp.e(t) \quad (4.1)$$

Oransal kontrol cihazı sistemin istediği enerjiyi aralıksız olarak ayarlayarak verir. İhtiyaç duyulan enerji ile verilen enerji arasında bir denge vardır. Elektrik enerjisi kullanılarak soğutma yapılan bir sistemde, oransal kontrol cihazı soğutucunun elektrik enerjisini sistemin sıcaklığını set edilen değerde tutabilecek kadar, sistemin ihtiyaç duyduğu kadar verir. Enerjinin % 0'dan % 100'e kadar ayarlanabildiği, oransal kontrol yapılabilen sıcaklık aralığına oransal band denir. Set değeri ile sistemin sabitlendiği sıcaklık arasındaki farka kayma (off-set) denir. Kaymayı azaltmak için oransal band azaltılabilir. Ancak, daha önce de belirtildiği gibi oransal band azaldıkça, açık-kapalı kontrolle yaklaşıldığı için set değeri etrafında salınımlar çoğalabilir. Geniş oransal band da kaymanın fazla olacağı düşünülerek sisteme en uygun oransal bandın seçilmesi gerekir. Sıcaklık azalır, birkaç kere set değeri etrafında salınım yaptıktan sonra set değerinin üzerinde veya altında sabit bir sıcaklık farkı ile gelip sabitlenir. Kayma artı veya eksi değer alabilir.

4.3.3. Diferansiyel (Türev) kontrol

Diferansiyel kontrolde hatanın türevi alınır ve çıkış elde edilir. Diferansiyel kontrol ancak negatif ve pozitif değişmelerde devreye girer. Ölçülen değer sabit olduğunda diferansiyel kontrolün etkisi olmaz. Diferansiyel etki denetleyicilerde tek başına kullanılmaz. Kd diferansiyel zamanıdır. Yüksek gürültü içeren yerlerde ve gecikmenin çok olduğu sistemlerde diferansiyel kontrol yöntemi kullanılmamalıdır. Diferansiyel kontrolün zaman

cinsinden ifade edilmektedir. Zamanı büyüdükçe diferansiyel kontrolün tepkisi çoğalır. 0ffsetler ve overshootlar düşmektedir.

$$m(t) = K_d \cdot \frac{d_e}{d_t} \tag{4.2}$$

4.3.4. İntegral kontrol

İntegral kontrolde hatanın integrali alınır ve giriş değeri elde edilir. Ki integral zamanıdır. Orantı sonucunda ortaya çıkan hatayı gidermenin yolu, integral elemanına hatanın integralini alan denetim mekanizması ilave edilir. İntegral etki çıkışı geçmişte meydana gelen hataların toplamı ile doğru orantılıdır. Hatanın integralinin büyük olduğu anlarda, düzeltme etkisi de büyük olmaktadır. Genellikle oransal etki ile beraber kullanılır.

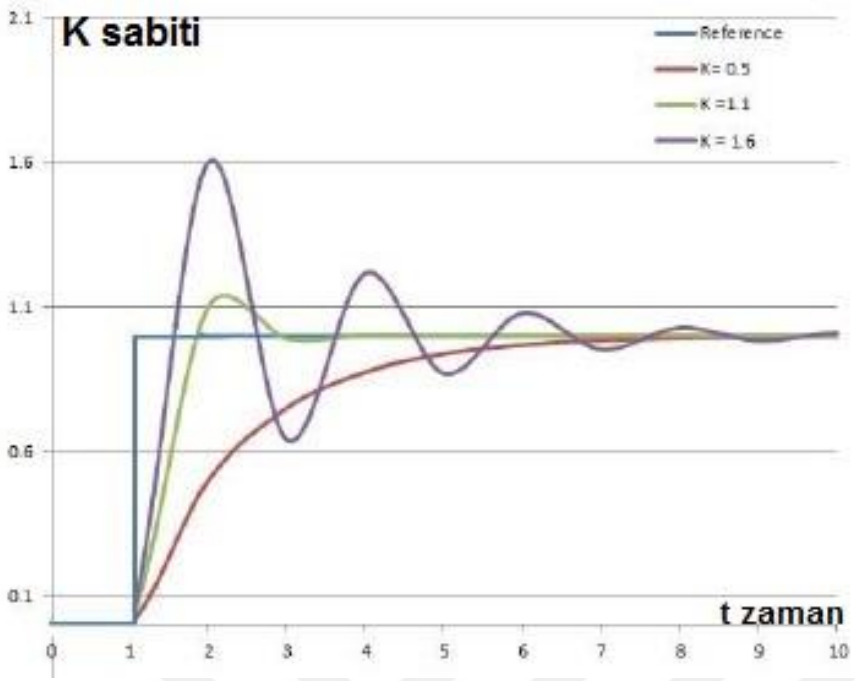
$$m(t) = K_i \int_0^t e(t) \cdot dt \tag{4.3}$$

4.3.5. Oransal + integral + türevsel kontrol (PID)

Kontrolün yeterli bulunmadığı sistemlerde Oransal+integral+Türevsel kontrol tercih edilmelidir. Oransal kontrolde oluşan kaymalar (offset) oransal ve integral kontrol ile giderilir. Ancak, meydana gelen aşmalar (overshootlar) bu kontrole türevsel ile en aza indirilir.

$$m(t) = K_i \int_0^t e(t) \cdot dt \tag{4.4}$$

Şekil 4.3'de oransal+integral+türevsel kontrolün reaksiyon eğrisi verilmektedir. PID kontrolde diğer kontrol türelerine oranla yok denecek kadar az aşma/azalma vardır ve kayma oldukça azaltılmış durumdadır. P, I, D parametrelerinin iyi ayarlanıp ayarlanmasına bağlı olarak elde edilen kontrol eğrisi değişmektedir.

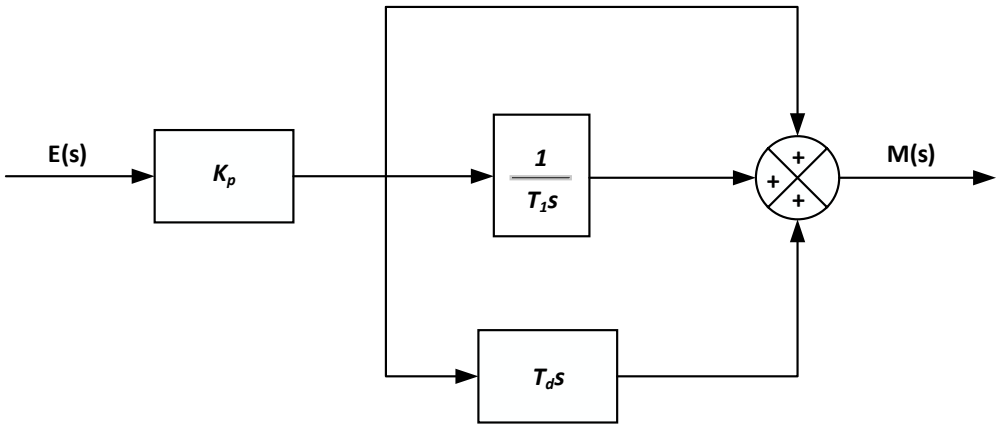


Şekil 4.3. Farklı K_P değerlerinin sistem cevabına olan etkisi

Şekil 4.4'de PID Kontrol Sistemi Blok Şeması bulunmaktadır. Burada görüldüğü üzere denetleyiciden geçen m(t) sinyali, oransal kazanç (Kp) ile hata değeri, integral kazancı (Ki) ile hatanın integrali, türevsel kazanç (Kd) ile hatanın türevi çarpımlarının toplamıdır.

$$m(t) = K_p e + Ki \int_0^l e.dt + Kd \frac{de}{dt} \tag{4.5}$$

Oransal denetleyicilerin (Kp), yükselme zamanını azaltmada etkisi vardır ancak kararlı hal hatasını tamamen yok etmez.



Şekil 4.4. PID kontrol blok şeması

(Ki) kararlı hal hatasının azaltılmasında etkisi vardır ancak bu durum geçici tepkinin daha kötü hale gelmesine sebebiyet verebilir. Türevsel denetleyicinin (Kd) sistemin kararlılığının artmasında etkisi vardır, aşmayı azaltır ve geçici tepkiyi düzeltir. Kapalı döngülü bir sistemde, her bir denetleyicinin etkisi Kp, Kd ve Ki aşağıdaki Çizelge 4.1'de özet olarak gösterilmiştir.

Çizelge 4.1. Kp, Ki, Kd Katsayılarının çıkışa etkisi.

Katsayı	Yükselme Zamanı	Aşma	Yerleşme Zamanı
K_p	Azalır	Artar	Az Değişir
K_i	Azalır	Artar	Artar
K_d	Az Değişir	Azalır	Azalır

Çünkü Kp, Ki ve Kd birbirlerine bağımlı değerlerdir bu nedenle bu düzeltmeler tam olarak geçerli değildir. Değişkenlerden birinin değişimi diğer ikisinin etkisinin değişmesine sebep olabilir. PID kontrol en eski kontrol tekniklerinden olmakla birlikte teknik çeşitli sebeplerden dolayı yaygın olarak kullanılmaktadır.

4.4. Adaptif Kontrolün CNC'ye Uyarlanması

Adaptif bir kontrol sistemi, değişen ortamlarda özelliklerini ayarlamaya yetecek kadar akıllı bir geri besleme kontrol sistemi olarak tanımlanabilir, böylece belirli performans hedeflerinin yerine getirilmesi sağlanır.

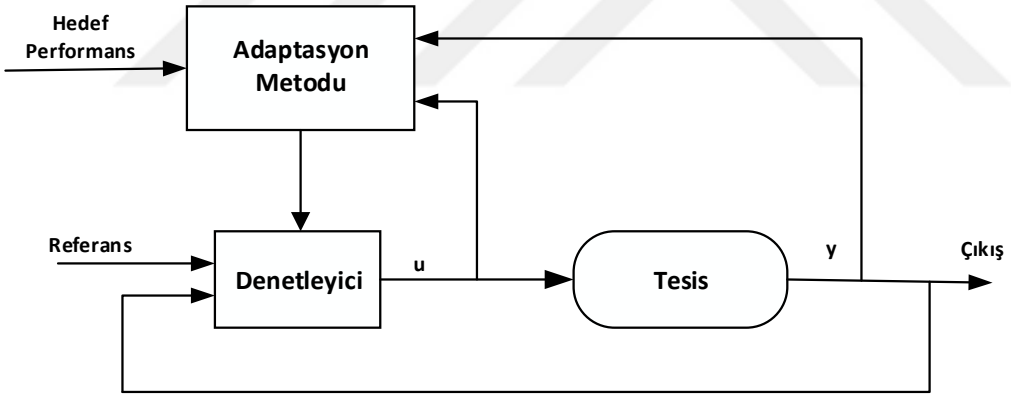
Bir sistemde en iyi performansı elde edebilmek amacıyla, ölçülen değişken verilerinin denetleyiciye iletilir ve geri beslemeli hale gelmesi sağlanır. Yeni değişken verilerin otomatik olarak uyarlandığı bir yöntemidir. Sistemden anlık olarak haberdar olunur ve ortaya çıkabilecek hatalara yönelik olarak gerekli önlemlerin alması yani adaptif olması sağlanır.

Adaptif kontrol sistemi, kontrol sisteminin özelliklerini ayarlayarak genel sistem performansının aynı kalmasını veya en uygun seviyede tutulmasını sağlayacak şekilde ayarlayarak sistem dinamikleri değişikliklerini otomatik olarak telafi eder.

Bu kontrol sistemi zaman içerisinde tesis performansındaki herhangi bir bozulmayı dikkate almaktadır. Ayarlanabilir kontrol sistemi, sistem dinamiklerini ölçmek (veya tahmin etmek) ve denetleyici özelliklerini buna göre değiştirmek için diğer yardımcı ekipmanları içerir. Denetleyici, genel sistem performansını koruyacak şekilde kontrol özelliklerini ayarlar.

Adaptif kontrol ile tesisin dinamik modelinin parametreleri bilinmediğinde veya zamanla değişme durumunda, performans seviyesinin korumak için denetleyicilerin gerçek zamanlı ve otomatik olarak ayarlanması amaçlanmaktadır.

Adaptif kontrol sistemi, optimal kontrol sistemden veya geri beslemeli kontrol sistemlerinden farklıdır. Geri beslemeli ya da optimal sistemler bilineni gerçekleştirirler. Eğer çevredeki değişimler çok hassas ise bu sistemler istenilen biçimde tasarlanamaz. Adaptif sistem çevreyi ve sistemin performansını değerlendirir ve kontrol karakteristiğinde gerekli değişikliği yapar.



Şekil 4.5. Adaptif kontrol

Şekil 4.5'de sunulan bloğa göre denetleyici parametreleri u ve y'ye bağlı olduğundan adaptif bir kontrol sistemi doğrusal değildir. Adaptif kontrol tanımlama, karar ve değişim fonksiyonlarını içerir: Bunları herhangi bir adaptif kontrol sisteminde bileşenlerine ayırmak zordur. Bu üç fonksiyon adaptasyonun meydana gelmesi için sistemde bulunmalıdır[22].

4.4.1. Tanımlama fonksiyonu

Sistemin uygun performansının belirlenmesini ifade eder. Normalde sistemin performansı performans indeksi ile belirlenir. Tanımlama fonksiyonu sistemden veri geri beslemesi kullanarak performans ölçümünün, mevcut değerinin belirlenmesiyle ifade edilir. Sistemin performansı çevre faktörü ile zamanla değişmesi sebebiyle tanımlama fonksiyonu anlık değişim göstermektedir. Sisteme girilen tanımlamalar birçok ölçülebilen faaliyet içerebilir. Sistem değişkenlerinin ölçümünden performans indeksinin hesabı veya sistemin uygun matematik modeli belirlenir. Tanımlama fonksiyonu, geçerli performans kalitesi ile hedeflenen ortalama performansın karşılaştırılmasını ifade eder.

Genellikle adaptif kontrolün tanımlama fonksiyonu, performans indeksi değerini ya da problemi tarif eden matematiksel modelin değerinden oluşur. Örneğin talaş kaldırmada, performans indeksi, matematiksel denklem olarak kaldırılan talaş hacminin birim maliyeti olabilir ve sistemde matematik modelin etkilerine bakılmalıdır.

Pratik problemlerde değişikliklere genelde izin verilir. Sistemde performans değişimi varsa adaptif kontrol yapılmaya adaydır. Adaptif kontrol sistemlerinde en zor problem, tanımlama problemi olmakla birlikte çözümü çeşitli varsayımlara ve kabullere dayanmaktadır. Tanımlama fonksiyonu verilen sistemi en uygun şekilde temsil etmelidir. Tanımlama fonksiyonu Sistemin tanımlandığı yerde uygun olabildiği gibi sistemin tanımlandığı yerde uygun olmayabilir. Bu durumda sistemin matematik modelinin kullanımı uygun olmayabilir ve yapılan ölçümler ile sistemin performans kalitesi değerlendirilebilir.

4.4.2. Karar fonksiyonu

Adaptif kontrol fonksiyonunun performansı iyileştirmek için sistemin hangi parametrelerle ayarlanacağına karar verilmeden önce sistem performansını belirlemek gerekir. Bu karar, tasarımcının sağladığı ön programlar ile yerine getirilir. Denetleyicinin iç parametrelerinin bazıları bu kararla bazıları da ilerde verilecek kararlarda değiştirilebilir.

4.4.3. Düzenleme fonksiyonu

Düzenleme fonksiyonu sistemde çevresel şartlar sonucunda meydana gelebilecek mekanik ve fiziksel değişimleri değerlendirir. Adaptif fonksiyonun kararını yürütmek için kullanılmaktadır. Karar fonksiyonu yazılımsal bir fonksiyon olarak, düzenleme fonksiyonunu ise donanım fonksiyonu olarak düşünebiliriz. Düzenleme sistemin daha ortalama bir hale getirmek için sistem parametrelerinin değişimini içermektedir. Sistemin zamanla çevreden etkilendiği kabul edilir. Adaptif kontrolün düzgün bir şekilde yapılabilmesi için mekanik sistemde gecikmeler fazla olmamalı ve sistem dinamiği anlaşılır biçimde tasarlanmalıdır.

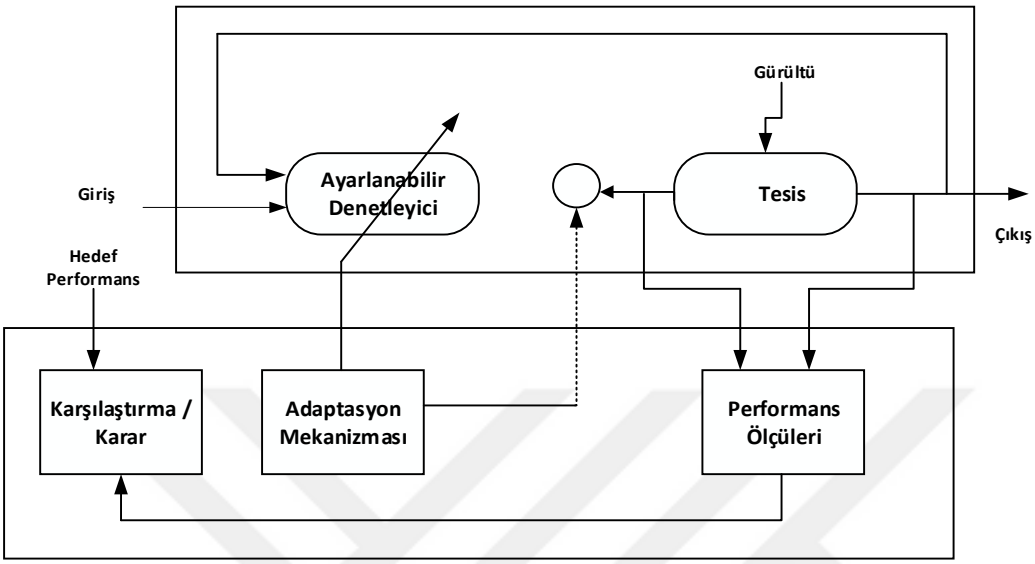
Bir talaş kaldırma ele alındığında, Adaptif Kontrol Sistemi (AKS) tanımlanan çıkış sistem değişkenlerini ölçer ve bunları hız, ilerleme, kesme derinliği v.s.yi kontrol elemanları olarak kullanılır. Adaptif kontrolde kullanılan bazı sistem değişkenleri iş sehimi, kuvvet, tork, kesme sıcaklığı, titreşim genliği ve güçtür. Diğer bir deyişle hemen hemen tüm metal kesme değişkenleri deneysel adaptif kontrol sistemi ile ölçülebilir. Adaptif kontrollü talaş kaldırma işlemi, sistemi daha verimli yapmaya çalışır. İşleme performansının ölçümünde birim hacim talaş kaldırma maliyeti veya Talaş kaldırma hızı kullanılabilir. Adaptif kontrollü işleme konusu, nümerik kontrolün mantıksal bir göstergesidir [23].

Adaptif kontrol hem geri beslemeli sistem hem de ortalama kontrol sistemi özelliklerine sahiptir. Bir geri beslemeli sistem gibi belirli sistem değişkenleri ölçülür. Ayrıca optimal sistem gibi toplam performans ölçümünde kullanılır. Diğer iki tipten ayırt edici özelliği zamanla değişen ortam şartları için tasarlanabilirler. Çevredeki değişimi telafi etmek için bir adaptif kontrol sistemi tasarlanır.

Adaptif kontrol, istenen sistem performansını elde etmek için parametrik, yapısal ve çevresel belirsizlikleri olan bir sistem için denetleyiciyi ayarlayan adaptasyon mekanizmaları sağlar. Yük değişimi veya bileşen yaşlanması parametrik belirsizliklere neden olur, bileşen arızası yapısal belirsizliklere yol açar ve dış gürültüler tipik çevresel belirsizliklerdir.

Sistem parametrelerinin olası tüm değerleri için, belirtilen performans ölçülerinin denetleyici parametrelerini uygun değerleri hedef performans değerleri ile karşılaştırarak elde edilebileceği kalıplaşmış bir yapıya ve karmaşıklığa sahip adaptasyon mekanizması

vardır. Adaptasyon döngüsünün görevi sadece denetleyici parametrelerinin ‘iyi’ değerlerini aramaktır. Şekil 4.6’da Adaptif Kontrolün Temel Konfigürasyonu sunulmuştur.



Şekil 4.6.Adaptif Kontolün Temel Konfigürasyonu

4.5. Adaptif Kontrol Sınıflandırmaları

Yüksek performanslı kontrol sistemlerinde; tesisin kontrol parametreleri bilinmediğinde veya zamana bağlı olarak değiştiğinde kontrol cihazının hassas şekilde ayarlanmasını sağlar. Adaptif Kontrol teknikleri, denetleyici parametrelerinin otomatik olarak eşzamanlı ayarlanması için sistematik bir yaklaşım sunar. Adaptif kontrol teknikleri bazı lineer olmayan ve pratikte çözülemeyen kontrol problemlerinin yaklaşımı olarak düşünülebilir. Adaptif kontrolün amacı tesis parametreleri bozulduğunda kabul edilebilir performans seviyesini elde etmektir.

4.5.1. Basit Adaptif Kontrol Sistemleri

İki farklı adaptif kontrol sistemi sınıfı bulunmaktadır. Doğrudan adaptif kontrole dayalı olanlar ve dolaylı adaptif kontrole dayalı olanlar.

Doğrudan adaptif kontrol

Denetleyici parametrelerinin doğrudan adaptif bir yasaya göre düzenlendiği kontrol sistemleridir. Doğrudan adaptifı kontrol için yaygın olarak kullanılan iki tasarım bulunmaktadır. Bunlar Lyapunov tasarımı ve gradyan tasarımlarıdır.

Dolaylı uyarlamalı kontrol

Doğrudan adaptif kontrolün aksine önce fabrika parametreleri ve denetleyici parametreleri uyarlanır, daha sonra bilinmeyen tesis parametreleri, tahminler kullanılarak bazı cebirsel tasarım denklemleri kullanılarak hesaplama yapılır.

Kontrol sistemi analizi için sistem modellemesi, sinyal ölçümü, geri bildirim yapısı, geri bildirim kazancı, sinyal sınırlılığı, sistem kararlılığı ve hata yakınsama bulunur. Uyarlanabilir kontrol için bazı önemli konular, denetleyici yapılarının seçimi ve parametrelendirilmesi, hata sistemlerinin türetilmesi, tahmin hatalarının seçimi, hata modellerinin ifadesi, uyarlanabilir güncelleme yasalarının geliştirilmesi, kararlılık ve sağlamlık analizi ve sistem performansının değerlendirilmesidir. Birçok sistem parametresi belirsiz bağların varlığında hem dengeleme hem de izleme elde etmek için benzersizdir [23].

4.6. Adaptif Kontrol Yöntemleri

CNC tezgahlarda yapılan tralaş kaldırma işlemlerinde kullanılan adaptif yöntemler; Optimizasyonlu Adaptif Kontrol, Geometrik Adaptif Kontrol, Kısıtlı Adaptif Kontrol yöntemleri şeklinde sıralanabilir.

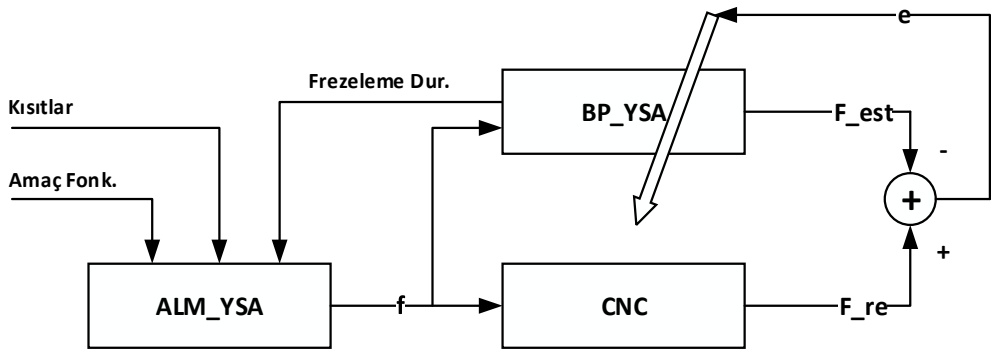
4.6.1. Optimizasyonlu adaptif kontrol (OAK)

Optimizasyonlu Adaptif Kontrol sisteminde; üretim hızı en yüksek, işleme zamanı en düşük, en az işleme maliyeti gibi kriterlere bağlı olarak belirlenen ortalama kesme hızı, takım ömrü, takım aşınması gibi bir hedef fonksiyonu belirlenir ve tezgâhın çalışması bu kritere göre sağlanır.

Performans indeksi (birim hacim talaş kaldırma maliyeti veya maksimum üretim hızı) tüm sistemin performansını gösterir. Optimizasyonlu adaptif kontrol ile hedeflenen, uygulamadaki ilerleme hızının ayarlanması ve performans indeksinin optimize edilmesidir.

Şekil 4.7'de blok şeması verilmiştir. Burada bir frezeleme işleminde ölçülen kesme kuvvetleri (F_real) ile tahmin edilen kesme kuvvetleri (F_est) arasındaki fark (e) geriye yayılma algoritması kullanan YSA'nın ağırlıklarının ayarlanmasında kullanılmıştır. Çıkışta

frezeleme durumuna ilişkin gerçek zamanlı bilgiler Optimizasyon Algoritmasıyla beraber 2. bir YSA'ya (ALM_YSA) verilmiştir.[24]



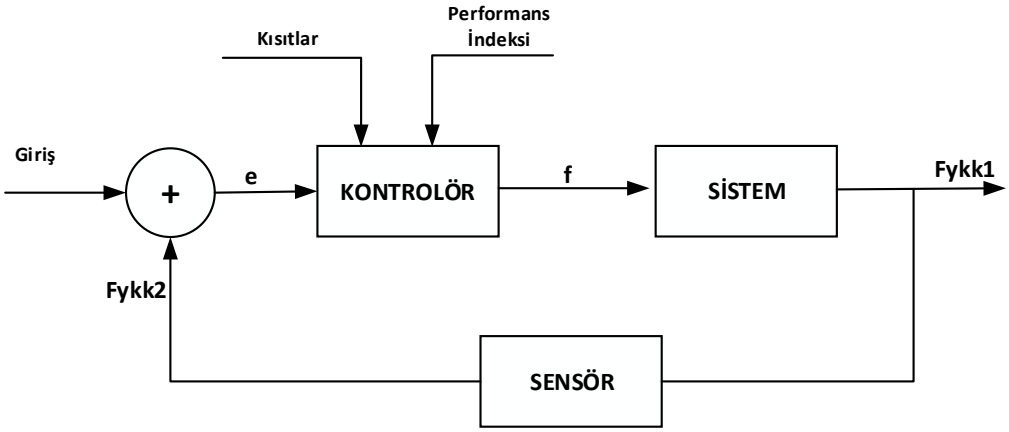
Şekil 4.7. Optimizasyonlu adaptif kontrol [24]

4.6.2. Geometrik adaptif kontrol (GAK)

Geometrik Adaptif Kontrol Sistemi genellikle sonlandırma uygulamalarında kullanılır. Bu sayede takım aşınması ve malzeme yapısındaki düzensizliklere rağmen belirlenen parça kalitesinde üretimin sürekliliği sağlamış olur.

4.6.3. Kısıtlı adaptif kontrol (KAK)

Kısıtlı Adaptif kontrol sisteminde; sistemi etkileyen bir değişken, belirlenen bir değerde sabit tutulmakta aynı anda diğer değişkenler sınırlanmaktadır. Örneğin talaş kaldırma sisteminde, sabit tutulan parametre: kesme kuvveti, motor gücü, yüzey kalitesi vb. olabilir. Şekil 4.8'de Tornalama işleminde KAK uygulanmasına ait blok şema görülmektedir. Burada f İlerleme, e hata, Frkk Referans kesme kuvveti, Fykk yeni kesme kuvvetidir.



Şekil 4.8. Kısıtlı adaptif kontrol

Hedef zorunlu sınır değerleri içinde ölçülebilen değişkenleri sürdürebilmek için ilerleme ve kesme hızı ayarlamaktır. Burada performans indeksi olarak iş parçasından talaş kaldırma hızının takım aşınma oranına bölünmesi ile elde edilen değerdir. Performans İndeksinin maksimum olması hedeflenir.

$$P\dot{I} = \frac{TKH}{TAH} \tag{4.6}$$

Pİ = Performans İndeksi
TKH = TalaşKaldırma Hızı
TAH = Takım Aşınma Hızı

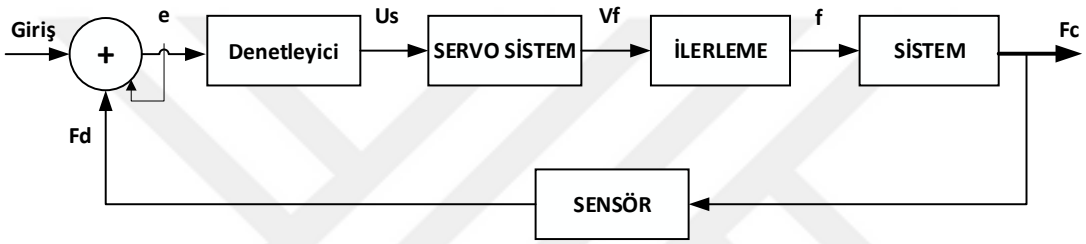
İşlemede Kısıtlı Adaptif Kontrol şu şekilde sınıflandırılmıştır [20].

-Geri Beslemeli Denetleyici Tabanlı KAK,
-Kendi Kendini Ayarlayan Kontrol Tabanlı KAK,
-Model Referans Adaptif Kontrol Tabanlı KAK,
-Değişken Yapılı Sistem Tabanlı KAK,
-Yapay Zeka Tabanlı KAK ‘dür.

Geri beslemeli denetleyici tabanli KAK sistemi

KAK Sistemlerinde sabit kazançlı geleneksel geri beslemeli denetleyici kullanılmaktadır. Bu geri beslemeli sistemin denetleyicisi bir PID, PI veya PD dir. PID Denetleyici aşağıdaki gibi tanımlanabilir. Şekil 4.9'da gösterilmiştir. Denklemde; *Kp* oransal kazanç, *Ti* integral zaman sabiti, *Td* türev zaman sabitidir.

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) \cdot dt + T_d \frac{de(t)}{dt} \right] \quad (4.7)$$



Şekil 4.9. İntegral kısıtlı adaptif kontrol

Buradaki integral denetleyicili KAK sabit tornalama kuvvetini temel almaktadır. Sistem bir integral denetleyici, servo sistem, ilerleme, kesme prosesi ve sensörlerden oluşur. Sistemin giriş ve çıkışları aşağıdaki gibi açıklanabilir [26].

Burada; ***Us*** Kontrol çıkışı, ***Kc*** Denetleyici kazancı, ***T*** Kontrol çevrimi, ***e*** Çıkış hatası, ***Vf*** İlerleme komutu, ***Km*** servo kazanç, ***ε***servo sönümleme oranı, ***ωn*** servo dogal frekansı, ***f*** İlerleme, ***n*** devir sayısı, ***Fc*** Gerçek Kesme kuvveti, ***ap*** kesme derinliği, ***Ks*** kesme ilerlemesi kazancı, ***u*** nonlineer indeks, ***Fd*** Ölçülen kesme kuvveti, ***Fr*** Referans kesme kuvveti ve ***KD*** dönüşüm faktörüdür. Yukarıdaki eşitliklerden sistemin transfer fonksiyonu;

$$U_s(i) = K_c \, T \sum E(i) \quad (4.8)$$

$$U_s(i) = K_c \, T \sum E(i) \quad (4.9)$$

$$f = \frac{60}{n} \cdot V_f \quad (4.10)$$

$$F_c = K_s \cdot a_p \cdot f^u \quad (4.11)$$

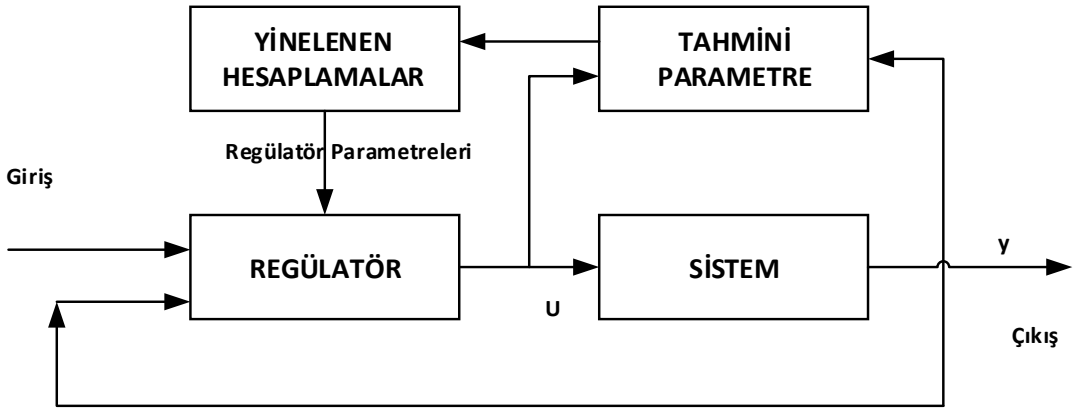
$$F_d = K_D \cdot F_c \tag{4.12}$$

$$G(s) = \left[\frac{K_m \cdot \omega_n^2}{s^2 + 2 \cdot \varepsilon \cdot \omega_n \cdot s + \omega_n^2} \cdot G_u(s)\right] \tag{4.13}$$

Burada *K=(60/n).Kc.T.Km.KD.Ks.ap* sistemin açık döngü kazancı, *Gu(s)* sistemin lineer olmayan bölümüdür. KAK sistemlerinde ana problem kazancı ayarlamadır. Anlık değişen sistemlerde kazanç ayarlamak zordur bu durumda sistem düşük performanslı ve kararsız çalışır.

Kendi kendini ayarlayan denetleyici tabanli kak

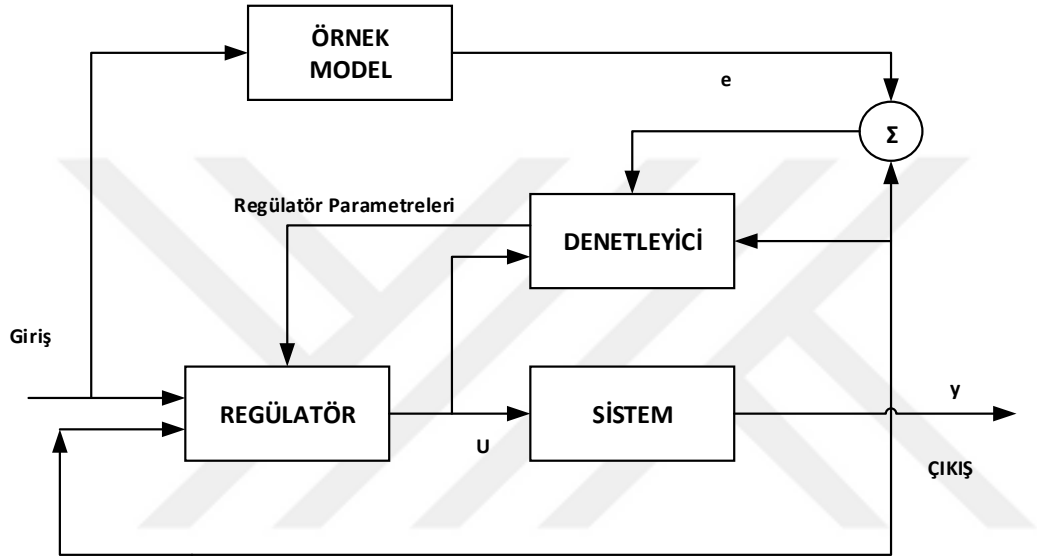
Sistemin değişen şartlarda parametreleri tahmin edilerek, her ölçüm aralığında düzenleyici parametrelerinin tekrar hesaplanması gerçekleştirilmektedir. Sistemin sabit olduğu, parametrelerin bilinmediği kabul edilir. İstenen performans parametrelerini set edecek düzenleyici tasarlanır. Şekil 4.10'da Kendini Ayarlayan Adaptif Kontrol Blok şeması verilmiştir.



Şekil 4.10. K.A. Adaptif kontrol blok şeması

Model referans adaptif kontrol tabanli KAK(MRAK)

Hedeflenen performans referans model olarak belirlenir. Sistem, denetleyici ve geri beslemeye sahiptir. Denetleyici parametreleri "e" hata işaretinden alınan geri besleme ile değiştirilir. Geri besleme iç çevrim, parametre ayarına ait geri besleme dış çevrim olarak tasarlanır. Şekil 4.11'de Model referans adaptif kontrol blok şeması verilmiştir.



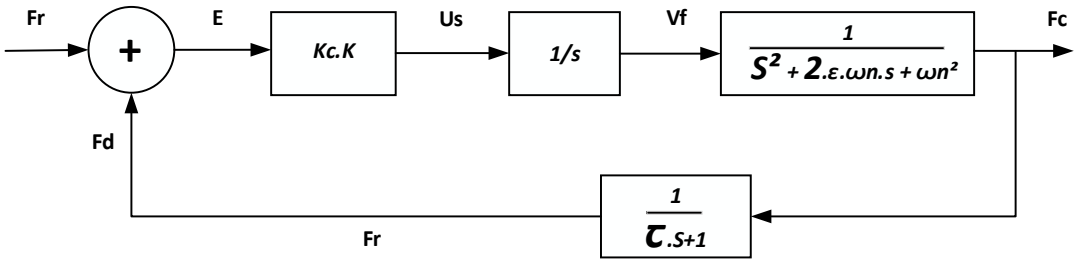
Şekil 4.11. M.R. adaptif kontrol blok şeması

Değişken yapili sistem tabanli KAK

Değişken yapılı sistemin amacı, durum uzayında kayar yüzey olarak isimlendirilen bir yörünge oluşturmaktır. Kayar Mode Kontrol olarak adlandırılan değişken yapılı sistem kayar mode göre negatif bir kontrol çıkışı ile pozitif bir kontrol çıkışı arasında anahtarlama yapar. Değişken yapılı kontrol sistemi, hızlı cevap ve doğru performans sağlayabilir. Bu özelliğinden dolayı işleme sistemlerine uygulanabilir[20]. Freze sistemlerinde Değişken Yapılı Kontrol Şekil 4.12'de verilmiştir. Her model ünitesinin farklı tornalama sistemleri için farklı transfer fonksiyonu vardır. Uzama ölçer tipi bir dinamometre tarafından kesme kuvvetleri ölçüldüğü zaman, birinci dereceden model takip edildiği için Eş.4.14'de yer alan kesme kuvveti sensörünün transfer fonksiyonu aşağıdaki gibi olur.

$$F_d = \frac{K_d}{\tau \cdot S+1} \cdot K_c \qquad (4.14)$$

Burada τ dinamometrenin zaman sabitidir. *Kc* parametre değişimine karşı Kararlılığı sağlamak için anahtarlama yapan kontrol kazancıdır.



Şekil 4.12. Tornalamada değişken yapılı kontrol.

Yapay Zeka Tabanlı KAK

Yapay Zeka Tabanlı KAK'de kontrol stratejileri planlanır ve yeni fonksiyonlar öğretilir ne zaman çevrede gerçekleşen ortam şartlarına göre kendisini hazırlamış olur.

4.7. Takım Tezgâhlarında İzleme Yöntemleri

Tezgâhın ve çevre şartlarının izlenmesi ve bir bilgisayar tarafından verilerin işlenmesi için birçok sensör türü kullanılmaktadır. Sensör seçimi kullanılacağı yer, uygulama amacı, uygulama şartlarına göre yapılmalıdır. İzleme yöntemleri ile takım, tezgâh ve işleme durumu ile ilgili sağlıklı veri akışı sağlamak mümkündür.
Ölçme işleminden beklenen sonucun alınması için kullanılacak sensörlerin montaj yeri ve şekli, hassasiyeti, güvenilirliği, ölçme aralığı, örnekleme zamanı v.s..gibi unsurlara dikkat edilmelidir. Bu sensörlerden biri veya birkaçı kullanılarak ölçme yapılabilir. Ölçme yöntemleri aşağıdaki gibidir.

1. Kesme Kuvveti Ölçümü
2. Titreşim Ölçümü
3. Akustik Emisyonların Ölçümü (AE)
4. Elektrik Direncinin Ölçümü
5. Motor Akımının Ölçümü

6. Ultrasonik Ölçüm
7. Yüzey Pürüzlülüğü Ölçümü
8. Yakınlık (Proximity / Touch) Ölçümü
9. Sıcaklık Ölçümü
10. Optik Sensör Ölçümü
11. Ses Ölçümü

Hazırlanan CNC Tezganında akım ölçümü detaylarına yer verilmiştir. Motor akımı, kesme kuvvetleri ve takım aşınması ile doğru orantılıdır. İlerleme kuvvetini ve takım durumunu tahmin etmek için en ekonomik metotlardan biridir. Yüksek hızlı frezeleme işleminde akım izleyerek takım kırılmasını tespit edilmiştir [26]. Bir tornalama işlemi için YSA-BM tekniklerini kullanarak kolaylıkla akımdan kuvvetleri ölçebilmiştir [27].

Söz konusu projede SoMachine programında yer alan fonksiyon blokları kullanılarak farklı değişkenler altındaki akım değerleri izlenmiştir.

4.8. Adaptif Kontrollü Talaş Kaldırma İşlemi

Adaptif Kontrollü Talaş kaldırma işlemi için programcı kesme hızı ve ilerleme hızını belirlemek zorundadır. Bu kesme parametrelerinin tanımlanması tecrübenin yanında iş parçası, takım malzemesi, makine özellikleri, soğutma etkileri gibi faktörlere göre değişmektedir. Bu parametrelerinin seçimi direk olarak, ürünün boyutsal doğruluğunu, yüzey düzgünlüğünü, takım aşınma oranını, ekonomikliğini ve takımın kırılmasını etkiler. Bu üretim ve ürün kalitesine bağlı faktörler, işleme sırasında daha iyi hale getirilebilir. Bu işlem değişkenlerinin gerçek zamanlı olarak ölçülerek kontrol edilmesi adaptif kontrolün talaş kaldırma işlemlerine uygulanmasıdır. Çoğu üretim operasyonları dinamik proses değişkenleri tarafından karakterize edilir. Bunlar giriş ve çıkış değişkenleri olarak iki temel tiptedir. Değişkenlerin sınıflandırılması farklı operasyonlar için farklı olacaktır. Örneğin ilerleme bir takım tezgâhında kontrol edilebilen giriş değişkeni olurken diğer takım tezgâhları için sabitleşmiş değişken olabilir [23].

4.8.1. Giriş değişkenleri

Kontrol edilebilir giriş değerleri akış hızı, sıcaklık ayarları v.s dir. Kontrol edilemeyen giriş değişkenleri takım keskinliği, iş parçası malzeme sertliği ve iş parçası geometrisi kontrol edilemeyen değişkendir.

4.8.2. Çıkış değişkenleri

Ölçülebilir Çıkış Değişkenleri işleme esnasında anlık olarak karakteristikleri ölçülebilen değişkenlerdir. Örnek olarak talaş kaldırma işlemlerinde kesme kuvvetleri, titreşim, sıcaklık, akım vs. olabilir.

Performans Değerlendirme Değişkenleri tüm sistemin performansının değerlendirilmesinde kullanılır. Genellikle hem sistemin ekonomikliği hem de üretilen ürünün kalitesi ile ilişkilidir. Örneğin talaş kaldırarak işlemedeki performans değerlendirme değişkenleri "birim üretim basına maliyet" ve "üretim hızı"dır. Bazen ürün kalitesi, yüzey pürüzlülüğü ve parça boyutu vs. de performans değerlendirmede kullanılabilir.

4.8.3. Adaptif kontrolün talaş kaldırma işleminde gerekliliğinin nedenleri

Kesme geometrisindeki değişim

Kesme derinliği veya genişliğinin değişiklik gösterdiği durumlardır. İlerleme genellikle kesit değişimini telafi etmek için ayarlanır.

Değişken iş parçası sertliği ve değişken işlenebilirlik

Takım iş parçasındaki homojen olmayan alanlarla karşılaşıldığı zaman aşınmanın azaltılması için, önceden ilerlemenin azaltılması gereklidir.

Değişken İş Parçası Rijitliği

İşlemenin kararlı olması için Yetersiz rijitlik sonucu iş parçası sehim yaparsa ilerlemenin azaltılması gereklidir.

Takım Aşınması

Kesme kuvvetlerinde artış ile takım aşınmaları orantılı olduğu bilinmektedir. Kesme kuvvetlerini daha fazla artırmamak için adaptif denetleyici ilerlemenin azaltılmasını sağlamalıdır.

Kesme Esnasındaki Boşluklar

İşlenecek parçada boşlukların olduğu bazı bölgelerde işlemeye ihtiyaç duyulmayabilir. Eğer takım hava boşluklu yerlerde aynı ilerlemeyle devam ediyorsa zaman kaybı olur. Bundan dolayı hava boşlukları ile karşılaşıldığında ilerlemeyi artırmak için özel için döngü gereklidir.

4.8.4. Talaş kaldırmada adaptif kontrolün faydaları

Üretim hızlarını artırır, Takım ömründe artış sağlar, daha az operatör müdahalesi gerektirir, Programlama daha kolaydır.

4.9. Talaş Kaldırma İşlemlerinde Adaptif Kontrol Uygulaması

Maksimum üretim kapasitesi tezgâhların en yüksek hız değerlerinde elde edilir. Talaş kaldırmanın maksimum olması için kesme derinliği ve genişliği gibi faktörlere bağlı olarak kesme kuvvetinin mevcut kesme şartlarında verilen limitler içinde kalması zorunludur. Uygulamalar esnasında sabit bir kesme kuvveti sürdürülmeye çalışılır. İş parçası sertliği, kesme genişliği ve derinliği artışından dolayı kuvvet arttığı zaman ilerleme hızı düşürülerek telafi edilecektir. Birim zamanda kaldırılan talaş hacminin maksimum olabilmesi için parçadaki hava boşluklarında kuvvet azaldığı zaman ilerleme hızı arttırılmalıdır.

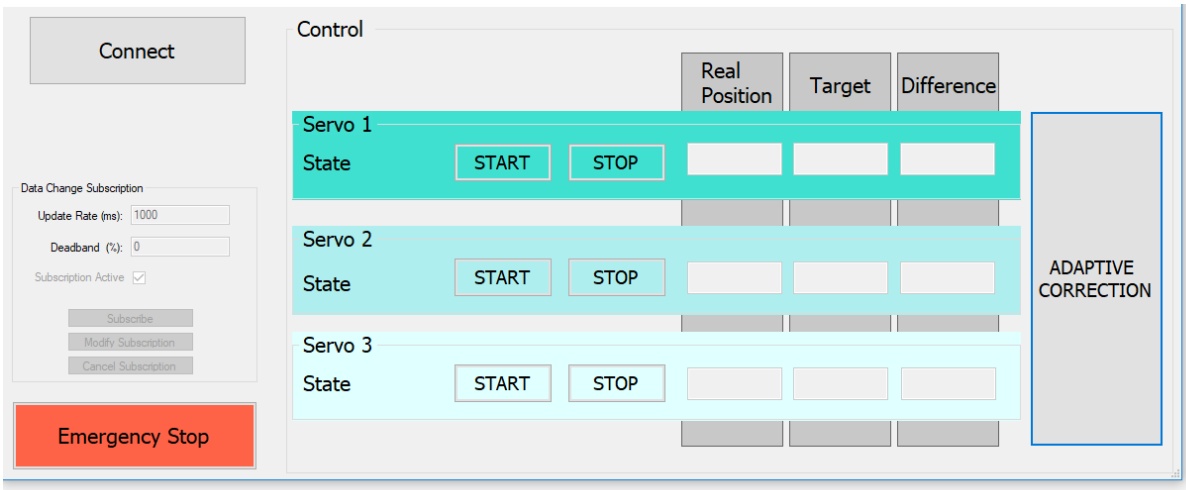
Eğer kesici takım, hava boşluğu veya iş konturu boyunca hareket ediyorsa akım değerleri izlenerek bir hava boşluğu olduğu tespit edilir. Eğer akım bu eşik değer altında ise denetleyici kesici takımın bir hava boşluğu boyunca geçtiğini varsayar. Eğer hava boşluğu artarsa ilerleme 1,5 veya 2'ye katlanıp kesmesiz ilerleme değeri maksimum edilip zamandan kazanç sağlanır [22].

Yapılan incelemeler ve aşağıda sunulan grafikler doğrultusunda kesme hızı artışının yüksek güç gerektirdiği ve motor besleme gerilimin sabit olması sebebiyle akımın arttığı sonucu çıkarılmıştır. İlerleme artışıyla birim zamanda kaldırılan talaş hacmi arttığı için artan güç gereksinimi motorun çektiği akımın artmasına sebep olmuştur.

Bu çalışmada adaptif kontrol iki aşamalı olarak gerçekleştirilmiştir. Bunların birincisi harici konum algılayıcı yardımı ile mil boşluğu anlık ölçümlerine dayalıdır. İkincisi ise motorun kp değerinin anlık ölçülmesi esasına dayalıdır.

4.10 Harici Konum Algılayıcı İle Yapılan Adaptif Kontrol

Bu çalışmada harici konum algılayıcı ile dahili konum algılayıcıdan gelen konum bilgileri kıyaslanarak, istenilen konuma ulaşıp ulaşmadığı kontrol edilir. Her iki konum algılayıcı arasında mil boşluğundan kaynaklanan bir fark varsa motorların hesaplanan fark telafi edilene kadar hareket etmesi sağlanır. Haberleşme ayarları yapıldıktan sonra Connect Butonuna basılarak haberleşme başlatılır. Resim 4.1 ile gösterilen ekranda Real Position ile gösterilen alanlarda X,Y,Z eksenlerinin gerçek konum bilgisi harici konum algılayıcıdan alınarak okunur. Target ile gösterilen alanlarda X,Y,Z eksenlerinin gitmesi hedeflenen konum bilgisi yer alır. Difference ile gösterilen alanda ise X,Y,Z eksenlerinin harici ve dahili konum algılayıcıdan gelen konum bilgileri kıyaslanır ve mil boşluğundan kaynaklanan fark mm cinsinden yer alır. Adaptive Correction butonuna basıldığında oluşan bu fark motordun yeniden konumlanması için offset değeri olarak atanır ve sistem adaptif hale getirilir.



Resim 4.1. VisualStudio adaptif kontrol ekranı-1

Resim 4.2'de gösterilen ekranda X,Y,Z eksenlerine hareket veren Servo1,Servo2 ve Servo3 motorlarına güç verilmesi sağlanmıştır. Emergency Stop düğmesine basılması halinde motorlar hareket halindeyken üçü birden durdurulabilecektir.



Resim 4.2. VisualStudio adaptif kontrol ekranı-2

Sürücülere verilen gücün durdurulabilmesi için State kısmında yer alan butona basılır ve Resim 4.3.'deki gibi Power OFF olarak görülür. Daha sonra haberleşmeyi kesmek için Disconnect butonuna basılır.



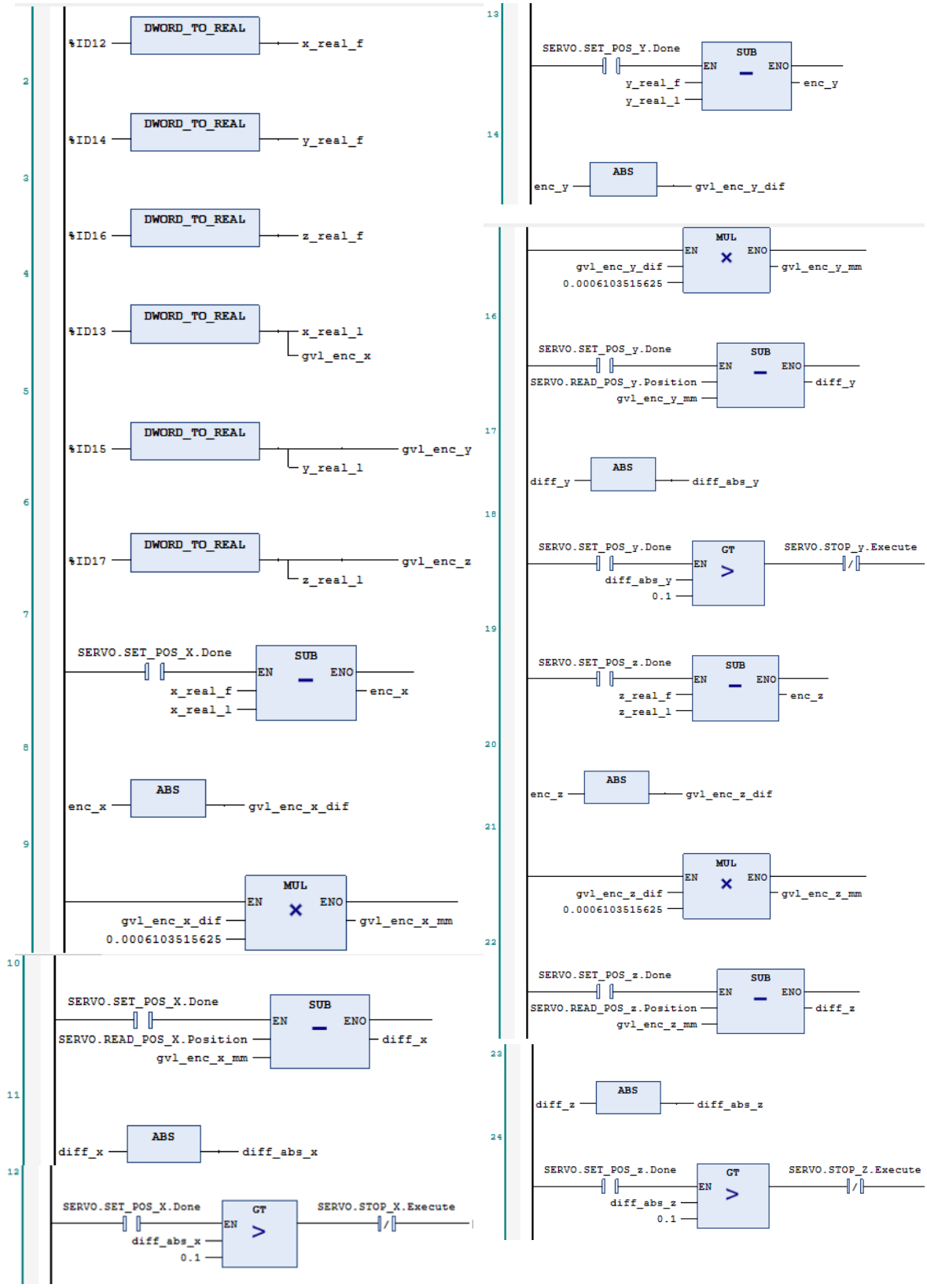
Resim 4.3. VisualStudio adaptif kontrol ekranı-3

Harici konum algılayıcı ile Adaptif Programlama yapılabilmesi için öncelikle Konum algılayıcı programlama dosyası olan ENC(Prg) program menüsü açılır. Burada iki konum algılayıcı arasındaki fark, SUB komutu ile birbirinden çıkartılır, çıkan değer ABS komutu

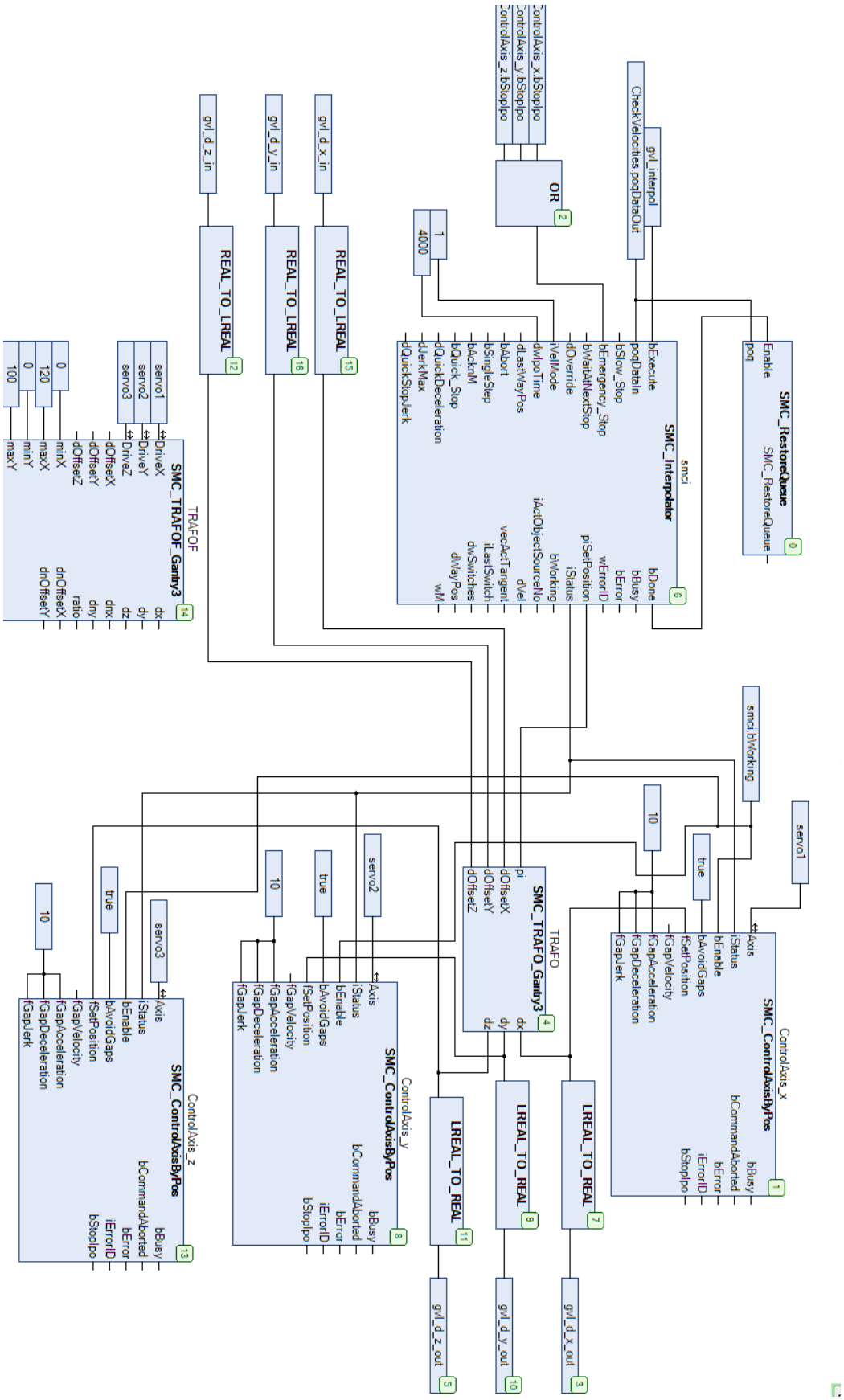
ile mutlak hale getirilir, Çıkan mutlak sonuç 1 konum algılayıcı sinyalinin mm karşılığı olan 0,00613515625 değeri ile çarpılır, milimetre cinsinden elde edilen sonuç0,1 mm den büyük olup olmadığını tespit etmek için GT (Greather Than)komutu kullanılır. Çıkan sonucun milimetreden büyük olması halinde motora dur komutu verilir veya offset ayarı yapılarak eksik yapılan ilerleme miktarının telafisi sağlanır. Bunun yapılabilmesi interpolarasyon yapılan Prg(4) dosyası program menüsünden açılır. SMC_TRAFO_Gantry3

Program bloğunun dOffsetx, dOffsety, dOffsetz girişlerine REAL'dan LREAL'a çevrilmiş olan gvl_d_x_in, gvl_d_y_in, gvl_d_z_in girişleri bağlanır.

Konum algılayıcılar arasındaki farkın 0,1 mm2yi geçmesi halinde güvenlik amacıyla motorun durdurulabilmesi için Prg(1) dosyası program menüsünde yer alan ve daha önce iç değişken olan STOP_X adındaki MC_STOP FB'nun execute girişi kapalı kontak olarak GT(Grather Tahan) FB'nun çıkışına bağlanır. Yapılan ilave programlama sonucunda elde edilen programın son hali Resim 4.4 ve Resim 4.5'de sunulmuştur.



Resim 4.4. Harici konum algılayıcı ile adaptif programlama-1



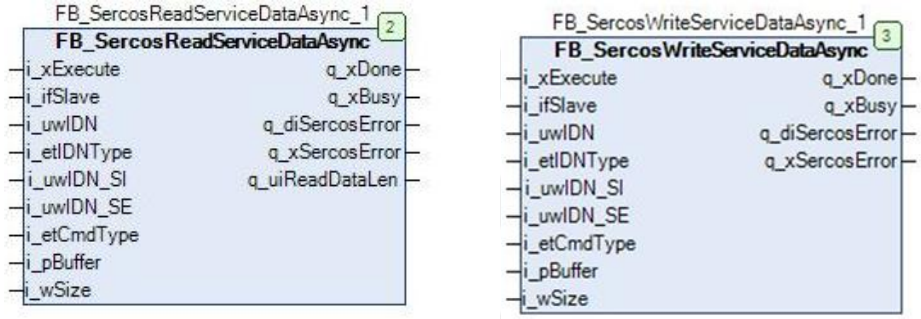
Resim 4.5. Harici konum algılayıcı ile adaptif programlama-2

4.11 Hız Regülatörü P Faktörü İle Yapılan Adaptif Kontrol

4.11.1. SoMachine içerisinde kullanılan fonksiyon blokları

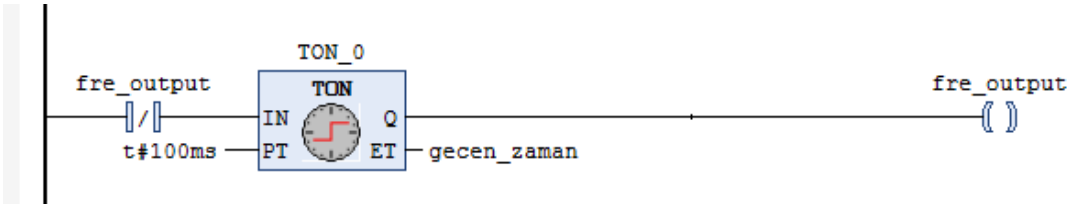
Servo motorların çektiği akımı okuyabilmek ve p faktörünü okumak için SoMachine ara yüzünde yer alan FB_SercosReadServiceDataAsync fonksiyon bloğu, p faktörünü değiştirebilmek için FB_SercosWriteServiceDataAsync kullanılmıştır.

FB_SercosReadServiceDataAsync ve FB_SercosWriteServiceDataAsync komutları

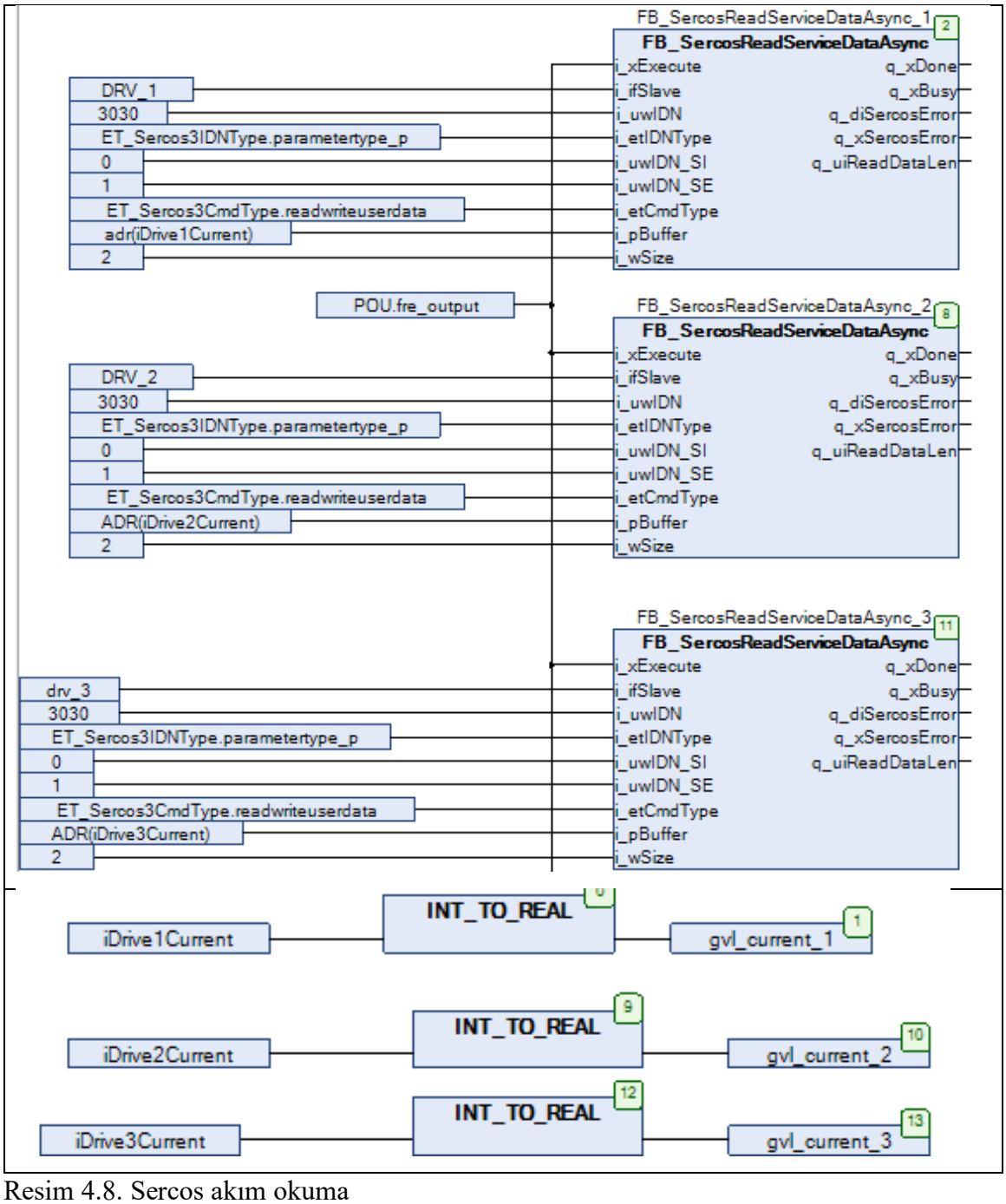


Resim 4.6. FB_SercosReadServiceDataAsync ve FB_SercosWriteServiceDataAsync

Söz konusu FB'larında sürekli okumanın veya yazma işleminin aralıklı sağlanabilmesi için i_xExecute girişine sürekli olarak her 100 ms'de 1 puls gönderilmiştir. Puls girişinin sağlanabilmesi için bir frekans üreticine ihtiyaç duyulmuştur, bunun için SoMachine içerisinde yer alan TON FB kullanılmıştır. Hazırlanan frekans üreticinin görseli Resim 4.7'de sunulmuştur. I_ifslave girişine hangi servo motordan bilgi okuyacaksak onun bağlı bulunduğu sürücüye ait değişken adı atanır. Okunması talep edilen bilgiye ait parametreler, parametre listesinden bulunur. Örneğin akım değerini okumak için IDN P-0-3030.0.1 değerleri kullanılmalıdır. İ_uwIDN girişine dört haneli 3030 değeri girilmiş, i_uwIDN_SI girişinde yanında bulunan 0 değeri ve i_uwIDN_SE girişine 1 değeri verilmiştir. Fonksiyon bloğunda yer alan i_etIDNType girişine ET_Sercos3IDNType.parametertype_p, i_etCmdType girişine ET_Sercos3CmdType.readwriteuserdata yazılır. İ_pBuffer girişine adr(iDrive1current) değişkeni atanmış ve bilgiler iDrive1current değişkeni içerisinden okunabilmiştir. Okunan değerler move komutu ile gvl kütüphanesinde yer alan gvl_current_1 dış değişkenine INT_TO_REAL FB kullanılarak öncelikle integer değerden real değere çevrilmiş ve elde edilen real değer atanarak kepserver üzerinden okunabilir hale getirilmiştir.

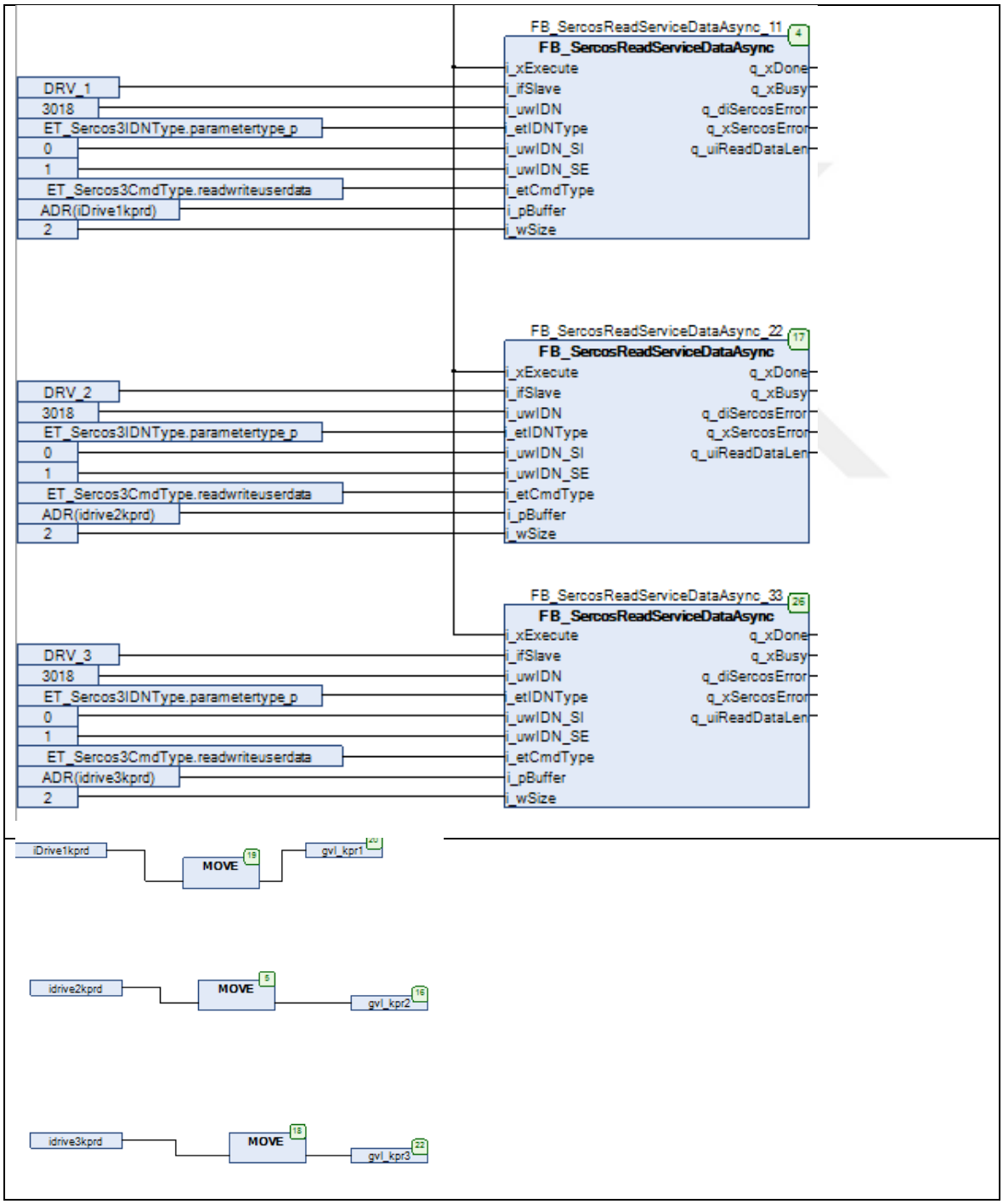


Resim 4.7 Frekans üreteci



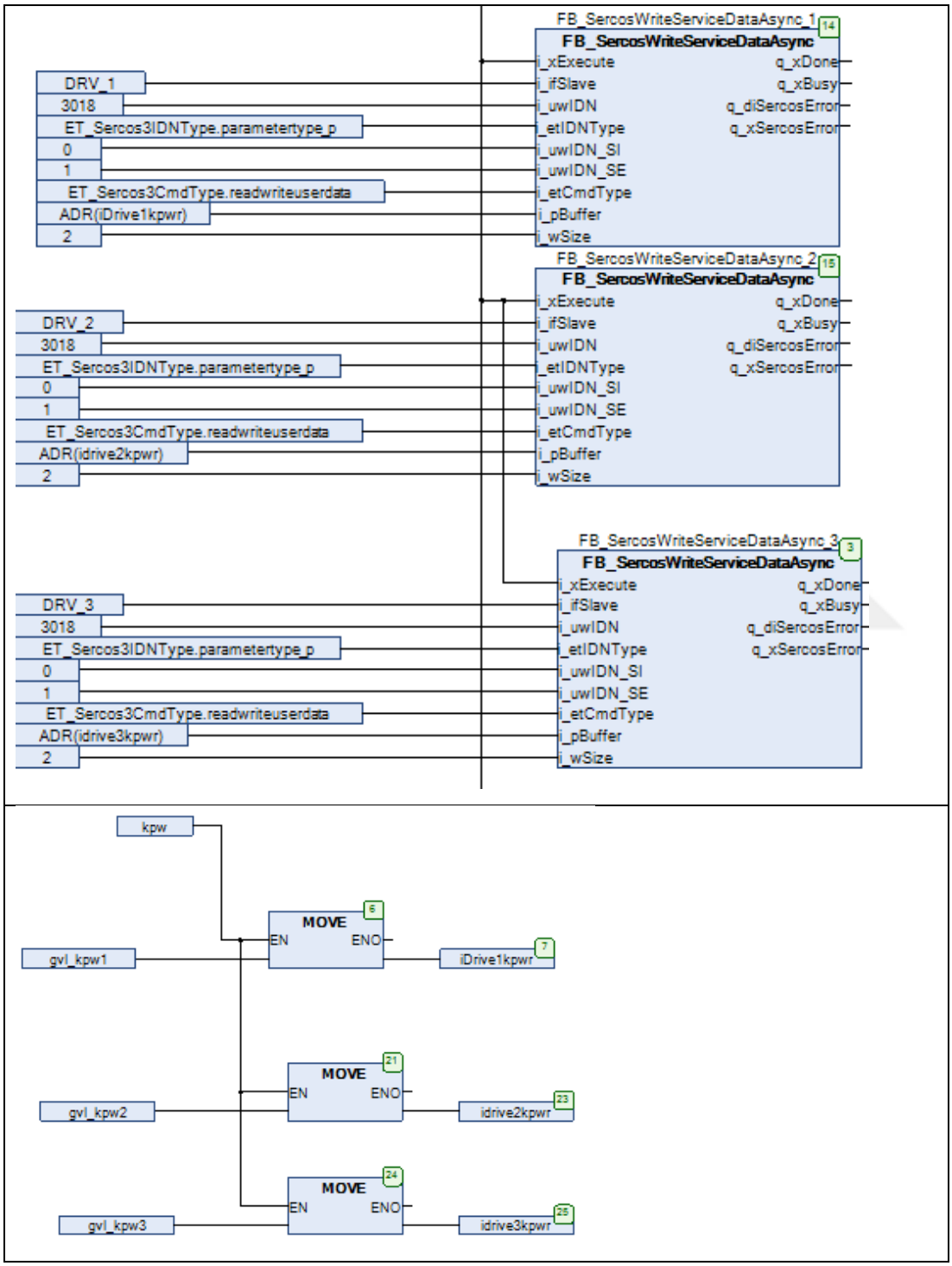
Resim 4.8. Sercos akım okuma

Regülatör parametre seti arasında değişimde değerlerin uyarlanması lineer biçimde CTRL_ParChgTime parametresinde ayarlanan süre üzerinden gerçekleşir. Değiştirilen ayarlar derhal kabul edilir. P faktörünün okunması SoMachine programı üzerinden IDN P-0-3018.0.1 P parametreleri FB_SercosReadServiceDataAsync fonksiyon bloğuna girilerek yapılmıştır. P faktörünü okumak için yapılan programlama Resim 4.9'da sunulmuştur.



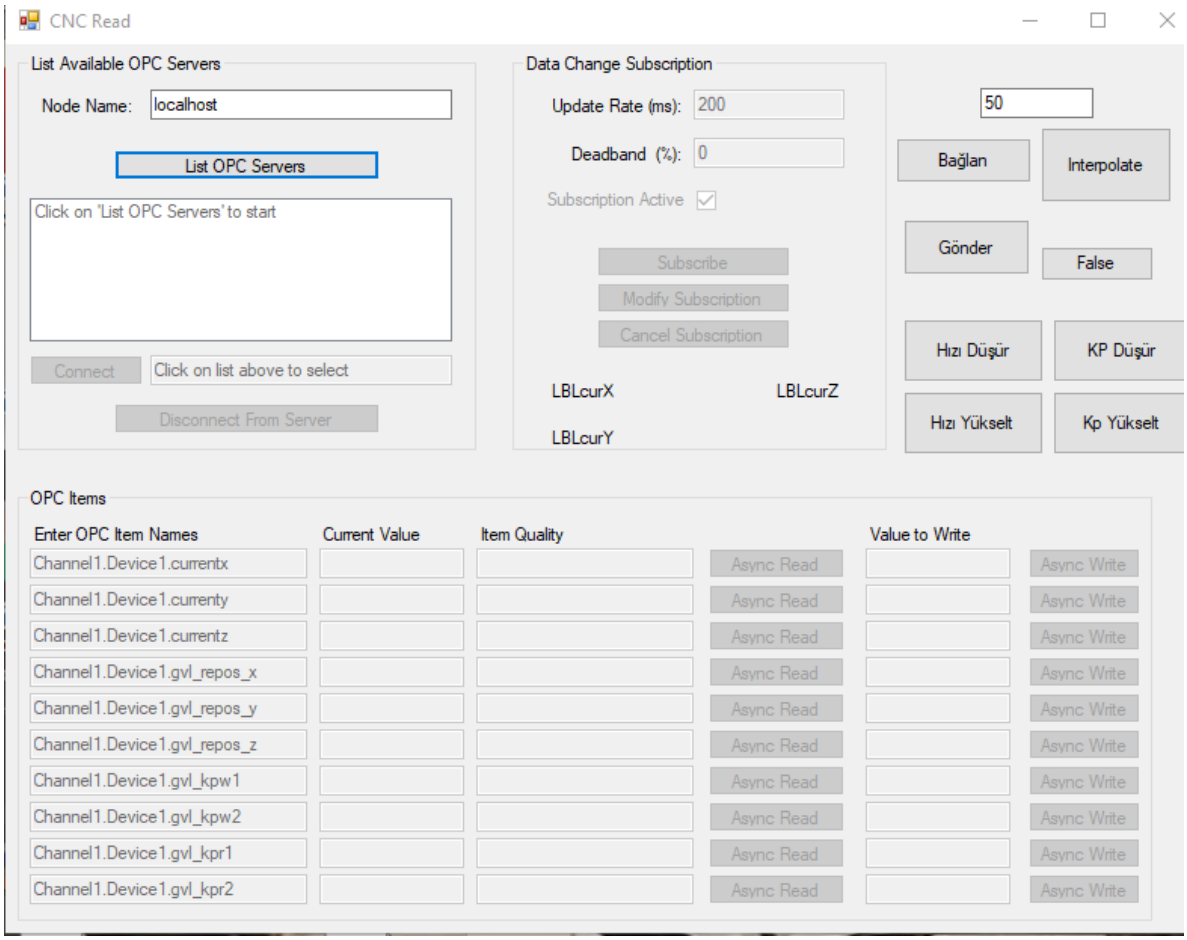
Resim 4.9. Sercos P faktörü okuma

P faktörünün yazılması SoMachine programı üzerinden IDN P-0-3018.0.1 P parametreleri FB_SercosWriteServiceDataAsync fonksiyon bloğuna girilerek yapılmıştır. P faktörünü yazmak için yapılan programlama Resim 4.10'da sunulmuştur.



Resim 4.10. Sercos P faktörü yazma

So machine programında okunan P faktörü Kepserver programı aracılığıyla okunarak C# programında hazırlanan program ve ara yüz vasıtası ile adaptif hale getirilmesi sağlanır. Hazırlanan program ara yüzü Resim 4.11'de sunulmuştur.

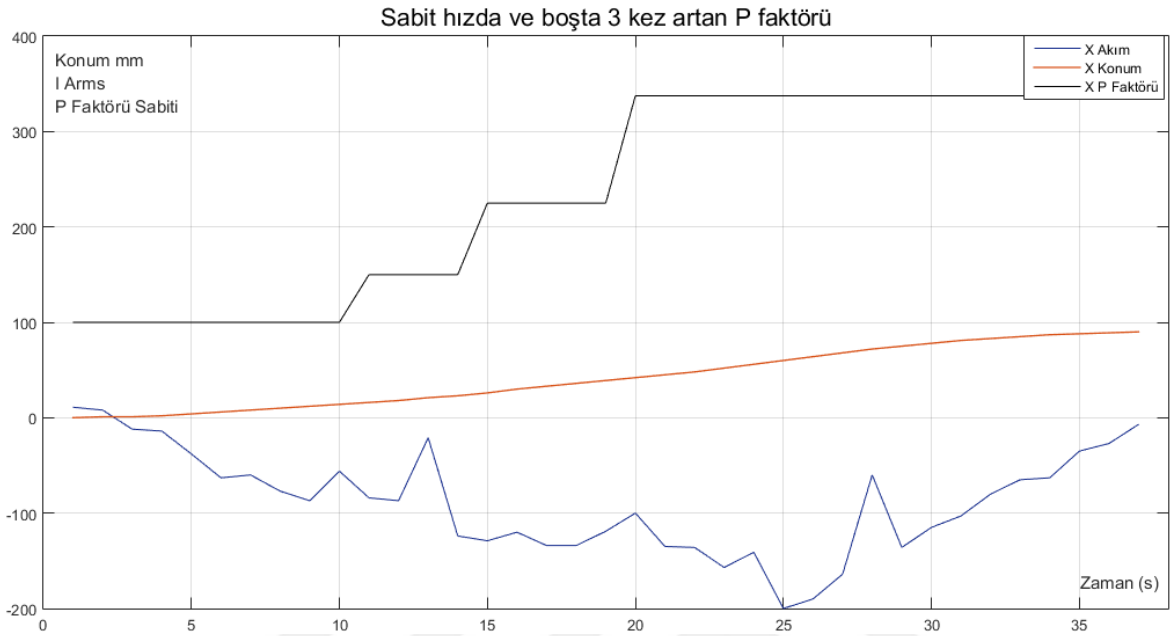


Resim 4.11. P Faktörü program ara yüzü

4.11.2 Sabit hızda P faktörünün akım değerine etkisi

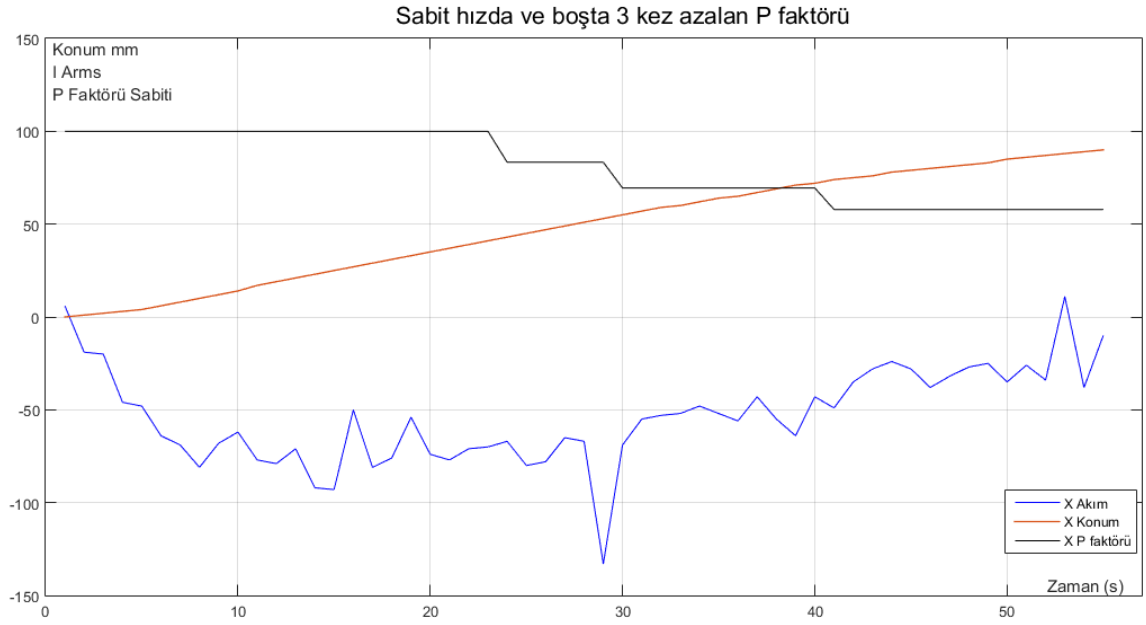
So machine programında yer alan FB_SercosReadServiceDataAsync fonksiyon bloğuna IDN P-0-3018.0.1 p parametreleri girilerek p faktörü değerleri, IDN P-0-30108.0.1 p parametreleri girilerek akım değerleri okunur, okunan akım ve p faktörü değerine ait değişkenler GVL kütüphanesine real değer ile atanır. Kepserver programı ile bu değişken değerleri bilgisayar üzerinden okunması ve değiştirilmesi FB_SercosWriteReadService DataAsync fonksiyon bloğu ile mümkündür. C# programında yazılan program vasıtası ile akım, konum, p faktörü, süre değerleri kayıt altında tutulmuştur. Yapılan bu işlem sonucunda akım –hız-konum- p faktörü arasındaki ilişkiyi ortaya koymak mümkün olmuştur.

Sabit hızda ve boşta 3 Kez Artan, P faktörü ve akıma etkisini gösteren grafik Şekil 4.13'de sunulmuştur.



Şekil 4.13. Sabit hızda ve boşta 3 kez artan P faktörü

Sabit hızda ve boşta 3 Kez Azalan, P faktörü ve akıma etkisini gösteren grafik Şekil 4.14'de sunulmuştur.



Şekil 4.14. Sabit hızda ve boşta 3 kez azalan P faktörü

Sabit hızda ve yükte 3 Kez Artan, P faktörü ve akıma etkisini gösteren grafik Şekil 4.15'de yükte yapılan işlemenin görseli ise Resim 4.10.a'da sunulmuştur.



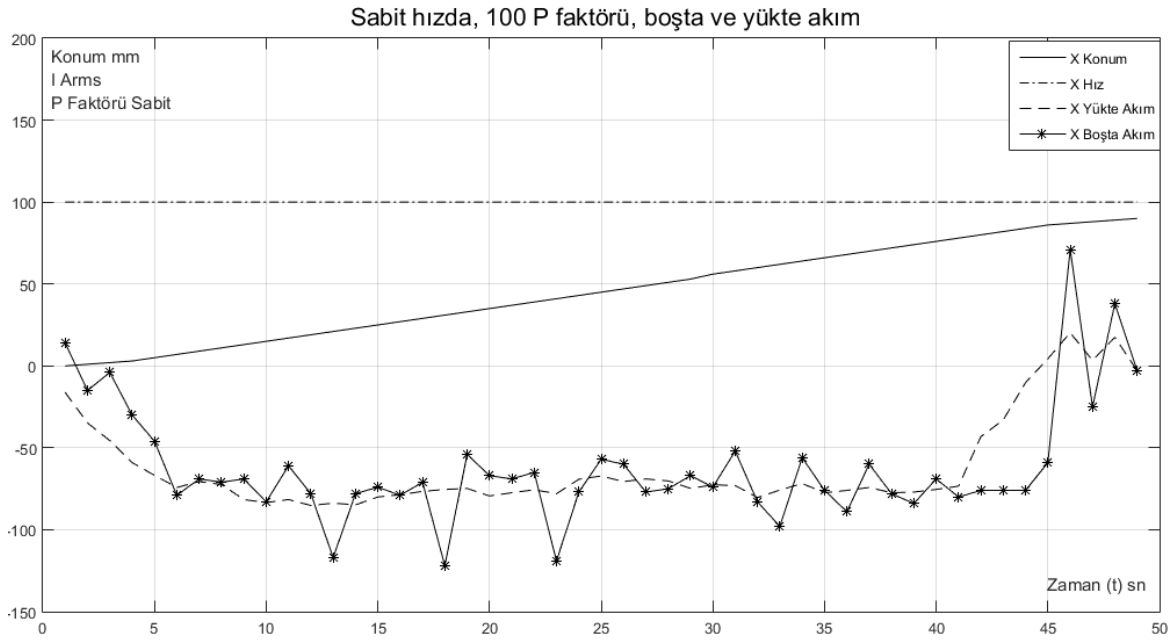
Şekil 4.15. Sabit hızda ve yükte 3 kez azalan P faktörü

Sabit hızda ve yükte 3 Kez Azalan, P faktörü ve akıma etkisini gösteren grafik Şekil 4.16'da yükte yapılan işlemenin görseli ise Resim 4.10.b'de sunulmuştur.



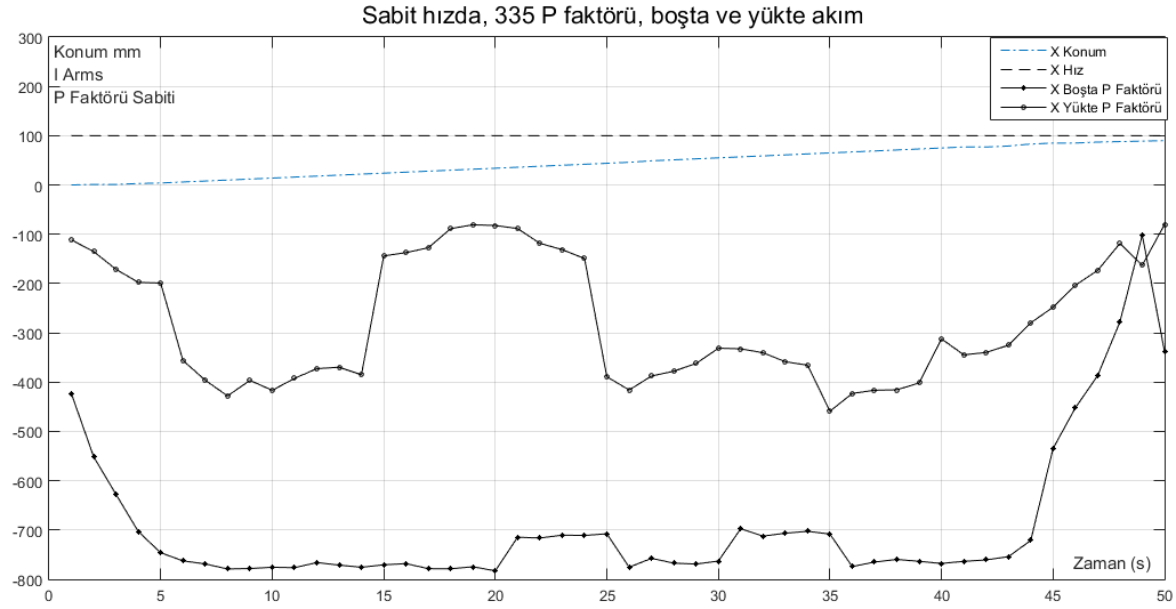
Şekil 4.16. Sabit hız, yükte 3 kez azalan P faktörü

Sabit hızda ve Sabit 100 P Faktörü, yükte ve boşta akıma etkisini gösteren grafik Şekil 4.17'de yükte yapılan işlemenin görseli ise Resim 4.10.c'de sunulmuştur.

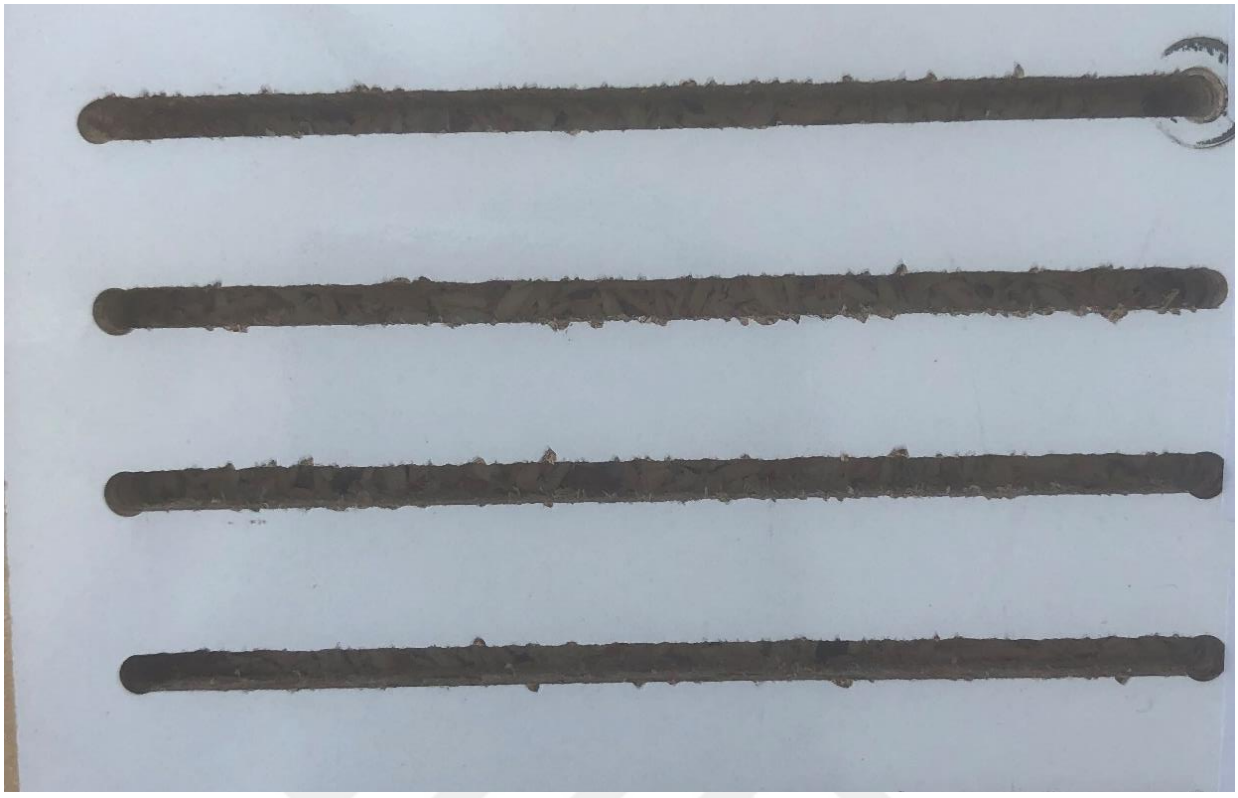


Şekil 4.17. Sabit hızda, 100 P faktörü, boşta ve yükte akım

Sabit hızda ve Sabit 335 P Faktörü, yükte ve boşta akıma etkisini gösteren grafik Şekil 4.18'de, yükte yapılan işlemenin görseli ise Resim 4.12.d'de sunulmuştur.

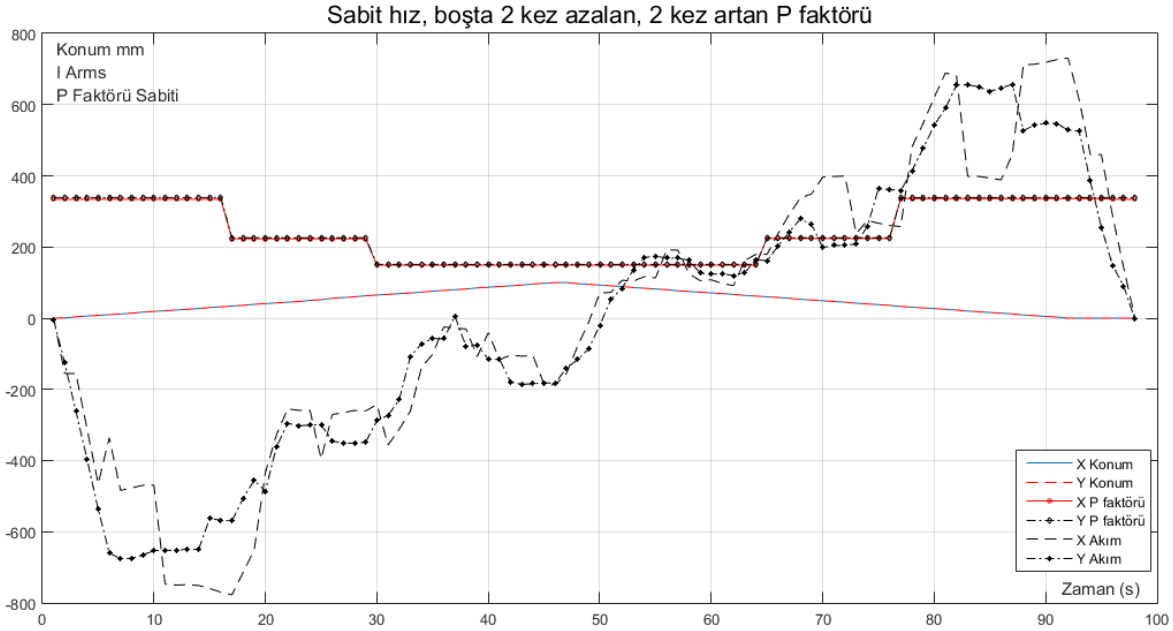


Şekil 4.18. Sabit hızda, 335 P faktörü, boşta ve yükte akım



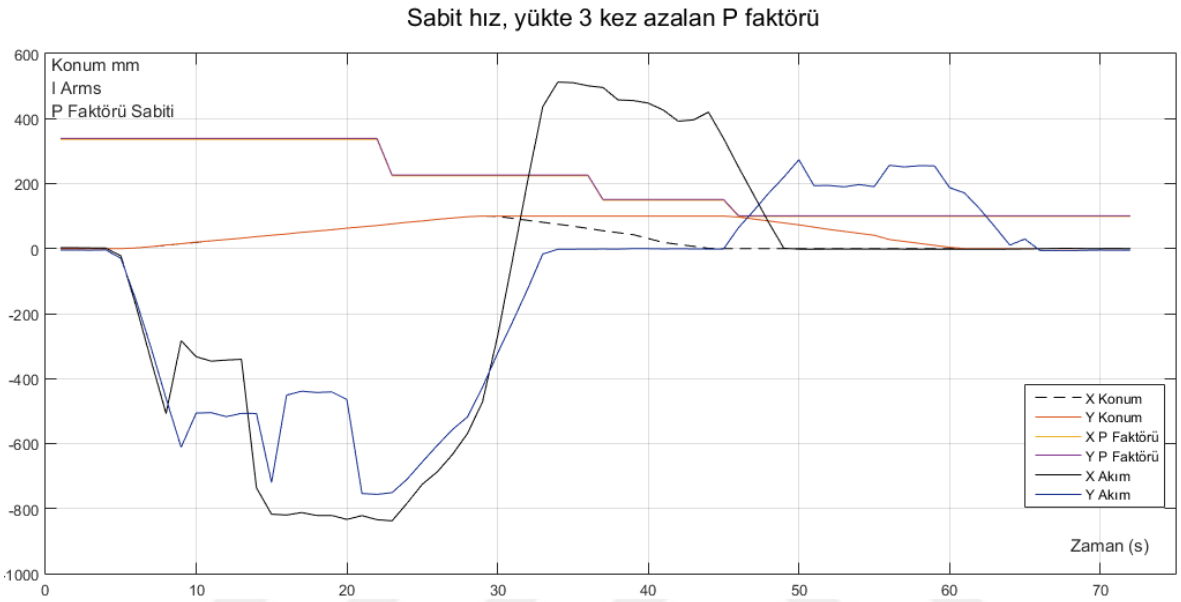
Resim 4.12. X ekseninde değişken P faktörü ile doğrusal işleme

Sabit hızda ve boşta 2 Kez Azalan, 2 Kez Artan P faktörü ve akıma etkisini gösteren grafik Şekil 4.19'da sunulmuştur.

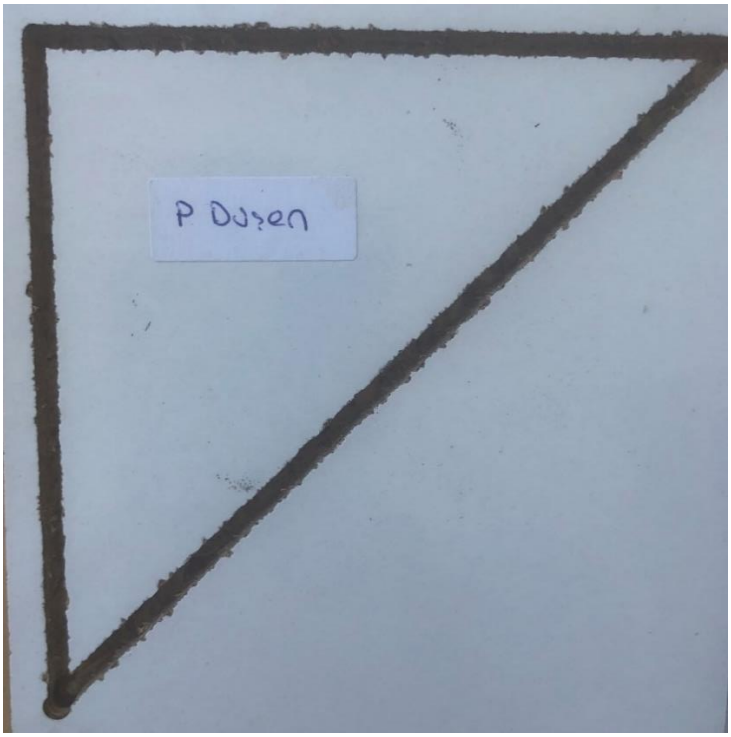


Şekil 4.19. Sabit hız, boşta 2 kez azalan, 2 kez artan P faktörü

Sabit hızda ve Yükte 3 Kez Azalan P faktörü ve akıma etkisini gösteren grafik Şekil 4.20'de, yükte yapılan işlemenin görseli ise Resim 4.13'de sunulmuştur. Burada yapılan işleme esnasında işlemeyi yapan motor yeterli güçte olmadığından ve sistemde yer alan titreşimden dolayı konum başlangıç noktalarında kaymalar meydana gelmiştir.

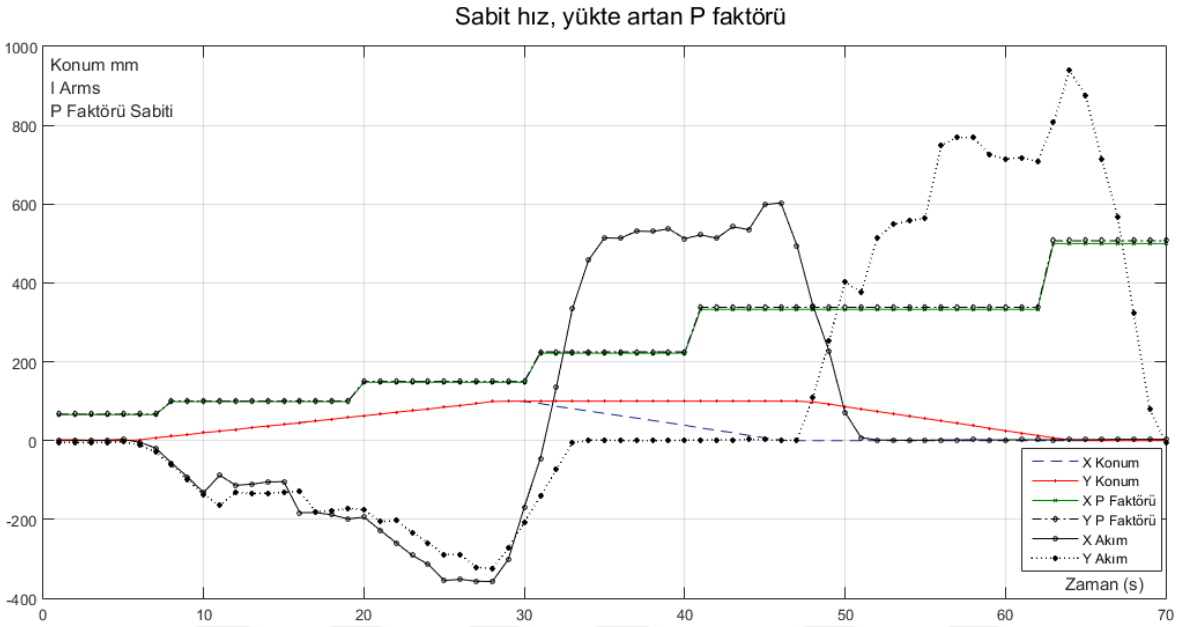


Şekil 4.20. Sabit hız, yükte 3 kez azalan P faktörü

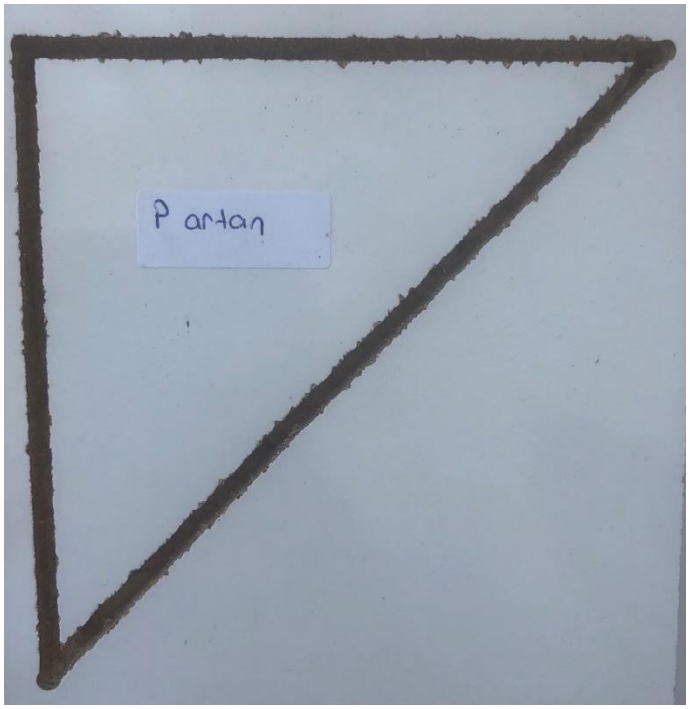


Resim 4.13. X-Y ekseninde azalan P faktörü ile doğrusal işleme

Sabit hızda ve yükte 3 Kez Artan P faktörü ve akıma etkisini gösteren grafik Şekil 4.21'de, yükte yapılan işlemenin görseli ise Resim 4.14'de sunulmuştur. Burada yapılan işleme esnasında işlemeyi yapan motor yeterli güçte olmadığından ve sistemde yer alan titreşimden dolayı konum başlangıç noktalarında kaymalar meydana gelmiştir.



Şekil 4.21. Sabit hız, yükte artan P faktörü

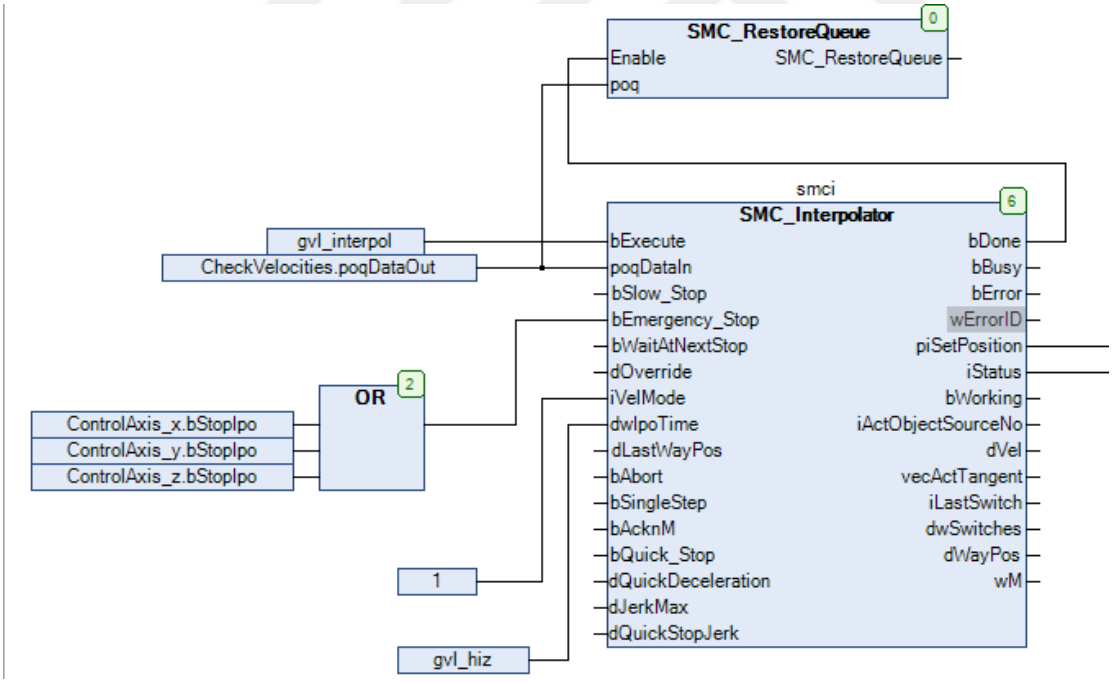


Resim 4.14. X-Y ekseninde artan P faktörü ile doğrusal işleme

Yukarıdaki tablolarda görülmektedir ki sabit hızda, boşta ve yükte olmak üzere P faktörü değişimi ile yapılan ölçümler neticesinde akım değerlerinde azımsanmayacak miktarda değişimler gözlemlenmiştir. Buna göre örneğin frezeleme işlemi esnasında boşlukta geçen süre boyunca fazla akım çekmesinin gerilimin sabit olduğu düşünüldüğünde fazla güç sarfiyatına neden olacağı tespit edilmiştir.

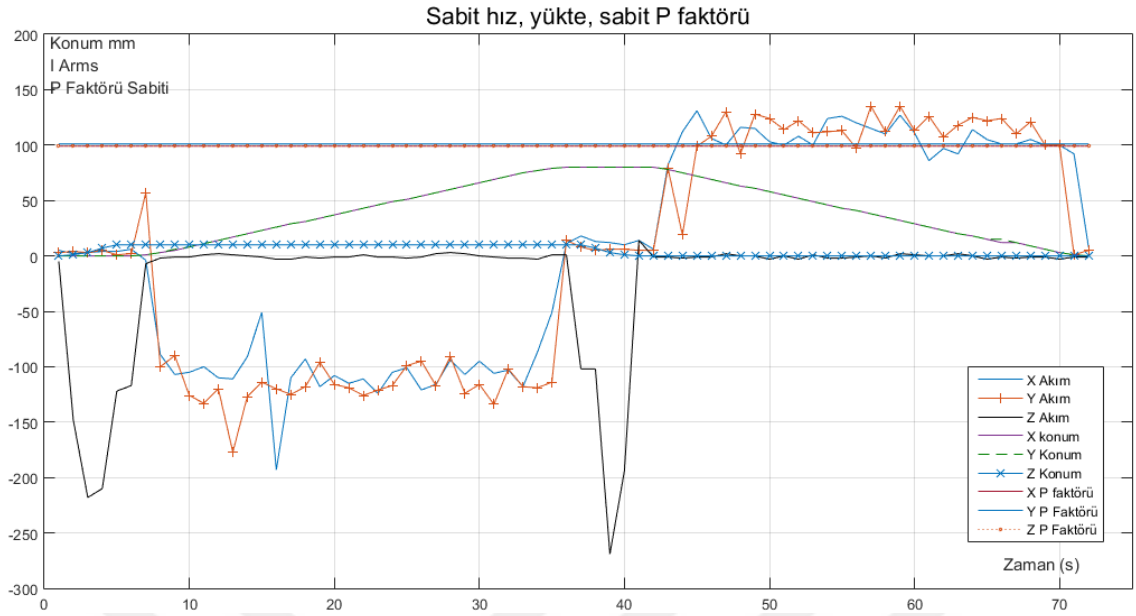
4.11.3 Değişken hızlarda P faktörünün akım değerine etkisi

Değişken hızlarda P faktörünün etkisini tespit edebilmek için mevcut programlamada interpolasyon yapmak amacıyla kullanılan SMC_Interpolator fonksiyon bloğunun dwIpo girişine gvl kütüphanesinden bir değişken atanmıştır. Değişkenin gvl kütüphanesinde olması sayesinde hazırlanan arayüz sayesinde, C# ve kepserver aracılığı ile hız değişimi yapılabilmiştir. Programda yapılan değişikliğe ait ekran görüntüsü Resim 4.15'de sunulmuştur.



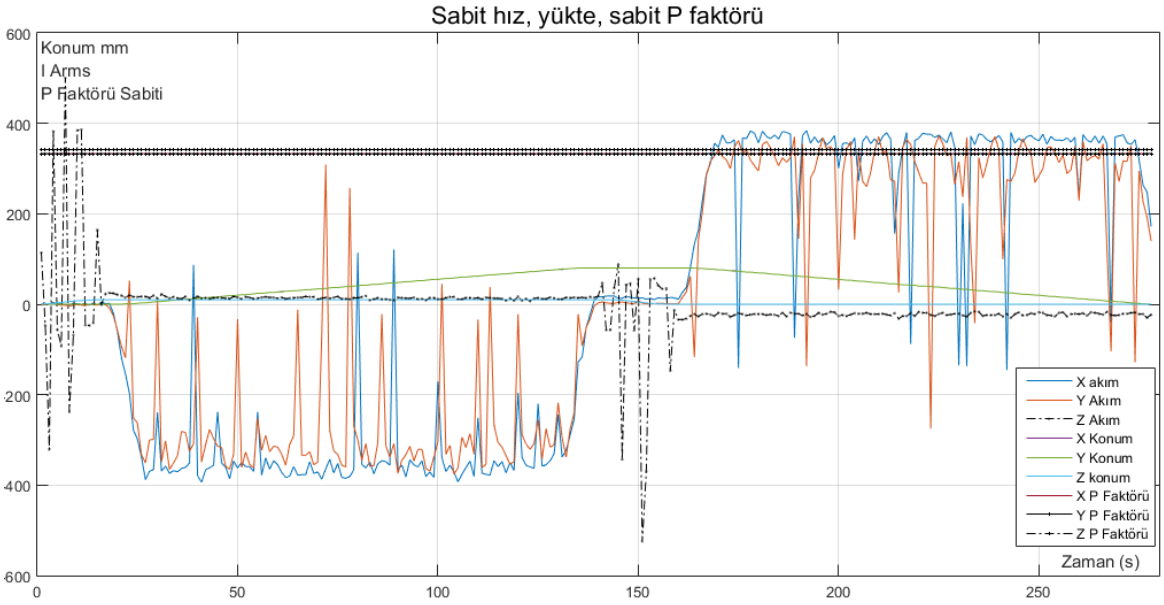
Resim 4.15. Hız değişimi programlanması

Doğrusal 2 eksende, sabit 100 hızda, sabit p faktörü 100'de ve yükte ve akıma etkisini gösteren grafik Şekil 4.22'de, yükte yapılan işlemenin görseli ise Resim 4.16.a'da sunulmuştur.



Şekil 4.22. Sabit hız, yükte 3 kez artan P faktörü

Doğrusal 2 eksende, sabit 100 hızda, sabit p faktörü 335'de ve yükte ve akıma etkisini gösteren grafik Şekil 4.23'de, yükte yapılan işlemenin görseli ise Resim 4.16.b'de sunulmuştur.

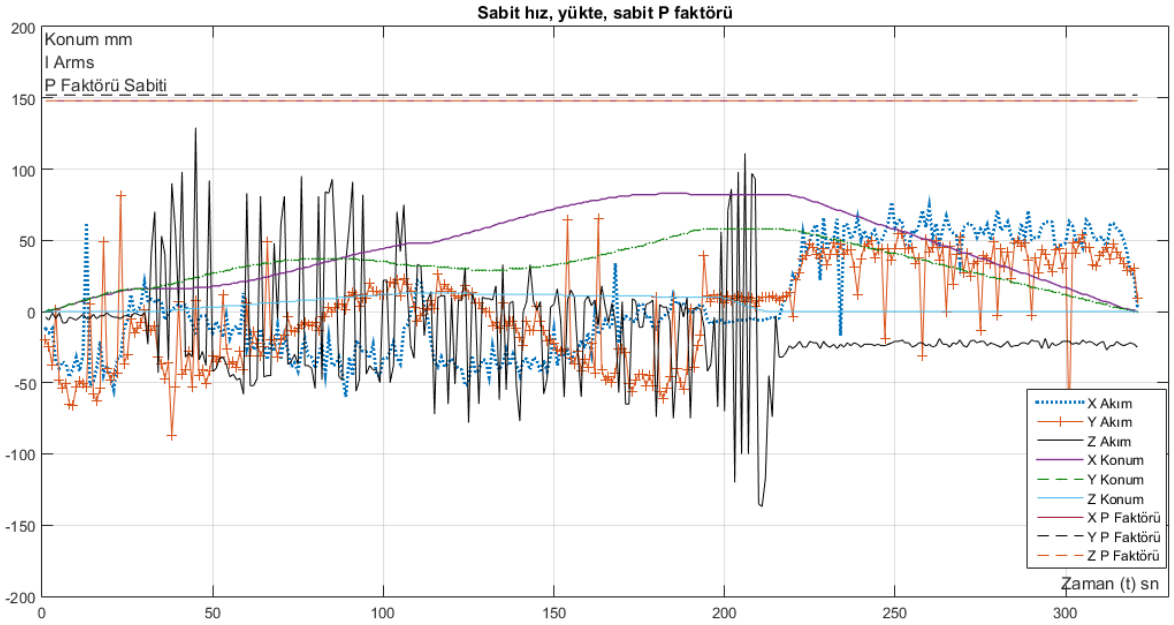


Şekil 4.23. Sabit hız, yükte, sabit P faktörü



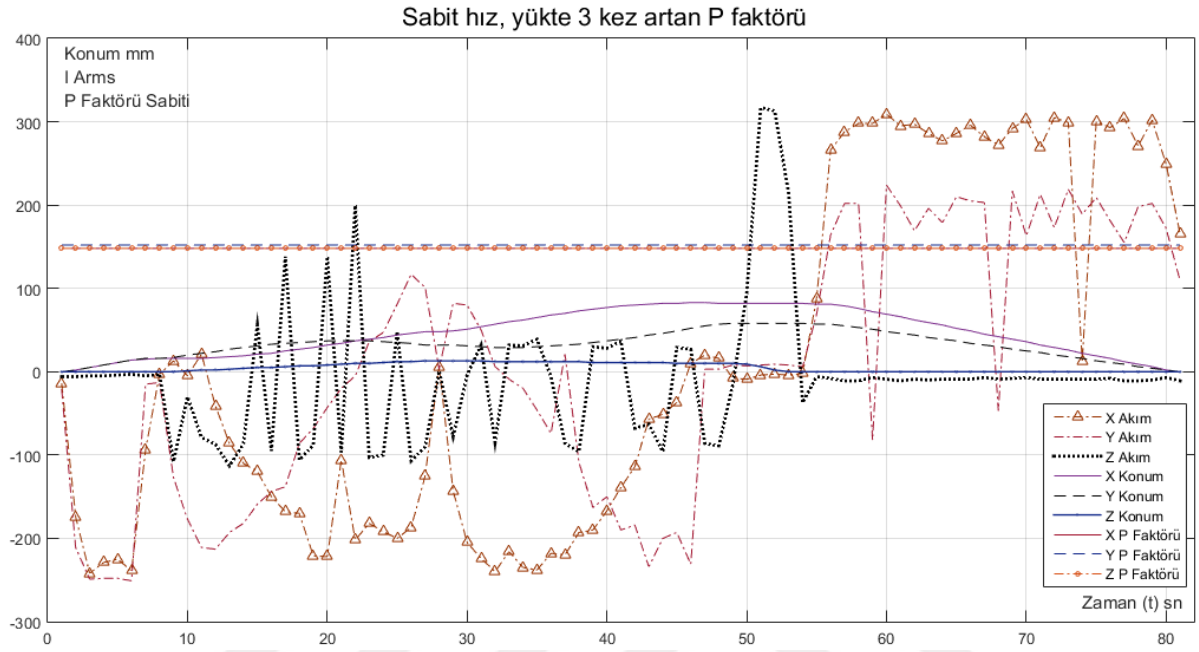
Resim 4.16. Düşük Hız Doğrusal İşleme

Dairesel 3 eksende, sabit 25 hızda, sabit p faktörü 335'de ve yükte ve akıma etkisini gösteren grafik Şekil 4.24'de, yükte yapılan işlemenin görseli ise Resim 4.17'de sunulmuştur.



Şekil 4.24. Sabit hız, yükte P faktörü

Dairesel 3 eksende, sabit 100 hızda, sabit P faktörü 335'de ve yükte ve akıma etkisini gösteren grafik Şekil 4.25'de, yükte yapılan işlemenin görseli ise Resim 4.18'de sunulmuştur.



Şekil 4.25. Sabit hız, yükte P faktörü

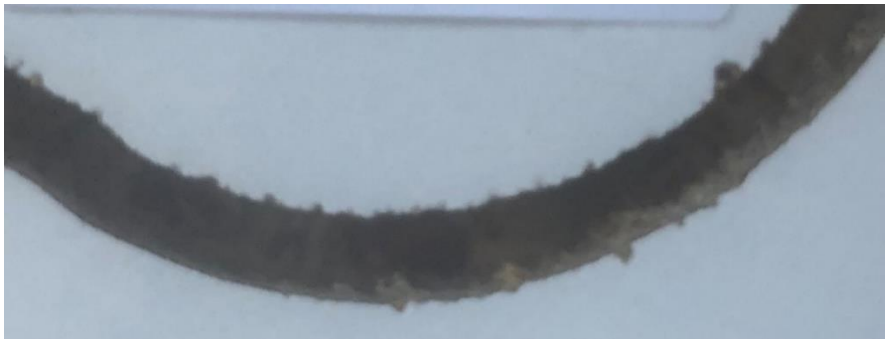


Resim 4.17. Yüksek hız dairesel işleme-1



Resim 4.18. Düşük hız dairesel işleme-1

Ayrıca Resim 4.17 ile Resim 4.18'de yer alan işlemelerin Resim 4.19 ve Resim 4.20'de detaylı büyütülmüş görseli bulunmaktadır. Buna göre yavaş hızda yapılan işlemelerin, birim alan başına düşen dönme sayısı artmış olduğundan, her ne kadar yüzey pürüzsüzlüğü ölçebilen cihazın bulunmaması nedeniyle ölçülememiş ancak daha düzgün bir görüntüye sahip olduğu gözlemlenmiştir.



Resim 4.19. Yüksek hız dairesel işleme-1



Resim 4.20. Düşük hız dairesel işleme-2

Yukarıda sunulan tablolardan elde edilen sonuca göre sabit p faktörü varken; yük altında akım değerinin daha çok salınımda bulunduğu, yüksek hızda ise daha çok yüksek değerler aldığı tespit edilmiştir. Elde edilen sonuca göre pürüzsüzlüğün önemli bir fark yaratmadığı yüzeylerde daha yüksek hızlarda işleme yapılarak zamandan kazanç sağlama, yüksek hızda işleme gerektirmeyen işlemelerde daha az hızda gidilerek harcanan akım değerinden, dolayısıyla harcanan toplam güçten kazanç sağlamak mümkündür. Hazırlanan program ile hız değerleri anlık değiştirilebildiğinden hedeflenen kazançları sağlamak mümkün hale gelmiştir.

5. SONUÇ VE ÖNERİLER

Bu çalışmada adaptif kontrollü 3 eksenli CNC tezgâh prototipi hazırlanmıştır. Çalışmada, önce 3 adet servo motorun eşzamanlı olarak kontrol programı interpolarasyon ve noninterpolarasyon özelliği kullanılarak hazırlanmıştır. İkinci aşamada motorların adaptif kontrolü sağlanmıştır. Adaptif kontrollü CNC makine prototipi; 3 servo motor, 3 motor sürücüsü, dokunmatik panel, plc ve haberleşme kabloları, güç kaynağı gibi yardımcı aparatlar, 3 eksenli CNC makinesine montajı yapılarak oluşturulmuştur. Harici konum algılayıcılar ile sistemin adımlarının da kontrolü geri besleme ile sağlanarak olası hatalara karşı arındırılmıştır. Sistemin güvenlik kontrolü, adaptif hale getirilebilmesi ve PLC ile veri alışverişi yapılabilmesi KEPServerEX OPC programı kullanılmıştır. Yapılan prototip için hazırlanan programlar ve arayüz sayesinde; motorun P faktörü değerleri değiştirilerek akım değerlerinden dolayısıyla harcanan güçten kazanç sağlanabilmiş buna ilave olarak anlık hız değişimleri yapılabildiğinden zamandan ve hıza bağlı değişen akım değerlerinden kazanç sağlanmıştır.

Yapılan çalışmalara ek olarak, tezgahın malzeme cinsini tanımlayabilmesi ve buna bağlı olarak işleme modunun belirlenmesi, takım ucunu değiştirmesi, kaba işlemeler için ekonomik işleme modunun seçilebilmesi vb. programlamaların arttırılarak zaman ile etkinliğin arttırılması hedeflenmektedir. Ayrıca CNC ‘ye eklenilecek olan kamera ile yapılan işlemeler eşzamanlı olarak gözlemlenebilecek ve olası hata durumlarında güvenlik önlemleri alınabilecek, malzemenin olası fire miktarı azaltılabilecektir. CNC makinesinde yapılacak olan ses ölçümü, titreşim ölçümü gibi yöntemler kullanılarak elde edilecek olan geri beslemeler ile adaptifliği daha iyi hale getirmek mümkün olacaktır.

KAYNAKLAR

1. Besyaprak, Y. (2016). *Matlab Opc Araçlari Kullanilarak Servo Motorun Pozisyon Kontrolü*, Yüksek Lisans Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara.

2. Zhang, L. X., Sun, R. Y., Gao, X. S., ve Li, H. B. (2011). High speed interpolation for micro-line trajectory and adaptive real-time look-ahead scheme in CNC machining. *Science China-Technological Sciences*, 54(6), 1481-1495.

3. Jee, S., ve Koren, Y. (2004). Adaptive fuzzy logic controller for feed drives of a CNC machine tool. *Mechatronics*, 14(3), 299-326.

4. Ambrizal, N. H. B., Farooqi, A., Alsultan, O. I., ve Bin Yusoff, N. (2017). *Design and Development of CNC Robotic Machine Integrate-able with Nd-Yag Laser Device.* Advances in Material ve Processing Technologies Conference, 184, 145-155.

5. Pejryd, L., Repo, J., ve Beno, T. (2012*). Machine Tool Internal encodes Sensors for the Detection of Tool Wear*. 3rd Cirp Conference on Process Machine Interactions, 4, 46.

6. Büyüksah, U. (2005). *3 Eksenli CNC Tezgah Tasarımı,* Yüksek Lisans Tezi, Yıldız teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.

7. Kayalık, M. (2009). *Bilgisayar Kontrollü Dik İslem Tezgahı Tasarımı ve Uygulaması*, Yüksek Lisans Tezi, Dokuz Eylül Universitesi Fen Bilimleri Enstitüsü, İzmir.

8. Y., T., Korkut, İ., ve Akıncı, M. T. (2004). CNC Torna Tezgahları İçin Dialog Metodu Kullanılarak Nc Kod Türetilmesi, *Pamukkale Üniversitesi Mühendislik Fakültesi Dergisi*, 10(3), 299-304.

9. Lin, F. J., Shieh, H. J., Shieh, P. H., ve Shen, P. H. (2006). An adaptive recurrent-neural-network motion controller for X-Y table in CNC machine, *Ieee Transactions on Systems Man and Cybernetics Part B-Cybernetics*, 36(2), 286-299.

10. Mo, W., Wang, M., Lin, J. S., Zan, H., Sun, G. (2014). *Control system based on PLC for winding machine.* 2014 International Symposium on Computer, Consumer and Control, 74-77.

11. Birok, E. (2013). *Yüksek mertebeden teorik bir sistemin S7-300/400 tipi PLC için sayısal benzetim ile dijital PID kontrolör tasarımı*, Yüksek Lisans Tezi, Kocaeli Üniversitesi Fen Bilimleri Enstitüsü, Kocaeli.

12. Bayrak, G., Murr, P. J., Ulewicz, S., Vogel-Heuser, B. (2012). *Comparison of a transformed Matlab/Simulink model into the programming language CFC on different IEC 61131-3 PLC environments.* 2012 IEEE 17th Conference on Emerging Technologies ve Factory Automation (ETFA).

13. Bedir, Y. (2010). *Manyetik alan kuvveti ile ileri-geri doğrusal hareket yapan bir piston hareketinin PLC-SCADA kumanda sistemi ile kontrolü*, Yüksek Lisans Tezi, Mustafa Kemal Üniversitesi Fen Bilimleri Enstitüsü, Hatay.

14. Akçura, D. (2010). *PLC ve asenkron motoru ile garaj kapısının kontrolü*, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 50.

15. Cordes, M., Hintzea,W., Altintas, Y., (2019).Chatter stability in robotic milling. *Robotics and Computer Integrated Manufacturing* 55 (2019) 11–18.

16. Namazia, M., Altintas, Y., Abeb,T., Rajapakse, T., (2007). Modeling and identification of tool holder–spindle interface dynamics. *International Journal of Machine Tools ve Manufacture,* 47 (2007), 1333–1341.

17. Comak,A., Altintas, Y., (2017). Mechanics of turn-milling operations. *International Journal of Machine Tools ve Manufacture,* 121 (2017), 2–9.

18. Aslan,D., Altintas, Y., (2018). On-line chatter detection in milling using drive motor current commands extracted from CNC. *International Journal of Machine Tools and Manufacture,* 132 (2018), 64–80.

19. Yang, J., Aslan,D., Altintas, Y.,(2018). A feedrate scheduling algorithm to constrain tool tip position and tool orientation errors of five-axis CNC machining under cutting load disturbances. *CIRP Journal of Manufacturing Science and Technology,* 23 (2018), 78–90.

20. Dursun, M., Polat Tektaş, E., Özden, S., (2019). *A New Protoype Desıgnıng Of Cnc Machıne By Usıng Adaptıve Control*. ECSAC'19 5 th Europan Conference on science, art culture, (2019) 433.

21. Yüksel, İ.,(2006). *Otomatik Kontrol-Sistem Dinamiği ve Denetim Sistemleri* (5. Baskı). Türkiye: Nobel Yayın ve Dağıtım, 38-39.

22. Groover, P.,(1987). *Automation Production Systems and Computer Integrated Manufacturing*. Prentice Hall, London.

23. Asiltürk, İ., (2007). *Testere İle Kesme İşleminde Yapay Zeka Tabanlı Adaptif Kontrol Uygulaması*, Yüksek Lisans Tezi, Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya.

24. Liu,Y., Wang, C., (1999). Neural Network Based Adaptive Control and Optimization in the Milling Process. *Journal of Advanced Manufacturing Tech*., 15, 791-795.

25. Liu,Y., Cheng,T., Zuo,L., (2001). Adaptive Control Constraint of Machining Processes. *Journal of Advanced Manufacturing Technology*, 17, 720 - 726.

26. Altintas, Y., (1992), Prediction of Cutting Forces and Tool Breakage in Milling from Feed Drive Current Measurements. *Transactions of ASME, Journal of Engineering for Industry*, 386-392.

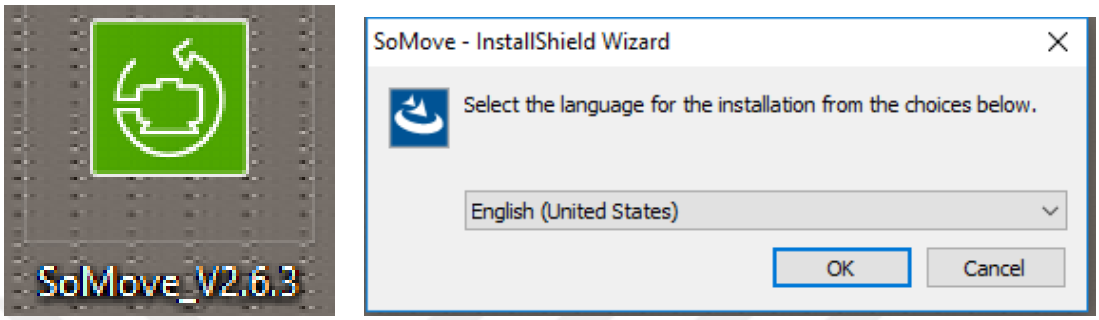
27. Li, X., Alexandar D., Patri K. V., (2000). Current sensor based feed cutting force intelligent estimation and tool. wear condition monitoring. *IEE Transactions on instrumentation and Measurement, 51(1), 289-296.*

28. Internet Örnek G-Code :https://www.tezmaksan.com.tr/CNC-torna-programlamada-kullanilan-g-ve-m-kodlari-758-haberdetay-585 Son Erişim Tarihi: 30.05.2019

EKLER

EK-1. SoMove programının kurulması

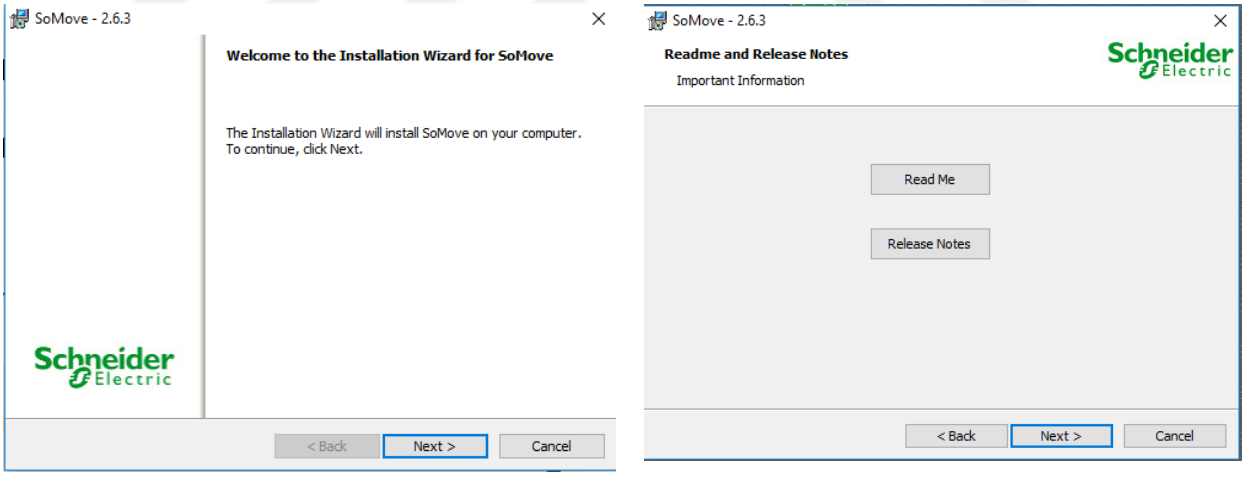
SoMove programının kurulumunu başlatabilmek için Resim 1.1.a'da yer alan ikon üzerine tıklanır. Resim 1.1.b'de dil seçimi yapılarak Ok butonu tıklanır.



(a) (b)

Resim 1.1.(a) SoMove yükleme ikonu (b) SoMove dil seçimi

Resim 1.2.a ve b'de yer alan yükleme ekranlarında next butonuna basılarak devam edilir.

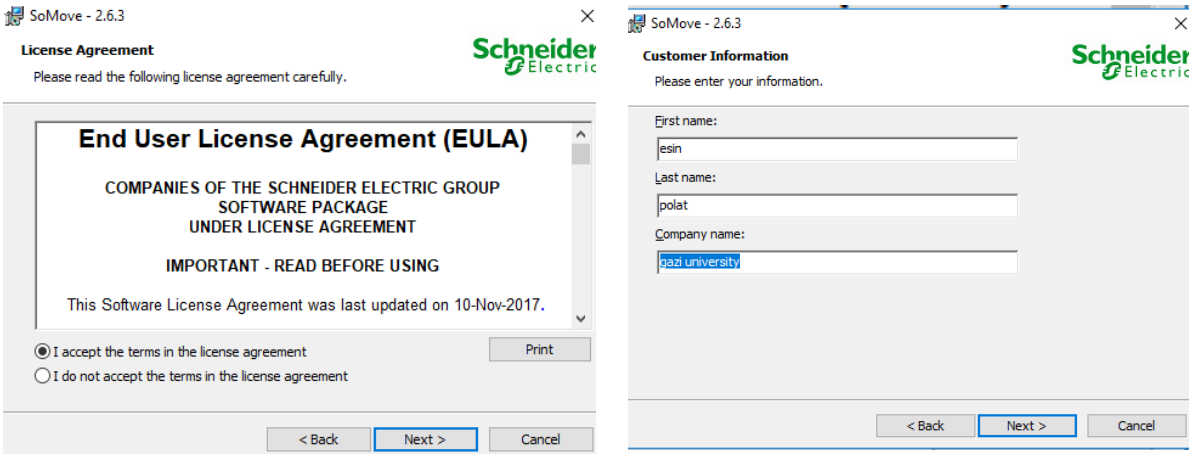


(a) (b)

Resim 1.2.(a) SoMove yükleme ekranı-1(b) SoMove yükleme ekranı-2

Resim 1.3.a ve b'de yer aldığı gibi onay ikonu tıklanıp, gerekli boşluklar doldurulup, next butonuna basılarak devam edilir.

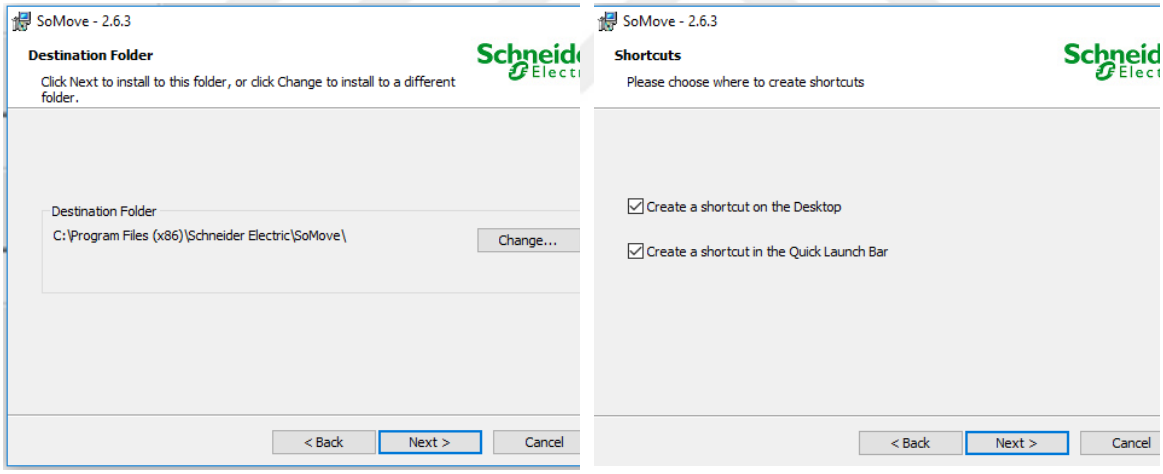
EK-1. (devam) SoMove programının kurulması



(a) (b)

Resim 1.3.(a) SoMove yükleme ekranı-3 (b) SoMove yükleme ekranı-4

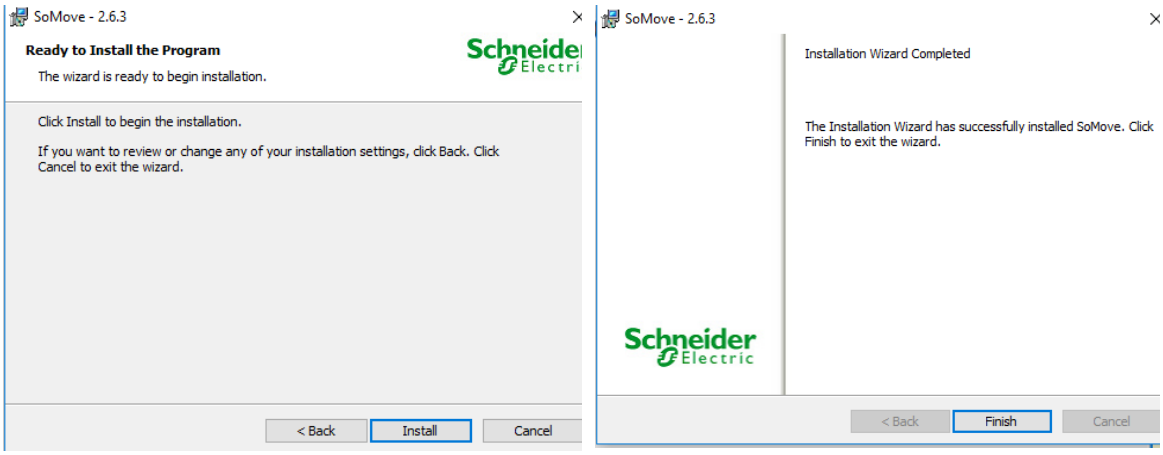
Resim 1.4.a ve b'de yer alan yükleme ekranlarında next butonuna basılarak devam edilir.



(a) (b)

Resim 1.4.(a) SoMove yükleme ekranı-5 (b) SoMove yükleme ekranı-6

EK-1. (devam) SoMove programının kurulması

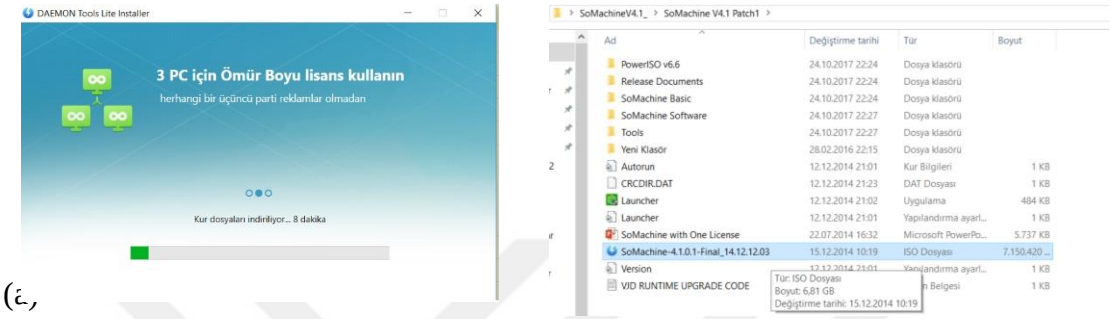


(a) (b)

Resim 1.5.(a) SoMove yükleme ekranı-7 (b) SoMove yükleme ekranı-8

EK-2. SoMachine programının kurulumu

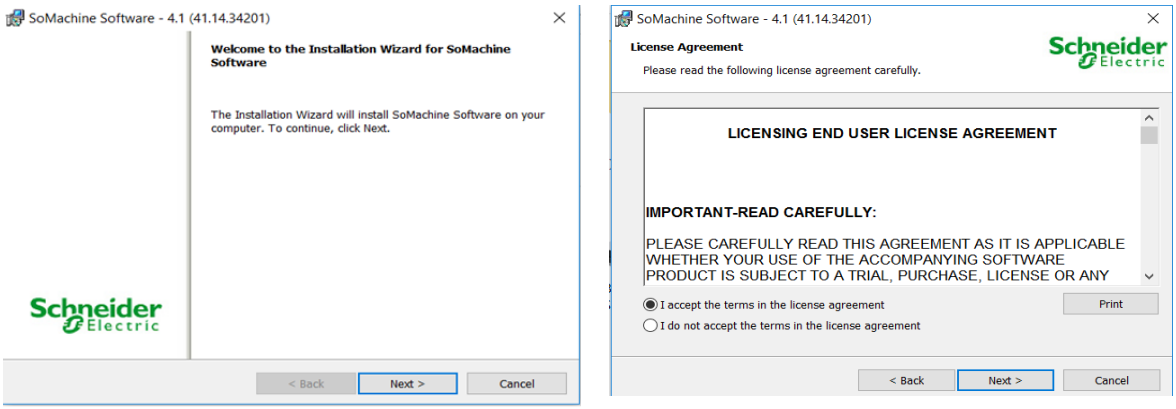
İlk olarak sanal disk kurulumu gerekmektedir. Resim 2.1.a'da görüldüğü gibi sanal disk oluşturulduktan sonra Resim 2.1.b'deki gibi SoMachine 4.1.0.1.Final tıklanarak kurulum başlatılmaktadır.



(a) (b)

Resim 2.1.(a) Sanal disk kurulumu (b) SoMachine kurulum ekranı-1

Resim 2.2.a'da yer alan ekranda Next tuşuna basılır ve Resim 2.2.b'de lisans sözleşmesini onaylayıp tekrar Next tuşuna basılır.

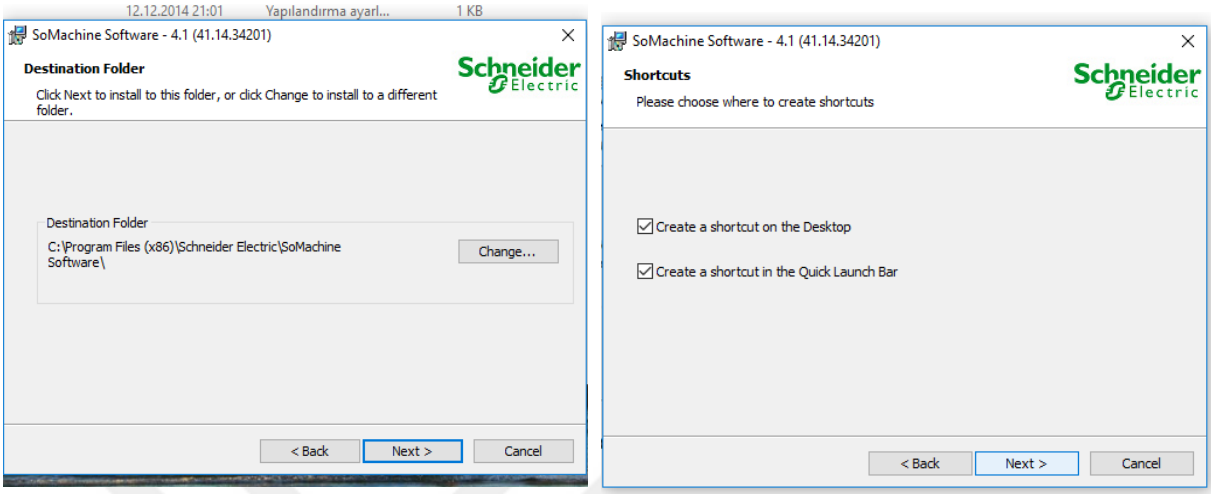


(a) (b)

Resim 2.2.(a) SoMachine kurulum ekranı-2 (b) SoMachine kurulum ekranı-3

Resim 2.3.a'da yer alan ekranda Next tuşuna basıyoruz ve Lisans Sözleşmesini onaylanır vetekrar Next tuşuna basılır. Resim 2.3.b'de yer alan Next tuşuna basıyoruz. İkonlar tıklanıp Next tuşuna basılır.

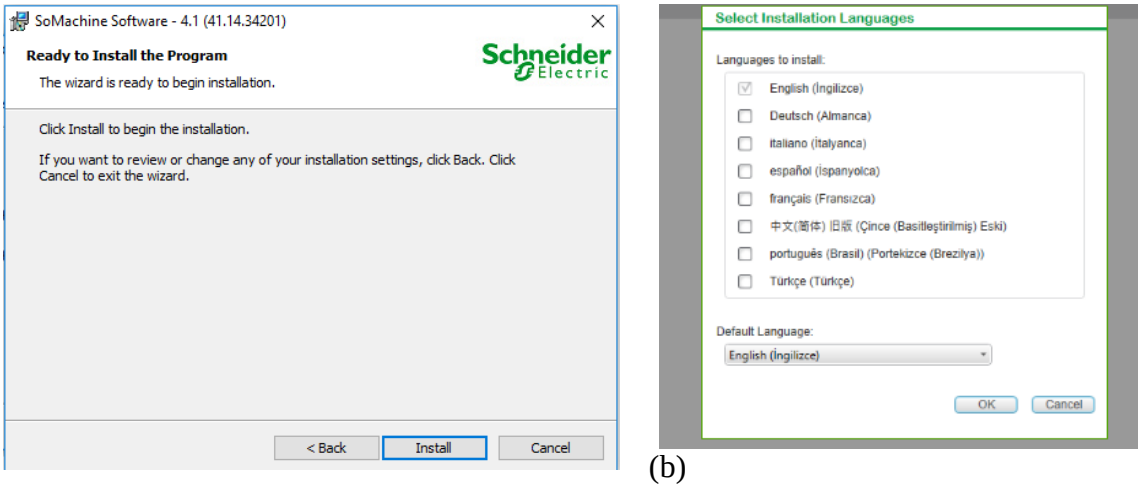
EK-2. (devam) SoMachine programının kurulumu



(a) (b)

Resim 2.3.(a) SoMachine kurulum ekranı-4 (b) SoMachine Kur kurulum ekranı-5

Resim 2.4.a'da yer alan install tuşuna basılır. Ardından çıkan ekranda finish e basıyoruz. Install New versiyon seçilir ve Resim 2.4.b'de dil seçimi yapılır.

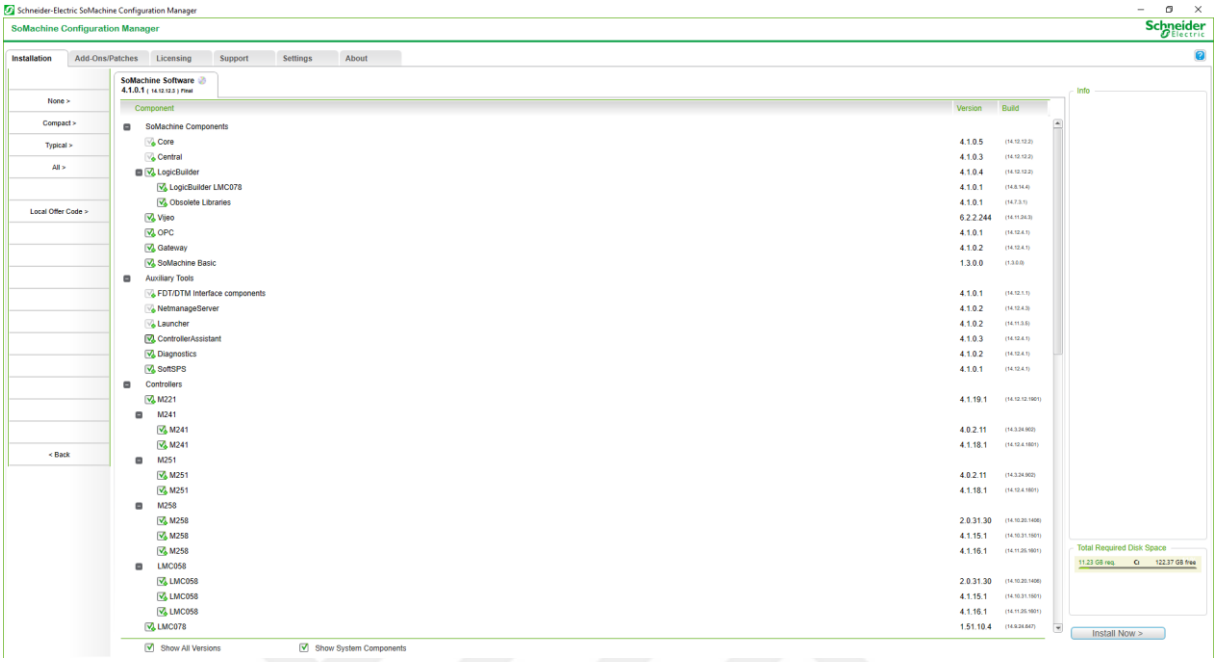


(a) (b)

Resim 2.4.(a) SoMachine kurulum ekranı-6 (b) SoMachine kurulum ekranı-7

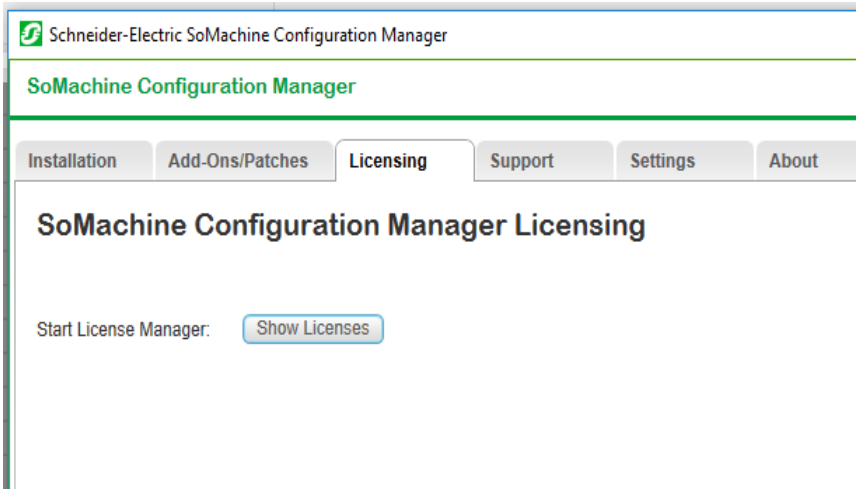
Resim 2.5'de yer alan ekranda Components kısmında boşlukta olan kutucuklar tıklanarak içinde yeşil tik işareti oluşması sağlanır ve install now a basılır. Daha sonra yükleme ekranı çıkacaktır. Burada dikkat edilmesi gereken husus kullanılacak PLC modelinin seçili olması gerekmektedir.

EK-2. (devam) SoMachine programının kurulumu



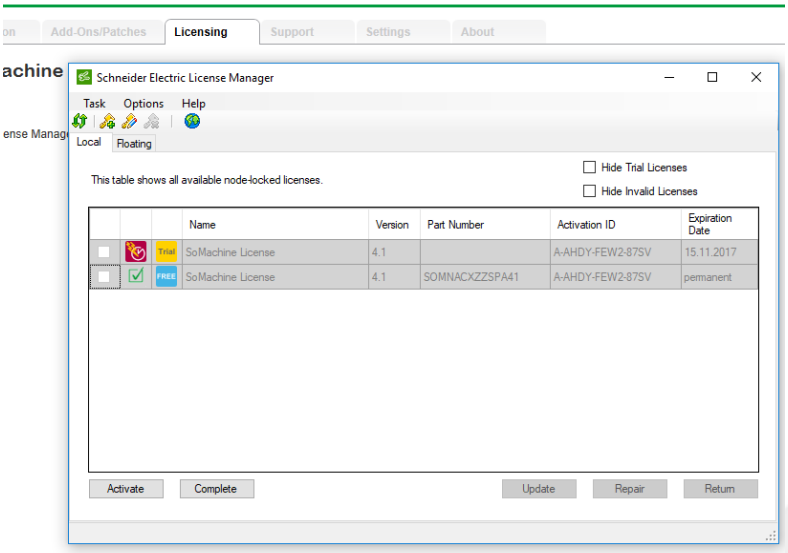
Resim 2.5. SoMachine kurulum ekranı-8

Programın lisans tanımlamasının yapılabilmesi için Resim 2.6'da yer alan ekranda Licencing menüsünden Show Licenses butonuna tıklanır. Resim 2.7'de çıkan ekranda deneme lisansının bitiş süresi, aktif olup olmadığı buradan gözlemlenebilir. Burada Activation ID lisanslama işleminin tamamlanabilmesi için not edilmelidir.



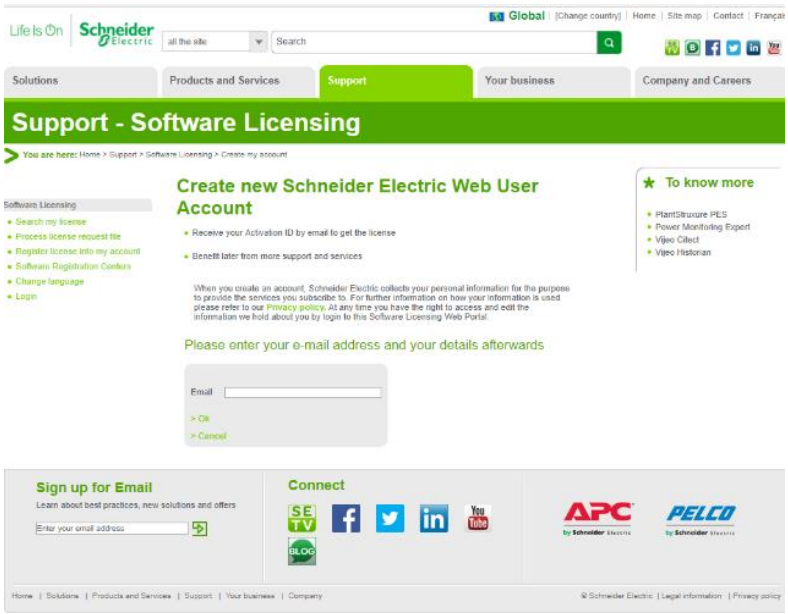
Resim 2.6. SoMachine lisans alma-1

EK-2. (devam) SoMachine programının kurulumu



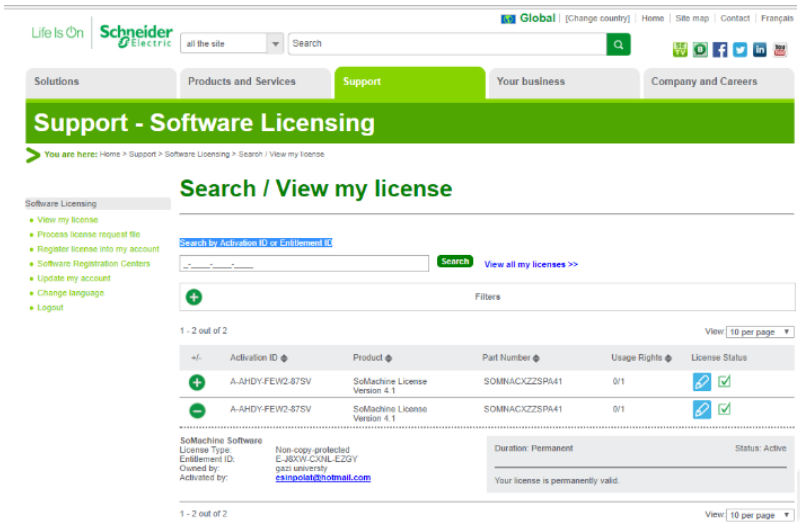
Resim 2.7 SoMachine lisans alma-2

Resim 2.8 ve Resim 2.9'da yer alan ekranlarda www.schneider-electric.com web sitesinden Support – Software/// Licensing Create new Schneider Electric Web User Account ile yeni bir hesap oluşturulur. Oluşturulan hesaba giriş yapıldıktan sonra daha önce kaydedilen Activation ID girilir ve aktifleştirilir.



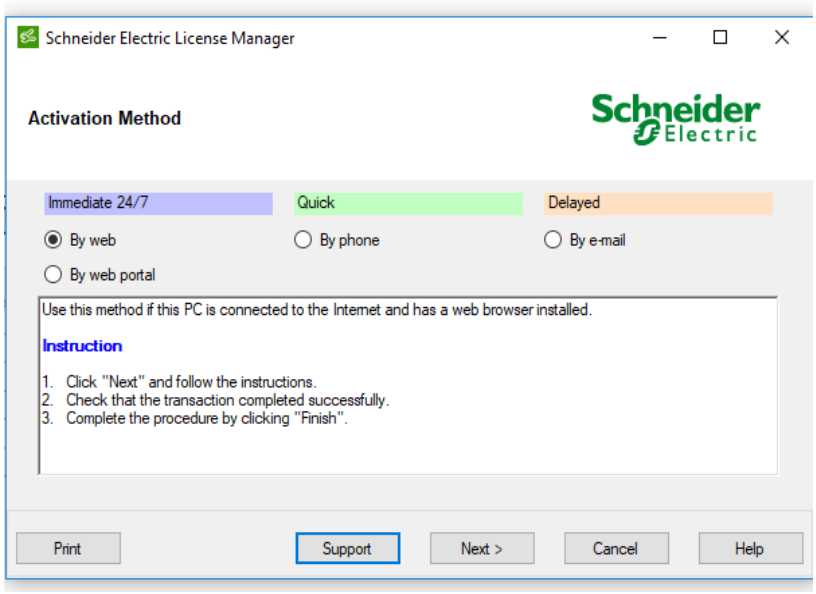
Resim 2.8. SoMachine lisans alma-3

EK-2. (devam) SoMachine programının kurulumu



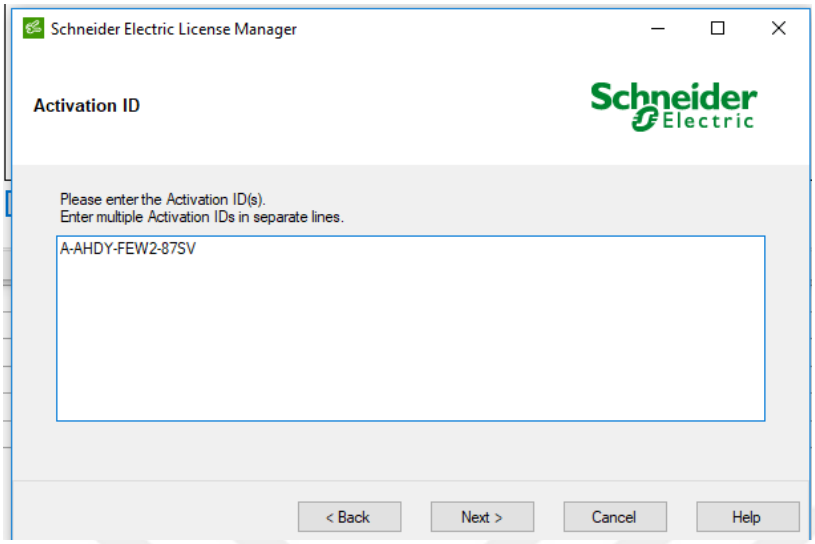
Resim 2.9. SoMachine lisans alma-4

Resim 2.10'da olduğu gibi tekrar Lisans Manager ekranı açılır By Web tıklanır ve next butonuna basılır. Resim 2.11 'de Activation ID yazılarak next tuşuna basılır.



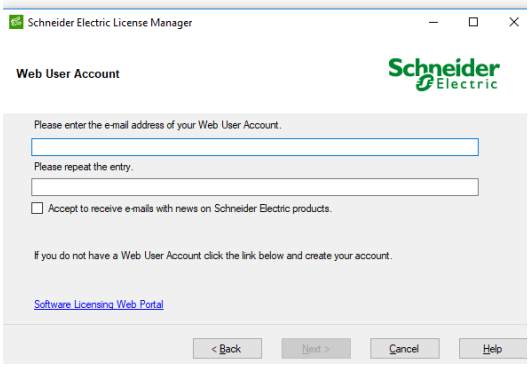
Resim 2.10. SoMachine lisans alma-5

EK-2. (devam) SoMachine programının kurulumu

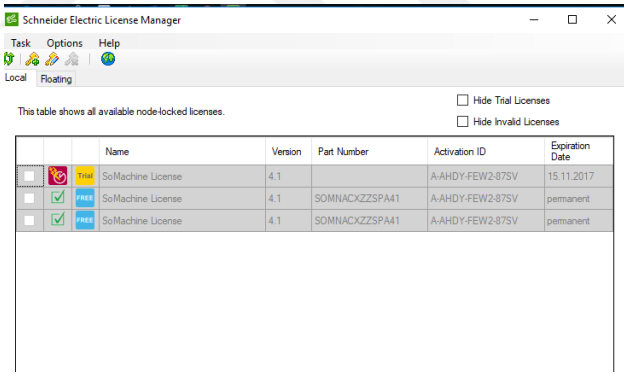


Resim 2.11. SoMachine lisans alma-6

Resim 2.12.a'da yer alan ekranda Web user account kısmına yukarıda hesap oluşturmak için mail adresi girilir ve next tuşuna basılır. Çıkacak olan ekranda finish butonuna basılarak lisans alma işlemi tamamlanır. Resim 2.12.b'deki ekranda ise Lisans onaylanmış olup, kalıcı (permanent) olduğu belirtilmektedir.



(a)



(b)

Resim 2.12.(a) SoMachine lisans alma-7 (b) SoMachine lisans alma-8

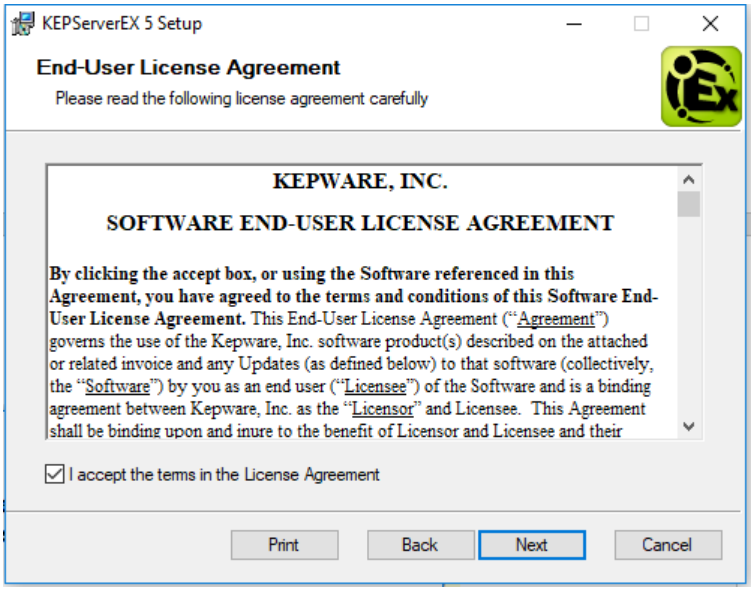
EK-3. Kepserver Programının Kurulması

Kepserver programının kurulumu ve programlaması için gerekli maddeler aşağıda sunulmuştur. Kepserver kurulum dosyası başlatılır, Resim 3.1'deki gibi next tuşuna basılır.



Resim 3.1. Kepserver kurulumu-1

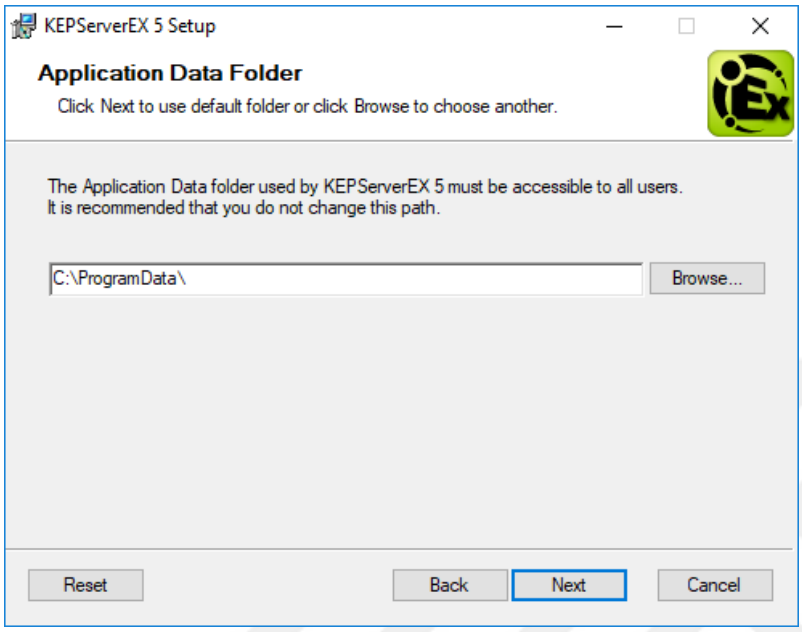
Resim 3.2'deki gibi accept seçeneğine aktif edilerek next tuşuna basılır.



Resim 3.2. Resim Kepserver kurulumu-2

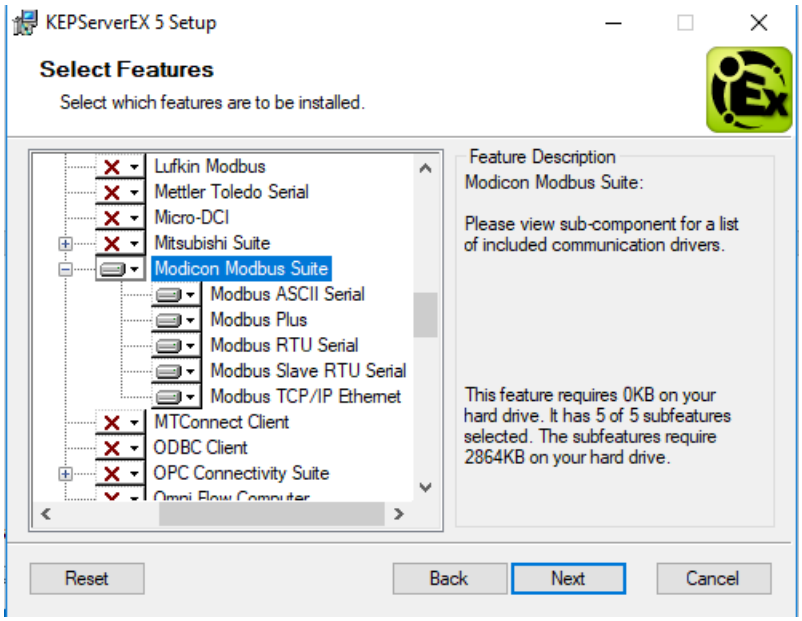
EK-3. (devam) Kepserver Programının Kurulması

Resim 3.3'de Programın bilgisayar içinde kurulacağı alan seçimi yapılır.



Resim 3.3. Kepserver kurulumu-3

Haberleşmesini planladığımız portların seçimi Resim 3.4'deki gibi yapılır. Daha sonra gelecek ekranlarda next tuşuna basılarak kurma işlemi tamamlanır.



Resim 3.4. Kepserver kurulumu-4

EK-4. G-CODE ve M-CODE ile programlama

G-code geometric code'nin kısaltılmış halidir. G00 – G99 arasında numaralandırılan G kodları, talaş kaldırma işlemlerinde **CNC** programlarının yazımında en önemli kodların başında gelmektedir. En sık kullanılan G-code 'lar aşağıda sıralanmıştır.

GOO – GO : Boşta hızlı ilerleme, kesici takımın talaş kaldırmadan boştaki hızlı hareketi
GO1 – G1 : Kesici takımın iş parçası ile temas halinde, talaş kaldırarak yaptığı hareket
GO2 – G2 : Saat dönüş yönünde (CW) eğrisel interpolasyon
G03 – G3 : Saat dönüş yönünün tersi yönde (CCW) eğrisel interpolasyon
G04 – G4 : Geçici bekleme yapma
G17 : X – Y düzlem seçimi yapma
G18 : X – Z düzlem seçimi yapma
G19 : Y – Z düzlem seçimi yapma
G20 : İnç ölçü sisteminde giriş
G21 : Metrik ölçü sisteminde giriş
G27 : Tezgah referans noktasına dönüş kontrolü
G28 : CNC Tezgahı sıfır noktasına (Zero Point) gönderme
G31 : Satır atlama komutu
G32 : Diş açma komutu
G34 : Değişken adımlı diş açma komutu
G40 : Kesici takım ucu radyüs telafisi iptali
G41 : Kesici takım ucu radyüs telafisi (Solda)
G42 : Kesici takım ucu radyüs telafisi (Sağda)
G50 : Maksimum tezgah mili devir sayısı sınırlama
G54 : İş parçası sıfır noktası
G55 - G59 : İş parçası farklı referans noktaları seçimi
G70 : Finiş tornalama çevrimi
G72 : X eksenine paralel olarak otomatik kaba dışçap tornalama
G73 : Profile göre dış çap tornalama çevrimi
G74 : Z ekseninde delik delme çevrimi
G75 : X ekseninde kanal açma ve kesme çevrimi
G76 : Çok pasolu otomatik diş açma çevrimi
G80 : Çevrimlerin iptal edilmesi
G83 : Kademeli olarak delik delme çevrimi

EK-4. (devam) G-CODE ve M-CODE ile programlama

CNC Torna Tezgahı M kodları (Mnn)

M kodları, CNC torna programlarının yazımında kullanılan yardımcı fonksiyon kodlarıdır. Bu kodlar kesme işlemini kolaylaştıran kodlar olup aynı satırda iki M kodu kullanılmaz. M harfi ve bunu takip eden 2 haneli rakamlardan oluşur.

M00 – M0 : Program durdurma. Bu komut okunduğunda program durur. Devam etmek için Cycle Start düğmesine basılır.

M01 – M1 : isteğe bağlı durdurma (Optional Stop Switchi aktif ise program durur). Devam etmek için Cycle Start düğmesine basılır.

M02 – M2 : Program sonu (Tek parça imalatında)

M03 – M3 : Tezgah milini saat yönünde (CW) döndürme

M04 – M4 : Tezgah milini saat yönünün tersine(CCW) döndürme

M05 – M5 : Tezgah milini durdur

M06 – M6 : Kesici takımı değiştir

M08 – M8 : Soğutma sıvısını açma

M09 – M9 : Soğutma sıvısını kapatma

M18 : Ayna çözme

M19 : Ayna kilitleme

M30 : Programı sonlandırma ve tekrar program başına dönme

Çizelge 4.1'de G-code dizilimi ve program kodları verilmiştir.

EK-4. (devam) G-CODE ve M-CODE ile programlama

Çizelge 4.1 G-Code Dizilimi

Aciklama	Simge	Simgenin Anlami
Program numarasi	O (EIA)/(ISO)	Program numarasi
Blok (Satir) sira numarasi	N	Sira numarasi
Hazirlik fonksiyonu	G	Bir hareket modunu gosterir (Dogrusal, dairesel, vs.)
Pozisyonlama fonksiyonlari	X, Z	Her bir eksenin hareket komutlari (Mutlak Tip)
	U, W	Hareket mesafe ve yon komutlari(Artimsal Tip)
	I, K	Baslangic noktasinin yayin merkezine olan uzakliklari
	R	Daire yaricapi,kose R,kenar R
Ilerleme Fonksiyonu	F	Kesme ilerlemesi
Yardimci fonksiyonlar	M	Makinanin calisan fonksiyonlarini acip kapatan komutlar
Fener mili hiz fonksiyonu	S	Ana fener milinin devir sayisini belirler
Takim fonksiyonu	T	Takim numarasi ve takim ofset numarasi
Bekleme	P, U, X	Bekleme zamani
Alt program Numarasi	P	Alt program cagirma numarasi
Sira Numarasi	P, Q	Birlesik tekrar cevriminde baslangic ve bitis satir numaralari
Tekrar Sayisi	L	Alt program tekrar sayisi
Parametreler	A, D, I, K	Sabit cevrimlerdeki parametreler

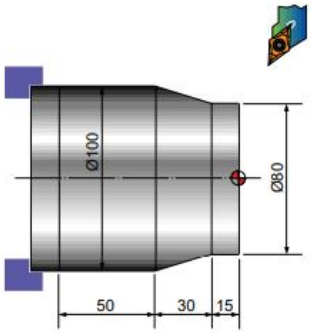
Bir blok (Satir)

N	G	X Z	F	S	T	M		:
Satir Numarasi	Hazirlik fonksiyonu	Pozisyonlama fonksiyonu	Ilerleme fonksiyonu	Fener Mili devir fonksiyonu	Takim fonksiyonu	Yardimci fonksiyon		Satir Sonu

G-Code ile yazılmış bir program Resim 4.1’de gösterilmiştir.

G01 PROGRAM

Ornek1)



O0001 :

N10 G50 S1500 T0100 M42 :
G96 S180 M03 :
G00 X100.5 Z5.0 T0101 M08 :
G01 Z-95.0 F0.25 :
G00 U2.0 Z0.5 :
G01 X-1.6 F0.2 :
G00 X95.0 W1.0 :
G01 Z-37.3 F0.25 :
X100.0 Z-45.5 :
G00 Z1.0 :
X90.0 :
G01 Z-29.8 :
X95.0 Z-37.3 :
G00 Z1.0 :
X85.0 :
G01 Z-22.3 :
X90.0 Z-29.8 :
G00 Z1.0 :
X80.5 :
G01 Z-15.55 :
X85.0 Z-22.3 :
G00 X200.0 Z200.0 M09 T0100 :
M01 :

N20 G50 S2000 T0300 :
G96 S200 M03 :
G00 X85.0 Z5.0 T0303 M08 :
Z0 :
G01 X-1.6 F0.2 :
G00 X80.0 Z3.0 :
G42 Z1.0 :
G01 Z-15.0 F0.18 :
X100.0 Z-45.0 :
Z-95.0 :
G40 U2.0 W1.0
G00 X200.0 Z200.0 M09 T0300 :
M30 :

G50 : Fener mili max donus hizi ayarlama komutu
G96 : Sabit yuzey hizi kontrolu komutu
G40 : Takim uc yaricapi telafisi iptali
G42 : Saga takim uc yaricapi telafisi

Resim 4.1 Örnek G-Code

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı :POLAT TEKTAŞ, Esin

Uyruğu : T.C.

Doğum tarihi ve yeri : 07/07/1984 Niğde

Medeni hali : Evli

Telefon : 0 (543) 500 30 80

Faks : 0 (312) 417 08 25

e-mail : esinpolat@hotmail.com



Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Yüksek lisans	Gazi Üniversitesi /Elk. ve Elktr. Müh.	Devam ediyor
Lisans	Niğde Üniversitesi /Elk. ve Elktr. Müh.	2006
Lise	Niğde Anadolu Lisesi	2002

İş Deneyimi

Yıl	Yer	Görev
2007-Halen	Milli Savunma Bakanlığı	Milli Savunma Uzmanı
2006-2007	RCL Mühendisi	Proje Mühensisi

Yabancı Dil

İngilizce

Yayınlar

Dursun, M., Polat Tektaş, E., Özden, S., 2019. *A New Protoype Desıgnıng Of Cnc Machıne By Usıng Adaptıve Control*. ECSAC'19 5 th Europan Conference on science, art culture, (2019) 433.

Hobiler

Yüzme, Gezme, Araştırma



GAZİ GELECEKTİR..