

**SAYISAL İŐARET İŐLEMCİLER İÇİN
GÖMÜLÜ SİSTEM DENEY SETİ TASARIMI**

**2011
YÜKSEK LİSANS TEZİ
ELEKTRONİK VE BİLGİSAYAR EĞİTİMİ**

Funda DEMİR

**SAYISAL İŐARET İŐLEMCİLER İÇİN
GÖMÜLÜ SİSTEM DENEY SETİ TASARIMI**

Funda DEMİR

**Karabük Üniversitesi
Fen Bilimleri Enstitüsü
Elektronik ve Bilgisayar Eğitimi Anabilim Dalında
Yüksek Lisans Tezi
Olarak Hazırlanmıştır**

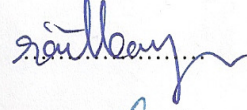
KARABÜK

Ocak 2011

Funda DEMİR tarafından hazırlanan “SAYISAL İŞARET İŞLEMCİLER İÇİN GÖMÜLÜ SİSTEM DENEY SETİ TASARIMI” başlıklı bu tezin Yüksek Lisans Tezi olarak uygun olduğunu onaylarım.

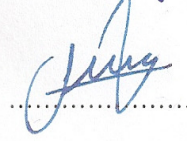
Doç. Dr. Raif BAYIR

Tez Danışmanı, Mekatronik Mühendisliği Anabilim Dalı



Yrd. Doç. Dr. Fecir DURAN

Tez Danışmanı, Elektronik ve Bilgisayar Eğitimi Anabilim Dalı

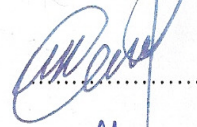


Bu çalışma, jürimiz tarafından oy birliği ile Elektronik ve Bilgisayar Eğitimi Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir. 01/02/ 2011

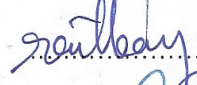
Ünvanı, Adı SOYADI (Kurumu)

İmzası


Başkan: Yrd. Doç. Dr. Rahmi CANAL (GÜ)



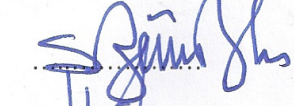
Üye :Doç. Dr. Raif BAYIR (KBÜ)



Üye : Yrd. Doç. Dr. Fecir DURAN (GÜ)



Üye : Yrd. Doç. Dr. Salih GÖRGÜNOĞLU (KBÜ)



Üye : Yrd. Doç. Dr. Hüseyin DEMİREL (KBÜ)

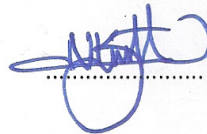


15.03/2011

KBÜ Fen Bilimleri Enstitüsü Yönetim Kurulu, bu tez ile, Yüksek Lisans derecesini onamıştır.

Doç. Dr. Nizamettin KAHRAMAN

Fen Bilimleri Enstitüsü Müdürü



“Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim.”

Funda DEMİR

ÖZET

Yüksek Lisans Tezi

SAYISAL İŞARET İŞLEMCİLER İÇİN GÖMÜLÜ SİSTEM DENEY SETİ TASARIMI

Funda DEMİR

Karabük Üniversitesi

Fen Bilimleri Enstitüsü

Elektronik ve Bilgisayar Eğitimi Anabilim Dalı

Tez Danışmanı

Doç. Dr. Raif BAYIR

Ocak 2011, 87 sayfa

Tez Danışmanı

Yrd. Doç. Dr. Fecir DURAN

Ocak 2011, 87 sayfa

Sayısal işaret işlemciler (Sİİ) günümüzde, askeri elektronik, tıp elektroniği, telekomünikasyon, enstrümantasyon, denetim sistemleri, ses ve görüntü işleme alanlarında yaygın olarak kullanılmaktadır. Yüksek hızda ve doğrulukta işlem yapabilmeleri ve karmaşık denetim yöntemlerinin uygulanmasında başarılı olduklarından tercih edilmektedirler. Bu çalışmada, teknik ve mühendislik eğitiminde 32 bit sayısal işaret işlemcilerin kullanımını ve yaygınlaşmasını sağlamak için prototip deney seti gerçekleştirilmiştir. Bu deney seti sayesinde, sayısal işaret işlemcilerin eğitiminde gösteri ve uygulama yöntemlerinin kullanımı etkin hale getirilmektedir. Bu sayede etkin ve kalıcı bir öğrenme olanağı sağlanmaktadır.

Deney setinde Texas Instrument firmasının TMS320F2812 sayısal işaret işlemcisi kullanılmaktadır. Gerçekleştirilen deney setinde kullanıcılar, denetim algoritmalarını ve sistem modellerini Matlab/Simulink ortamında kurarak, benzetimlerini ve uygulamalarını gerçekleştirmektedir. Sayısal işaret işlemci deneyleri için Matlab'ın Embedded Target for TC2000 DSP ve Real Time Workshop (RTW) araçları kullanılarak, yazılıma ihtiyaç duyulmadan, kodlarının üretimi otomatik olarak yapılmaktadır. Otomatik üretilen kodlar Texas Instrument firmasının Code Composer Studio 2000 yazılım geliştirme ortamında derlenerek F2812 eZdsp kartına yüklenmektedir. Kullanıcılar, F2812 eZdsp için geliştirilen deney seti ile analog ve sayısal elektronik deneylerini gerçekleştirebilmektedir. Bu deney seti fakülte ve yüksekokullarda sayısal işaret işleme eğitiminde rahatlıkla kullanılabilir.

Anahtar Kelimeler : Sayısal işaret işleme, TMS320F2812 deney seti, matlab simulink.

Bilim Kodu : 702.3.020

ABSTRACT

M.Sc. Thesis

Funda DEMİR

**EMBEDDED SYSTEM EXPERIMENT KIT DESIGN
FOR DIGITAL SIGNAL PROCESSORS**

**Karabük University
Graduate School of Natural and Applied Sciences
Department of Electronic and Computer Education**

Thesis Advisor

Assoc. Prof. Dr. Raif BAYIR

January 2011, 87 pages

Thesis Advisor

Assist. Prof. Dr. Fecir DURAN

January 2011, 87 pages

Digital signal processors are widely used in the fields of military electronics, medicine, telecommunications, instrumentation, control systems, audio and image processing today. They are preferred for their capability of high speed and accurate transaction and being successful in using complex control algorithms. In this study a prototype experiment kit performed to provide spread and use of 32-bit digital signal processors in technical and engineering education. With this experiment kit, using digital signal processors in demonstration technics of education is effectived. In this way effective and lasting learning is made. TMS320F2812 eZdsp produced by Texas Instrument is used in this study as a digital signal processor. The users make their

simulation and application with the developed experiment kit by establishing control algorithms and system models with Matlab/Simulink. In this study, generating code for digital signal processing experiments, performed automatically without software by using Embedded Target for TC2000 DSP ve Real Time Workshop. The automatically generated code is installed to the F2812 eZdsp board by complying on Texas Instrument Code Composer Studio 2000 software development environment. The users, with developed experiment kit for F2812, can be realized analogue and digital electronics experiments easily. This experiment kit can use easily for educating digital signal processing at faculties and colleges.

Keywords : Digital signal processing, eZdsp TMS320F2812 experiment kit, matlab simulink.

Science Code : 702.3.020

TEŐEKKÜR

Bu tez alıŐmasının planlanmasında, araŐtırılmasında, yürütülmesindeki tüm aŐamalarda ilgi ve desteęini esirgemeyen, tez danıŐmanlarım Sayın Do Dr. Raif BAYIR ve Sayın Yrd. Do. Dr. Fecir DURAN' a teŐekkürlerimi sunarım.

Deneysel alıŐmalar sırasında yardımlarını esirgemeyen baŐta eŐim Batıkan Erdem DEMİR olmak üzere, Öğr. Gör. Ali UYSAL, ArŐ. Gör. Emel KOAK ve Öğr. Gör. Serdar DERİCİ'ye teŐekkürü bir bor bilirim.

Her zaman beni destekleyen aileme sonsuz teŐekkürlerimi sunarım.

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	iv
ABSTRACT	vi
TEŞEKKÜR	viii
İÇİNDEKİLER	ix
ŞEKİLLER DİZİNİ	xii
ÇİZELGELER DİZİNİ	xv
EKLER DİZİNİ	xvii
SİMGELER VE KISALTMALAR DİZİNİ.....	xvii
BÖLÜM 1.	1
GİRİŞ	1
BÖLÜM 2.	6
SAYISAL İŞARET İŞLEME.....	6
2.1. TMS320F2812 SAYISAL İŞARET İŞLEMCİSİ.....	8
2.1.1. Fonksiyonel Blok Diyagram	9
2.1.2. F2812 Matematiksel İşlemler Birimi.....	10
2.1.3. Veri Hafıza Erişimi.....	11
2.1.4. Dahili Veriyolu Yapısı.....	12
2.1.5. Atomik Aritmetik Lojik Birim	12
2.1.6. Komut İş Hattı.....	13
2.1.7. Hafıza Haritası	14
2.1.8. Analog Sayısal Dönüştürücü ve Ayarları	15
2.1.9. DGM İşaretleri Arasında Ölü Zaman Ayarı	16
BÖLÜM 3.	18
YAZILIM GELİŞTİRME ARAÇLARI.....	18
3.1. CODE COMPOSER STUDIO	18

Sayfa

3.2. MATLAB/SIMULINK	20
3.2.1. IQmath Kütüphane Desteđi.....	22
3.2.2. Matlab Simulink Target Support Package TC2 Blokları.....	23
3.2.2.1. Sayısal Giriş Blođu	23
3.2.2.2. Sayısal Çıkış Blođu	24
3.2.2.3. Seri İletişim Arabirim Blođu.....	26
3.2.2.4. Analog Sayısal Dönüştürücü Blođu	27
3.2.2.5. Donanım Kesmesi Blođu	29
3.2.2.6. Darbe Genişlik Modülasyonu Blođu	30
3.2.2.7. Kuadratik Kodlayıcı Darbe Blođu.....	33
BÖLÜM 4.	35
eZdsp F2812 DENEY SETİNİN TASARIMI VE GERÇEKLEŞTİRİLMESİ	35
4.1. GERÇEKLEŞTİLEN DENEY SETİ BLOK DİYAGRAMI	35
4.2. eZdsp F2812 DSP GELİŞTİRME KARTI GENEL ÖZELLİKLERİ	35
4.2.1. eZdsp F2812 Bağlantıları ve İşlevleri	36
4.2.2. eZdsp F2812 Üzerindeki Ledler.....	37
4.3. DENEY SETİ ÜZERİNDE BULUNAN DEVRE KARTLARI	37
4.3.1. eZdsp F2812 Geliştirme Kartının Genişletilmesi.....	37
4.3.2. Motor Hız Denetim Devre Kartı	38
4.3.2.1. Kuadratik Kodlayıcı Darbe	39
4.3.2.2. Aç-Kapa Denetim Yöntemi	41
4.3.2.3. PID Denetim Yöntemi	41
4.3.3. Sayısal Giriş Devre Kartı	43
4.3.4. Sayısal Çıkış Devre Kartı	44
4.3.5. Röle Denetim Devre Kartı	45
4.3.6. Analog Giriş Devre Kartı.....	46
4.3.7. Yedi Parçalı Gösterge Devre Kartı.....	47
BÖLÜM 5.	49
eZdspF2812 DENEY SETİNDE GERÇEKLEŞTİRİLEN UYGULAMALAR.....	49
5.1. KAYAN LED UYGULAMASI	50

	<u>Sayfa</u>
5.2. TOPLAMALI KAYAN LED UYGULAMASI	56
5.3. GÖMÜLÜ MATLAB FONKSİYONU UYGULAMASI.....	58
5.4. BİNARY SAYICI UYGULAMASI	60
5.5. YEDİ PARÇALI GÖSTERGE UYGULAMASI	61
5.6. RÖLE DENETİM UYGULAMASI	64
5.7. ADIM MOTOR DENETİM UYGULAMASI	66
5.8. ADC SICAKLIK SENSÖRÜ UYGULAMASI	68
5.9. POTANSİYOMETRELİ ADC UYGULAMASI	71
5.10. DA MOTOR HIZININ AÇ-KAPA İLE DENETİM UYGULAMASI.....	73
5.11. DA MOTOR HIZININ PID İLE DENETİM UYGULAMASI	79
BÖLÜM 6.	83
SONUÇ VE ÖNERİLER	83
KAYNAKLAR.....	84
ÖZGEÇMİŞ.....	87

ŞEKİLLER DİZİNİ

Sayfa

Şekil 1.1. Sii üreten firmaların 2007 yılı gelir paylaşımı.	2
Şekil 1.2. 2000-2010 yılları arası gerçekleştirilen tezlerin yıllara göre dağılımı.	4
Şekil 2.1. A/D çevirici.	6
Şekil 2.2. TMS320F2812 işlemcisi bacak bağlantıları ve üst görünümü.	8
Şekil 2.3. TMS320F2812 fonksiyonel blok diyagramı.....	10
Şekil 2.4. TMS320F2812 iş hattı.....	13
Şekil 2.5. Hafıza haritası.	14
Şekil 2.6. ADC biriminin sıralı dönüşüm kipi.....	15
Şekil 2.7. Aynı besleme hattına bağlı anahtarların ölü zamanlı sürülmesi.	16
Şekil 2.8. EVA olay yöneticisine bağlı ölü zaman birimleri blok diyagramı.....	17
Şekil 3.1. CCS ana ekran görüntüsü.	19
Şekil 3.2. Matlab/Simulink üzerinden otomatik gömülü kod üretimi.	20
Şekil 3.3. Örnek Simulink modeli.	21
Şekil 3.4. Sayısal giriş blok parametreleri.	23
Şekil 3.5. Veri çözücü blok parametreleri.....	24
Şekil 3.6. Sayısal çıkış blok parametreleri.....	25
Şekil 3.7. Veri çoğullayıcı blok parametreleri.	25
Şekil 3.8. SCI alıcı bloğuna ait parametreler.	26
Şekil 3.9. SCI verici bloğuna ait parametreler.	27
Şekil 3.10. ADC blok parametreleri, ADC denetim menüsü.....	28
Şekil 3.11. ADC blok parametreleri, giriş kanalları menüsü.	28
Şekil 3.12. Donanım kesmesi blok parametreleri.....	29
Şekil 3.13. DGM bloğunun zamanlayıcı menüsün parametreleri.	31
Şekil 3.14. DGM bloğunun çıkışlar menüsü parametreleri.....	32
Şekil 3.15. DGM bloğunun lojik menüsü parametreleri.....	32
Şekil 3.16. DGM bloğunun ölü zaman menüsü parametreleri.....	33
Şekil 3.17. Kuadratik kodlayıcı darbe bloğuna ait parametreler.	34
Şekil 4.1. Deney setine ait blok diyagram ve gerçekleştirilen devre kartları.....	35
Şekil 4.2. eZdsp F2812 geliştirme kartı dış görüntüsü.	36

Sayfa

Şekil 4.3. Geliştirme kartının görünüşü.	37
Şekil 4.4. Motor hız denetim kartı görünüşü.....	39
Şekil 4.5. Kodlayıcı diskin görünümü ve oluşan QEP sinyalleri.	40
Şekil 4.6. QEPQ, QEPB ve QEPI sinyalleri ve birbirlerine göre konumları.	40
Şekil 4.7. Aç-Kapalı denetim ölçülen değer ve referans değer.	41
Şekil 4.8. PID kontrolör içyapısı.	42
Şekil 4.9. Sayısal giriş devre kartı açık şeması.	43
Şekil 4.10. Sayısal giriş devre kartı görünüşü.	44
Şekil 4.11. Sayısal çıkış devre kartı görünüşü.....	44
Şekil 4.12. Röle denetim devre kartı açık şeması.....	45
Şekil 4.13. Röle denetim devre kartı görünüşü.	45
Şekil 4.14. Analog giriş devre kartı açık şeması	46
Şekil 4.15. Analog giriş devre kartı görünüşü.....	47
Şekil 4.16. Yedi parçalı göstergeli devre kartı açık şeması.	47
Şekil 4.17. Yedi göstergeli ekran devre kartı görünüşü.....	48
Şekil 5.1. eZdspF2812 genişletme kartının CCS programına tanıtılması.....	49
Şekil 5.2. Yeni bir simulink model penceresinin açılması.....	50
Şekil 5.3. Kayan led uygulaması.	51
Şekil 5.4. Switch case fonksiyon parametreleri	51
Şekil 5.5. Ledsec blok yapısı.....	52
Şekil 5.6. Limitli sayıcı blok parametreleri.....	52
Şekil 5.7. Sayısal çıkış devre kartının bağlantısı.	53
Şekil 5.8. Simulink modelinin CCS programına gönderilmesi.....	53
Şekil 5.9. Sayısal giriş devre kartının bağlantısı.	54
Şekil 5.10. Tetiklemeli altsistem oluşturulması	55
Şekil 5.11. Giriş butonunun ayarlanması.....	55
Şekil 5.12. Switch case action subsystem bloğu ayarları.....	56
Şekil 5.13. Ledlerin yanma düzenlerinin karşılaştırılması.....	57
Şekil 5.14. Gömülü matlab fonksiyonu ile kayan led uygulaması.	58
Şekil 5.15. Gömülü matlab fonksiyonu içerisine case yapısının yazılımı.	59
Şekil 5.16. Binary sayıcı uygulaması.....	60
Şekil 5.17. Binary sayıcı uygulamasında ledlerin yanma düzeni.....	61

Sayfa

Şekil 5.18. Yedi parçalı gösterge üzerindeki ledlerin harflerle ifadesi.	61
Şekil 5.20. Sayıcı bloğunun üst limitinin ayarlanması.	63
Şekil 5.21. Yedi parçalı gösterge devre kartının bağlantısı	63
Şekil 5.22. Röle denetim uygulaması.	64
Şekil 5.23. Darbe üretici blok parametreleri.	65
Şekil 5.24. Röle denetim devre kartının bağlantısı.....	65
Şekil 5.25. Adım motor denetim uygulaması.....	66
Şekil 5.26. Sayıcı bloğunda örnekleme zamanı ayarının yapılması	67
Şekil 5.27. ADC sıcaklık sensörü uygulaması.	68
Şekil 5.28. ADC sıcaklık sensörü uygulaması için blok parametreleri ayarları.....	69
Şekil 5.29. ADC sıcaklık sensörü blok parametreleri.....	70
Şekil 5.30. ADC devre kartının bağlantısı.	70
Şekil 5.31. Potansiyometreli ADC uygulaması.	72
Şekil 5.32. Uygulamasıda ayarlanan ADC blok parametreleri.	72
Şekil 5.33. DA motor hızının aç-kapa ile denetim uygulaması.....	73
Şekil 5.34. Referans hız blok parametreleri.	74
Şekil 5.35. Karşılaştırmalı kaydedici blok parametreleri.....	74
Şekil 5.36. Double sayı çevirici blok parametreleri.	75
Şekil 5.37. Kazanç bloğu parametreleri.....	75
Şekil 5.38. PWM blok parametreleri zamanlayıcı menüsü ayarları.	76
Şekil 5.39. PWM blok parametreleri lojik menüsü ayarları.....	76
Şekil 5.40. QEP blok parametreleri.	77
Şekil 5.41. Motor hız denetim devre kartının bağlantısı.....	78
Şekil 5.42. DA motorunun aç-kapa hız denetimi.	79
Şekil 5.43. Matlab simulink ortamında gerçekleştirilen sistem modeli.....	79
Şekil 5.44. PID kontrolör blok parametreleri.....	80
Şekil 4.45. CCS ekranında program parametrelerinin değiştirilmesi	81
Şekil 5.46. DA motoru PID hız denetimi.....	82

ÇİZELGELER DİZİNİ

	<u>Sayfa</u>
Çizelge 1.1. Mikro-lojik gelir dağılımı.	2
Çizelge 3.1. Donanım kesmesi.	30
Çizelge 4.1. eZdsp F2812 bağlantıları.	36
Çizelge 4.2. DS1 ve DS2 led durumları.	37
Çizelge 4.3. P, I ve D ayarlarının bağımsız olarak etkileri.	43

EKLER DİZİNİ

- Ek I. Motor Hız Denetim Devre Kartı Açık Şeması
- Ek II. Sayısal Çıkış Devre Kartı Açık Şeması

SİMGELER VE KISALTMALAR DİZİNİ

SİMGELER

μ s	:	Mikro saniye
A	:	Amper
dev	:	Devir
dk	:	Dakika
Hz	:	Hertz
K	:	Kilo
kword	:	Kiloword
MHz	:	Mega hertz
ms	:	Mili saniye
ns	:	Nano saniye
R	:	Direnç
rpm	:	Repeat per minute (Devir/dakika)
V	:	Volt

KISALTMALAR

A/D	: Analog/Digital
ACC	: Accumulator
ADC	: Analog to Digital Converter
ALU	: Arithmetic Logic Unit
ARAU	: Auxiliary Register Arithmetic Unit
ASM	: Automatic Storage Management
CCS	: Code Composer Studio
COMCONA	: Compare Control Register A
CPU	: Central Processing Unit
D/A	: Digital/Analog
DA	: Doğru Akım
DBTCONA	: Dead-Band Timer Control Register A
DP	: Data Page
DSK	: DSP Starter Kit
DGM	: Darbe Genişlik Modülasyonu
DSP	: Digital Signal Processor
EEPROM	: Electrically Erasable Programmable Memory
EVA	: Event Manager Modules
FIFO	: First In First Out
FIR	: Finite Impuls Response
HFD	: Hızlı Fourier Dönüşümü
I	: Integer
IIR	: Infinite Impulse Response
JMP	: John's Macintosh Project"
JTAG	: Joint Test Action Group
LPT	: Line Print Terminal
MAC	: Media Access Control
MEB	: Milli Eğitim Bakanlığı
MIPS	: Millions of Instructions Per Second
PID	: Proportional Integral Derivative
PIE	: Peripheral Interrupt Expansion Manager

PLL	: Phase Locked Loop
PWM	: Pulse Width Modulation
Q	: Quotient
QEP	: Quadratic Encoder Pulse
RAM	: Random Access Memory
ROM	: Read Only Memory
RTW	: Real Time Workshop
SRAM	: Static Random Access Memory
SARAM	: Single Access Random Access Memory
Sİİ	: Sayısal İşaret İşleme
TI	: Texas Instrument
TMS	: Times Microwave Systems

BÖLÜM 1

GİRİŞ

Sayısal işaret işleme, işaretlerin sayısal ve sayısal sembollerle gösterimi ve bunların değerlendirilip işlenmesi ile ilgilendir. Böyle bir işlemin amacı bu sinyalin karakteristik parametrelerini tahmin etmek ya da sinyali anlaşılabilir, tahmin edilebilir bir formda sayısal ortama aktarmaktır.

XX. yüzyılın öncesine kadar işaret işleme analog sistemler ve cihazlar ile gerçekleştirilmektedir. Çok karmaşık işaret işleme gerektiren durumlarda sinyaller, manyetik şeritlere kaydedilmekte daha sonra sayısal bilgisayarlarda sonradan işlenmektedir. Sayısal bilgisayarların gelişimi işaret işleme tekniklerine de değişik boyutlar getirmiş bir işaret işleme sistemini analog olarak gerçekleştirmeden önce sayısal bilgisayarlarda benzetimini yapmak oldukça sık başvurulan bir yöntem haline gelmiştir. Bu şekilde yeni bir işaret işleme algoritması ya da sistem, esnek ve deneysel bir ortamda geliştirilir, test edilir ve ekonomik mühendislik imkanları ile geliştirilerek imal edilir duruma getirilmiştir.

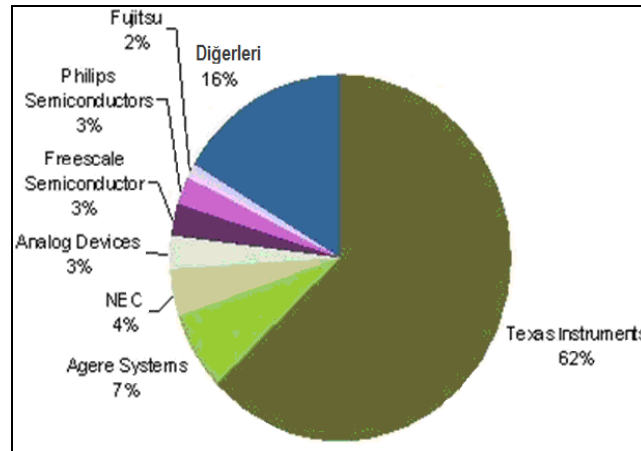
Sayısal bilgisayarların esnekliği, analog cihazlara olan ihtiyacı ortadan kaldırmaktadır. Böylece, tüm analog işlemlerin sayısal eşdeğerleri geliştirilmeye başlanmıştır. Bu yeni bakış “sayısal işaret işleme” alanında birçok algoritma geliştirilmesine imkan sağlamaktadır. Önceleri mikroişlemci veya mikrodenetleyici etrafında tasarlanan sayısal işaret işleme sistemleri, yüksek hız ve elektronik denetim gerektiren işlemler için yetersiz hale gelmektedir. Buradaki problem ileri denetleme algoritmalarındaki çok sayıda çarpma ve toplama işleminin gerçekleştirilememesidir. Bu algoritmaların yazılım ile gerçekleştirilmesi normal bir işlemci için çok fazla zaman almaktadır.

Modern denetim ve işaret işleme algoritmalarının gerektirdiği hız problemini çözmek amacıyla özel işlemci entegre devreleri tasarlanmıştır. Sayısal işaret işlemciler dediğimiz bu işlemciler, işaret işleme ve sayısal filtrelemede çok daha hızlı işlem yapabilecek özellikte olan mikroişlemcilerdir [1-3].

Günümüzde; askeri elektronik, tıp elektroniği, telekomünikasyon, enstrümantasyon, denetim sistemleri, ses ve görüntü işleme alanlarında yaygın olarak kullanılan sayısal işaret işlemciler yüksek hız ve doğrulukta işlem yapabilme özelliklerinden dolayı denetim uygulamalarında gün geçtikçe önemi artan elemanlar olmuştur. 2002 yılında, 8 bit mikrodenetleyiciler dünya işlemci gelirinde ikinci büyük paya sahip iken; Sİİ, genel mikrodenetleyici piyasasında hızla pay kazanmıştır. Bu yıllarda Sİİ, gelir payında en küçük paya sahip iken sonraki 5 yıl içerisinde yüzde 14'lük büyüme oranı ile, yüzde 10 büyümeye sahip mikroişlemci ve yüzde 8 büyüme oranına sahip mikrodenetleyicilerin önüne geçmiştir. Çizelge 1.1'de 2004-2011 yılları arası mikro-lojik gelir dağılımı yer almaktadır.

Çizelge 1.1. Mikro-lojik gelir dağılımı.

	2004	2005	2006	2007	2008	2009	2010	2011	ARTIŞ
MPU	30,488	34,970	37,605	41,141	45,280	52,020	56,154	61,331	%10
MCU	12,449	12,082	12,837	13,981	14,686	16,399	17,805	18,864	%8
Sİİ	7,798	7,635	9,083	11,046	11,391	13,205	15,996	17,664	%14
TOPLAM	50,734	54,687	59,526	69,168	71,358	81,624	89,956	97,859	%10

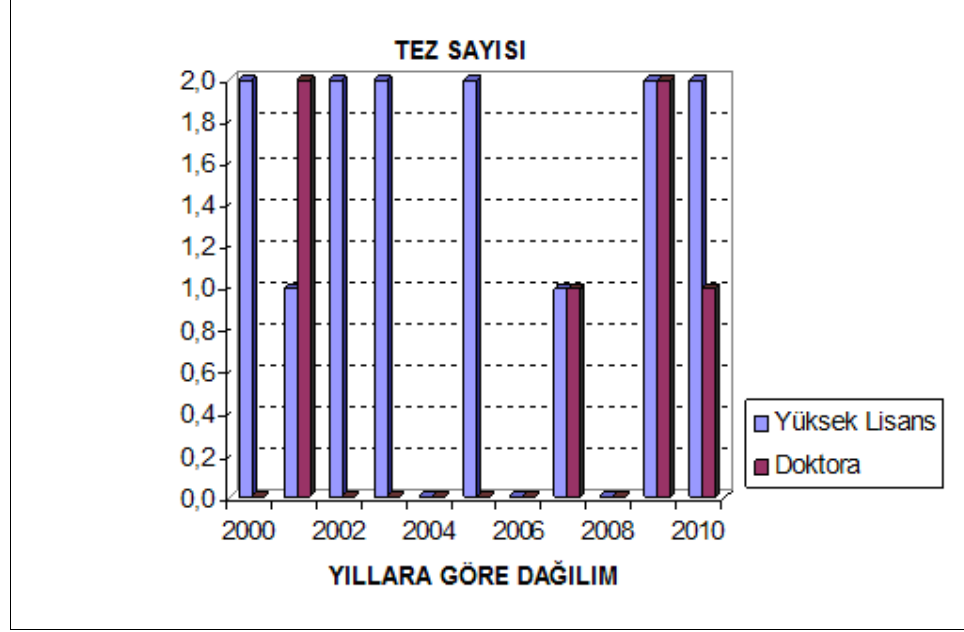


Şekil 1.1. Sİİ üreten firmaların 2007 yılı gelir paylaşımı.

Birçok alanda kullanılan Sİİ, yarı iletken pazarındaki asıl konumunu cep telefonu uygulamaları ile almıştır. Mobil telefon pazarında hızla gelişen Sİİ'nin 2011 yılında 12 milyar dolarlık bir gelire sahip olacağı düşünülmektedir. Bu da yılda yüzde 15'lik bir büyüme oranına işaret etmektedir. Üretilen her 3 telefonun ikisinde TI tarafından üretilen Sİİ'ler kullanılmakta iken 2007 Sİİ gelir paylaşımında firmaların pazar payı dağılımı Şekil 1.1'de görülmektedir [4].

Dünyada hızla yaygınlaşan Sİİ kullanımı, dünyada ve ülkemizde “Sayısal işaret işleme” derslerinin, teknik ve mühendislik eğitimi veren okulların ortaöğretim, lisans ve lisans üstü programlarında yerini almasını sağlamıştır. İçeriklerine bakıldığında bu derslerde işlenen konularının yaklaşık olarak aynı olduğu görülmüştür. Dersin içeriği, ayrık zamanlı işaretlerin ve sistemlerin oluşturulması ile başlarken dönem sonunda IIR (Infinite Impuls Response) ve FIR (Finite Impuls Response) filtrelerin tasarımı ile son bulmaktadır. İstanbul, Atılım, Dumlupınar, Tunceli, Anadolu, Dicle, Osmangazi, Uludağ, Ege, Dokuz Eylül, Kocaeli, Karabük, Süleyman Demirel, İstanbul Teknik Üniversitelerinde ve Milli Eğitim Bakanlığı'na (MEB) bağlı teknik liselerde bu dersi veren öğretmenlerden alınan bilgilere dayanarak sayısal işaret işleme dersinin teorik eğitimle birlikte genellikle matlab ortamında uygulamaların yapıldığı; Atılım, Karabük, Anadolu, İstanbul, İstanbul Teknik, Dicle ve Uludağ üniversitelerinde ve MEB'e bağlı teknik liselerde ise TMS320LF2510, TMS320C6711, TMS320C6713, TMS320C6416, TMS320F2812 işlemcilerinin geliştirme kartlarıyla uygulamaların ve projelerin yapıldığı görülmüştür [5-7].

Konuyla ilgili gerçekleştirilen akademik çalışmalar araştırıldığında son on yıl içerisinde Yükseköğretim Kurumu (YÖK) ulusal tez merkezi internet sitesine kayıtlı olan 14 adet yüksek lisans tezinin ve 6 adet doktora tezinin olduğu görülmüştür. Bu tezler haricinde YÖK merkezinden ulaşılamamış, internet ortamına aktarım aşamasında olan tezler de mevcuttur. Elde edilen verilere göre Yüksek lisans ve doktora tezlerinin yıllara göre dağılımı Şekil 1.2'de görülmektedir [8].



Şekil 1.2. 2000-2010 yılları arası gerçekleştirilen tezlerin yıllara göre dağılımı.

Gerçekleştirilen çalışma, TMS320F2812 işlemcisinin eğitimde ve sayısal işaret işleme çalışmalarında kullanımını kolaylaştırmak ve yaygınlaştırmak amacıyla gerçekleştirilen bir deney setidir. Eğitimde deney setinin kullanımı, etkili bir eğitim-öğretim ortamı hazırlayarak, algılama ve öğrenmeyi kolaylaştıran, ilgi uyandıran, sınıfa canlılık getiren bir uygulamadır. Ayrıca; öğrenme zamanını kısaltarak bilgiyi pekiştiren ve kalıcılığa yardım eden deney setleri, öğrencilerin konuya katılımını sağlayarak okuma ve araştırma arzusu uyandırdığı için, öğrencilerin öngörülen hedeflere daha kolay ulaşmalarını sağlamada ve yürütülen programın başarıya ulaşmasında önemli rol oynarlar [9]. Temel seviyeden ileri seviyedeki birçok uygulamanın gerçekleştirilebilmesi için tasarlanan ve gerçekleştirilen bu deney setinin daha önce yapılmış bir örneği de bulunmadığından 32 bit sayısal işaret işleme konusunda etkin bir materyal olacağı düşünülmektedir.

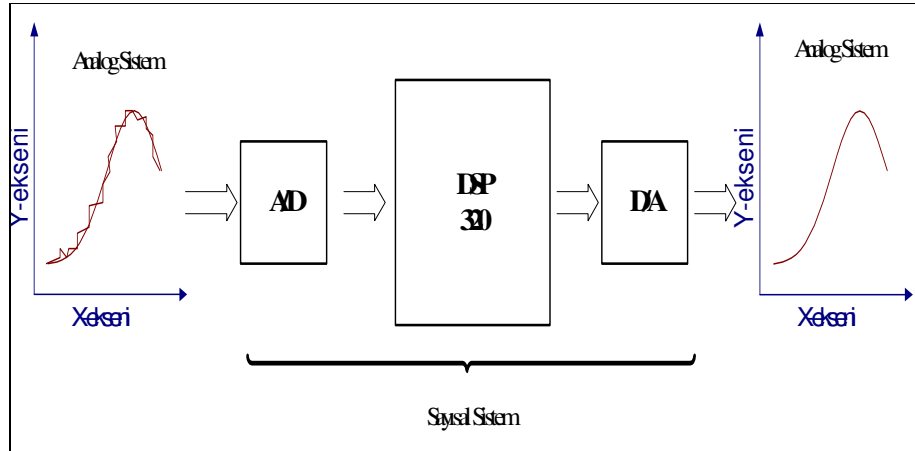
Tezin giriş bölümünde sayısal işaret işlemcilerin tarihçesi ve çalışmada kullanılan sayısal işaret işlemcisinin kullanım alanları açıklanmıştır. Bununla birlikte günümüzde sayısal işaret işleme eğitiminde kullanılan işaret işlemcilerine dair yapılan araştırmalar verilmektedir. İkinci bölümde; Sayısal işaret işlemcilerin genel özellikleri, TMS320F2812 işlemcisinin fonksiyonel blok diyagramı, merkezi işlem

birimi, dahili veri yolu yapısı ve komut iş hattı yapıları hakkında genel bilgiler verilmektedir. Üçüncü bölümde programlama ara yüzüne ait bilgiler açıklanmaktadır. CCS programlama ara yüzü, model tabanlı otomatik gömülü kod üretimi ve IQMath kütüphane desteğinden nasıl yararlandığı bu bölümde açıklanmıştır. Dördüncü bölüme, eZdsp F2812 başlama kitinin tanıtılmasıyla başlanmıştır. eZdsp F2812'ye ait bağlantı noktaları, kart üzerinde bulunan jumper (atlama) ayarları ve ledlerin işlevleri açıklanmaktadır. Ayrıca deney seti üzerinde bulunan devre kartları açık şemalarıyla birlikte yer almaktadır. Beşinci bölüm deney seti ile yapılan uygulamaları ve amaçlarını içermektedir. Altıncı bölümde ise gerçekleştirilen çalışmada elde edilen sonuçlar yer almaktadır.

BÖLÜM 2

SAYISAL İŞARET İŞLEME

Sayısal işaret işleme, işaretlerin sayısal bilgisayar veya özel amaçlı donanımda bir sayılar dizisi olarak gösterilmesi ve bu işaret dizisi üzerinde çeşitli işlemler yaparak, istenen bir bilgi yada büyüklüğün bu diziden çıkarılmasına dayanmaktadır. Bir sayısal işaret işleme sisteminde, dış dünya ile haberleşebilmek için A/D (Analog to Digital) ve D/A (Digital to Analog) çeviriciler bulunmalıdır (Şekil 2.1). Analog işaret işlemede işaret, sürekli zaman ve sürekli genlik seviyeleri ile işlenir. Bu sistemde ise, analog işaret artık zaman aralıklarında örneklenir. Örnek alınmış analog işaret genlikleri, bir A/D çevirici ile sayısal değerlere çevrilir ve işaretin sayısal eşdeğeri, sayısal ekseninde sayı dizileri halinde, işlemcinin kendisi tarafından işlenir. Bu işlemler; işaretlerin analiz edilmesi veya sentezlenmesi (HFD'ler, konuşma sentezi gibi), işaretlerin frekans veya genlik cevaplarının değişikliğe uğratılması (süzgeçler, denetleyiciler gibi) ve hatta sistemlerin veya işaretlerin kestirimi (tahmin edilmesi) sonuçları ortaya çıkacaktır. Burada veri Sİİ üzerinden işlem yaptıktan sonra, D/A çeviriciler yardımı ile analog biçime dönüştürülür.



Şekil 2.1. A/D çevirici.

İşaretlerin, sayısal işaret işleme teknikleri ile sayısal ekseninde işlenmesinin, analog ekseninde işlenmesine göre birçok üstünlükleri vardır. Yıpranma, sıcaklık ile parametre değerlerinin kayması gibi ortam koşullarına bağlı analog elemanların tersine, Sİİ elemanları ortam koşullarından daha az etkilenirler ve bu halleriyle daha kararlı ve daha güvenilir bir performans sağlarlar. Bunun yanı sıra; bir takım karmaşık işaret işleme teknikleri vardır. Bunları analog yöntemlerle işlemek ya çok zordur yada tümden olanaksızdır.

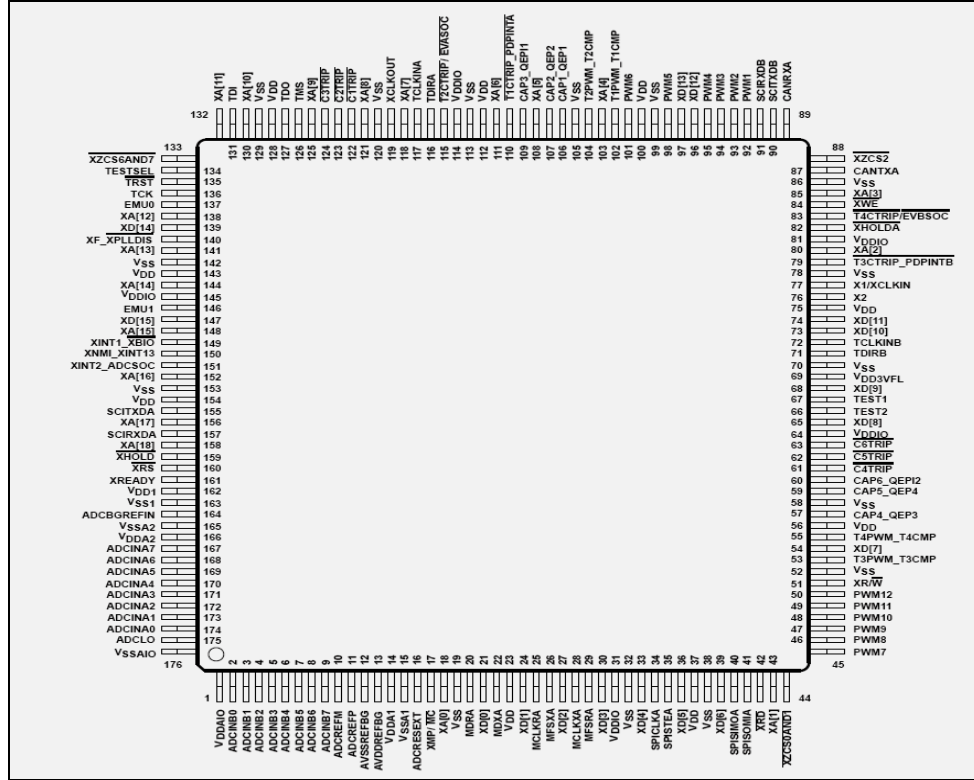
Sİİ'ler programlanabilir olduğundan, sistemler üzerinde değişikliklerle veya güncelleştirmelerle kolayca baş edilebilir. Bundan başka, tek bir Sİİ'yi, farklı sayıda işaret işleme için zaman içinde paylaşım yapmak da mümkündür.

Sİİ sistemleri sorunsuz değildir. Bu tür sistemlerde, analog işaretler işlenmeden önce sayısal biçime çevrildiğinden; zaman ve genliğin sürekli durumdan ayrık duruma çevrilmesi, çeşitli kısıtlamaları da beraberinde getirir. Eğer bu kısıtlamalara uyulmazsa, oldukça ciddi sorunlarla karşılaşılabilir. Burada göz önüne alınması gerekli en önemli nokta, örnekleme aralığının seçimidir [1,2]. Bu seçimi yapabilmek için unutulmaması gereken genel kural, analog işaret içerisindeki bilgiyi kaybetmemek için, Nyquist teoreminde bahsedildiği gibi, işaretin en yüksek frekanslı bileşenin en az iki katı frekans ile örnekleme yapılması gerekliliğidir [10]. Bu örneğin sayısal denetim sistemlerindeki örnekleme hızı, sistem bant genişliğinin 6 ile 10 katı aralığında bir değerde olmalıdır.

Gerçek zamanda işlemler yaparken, sayısal işaret işlemci bir sonraki örnek değerini almadan önce, tüm işlemlerini bitirip, hazır hale gelmelidir. Bu nedenle Sİİ son derece hızlı olmalıdır. Bu yüksek performansı sağlayabilmek için, işlemcilerin iç mimarileri donanımsal olarak birbirine bağlı birimlerden oluşur. Geleneksel mikroişlemci ve mikrodenetleyicilerde ise, iç mimari "mikrokod" denilen yöntemle oluşturulduğundan genel olarak yavaşlardır. İşte bu "donanımsal olarak birbirine bağlı birimlerden oluşan mimarisi" sayesinde Sİİ'ler komutların birçoğunu tek bir makine çevriminde yürütürler [1,2].

2.1. TMS320F2812 SAYISAL İŞARET İŞLEMCİSİ

Son yıllarda denetim sistemlerinde daha karmaşık yöntemler geliştirilmiş ve uygulama alanında da buna uyumlu ve daha hızlı mikroişlemciler geliştirilmiştir. Bunların son sürümlerinden biri de TMS320F2812 sayısal işaret işlemcisidir. TMS320F2812 Sİİ, sabit noktalı işlem yapabilme özelliğine sahip bir işlemcidir. Bu işlemci 32-bit 150MHz bir işlemci olup, toplamda 176 pine sahiptir. Bu pinlerden 16 adedi 6,67ns çözünürlüklü ölü zaman ayarlı ve programlanabilir darbe genişlik modülasyonu (DGM), diğer 16'sı, 12-bit 80ns dönüşüm zamanlı A/D çeviricidir. 4 adet sayısal yakalama girişi, 4 adetskare dalga kodlayıcı girişi bulunan işlemcinin 56 adet pini programlanabilir sayısal giriş-çıkış olarak kullanılabilir. 18 kword RAM (Random Access Memory) ve 128 kword flash EEPROM (Electrically Erasable Programmable Memory) hafızasıyla birlikte C/C++ programlama desteği de bu işlemcinin önemli özelliklerindedir [11]. İşlemcinin bacak bağlantıları ve üst görünümü Şekil 2. 2'de gösterildiği gibidir.



Şekil 2.2. TMS320F2812 işlemcisi bacak bağlantıları ve üst görünümü.

TMS320F2812 işlemci, sabit noktalı aritmetik işlem yapmasına rağmen IQmath kütüphane desteği ile kayan noktalı aritmetik işlem kolaylığında ve doğruluğunda işlem yapabilmektedir.

TMS320F2812 CPU (Central Processing Unit), kaydedicilere kayıt yapılmasını sağlayan komutların büyük çoğunluğunu ve mikrodenetleyiciler tarafından yaygın bir şekilde kullanılan komutların büyük çoğunluğunu çalıştırabilir. Aynı zamanda güçlü adresleme modları tarafından desteklenen bu mimari yapı, derleyicinin makine dili programlayıcı gibi neredeyse birebir C koduna benzeyen yoğun kodları üretmesine izin verir.

F2812, genellikle mikrodenetleyiciler tarafından gerçekleştirilen sistem denetiminde etkili olduğu gibi, matematiksel işlemlerde de oldukça etkilidir. Bu etki ikinci bir işlemcinin ihtiyacını ortadan kaldırır.

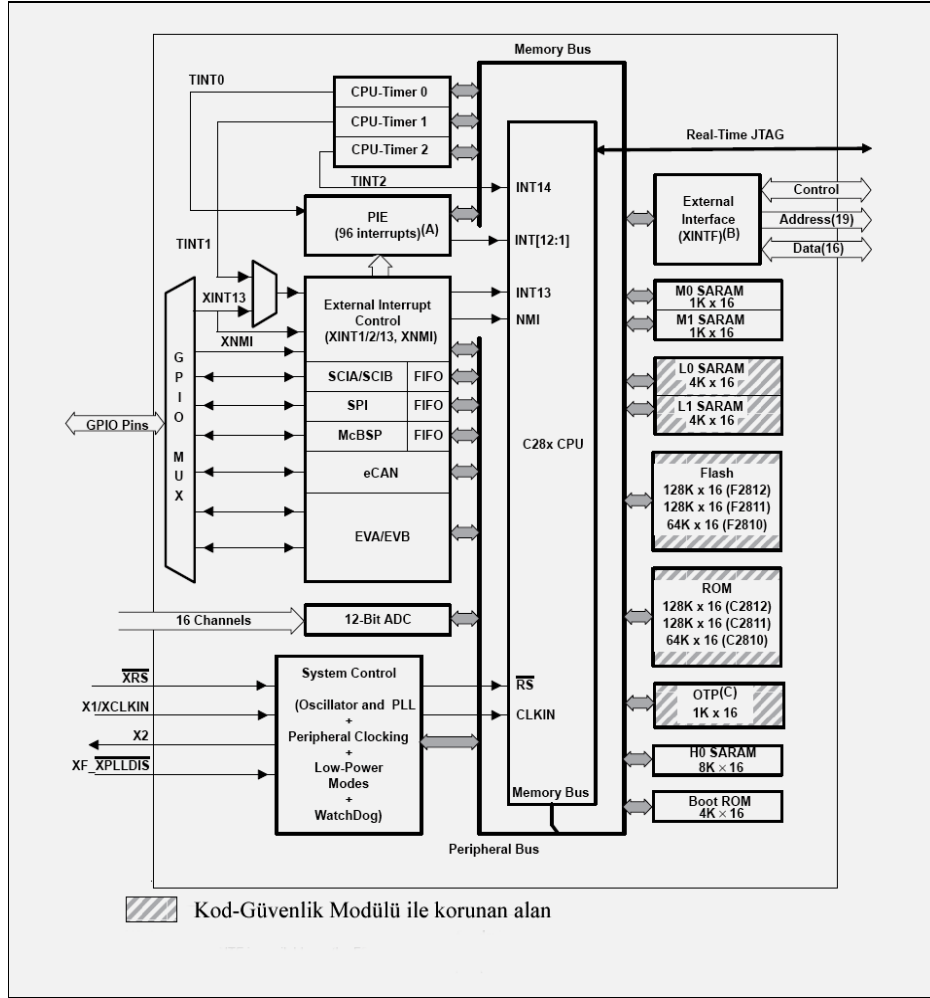
32 bitlik 3 adet zamanlayıcı genel zamanlayıcı amacıyla ya da gerçek zamanlı sistemler için üzerinde çalışılan donanıma yönelik zaman aralıkları üretmek amacıyla kullanılır. PIE (Peripheral Interrupt Expansion Manager) dahili ve harici kaynak sinyalleri için hızlı kesme cevabı sağlar. PIE tüm kaynaklar için kesme cevabı sağlar.

2.1.1. Fonksiyonel Blok Diyagram

TMS320F2812 işlemcinin yapısı 4 fonksiyonel bloktan oluşmaktadır: Fonksiyonel blok diyagram Şekil 2.3'te gösterildiği gibi, dahili ve harici veriyolu sistemi (internal and external bus), .merkezi işlem birimi (central processing unit), hafıza birimi (memory) ve çevresel birimlerden (peripherals) oluşmaktadır.

Merkezi işlem birimi ve hafıza arasındaki bilgi alışverişi için F2812'de iki bağımsız yol bulunmaktadır. Bunlar veri yolu ve program yoludur. Program yolu, program hafızasından komut kodu ve hızlı operantları taşır. Veri yolu ise merkezi Aritmetik Logical Unit (ALU) ve yardımcı kaydedici dosyası gibi değişik elemanlar ile veri hafızası arasındaki bilgi alışverişini sağlar. F2812 Sİİ de program kodları ve veriler farklı bloklar içerisinde yer alır. Böylece işlemcide işlenen bilgileri yalnızca veri

hafızasından değil, ayrıca program hafızasında da okuyabilme özelliğine sahiptir. Bu özelliğinden dolayı gelişmiş Harvard mimarisine sahip bir işlemci olduğu kabul edilir [12].



Şekil 2.3. TMS320F2812 fonksiyonel blok diyagramı.

2.1.2. F2812 Matematiksel İşlemler Birimi

Sayısal işaret işlemlerinin anlamı çarpanlar ve toplananlar ünitesidir. Bu üniteye kısaca Multiply and Accumulation (MAC) denir. MAC ünitesinde bu iki fonksiyon birlikte yapılır. Çünkü çarpmak için en az iki veri gereklidir ve aynı zamana ait iki verinin alınması sayısal işaret işlemlerinin verimini artırır. Böylece, iki veri yolu MAC ünitesi için hazır olacaktır.

F2812'nin 32x32 bitlik toplama ve çarpma (MAC) özelliği ve ayrıca 64 bit dahili işlem yapabilme kapasitesi bu işlemciye, daha maliyetli kayan noktalı (floating point) işlemcilere göre daha doğru ve hızlı bir şekilde çözüm sunma imkanı sağlar. Bununla birlikte 16x16 bitlik çarpma özelliği ve verilen komutlara anlık cevap verme özelliğine de sahiptir.

Çarpma işleminde, ilk sayıyı saklamak için XT kaydedicisi kullanır ve hafızadan yüklenen ikinci sayıyla çarpma işlemi yapılır. Eğer XT bir veri hafıza alanından yüklenmiş ve ikinci sayı bir program hafızasından alınamamış ise, tek çevrim bir çarpma oluşabilir. Çarpım sonucu P (Product) kaydedicisine yada doğrudan akümülatöre (ACC) yüklenir. Eğer 32x32 bitlik bir çarpma yapılırsa çarpım sonucu 64 bit olur.

2.1.3. Veri Hafıza Erişimi

Hafıza alanına erişiminde iki temel metot vardır:

- 1- Doğrudan adresleme modu.
- 2- Dolaylı adresleme modu.

Doğrudan adresleme modunda, bellek erişimi için 22 bitlik bir adres üretilir. Bu 22 bitlik adresin en yüksek 16 biti Data Page'den (DP), kalan 6 biti ise komuttan alınır. Bunun avantajı, DP çalıştığı sırada istenilen zamanda seçili sayfanın herhangi bir yerine müdahale edilebilmesidir. Dezavantajı ise, kodla başka bir sayfaya erişilmek istendiğinde, öncelikle DP'nin ayarlanması gerekliliğidir.

Dolaylı adresleme modunda, işlemcide 32 bitlik bir adres tutmak için toplamda 8 adet olan 32-bit XARn kaydedicisinden biri kullanılır. Bu işlemin avantajı, yardımcı kaydedici aritmetik birimi (Auxiliary Register Arithmetic Unit-ARAU) yardımıyla, hafızadaki bir veriye erişilirken aynı çevrim içerisinde pointer (işaretleyici) aritmetiğine de erişilebilmesi; dezavantajıysa veri hafızasına rastgele bir erişimin pointer kaydedicisinde yeni ayarlamalar gerektirmesidir.

AAU, aynı çevrimdeki pointer aritmetiğini veri hafıza alanına erişilmiş gibi paralel olarak çalıştırabilir. Bu paralelliğin sonucunda aritmetik ve lojik işlemlerin güçlü bir şekilde gerçekleştirilmesi ve bit deęiş tokuşu tek makine çevriminde gerçekleştirilir. AAU ayarları şunlardır:

- 1- Post- increment (arka artırma)
- 2- Pre- decrement (ön artırma)
- 3- Index addition (dizin ekleme)
- 4- Subtraction (çıkarma)
- 5- Stack relative operation (oranlı yığın işlemi)
- 6- Circular adres (daireesel adres)
- 7- Ek ayarlarla birlikte ters adresleme biti

2.1.4. Dahili Veriyolu Yapısı

Sİİ'lerin birçoğunda, hafıza alanı, çevresel birimler ve CPU arasındaki veri iletişimi için çoklu veriyolu kullanılmaktadır. F2812 veriyolu yapısı aşağıdaki birimlerden oluşmaktadır:

- 1- Program okuma yolu (22 bit adres ve 32 bit veriyolu).
- 2- Veri okuma yolu (32 bit adres ve 32 bit veriyolu).
- 3- Veri yazma yolu (32 bit adres ve 32 bit veriyolu).

Bu çoklu veriyolu yapısı işlemciye tek bir çevrim süresinde bilgi alma, veri deęeri okuma ve veri yazma imkanı vermektedir. Böylece tüm çevrebirimleri ve hafızalar, hafıza yoluna bağlanır ve önem sırasına göre işleme konur.

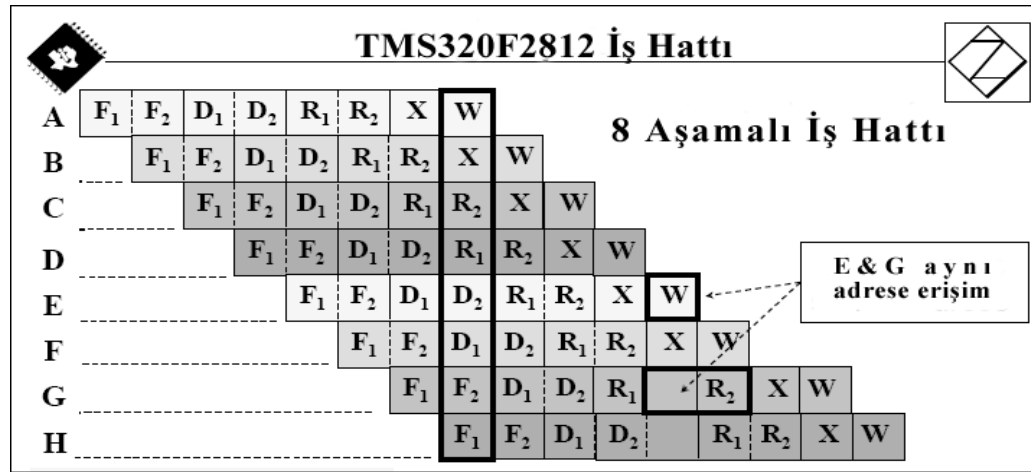
2.1.5. Atomik Aritmetik Lojik Birim

Atomikler küçük temel komutlardır ve engellenemezler. Atomik ALU kapasitesi tasklarla yönetilebilir ve işlemleri destekler. Buradaki komutlar normal programlamaya göre kod dizisini daha hızlı çalıştırabilmektedir.

2.1.6. Komut İş Hattı

TMS320F2812 işlemcilerde özel 8 basamaklı, korumalı iş hattı kullanılmaktadır. Korumalı iş hattı aynı yerden okuma ve yazma özelliğini desteklemektedir. Bu özelliği sayesinde harici bir hafıza kullanımına gerek kalmadan çok hızlı bir şekilde çalıştırılabilmektedir.

Komutlar sonlandırma işlemine kadar 8 işlemden geçer. Öncelikle iş hattı komutlarla doldurulur, bir işlem bir komut çevrim süresinde gerçekleştirilir. İş hattı basamakları Şekil 2.4’de gösterilmektedir.



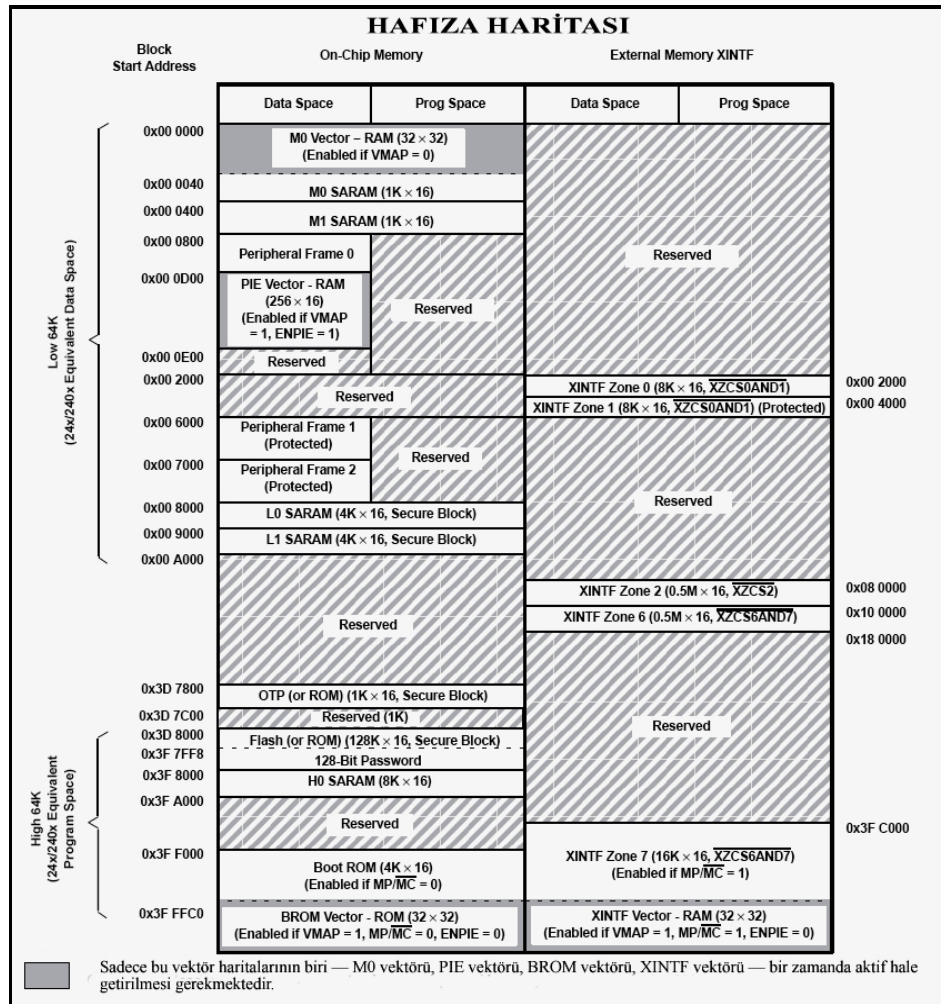
Şekil 2.4. TMS320F2812 iş hattı.

Komut iş hattı basamakları aşağıda gösterilmektedir.

1. F₁ :Program bus’da adres yolunda komut adresini üret.
2. F₂ :Komutu program bus veri yolundan oku.
3. D₁ :Kodu çöz.
4. D₂ :Operand komutlarının adres bilgisini hesapla.
5. R₁ :Operand adreslerini veri ya da program bus adres yoluna yükle.
6. R₂ :Operandı oku.
7. Y :Komutları çalıştır.
8. W :Sonucu veri hafızasına yaz.

2.1.7. Hafıza Haritası

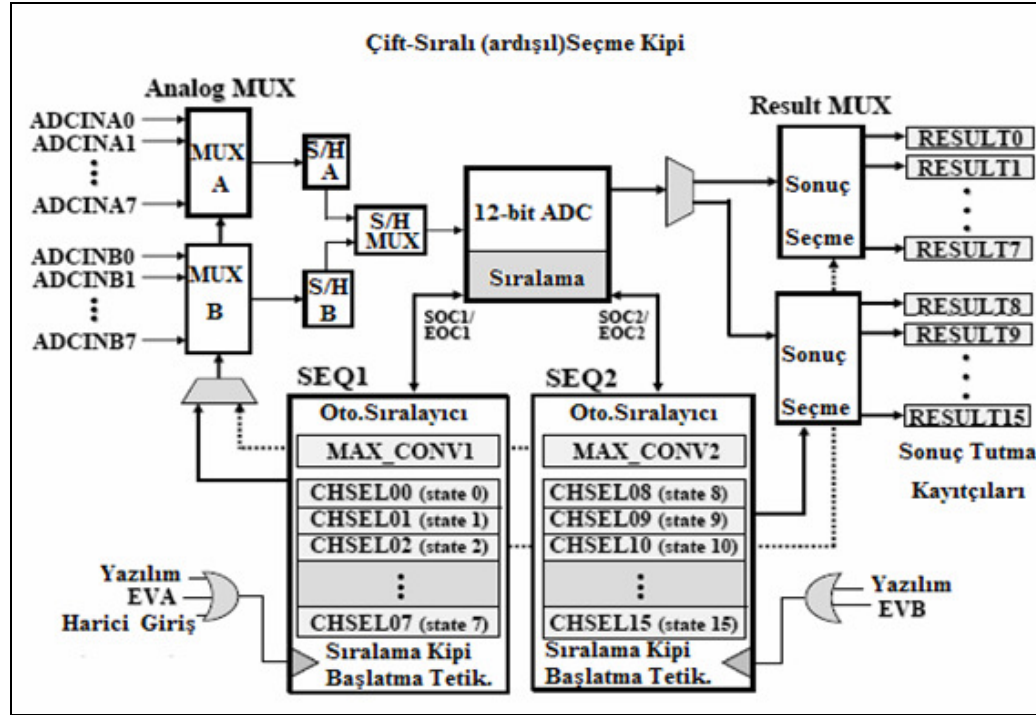
TMS320F2812 işlemcisindeki hafıza alanı program ve veri hafızası alanı olmak üzere ikiye bölünmüş durumdadır. Bu alan içerisinde de, Flash, SARAM (Single Access Random Access Memory), Expanded SARAM, Boot ROM (Read Only Memory) hafıza çeşitleri bulunmaktadır. Boot ROM fabrika tarafından programlanmış açılış yazılım yordamları ve matematiksel algoritmaları içerir. Bellek alan genişliği her zaman 16 bittir. Silinebilir hafıza 5 alana ayrılmıştır (M0, M1, L0, L1, H0). Bu hafıza alanları hem kod hem de veri hafızası olarak kullanılabilirler. PF0, PF1 ve PF2 status kaydedicileri tüm harici ünitelerin denetiminde kullanılır. Hafıza haritası ve hafıza alanları Şekil 2.5'te gösterilmiştir [12-15].



Şekil 2.5. Hafıza haritası.

2.1.8. Analog Sayısal Dönüştürücü ve Ayarları

F2812 işlemcisi, 12-bit çözünürlüklü 16 adet (2x8) 0-3V DA (Doğru Akım) analog girişli bir ADC'ye (Analog .Digital Converter) sahiptir. Programın işlemcide işlenmesi sırasında okunacak analog girişlerin sayısı ve analog işaretin hangi bacağa bağlandığı, dönüştürme sırası ve biçimi önceden ayarlanmalıdır. Örnekleme zamanı ve dönüştürülen işaretin sayısal değerinin ana program tarafından hangi sıklıkla alınacağı kesme vektörleriyle belirlenir. ADC'nin örnekleme hızı ise sistem saat hızına göre bağımsız olarak ayarlanabilmektedir (Şekil 2.6).



Şekil 2.6. ADC biriminin sıralı dönüşüm kipi.

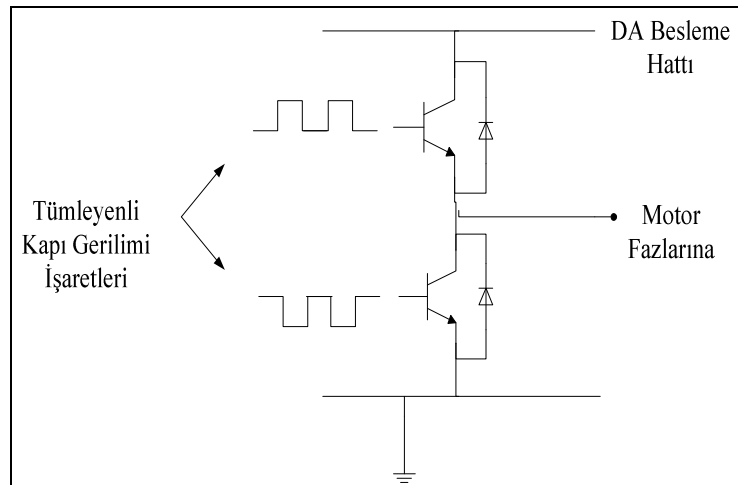
Bağlanmış analog kanal sayısı (ADC_MAXCONV), dönüşüm yapılırken hangi kanalın hangi sonuç kaydedicisinde tutulacağı (ADCSELSEQ_x ve ADCRESULT_x), hangi olay yöneticisinin dönüşümü yeniden başlatacağı (EVA_SOC_SEQ_x) ve kesme vektörünün yeniden yetkilendirilmesi (INT_ENA_SEQ_x) her TIMER1 periyoduna bağlı kesmeler için ayarlanmıştır. İlgili kaydediciler (ACTRL_x, ADC_MAXCONV_{xx}, ADCCHSELSEQ_x) AdcRegs içinde bulunur.

2.1.8.1. Analog İşaretlerin Örneklenmesi

F2812 Sİİ, 25MHz hızında çalışan ve 0-3V doğru gerilim değerini okuyup 12-bit sayısal dönüşüm yapan bir ADC birimine sahiptir. Bu sayısal değerler 16-bitlik kaydedicilerde tutulur. Akım, gerilim ve analog hız bilgileri okunurken kayıtçı verileri 4-bit sağa kaydırılarak kullanılır. Ancak alternatif genlikli işaretlerin bu ADC'den okunması için örneklenecek işaretler en fazla 1,5V genlikli olacak şekilde ayarlanmalı ve bir analog işaret bindirme devresinden geçirilmelidir. Böylece örneklenecek işaretin negatif genlikleri de ADC'ye verilebilir. Analog işaret bindirilerek örneklenmiş olan işaretlerde 1,5V'luk ekleme olduğundan ADC sonuç kaydedicilerinden okunan değerden 2048 değeri çıkartılarak normalize edilir. Bu durumda ADC için dönüştürülecek en büyük analog genlik 1,5V olacaktır.

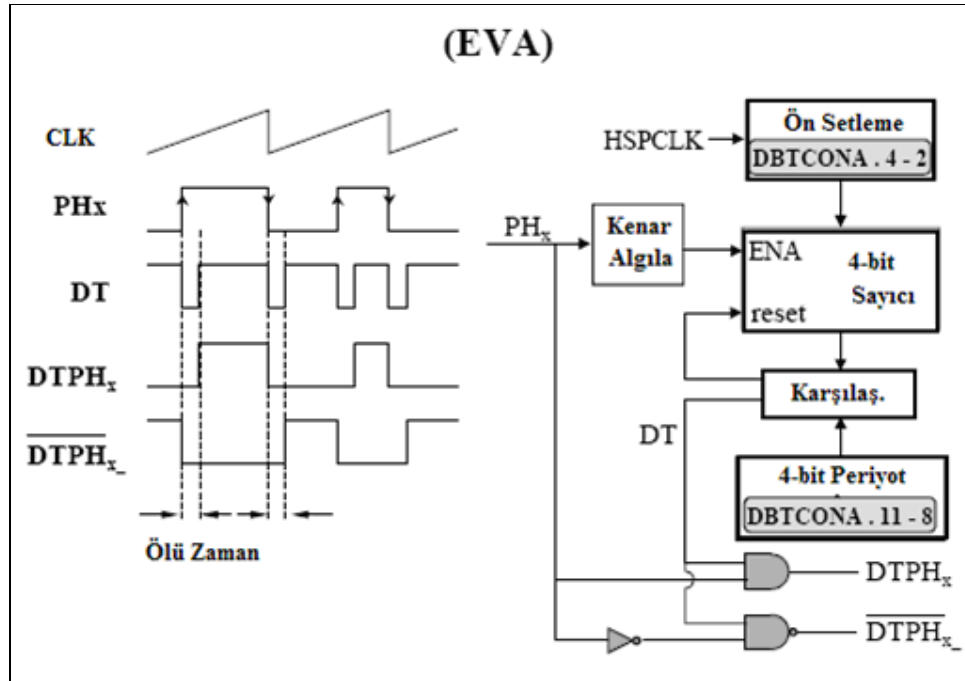
2.1.9. DGM İşaretleri Arasında Ölü Zaman Ayarı

Anahtarlamalı eviricilerde aynı hatta bağlı anahtarlar, biri diğerinden tümleyenli üretilmiş DGM işareti ile sürülürse sürücü işaretin düşen ve yükselen kenarlarında anahtar hızları yavaş kaldığı için, üstteki anahtar henüz tıkamaya gitmeden alttaki anahtar tetiklenmiş ve ilettime sokulmaya zorlanmıştır (Şekil 2.7). Bu durumda DA besleme hattı bu anahtar grubu tarafından kısa devre olmakla birlikte, anahtarlamalı elemanlar da kısa devre akımından etkilenerek yanmaktadır.



Şekil 2.7. Aynı besleme hattına bağlı anahtarların ölü zamanlı sürülmesi.

Bu durum basit elektronik devreler yada entegre devrelerle çözülebilsede haricen kullanılan devre elemanlarının hızı sayısal işaret işlemciye yetişemediği ve devre elemanlarının toleransları farklı olduğu için yine sorun çıkmaktadır. En doğru çözüm tümleyenli DGM işaretlerini, sayısal işaret işlemcisinde ölü zamanlı olarak üretmektir. F2812 işlemcisinde bunun için özel kayıtlar ve mantıksal birimler vardır (Şekil 2.8). İşlemcide öncelikle ölü zaman mantık birimi etkinleştirilmiştir. Daha sonra güç devresindeki anahtar elemanların iletme geçme ve tıkamaya gitme süreleri göz önüne alınarak uygun bit kombinasyonu EVA (Event Manager Modules) kaydedicilerinden DBTCONA (Dead-Band Timer Control Register A) ve COMCONA (Compare Control Register A) kayıtlarında ayarlanır.



Şekil 2.8. EVA olay yöneticisine bağlı ölü zaman birimleri blok diyagramı.

Şekil 2.8’de verilen EVA kayıtlarının 16-bitlik açılımları bilgi sayfalarından bakılarak ölü zaman kombinasyonu değiştirilebilir. İşlemci hızı 75MHz seçilirse yukarıdaki ayarlamalar 6µs’lik ölü zaman üretecektir. Bunun nedeni, DBTCONA zamanlayıcılarının işlemci saatini kullanmasıdır [16].

BÖLÜM 3

YAZILIM GELİŞTİRME ARAÇLARI

Texas Instruments firmasının ürettiği TMS320F2812 sayısal işaret işlemcisinin programlanması C++ ve ASM (Automatic Storage Management) programlama dilleri ile yapılabildiği gibi Matlab Simulink, VISSIM gibi özel paket programlar da kullanılabilir. Tez çalışmasında kullanılan eZdsp F2812 kartı satın alındığında, CCS platformu ile birlikte verilmektedir. Matlab Simulink ve VISSIM gibi programlar ile Sİİ için program ve gömülü sistemler tasarlanmakla birlikte, program kodları CCS aracılığıyla derlenmekte ve işlemciye yüklenmektedir. Bu işlemler için CCS, kullanılan program tarafından çağrılmakta ve program kodları üretmek için gerekli işlem basamakları CCS aracılığıyla yürütülmektedir.

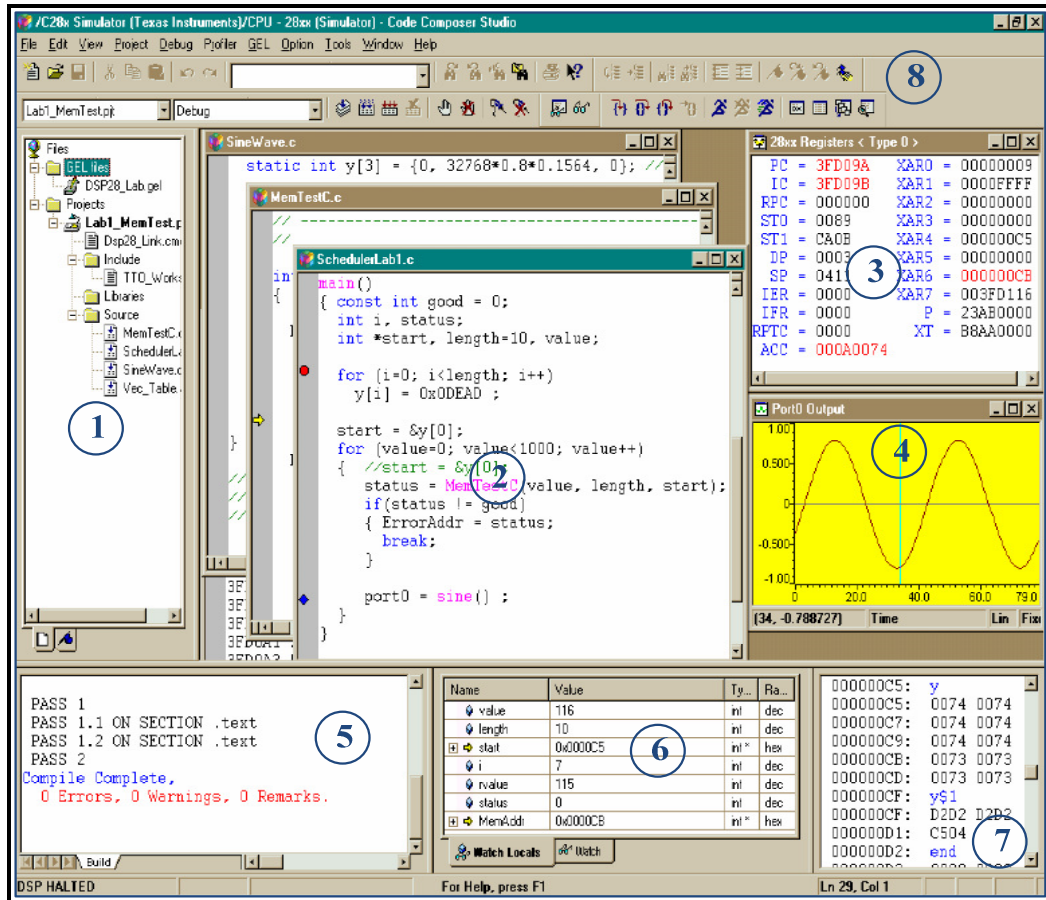
3.1. CODE COMPOSER STUDIO

Code Composer Studio, sayısal işaret işleme temelli uygulamaların tasarım sürecinde kullanılan, yazılım geliştirme ve derlenen yazılımları işlemciye yükleme aracıdır. Bu yazılım geliştirme aracı, TMS ailesi sayısal işaret işlemcilerinin kod geliştirme, derleme, hata ayıklama ve işlemci izleme özelliklerine sahiptir. Geliştirme aşamasındaki bir tasarımın uygulamalarla birlikte uyumluluğu ve kullanımı CCS platformunda test edilebilmektedir.

CCS ara yüzünde standart C/C++ proje oluşturma işlemleriyle, CCS örnek dosyalarından herhangi birinde ana program çatısı oluşturulabilir. CCS’de açılan her yeni proje için kullanılan işlemciye ilişkin “f2812.gel” dosyası eklenmelidir. F2812 işlemcisinin bütün birimlerinin adresleri ve adları C++ desteği ile nesne yönelimli olarak tanımlanmıştır. Bu nedenle standart olarak işlemci birimlerine ilişkin başlık dosyaları, kütüphane dosyaları ve kullanıcı tarafından değiştirilebilen kaynak dosyaları açılan projeye eklenmelidir. Bu dosyalar, kullanıcı tarafından oluşturulacak

yeni denetim yazılımları gibi kaynak dosyaları ile bağlanarak çıkış dosyası üretilebilmekte ve işlemciye yüklenebilmektedir [11].

İşlemcinin kullanılan birimlerine göre giriş/çıkış kapıları, ADC girişleri, DGM çıkışları, sayısal veri çıkışları, işlemci ve kullanılan birimlerin uygun çalışma hızları mevcut kaynak dosyalarında değiştirilebilmektedir. CCS'ye ait bir ekran görüntüsü Şekil 3.1'de gösterilmektedir.



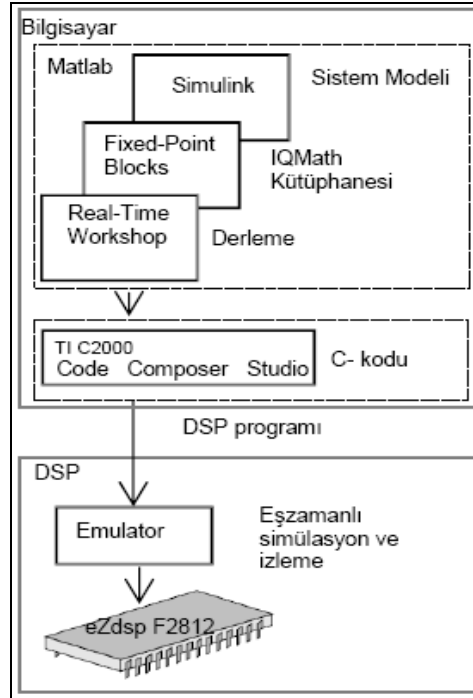
Şekil 3.1. CCS ana ekran görüntüsü.

Şekil 3.1'de 1 ile gösterilen alan "proje yöneticisi" alanıdır. Burada kaynak ve amaç dosyalarının yanı sıra derleyici linkleri yer almaktadır. 2 numarayla gösterilen alan "yönetim" alanıdır. Kaynak kodlar burada yazılıdır. 3 numaralı alan "CPU gözlem" alanı iken 4 numaralı alan "grafik gözlem" alanıdır. 5 numaralı alan "durum penceresi" alanıdır. Programının çalışma aşaması, kodlarda oluşan hata, uyarı ve

yorum mesajları burada yer alır. 6 numaralı alan programın aşamalı olarak takip edilmesini sağlayan “gözlem penceresi” alanı iken 7 numaralı alan “bellek izleme” alanıdır. Programa ait menü ve ikonlar ise 8 numaralı alanda yer almaktadır.

3.2. MATLAB/SIMULINK

Matlab/Simulink, denetim algoritmalarının blok diyagramlarla gösterilebilen benzetim ve gömülü sistemler için model tabanlı bir geliştirme ortamıdır. Blok diyagramlarla geliştirilen Sİİ tabanlı denetim sistemleri ve uygulamaları, Real Time Workshop (RTW), Link for Code Composer Studio Development Tools ve Embedded Target for TIC2000 Sİİ alt yazılımları ile hedef Sİİ’leri için makine diline çevrilmekte ve bu şekilde gerçek zamanlı uygulanması ve tasarım doğrulaması çok hızlı bir şekilde yapılabilmektedir. Şekil 3.2’de otomatik gömülü kod üretimi blok diyagramı verilmiştir.



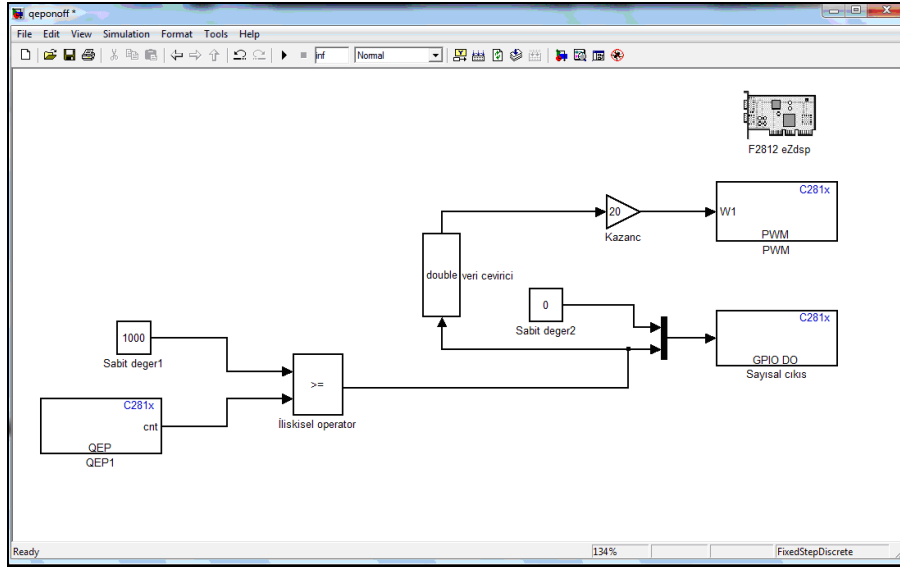
Şekil 3.2. Matlab/Simulink üzerinden otomatik gömülü kod üretimi.

RTW, Simulink ortamında oluşturulan grafiksel blok diyagramından C kodlarını otomatik olarak üretmektedir. Bu kodlar CCS yazılım geliştirme ortamına aktarılarak

diğer destek dosyalarıyla birlikte derlenmekte ve bilgisayarın paralel portuna bađlı F2812 Sİİ'nin üzerindeki emülatör aracılığı ile yüklenmektedir [17].

Tüm tasarım işlemleri tam otomatik, esnek ve oldukça kolay bir şekilde yapılabilmektedir. Ayrıca, herhangi bir programlama dili içine girmeden Sİİ blođu ve tasarım parametreleri gerçek zamanda deđiştirilebilmektedir. Grafikselsel blok diyagramların kullanımı, Sİİ üzerinden denetlenen sistem tasarımını, prototiplendirilmesini ve modifikasyonunu oldukça kolaylaştırmaktadır.

Bir Matlab ara yüzü olan Simulink'te bir modelleme işlemleri için, simge sürükleyip bırak mantığı ile taşınan bloklar kullanılmaktadır. Matlab kodu yazmak yerine, işlemleri blokları birbirine bađlanarak model diyagramları oluşturulmaktadır. Blok simgeleri, sistemin girişlerini, sistemin parçalarını veya sistemin çıkışlarını göstermektedir. Şekil 3.3'te matlab simulinkte gerçekleştirilen bir model görülmektedir.



Şekil 3.3. Örnek Simulink modeli.

Oluşturulan model derlenerek, Simulink'in otomatik kod oluşturma özelliđi ile kodlar oluşturulmakta ve oluşturulan kodlar Sİİ içerisine aktarılarak işlemci çalıştırılmaktadır.

3.2.1. IQmath Kütüphane Desteği

F2812 Sİİ sabit noktalı işlem aritmetiğine göre çalışır. Bu nedenle programda kullanılan bütün değerler işlemcinin tanıdığı en büyük tamsayıya göre ölçeklendirilir. Ondalıklı sayılar yerine tamsayılarla çalışılır. Bu durumda işlem başarı hızı azalmaz. Çarpma, toplama, çıkarma ve bölme gibi işlemler birlikte yapıldığında anlamlı bitlerin kaybedilmemesi için bit kaydırma yöntemleri kullanılsa da karekök, üstel ve trigonometrik işlemlerde genellikle okuma tabloları kullanılır. Bu değer tablolarının işlemciye yüklenmesi gerekir. Bu işlem programlamada zorluklara neden olur. Aynı zamanda elde edilecek analog değerlerde yuvarlamalara neden olur. Ancak bu işlemcilerin programlanmasında assembly yerine C/C++ kullanıldığında derleyici doğrudan yazılmış en karışık işlem basamaklarını, her türlü ondalıklı işlem için, işlemciye uygun olarak dönüştürse de bu oldukça zaman almaktadır. İşlem başarı hızını azaltmadan ondalıklı sayılarla da çalışabilmek için F2812 Sİİ üreticisi tarafından geliştirilen “IQmathLib.h” başlık dosyası, işlemci programlanırken sistem başlık dosyaları içine eklenir ve bağlı kütüphane kurallarına göre değişkenler tanımlanır. Bu durumda standart kayan noktalı aritmetik işlemler doğrudan C/C++ dilinde yazılabilir. C++ için “IQmathCPP.h” başlık dosyasıdır .

IQmath yaklaşımında “I” integer (tam), “Q” quotient (kesir) anlamında kullanılır. Buna göre farklı formatlar kullanılmakla birlikte geleneksel 32-bit IQmath yaklaşımında I8Q24 formatı kullanılırken Q19-Q26 formatı da işlem kararlılığına sahiptir. Eşitlik 3.1’de sekizlik tabanda verilen bir sayının I8Q24 formatındaki karşılığını gösterilmektedir [18,19].

$$-2^l + 2^{l-1} + \dots 2^1 + 2^0 \bullet 2^{-1} + 2^{-2} + \dots 2^{-q} \quad (3.1.)$$

Örneğin, 0x41200000 sayısının I8Q24 formatındaki karşılığı:

$$= 0100\ 0001.0010\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000b$$

$$= 2^6 + 2^3 + 2^{-3} = 65.125$$

olarak bulunur. Bu yöntemin avantajı tüm sayı aralıklarının aynı hassasiyette olması ve hızlı işlem yapılabilmesi iken, kayan noktalı aritmetik yapının tersine sınırlı dinamik aralığa sahip olması ise dezavantajdır.

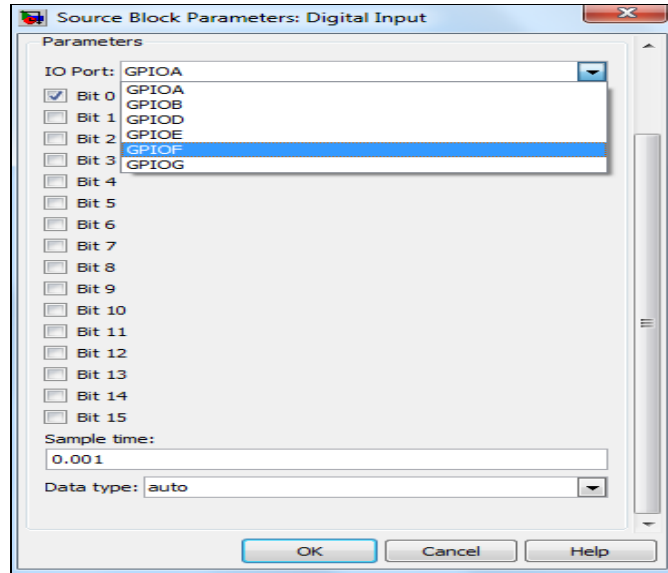
3.2.2. Matlab Simulink Target Support Package TC2 Blokları

Matlab Simulink kütüphanesinde “target support package TC2” menüsünün “C2000 target preferences” başlığında Texas firması tarafından üretilen, Custom board, F2808 eZdsp, F2812 eZdsp ve F28335 eZdsp geliştirme kartları bulunmaktadır. Kayan noktalı yapı özelliği sayesinde matlab programı tarafından desteklenen 28X serisi işlemciler günümüzde, özellikle denetim uygulamalarında oldukça sık kullanılmaktadır.

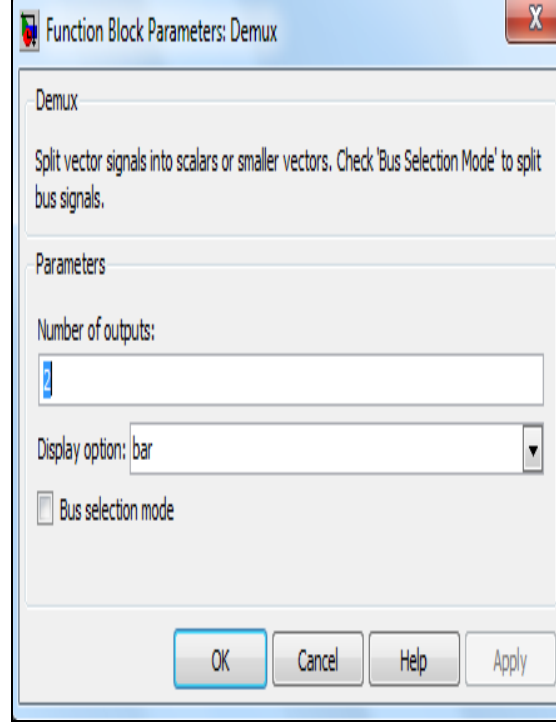
Bu çalışmada matlab ortamında, ses ve görüntü işlemede kullanılan C6000 serisinden sonra en çok kullanılan işlemci olan F2812 işlemcisi kullanılmıştır ve C281x yonga desteği altında verilen bloklardan ve bu bloklara ait parametre ayarlarından kısaca bahsedilmiştir.

3.2.2.1. Sayısal Giriş Bloğu

TMS320F2812 işlemcide sayısal giriş, olarak kullanılabilen A, B, D, E, F ve G olmak üzere 6 adet giriş portu ve toplamda 56 giriş pini bulunmaktadır. Sayısal girişe ait blok parametre özellikleri Şekil 3.4’te verilmiştir.



Şekil 3.4. Sayısal giriş blok parametreleri.

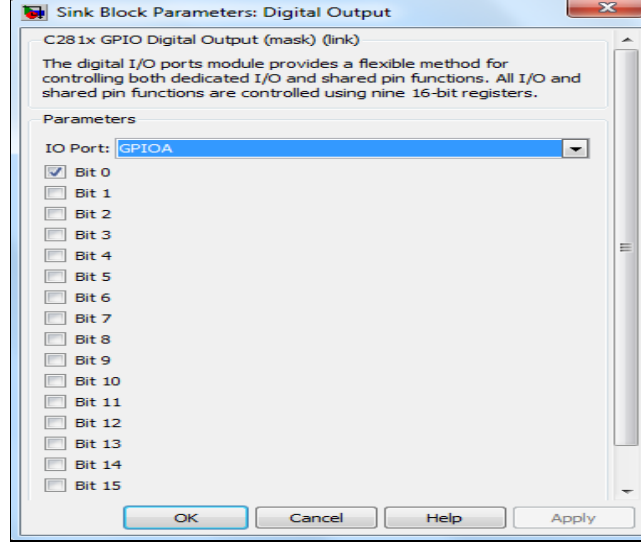


Şekil 3.5. Veri çözücü blok parametreleri.

Sayısal giriş bloğu ile ilgili yapılması gereken parametre ayarları, giriş olarak seçilen Sİİ pinlerinin, örnekleme zamanının ve veri tipinin ayarlanmasıdır. Veri tipi ayarlanmadığı durumda boolean'dır. Bu veri tipi diğer bloklarla uymadığı durumlarda "data type conversion" ile uygunlaştırılmalıdır. Bir sayısal giriş bloğu alınırken dikkate edilmesi gereken bir nokta giriş olarak seçilen pin sayısının çıkışta da aynı sayıda kalması gerekliliğidir. Sayısal giriş olarak seçilen pinler birden fazla ise bu pinler sayısal giriş bloğundan veri çözücü (demultiplexer) ile alınarak bir sonraki bloğa iletilir. Veri çözücü bloğuna ait en önemli parametre çıkış sayısının ayarlanmasıdır (Şekil 3.5).

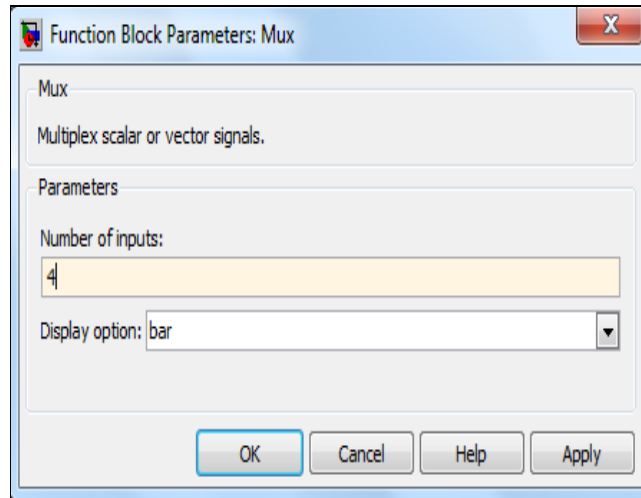
3.2.2.2. Sayısal Çıkış Bloğu

TMS320F2812 işlemcinin sayısal çıkış bloğu, sayısal girişde olduğu gibi A, B, D, E, F ve G portlarından oluşan 56 çıkış pinine sahiptir. Sayısal çıkış blok parametrelerinde, yalnızca kullanılan port ve pinlerin ayarlanması istenir. Şekil 3.6'da bu bloğa ait parametreler görülmektedir.



Şekil 3.6. Sayısal çıkış blok parametreleri.

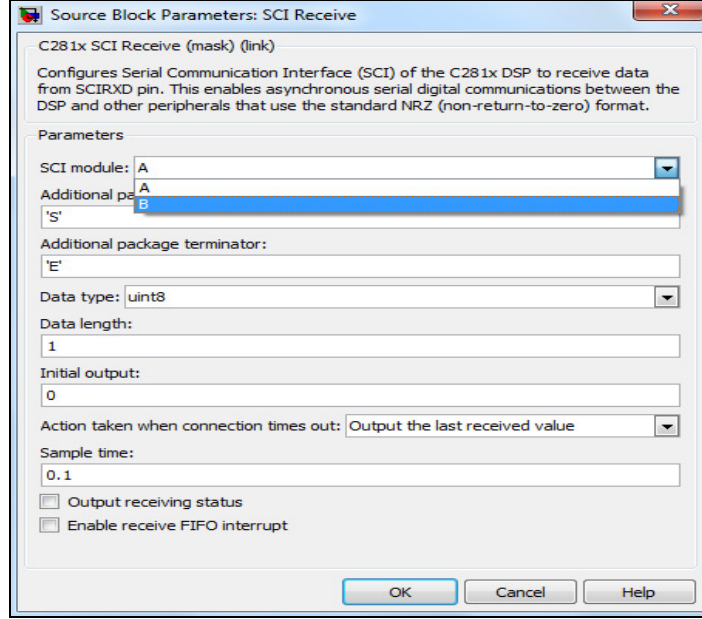
Burada dikkat edilmesi gereken nokta sayısal çıkış olarak kullanılacak olan pinlerin sayısal giriş olarak kullanılacak pinlerle aynı olmamasıdır. Örneğin A portunun belirli pinleri giriş olarak seçilmişse, aynı zamanda çıkış olarak da seçilemez. Ayrıca herhangi bir portu kullanıldığında o port için ikinci bir blok kullanımı desteklenmemektedir. Sayısal çıkış olarak kullanılacak çıkış sayısı belirlendikten sonra bu çıkışlar “veri çoğullayıcı” (multiplexer) ile sayısal çıkış bloğuna gelir. Çıkış sayısı çoğullayıcı bloğundaki parametreler Şekil 3.7’de görüldüğü gibi ayarlanır.



Şekil 3.7. Veri çoğullayıcı blok parametreleri.

3.2.2.3. Seri İletişim Arabirim Bloğu

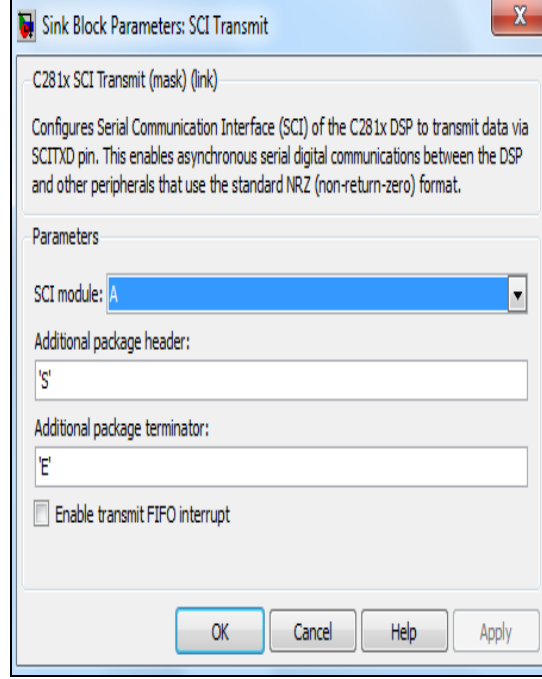
Seri iletişimde bir cihazdan bilgisayar ortamına veri almak için “SCI Receive” bloğu, bilgisayardan bir cihaza veri göndermek için de “SCI Transmit” bloğu kullanılmaktadır. SCI alıcı bloğuna ait parametreler Şekil 3.8’de görülmektedir.



Şekil 3.8. SCI alıcı bloğuna ait parametreler.

Blok içerisindeki ilk parametre SCI modülü seçimidir. Burada A ve B modülü olmak üzere iki modül mevcuttur. Additional package header (ek paket başlığı) olarak adlandırılan parametrede bilgisayara kabul edilecek bilgiye verilmek istenen ön ek belirlenirken, bu bilgiye verilecek son ek ise additional package terminator (ek paket bağlantısı) kısmından belirlenir.

Şekil 3.8’de görülen B modülünden, alınacak bilginin “S” ile başlarken “E” harfi ile sonlanması gerekir. Ön ek ve son ek kullanımı zorunlu değildir. Yalnız; uzaktan alınmak istenen bilginin, aynı frekanstaki başka bilgilerle karışma ihtimali varsa, istenen bilgi ön ek ve son eklerle etiketlenmelidir. “Data length” alınacak data uzunluğuyla ilgilidir. Burada her 100ms’de bir, bu 1 byte’lık veri alınmaktadır. 100ms örnekleme zamanını ifade eder.

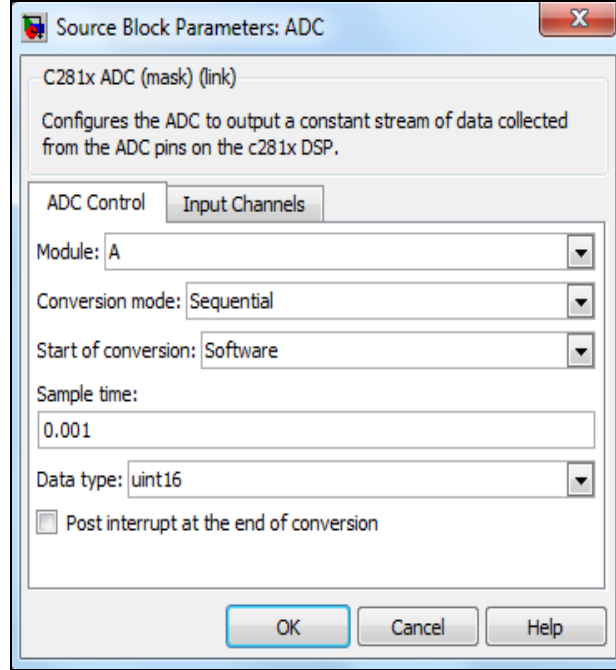


Şekil 3.9. SCI verici bloğuna ait parametreler.

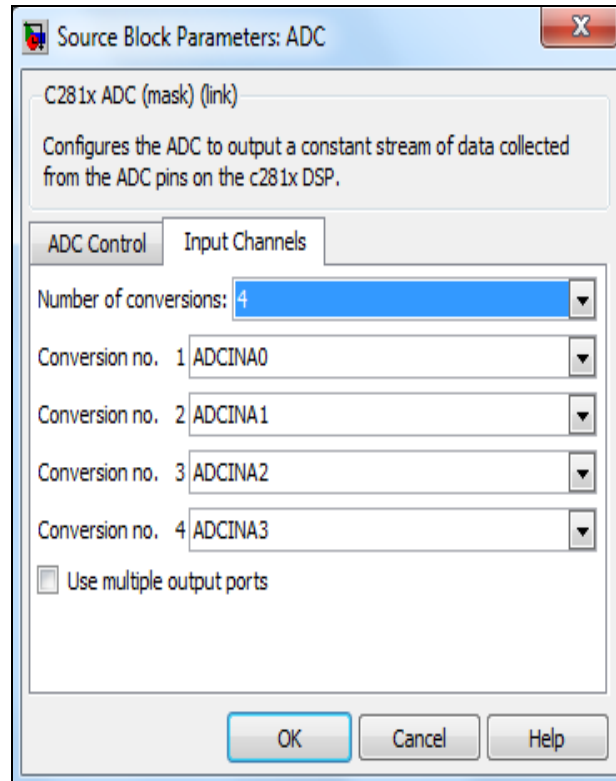
“SCI Transmit” bloğu bilgisayardan paketler halinde veri gönderimini sağlar. Şekil 3.9’da görüldüğü gibi bu blokta da verinin SII’nin hangi portundan gönderileceği (A veya B) seçilir. Ön ek ve son ek uygulaması burada da vardır ve gönderilecek verinin aynı frekanstaki başka verilerle karışması engellenebilir. “FIFO interrupt” seçili hale getirildiğinde, gönderilecek verilerin tıkanmaması ve belli bir sıraya göre gönderilmesi gerçekleştirilir. Böylece asenkron olarak seri sayısal iletişim gerçekleştirilmiş olur.

3.2.2.4. Analog Sayısal Dönüştürücü Bloğu

TMSF320F2812 işlemcide A ve B kanallarında toplam 16 kanal ADC bulunmaktadır. Burada, 0-4096 arası analog değerler alınarak 0-3V’luk sayısal değerlere çevrilmektedir. Parametre ayarlarında iki menü bulunur. ADC denetim menüsünde Şekil 3.10’da görüldüğü gibi SII’nin A ve B modüllerinden hangisinin kullanılacağı, dönüştürme modu (seri bir şekilde yada anlık olarak), dönüştürme başlangıç seçeneği, örnekleme zamanı ve işlem yapılacak veri tipi ayarları yapılır.



Şekil 3.10. ADC blok parametreleri, ADC denetim menüsü.



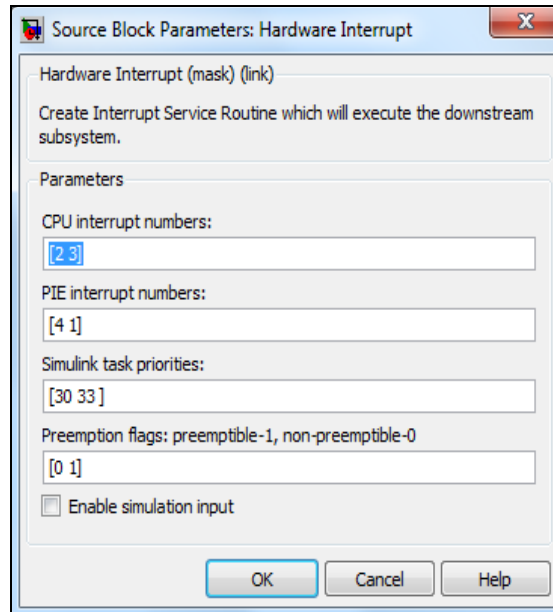
Şekil 3.11. ADC blok parametreleri, giriş kanalları menüsü.

ADC blok parametrelerinde ikinci menü Şekil 3.11’de görülen giriş kanal ayarlarıdır. Burada Sİİ’ye gönderilecek analog veri sayısı ve bunların hangi kanallarla gönderileceği belirlenir. Ayarlar yapıldıktan sonra ADC bir veri çözücü ile bir sonraki bloğa bağlanır.

3.2.2.5. Donanım Kesmesi Bloğu

Bilgisayarın merkezi işlem ünitesi olan CPU, çevre birimleri ile olan tüm iletişimleri başlatır ve onların yönetimini elinde tutar. Bu CPU’yu herhangi bir çevresel birimle iletişime sokmak ve yaptığı işlem kesilmek istendiğinde, donanım kesmesi kullanılabilir. TMS320F2812 işlemcisine ait donanım kesmesi blok parametreleri Şekil 3.12’de görüldüğü gibidir. Burada kullanıcıdan CPU ve PIE kesme numaraları ayarları ve simulinkte işletilme sıralarına ilişkin ayarlamalarının yapılması istenir.

Çizelge 3.1’de görülen, satır numarası CPU kesme değerini verirken; sütun sayısı da paralel kesmenin aktiveleştirme değerini verir. Çizelgeye göre belirlenen kesme değerleri blok parametresine girilir. Birden fazla kesme kullanılmak istendiğinde öncelik sıraları belirtilmelidir. Öncelik sırası 40’a kadardır.



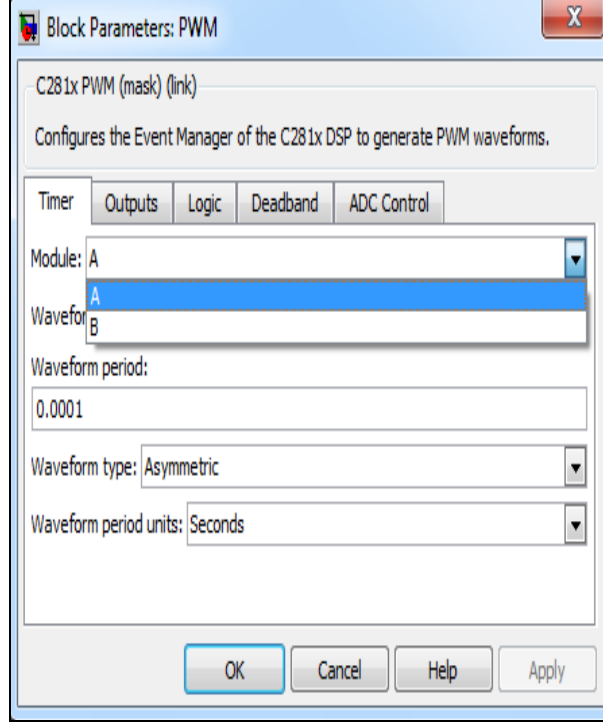
Şekil 3.12. Donanım kesmesi blok parametreleri.

Çizelge 3.1. Donanım kesmesi (hardware interrupt).

		Row numbers = CPU values / Column numbers = PIE values							
		8	7	6	5	4	3	2	1
1	WAKEINT (LPM/WD)	TINT0 (TIMER 0)	ADCINT (ADC)	XINT2	XINT1	Reserved	PDPINTB (EV-B)	PDPINTA (EV-A)	
2	Reserved	T1OFINT (EV-A)	T1UFINT (EV-A)	T1CINT (EV-A)	T1PINT (EV-A)	CMP3INT (EV-A)	CMP2INT (EV-A)	CMP1INT (EV-A)	
3	Reserved	CAPINT3 (EV-A)	CAPINT2 (EV-A)	CAPINT1 (EV-A)	T2OFINT (EV-A)	T2UFINT (EV-A)	T2CINT (EV-A)	T2PINT (EV-A)	
4	Reserved	T3OFINT (EV-B)	T3UFINT (EV-B)	T3CINT (EV-B)	T3PINT (EV-B)	CMP6INT (EV-B)	CMP5INT (EV-B)	CMP4INT (EV-B)	
5	Reserved	CAPINT6 (EV-B)	CAPINT5 (EV-B)	CAPINT4 (EV-B)	T4OFINT (EV-B)	T4UFINT (EV-B)	T4CINT (EV-B)	T4PINT (EV-B)	
6	Reserved	Reserved	MXINT (McBSP)	MRINT (McBSP)	Reserved	Reserved	SPITXINTA (SPI)	SPIRXINTA (SPI)	
7	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	
8	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	
9	Reserved	Reserved	ECAN1INT (CAN)	ECAN0INT (CAN)	SCITXINTB (SCI-B)	SCIRXINTB (SCI-B)	SCITXINTA (SCI-A)	SCIRXINTA (SCI-A)	
10	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	
11	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	
12	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	

3.2.2.6. Darbe Genişlik Modülasyonu Bloğu

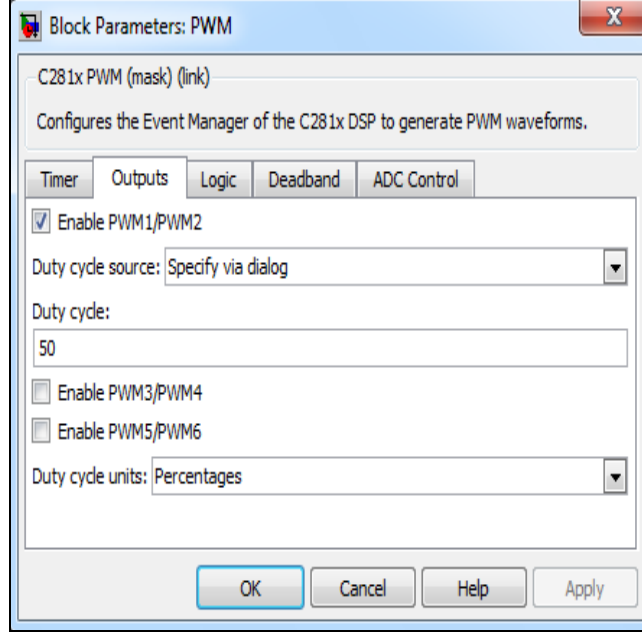
TMS320F2812 işlemcinin, darbe genişlik modülasyonu bloğu, darbe genişlik modülasyonlu işaretleriyle denetim işlemi gerçekleştirilebilen donanımlar için geliştirilmiş bir bloktur. Darbe genişlik modülasyonu bloğunun zamanlayıcı panelindeki modül belirleme seçeneğinde, A ve B olmak üzere iki modül bulunmaktadır. Bu modüllerden hangisi aktif edilmek istenirse o modül seçilerek, o modüle buluna DGM pinlerinin kullanımı tercih edilmiş olur. Uygulanacak olan işaretin periyodunun değeri iki farklı kaynak aracılığıyla belirlenebilir. Dalga şeklinin periyot değerinin belirlenmesinde (waveform period), kaynak olarak diyolog aracılığıyla belirle seçeneği seçilirse, panelde bulunan dalga şekli periyodu kısmına yazılan değer; eğer giriş portu seçilirse bloğun giriş portundan girilen değer dalga şeklinin periyodu olarak belirlenir. Dalga tipi (waveform type) seçeneğinden dalga tipi asimetrik veya simetrik olarak seçilebilir. Asimetrik dalga tipi seçildiğinde, işaret testere dişi dalga ile, simetrikte dalga tipi seçildiğinde ise üçgen dalda ile karşılaştırılarak üretilir. Şekil 3.13'te DGM bloğuna ait zamanlayıcı menüsünü görünümü verilmektedir.



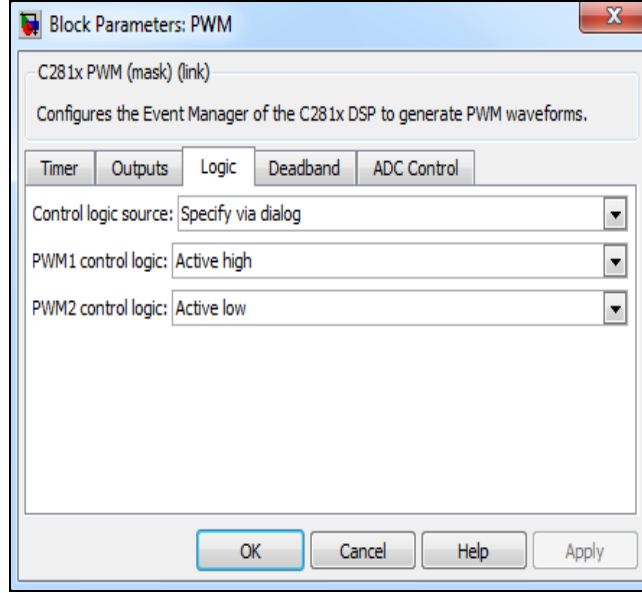
Şekil 3.13. DGM bloğunun zamanlayıcı menüsün parametreleri.

Darbe genişlik modülasyonu bloğunun çıkışlar menüsünde, kullanımı bir önceki menüde modül seçeneğiyle aktif edilmiş olan DGM çıkış çiftlerinin geçerli olduğunu gösterir. Panelde bulunan görev süresi değerinin (duty cycle source) belirlenmesinde kaynak olarak eğer diyalog aracılığı ile belirle seçeneği seçilirse, panelde bulunan görev süresi kısmına yazılan değer, eğer giriş portu seçilirse bloğun giriş portundan girilen değer işaretin görev süresi olarak belirlenir. Görev süresi (duty cycle) kısmına girilen değer, DGM işaretinin bir periyodu içerisinde, darbenin devam etme süresinin periyoda oranı verir. Şekil 3.14'te DGM bloğuna ait çıkışlar menüsünün görünümü verilmektedir.

Darbe genişlik modülasyonu bloğunun lojik panelinde, çıkışlar panelinde yapılan tercihler doğrultusunda geçerli olan tüm DGM'lerin denetimi yapılır. Denetim lojik kaynağı (control logic source) kısmında eğer diyalog aracılığıyla lojik denetim kaynağını belirle seçeneği seçilirse, panelde bulunan DGM lojik denetimler ile işaretin yüksek seviyede veya alçak seviyede aktif olma durumları belirlenebilir. Şekil 3.15'te DGM bloğuna ait lojik menüsünün görünümü verilmektedir.



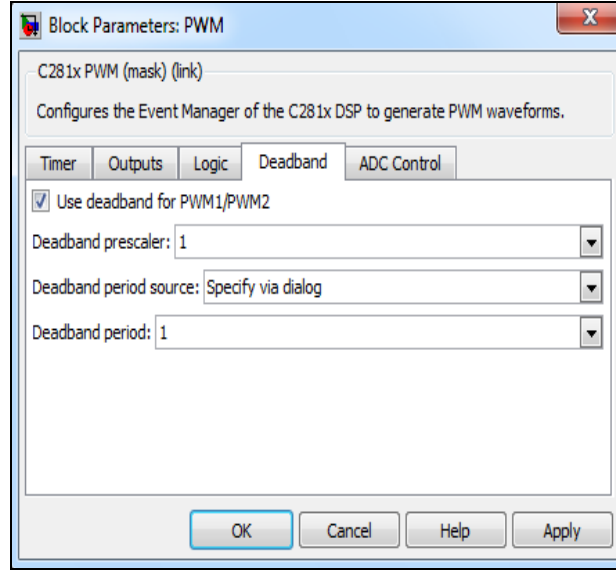
Şekil 3.14. DGM bloğunun çıkışlar menüsü parametreleri.



Şekil 3.15. DGM bloğunun lojik menüsü parametreleri.

Darbe genişlik modülasyonu bloğunun ölü zaman (deadband) menüsünde, DGM (PWM-Pulse Width Modulation) işaret çıkışı çiftleri için ölü zaman alanının kullanılma durumu belirlenir. Eğer ölü zaman kullanımı geçerliyse, ölü zaman ön bölücü değeri için seçenekte sunulan değerlerden biri seçilir. Bu değer ölü zaman

periyodu kısmına girilen periyot değeri ile çarpılır ve ölü zaman boyutu belirlenir. Şekil 3.16’da PWM bloğuna ait ölü zaman (deadband) menüsünün görünümü verilmektedir.

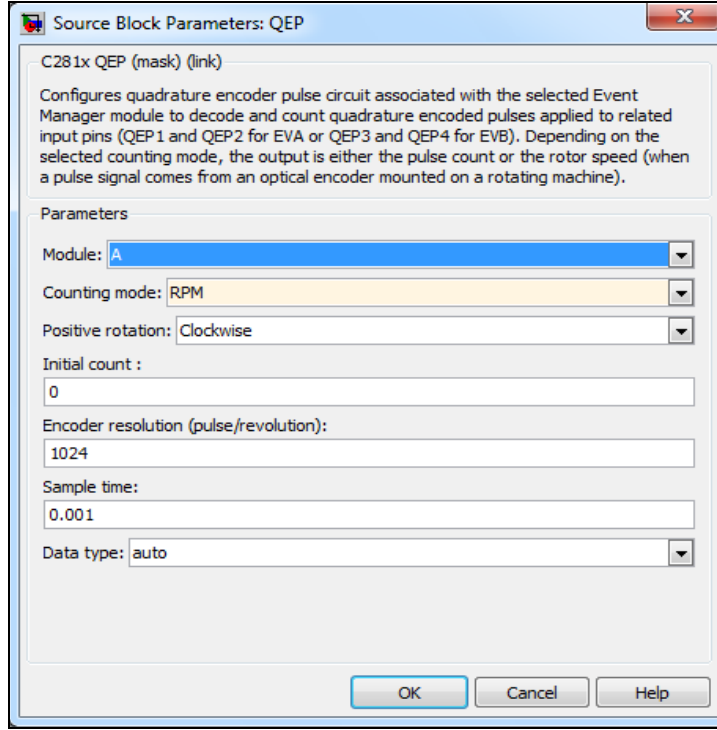


Şekil 3.16. DGM bloğunun ölü zaman menüsü parametreleri

3.2.2.7. Kuadratik Kodlayıcı Darbe Bloğu

TMS320F2812 işlemcinin, kuadratik kodlayıcı darbe bloğu, motor hız denetimleri için özel olarak geliştirilmiş bir bloktur. Parametreler panelindeki modül belirleme seçeneği ile kullanılacak olan QEP (Quadratic Encoder Pulse) pinleri belirlenmektedir. Eğer A modülü seçilirse, QEP1 ve QEP2 pinlerinin kullanımı, eğer B modülü seçilirse, QEP3 ve QEP4 pinlerinin kullanımı aktif hale getirilir. Sayma modunu (counting mode) belirleme seçeneği ile QEP darbelerinin nasıl sayılacağı belirlenir. Sayıcı (counter) seçeneği seçilirse, indeks verisi referans alınmadan doğrudan sayıcı modunda çalışılır. RPM (Repeat Per Minute-Dakikadaki Devir Sayısı) seçeneği seçilirse, 1 dakikadaki devir sayısı hesaplanmaktadır. RPM seçili durumdayken, pozitif dönme (positive rotation) seçeneğinden, dönme hareketi yönünün saat yönünde veya saat yönünü tersi yönde olma durumu belirlenir. Kodlayıcı çözünürlüğü (encoder revolution) kısmın darbe değeri girilmelidir. Örnekleme zamanı (sample time) kısmına girilen değer ile QEP pinlerinde bulunan

verilerin alınması için gerekli olan zaman aralığını belirler. Şekil 3.17’de kuadratik kodlayıcı darbe (quadratic encoder pulse) bloğuna ait parametreler görülmektedir [20,21].



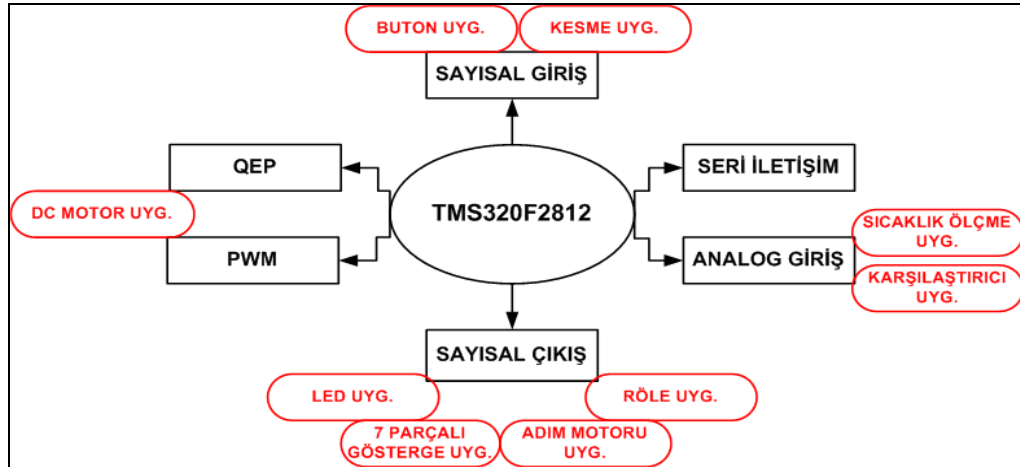
Şekil 3.17. Kuadratik kodlayıcı darbe bloğuna ait parametreler.

BÖLÜM 4

eZdsp F2812 DENEY SETİNİN TASARIMI VE GERÇEKLEŞTİRİLMESİ

4.1. GERÇEKLEŞTİRİLEN DENEY SETİ BLOK DİYAGRAMI

Bu çalışmada, 32 Bit sayısal işaret işlemcilerin kullanımının daha etkili ve kolay bir şekilde öğretilmesini kolaylaştırmak ve kullanımını yaygınlaştırmak amacıyla bir deney seti tasarımı ve yapımı gerçekleştirilmiştir. Çalışmada kullanılan TMS320F2812 Sİİ'nin, ADC, PWM, QEP, sayısal giriş-çıkış, analog giriş, seri iletişim ve kesme özelliklerinin gösterilebilmesi için Sİİ ara yüzlerine dışarıdan bağlanan çeşitli devreler tasarlanmıştır. Şekil 4.1'de deney setine ait blok diyagram ve özelliklere bağlı olarak gerçekleştirilen devre kartları görülmektedir.

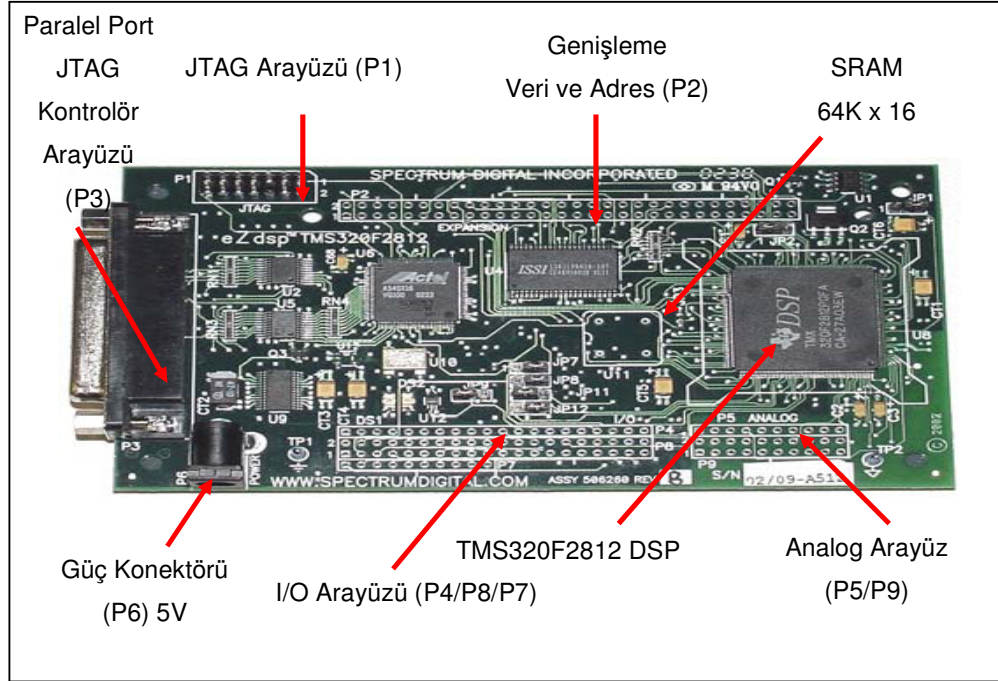


Şekil 4.1. Deney setine ait blok diyagram ve gerçekleştirilen devre kartları.

4.2. eZdsp F2812 Sİİ GELİŞTİRME KARTI GENEL ÖZELLİKLERİ

eZdsp F2812 geliştirme kartı, Sİİ mikroişlemcisinin çalıştırılmasını ve geliştirilen programların devre üzerinde emülasyonunu gerçekleştirmeyi sağlar. Code Composer

Studio IDE adlı programla birlikte çalışan bu devre kartı, bilgisayara paralel port (LPT) üzerinden bağlanmaktadır. eZdsp [2]. Şekil 4.2’de eZdsp320F2812’nin dış görünümü ve P ara yüzlerinin yerleşimi görülmektedir.



Şekil 4.2. eZdsp F2812 geliştirme kartı dış görüntüsü.

4.2.1. eZdsp F2812 Bağlantıları ve İşlevleri

eZdsp F2812 başlama kiti bağlantıları ve işlevleri Çizelge 4.1’de gösterilmektedir.

Çizelge 4.1. eZdsp F2812 bağlantıları.

Bağlantılar	İşlevleri
P1	JTAG ara yüzü
P2	Expansion (çoğullama)
P3	Paralel Port/JTAG denetim ara yüzü
P4/P8/P7	I/O ara yüzü
P5/P9	Analog ara yüzü
P6	Güç bağlantısı

4.2.2. eZdsp F2812 Üzerindeki Ledler

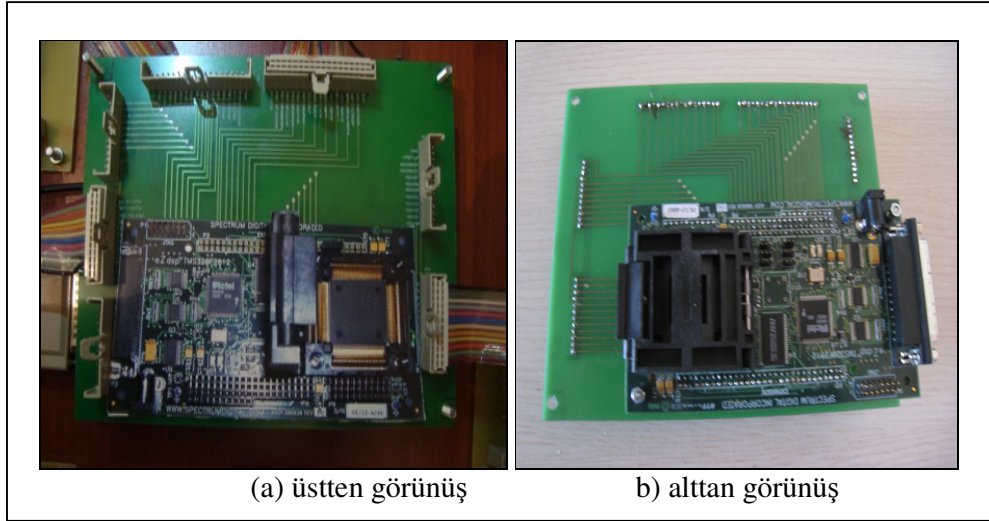
eZdsp F2812 başlama kitinin üzerinde iki adet led bulunmaktadır. Bunlardan DS1, 5 voltun varlığını göstermektedir ve eZdsp F2812 üzerinde güç varsa açık durumdadır. DS2, yazılım denetimi altındadır ve SII üzerindeki XF bacağına bir tampon ile bağlıdır. Bağlantı seçenekleri Çizelge 4.2’te gösterilmektedir [22].

Çizelge 4.2. DS1 ve DS2 led durumları.

LED	Sinyal	Denetim Sinyali
DS1	Yeşil	+5 volt
DS2	Yeşil	XF bit (XF high= on)

4.3. DENEY SETİ ÜZERİNDE BULUNAN DEVRE KARTLARI

4.3.1. eZdsp F2812 Geliştirme Kartının Genişletilmesi



Şekil 4.3. Geliştirme kartının görünüşü.

SII mikrodenetleyicisinin çalıştırılmasını ve geliştirilen programların devre üzerinde emülasyonunu gerçekleştirmeyi sağlayan eZdsp F2812 geliştirme kartı üzerindeki P1, P2, P3, P4, P5, P6, P7, P8 ve P9 ara yüz bağlantılarının geliştirme kartı

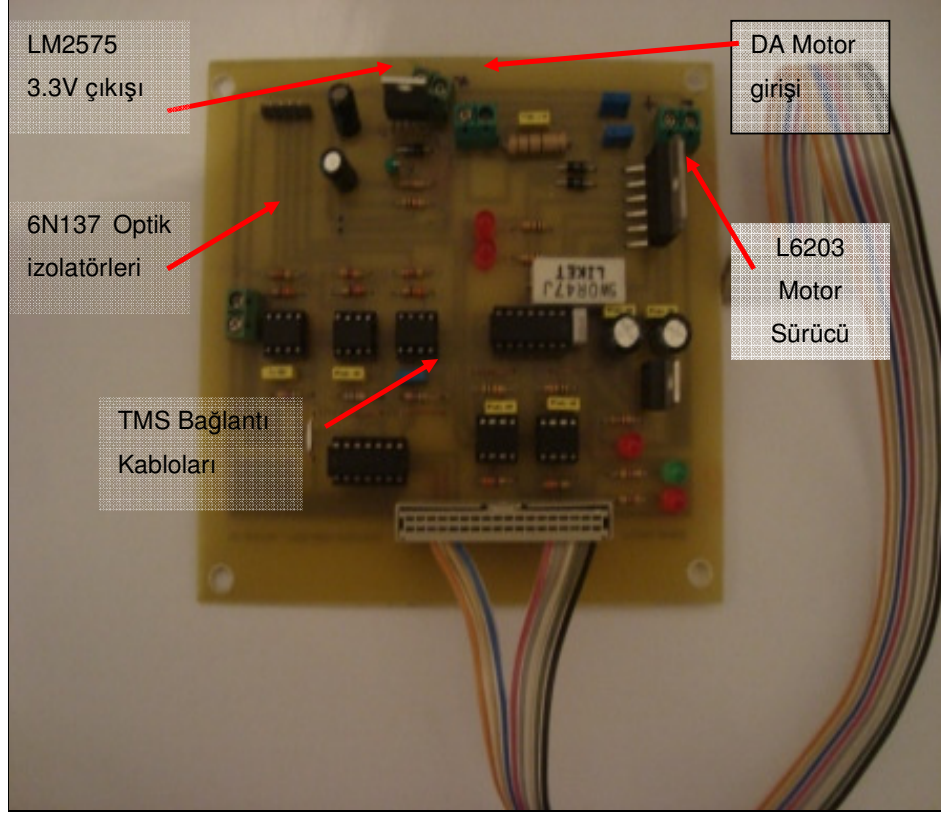
üzerindeki konumları gerçekleştirilmek istenen uygulamalar için uygun bulunmamıştır. Bu bağlantılarda kullanılacak olan P4, P5, P7, P8 ve P9 ara yüz bağlantılarının geliştirme kartına monte edilecek başka bir kart yardımıyla genişletilmesi çalışma alanını daha kullanışlı hale getirmiştir. Böylece kullanım amacına göre gerçekleştirilen diğer devreler, tasarlanan bu kart üzerine tunik soketler yardımıyla kolaylıkla bağlanabilmektedir. Kartın üstten ve alttan görünüşü Şekil 4.3'te gösterilmektedir.

4.3.2. Motor Hız Denetim Devre Kartı

Bu devre, Sİİ'nin QEP ve PWM özelliklerini incelemek amacıyla yapılmıştır. Devrede, Sİİ'den gelen PWM sinyaline göre DA (Doğru Akım) motorun hız ve yön denetimi gerçekleştirilir. DA motora bağlı yük değişimleri, motorun hızında değişimlere neden olur. Bu değişimler quadratic enkoder tarafından algılanır. Enkoderden alınan hız verileri (QEP), Sİİ'ye gönderilir ve burada kullanılacak olan denetim yöntemi devreye girer. Matlab ortamında bloklarla belirlenen denetim yöntemine göre motorda sabitlenmesi istenen hız değeri ayarlanır. t_{on} ve t_{off} sürelerinin değişimiyle motorun hızını değiştirebilen PWM sinyali yine Sİİ'den motora gönderilir. Böylece motorun çektiği yük değişikliklerine rağmen PWM sinyali ile sabit bir motor hızı elde edilmiş olur. Ek I'da devre kartının açık şeması verilmiştir.

Tasarlanan bu devrede, 6N137 optik izolatörleri kullanılarak Sİİ ile daha yüksek gerilimle çalışan motor sürücü katını birbirinden ayırarak işlemcinin zarar görmesi engellenmiştir. İşlemci için gerekli olan 3.3V'luk gerilim ise LM2575 entegresi ile sağlanmıştır. Şekil 4.4'de motor hız denetim devresi görülmektedir.

Devreye 12V gerilime sahip dakikada 2000 devir yapabilen bir DA motor bağlanmıştır. Motor üzerinde bulunan encoder (kodlayıcı) ise 512 devir/pals çözünürlüğüne sahiptir.



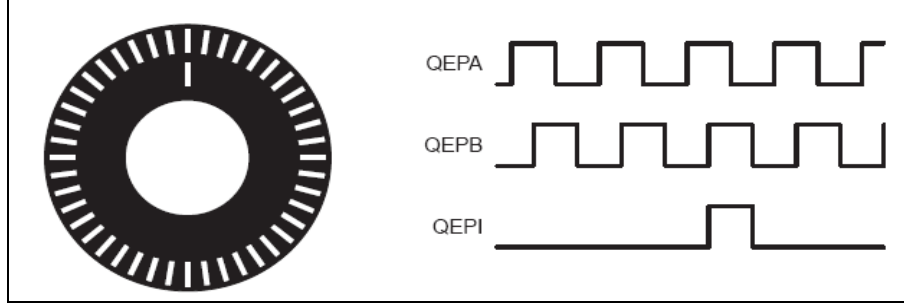
Şekil 4.4. Motor hız denetim kartı görünüşü.

4.3.2.1. Kuadratik Kodlayıcı Darbe

Kodlayıcı disk üzerinde bulunan ayrıık bölmeler, motor hareket ettiği sırada, içersinden geçen ışııkla birlikte hareketli bir model oluşturur (Şekil 4.5). Bu model, disk üzerindeki bölme sayısını hesaplamamıza yardımcı olur. Burada bölme sayısı, her devirde oluşan renk geçişleri olarak tanımlanır. Diskin attığı her bir turu hesaplamak için ise disk üzerindeki diğer sırada ikinci bir ayrıık bölme bulunur. Bu bölmedeki her ışık geçişi de bir tur olarak hesaplanırken oluşan bu sinyale indeks pulsü (QEPI) denir. Normalde 0'da duran indeks pulsünün her 1 olduğu konum, diskın bir dönüşünü ifade eder.

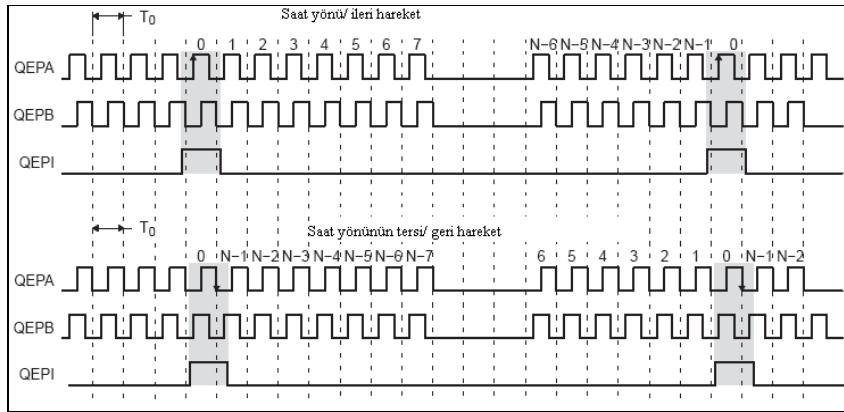
Yön bilgisini tayin etmek için, disk üzerindeki bölmeler iki farklı foto element tarafından okunur ve disk üzerindeki 1/4'lük mekaniksel yer deęiştirme dikkate alınır. Disk döndüğü sürece iki foto element birbirinden 90 derece faz farklı olan sinyaller üretir. Bunlar genellikle kuadratik QEPA ve QEPB sinyalleri olarak

adlandırılırlar. Birçok enkoder için saat yönü, QEPB den önce pozitif giden, QEPA olarak tanımlanır ve karşılıklı olarak Şekil 4.6’da gösterilir.



Şekil 4.5. Kodlayıcı diskin görünümü ve oluşan QEP sinyalleri.

Kodlayıcı tekerleği tipik olarak motorun her bir devri için, bir devir yapar. Bu yüzden, QEPA ve QEPB çıkışlarından gelen sayısal işaret frekansı, motorun hızına bağlı olarak oransal bir şekilde değişir. Örneğin, 2000 satırlık bir enkoder dakikada 5000 devir yapan bir motora doğrudan bağlandığında 166,6 kHz’lik bir frekansta cevap verir, dolayısıyla QEPA nın yada QEPB’nin çıkış frekansını ölçülerek motorun hızı tahmin edilmiş olur [23].

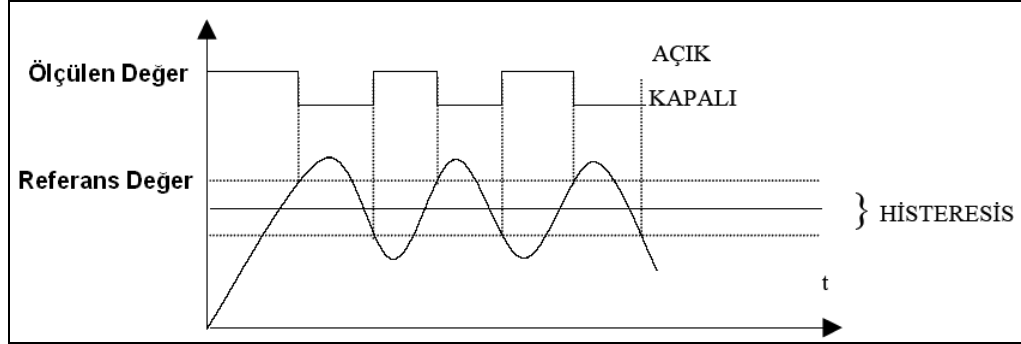


Şekil 4.6. QEPQ, QEPB ve QEPI sinyalleri ve birbirlerine göre konumları.

Gerçekleştirilen motor hız denetim devresinde DA motorunun farklı denetim yöntemleriyle çalıştırılması mümkündür. Bu çalışmada on-off (aç-kapa) ve PID (Proportional Integral Derivative) denetim yöntemleriyle uygulamalar yapılmıştır.

4.3.2.2. Aç-Kapa Denetim Yöntemi

Geleneksel denetim tekniği olan aç-kapa denetim yönteminde, ölçülen değer, referans değerinin üzerine çıktığında çıkış sinyali açılır, referans değerinin altına inmesiyle de çıkış sinyali kapatılır.

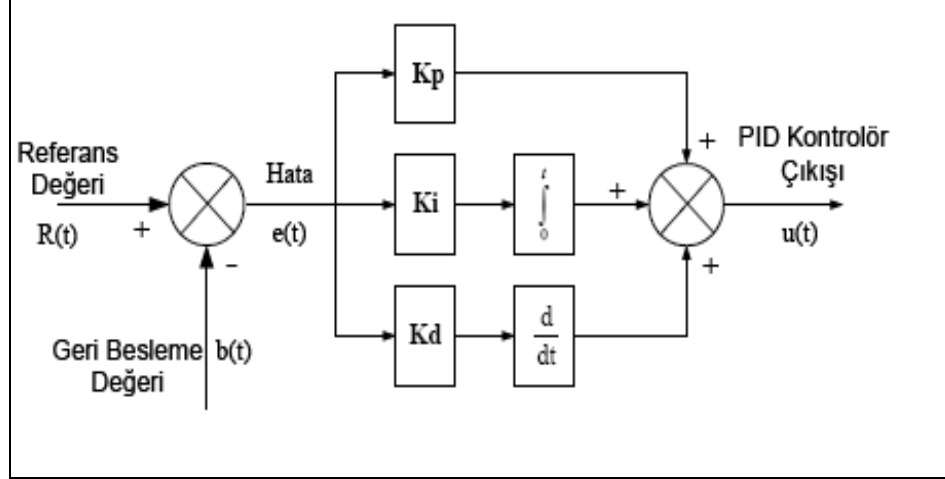


Şekil 4.7. Aç-Kapalı denetim ölçülen değer ve referans değer.

Sistemlerin sürekliliği nedeniyle, çıkış gücü kesildiği halde, ölçülen değer yükselmeye devam eder ve referans değeri üzerinde sürekli bir dalgalanma oluşur. Dalgalanmanın tepeden tepeye değişim ve sıklığı denetimi yapılan işlemin dinamiğine bağlıdır. Aç-Kapa denetim sistemlerinde genellikle, anlatılan sistemin, histeresis eklenmiş tipi kullanılmaktadır. Bu şekildeki bir çalışmada referans değeri etrafında histerisiz bantı oluşturulur. İşlem değeri, referans değerini geçer geçmez denetim cihazı çıkışı "kapalı" sinyali üretmez; ancak bu bant aşıldıktan sonra çıkış kapatılır. Aynı şekilde, ölçülen değer düşerken, referans değerinin altına düşünce değil, histeresis bandının dışına çıkınca "açık" sinyali üretilir (Şekil 4.7).

4.3.2.3. PID Denetim Yöntemi

PID kontrolörleri oluşturan kısımların her biri birer katsayı ile yönetilirler. Bu oransal, integral ve türev kazanç katsayıları (K_p , K_d , K_i) her sistem için ayrı değerler almaktadırlar. Şekil 4.8'de PID kontrolörün iç yapısına ait blok diyagramı verilmiştir. Bu blok diyagramından da görüldüğü gibi PID kontrolörün yapısı oransal kazanç, integral alıcı ve türev alıcı devrelerin birleşiminden oluşmaktadır.



Şekil 4.8. PID kontrolör içyapısı.

PID kontrolörün çıkışı $u(t)$ Eşitlik 5.1’de ifade edilmektedir.

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t) \quad (5.1)$$

şeklinde ifade edilir. Burada K_p oransal kazanç, K_i integral kazanç ve K_d türev kazanç katsayılarıdır.

Eşitlikte $e(t)$ olarak gösterilen hata değeri ise Eşitlik 5.2’de verilmektedir.

$$e(t) = R(t) - b(t) \quad (5.2)$$

yardımla hesaplanabilir.

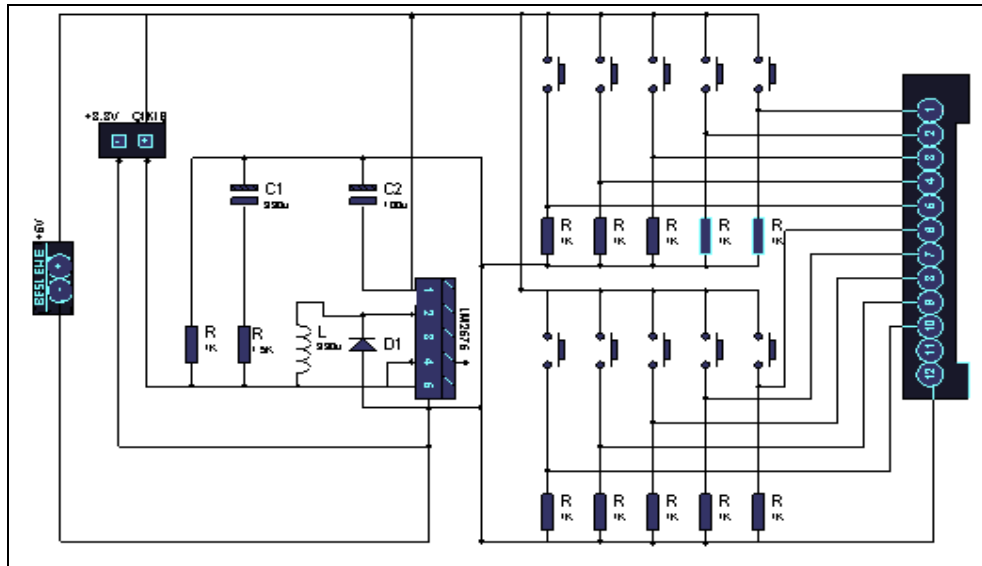
PID kontrolörü oluşturan oransal, integral, türev kazançlarının her biri sistemin çalışmasına çeşitli şekillerde etki etmektedir. Kapalı çevrim sisteminde kontrolörün her birinin etkisi (K_p, K_i, K_d) Çizelge 4.3’te özetlenmiştir[24]. Oransal etkinin artmasıyla kalıcı hal hatasının azaldığı fakat buna bağlı olarak titreşim ve aşımın arttığı, integral etkinin artırılmasıyla hatanın neredeyse sıfırlandığı ve bununla birlikte oransal ve türev etkide biraz daha arttığında aşım ve titreşimin de ortadan kaldırılabilceğini görülmüştür [24].

Çizelge 4.3. P, I ve D ayarlarının bağımsız olarak etkileri.

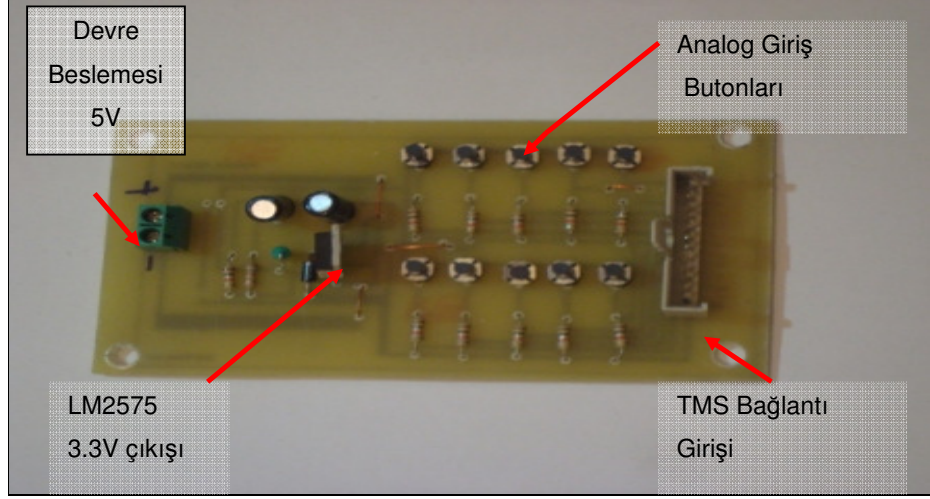
	Yükselme Zamanı	Max Aşım	Yerleşme Zamanı	Kalıcı Hal Hatası
K_p	Kısalır	Artar	Kısalır	Küçülür
K_i	Kısalır	Artar	Genellikle Artar	Sıfırlanır
K_D	Fazla değişmez	Azalar	Genellikle Kısalır	Etkilenmez

4.3.3. Sayısal Giriş Devre Kartı

Bu devrenin amacı Sİİ'ye sayısal veri gönderebilmektir. Tasarlanan sayısal giriş kartında 10 adet buton kullanılarak Sİİ'ye 10 farklı kanal aracılığıyla sayısal veri gönderilebilmektedir. Devreye pull_up olarak bağlanan butonlardan Sİİ'ye gidecek veri 3.3V'luk gerilim seviyesini aşmaması gerektiğinden bu devrede de LM2575 entegresinden yararlanılmıştır. Şekil 4.9'de açık şeması verilen devre kartının görüntüsü Şekil 4.10'daki gibidir.



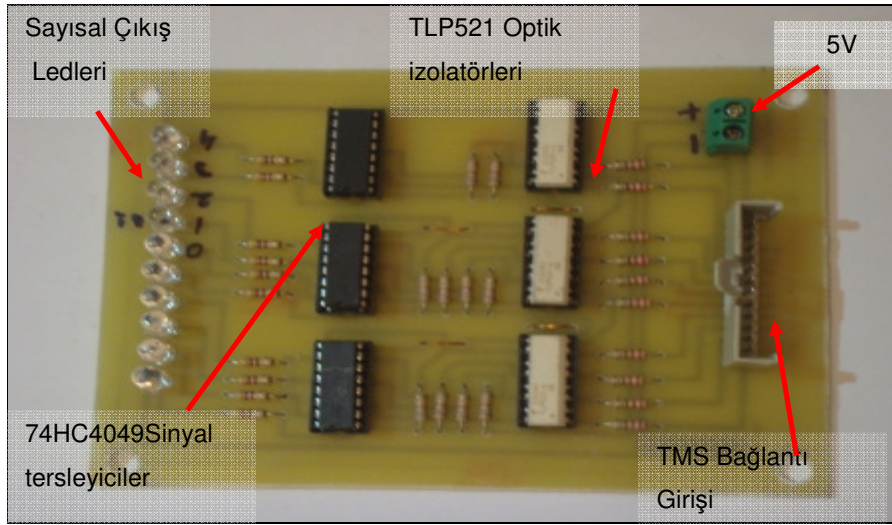
Şekil 4.9. Sayısal giriş devre kartı açık şeması.



Şekil 4.10. Sayısal giriş devre kartı görünüşü.

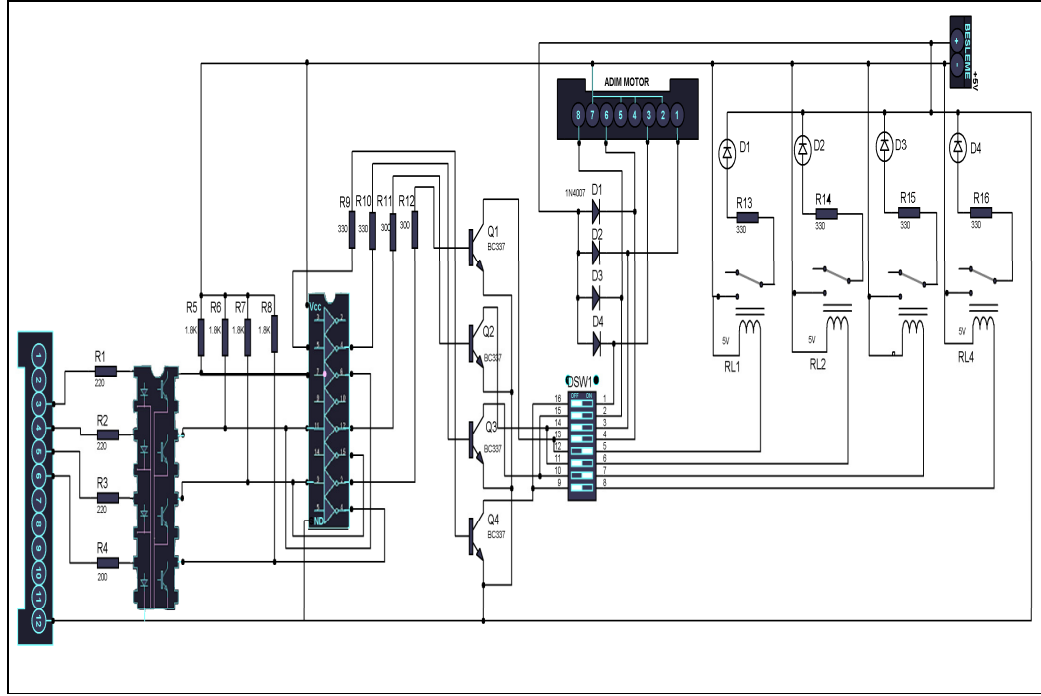
4.3.4. Sayısal Çıkış Devre Kartı

Bu devre Sİİ'den sayısal çıkış elde etmek amacıyla tasarlanmıştır. İşlemciden alınan 10 sayısal çıkışın gözlenebilmesi için devreye 10 adet led bağlanmıştır. Ledler ile Sİİ arasındaki optik yalıtımı sağlamak için devreye TLP521 optik izolatörü bağlanırken, oluşan sinyal terslemesi durumu 74HC4049 entegresi ile düzeltilmektedir. Şekil Ek II'de açık şeması verilen devre kartının görüntüsü Şekil 4.11'deki gibidir.

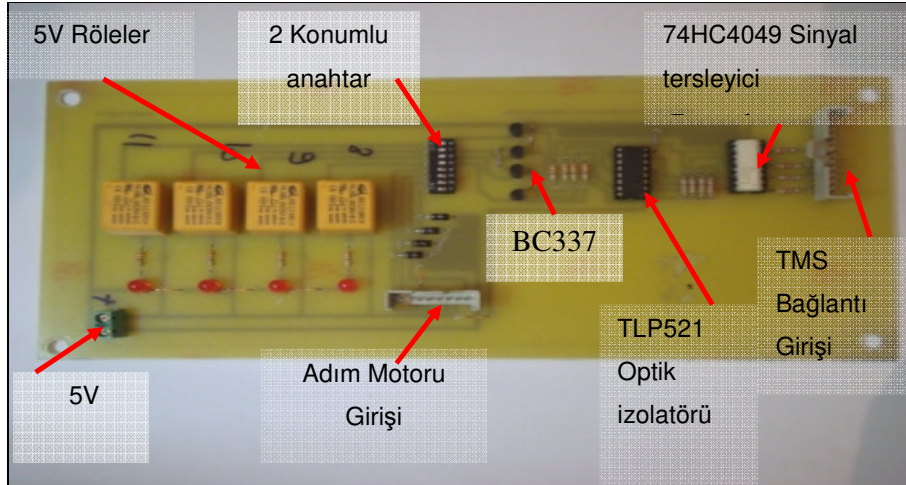


Şekil 4.11. Sayısal çıkış devre kartı görünüşü.

4.3.5. Röle Denetim Devre Kartı

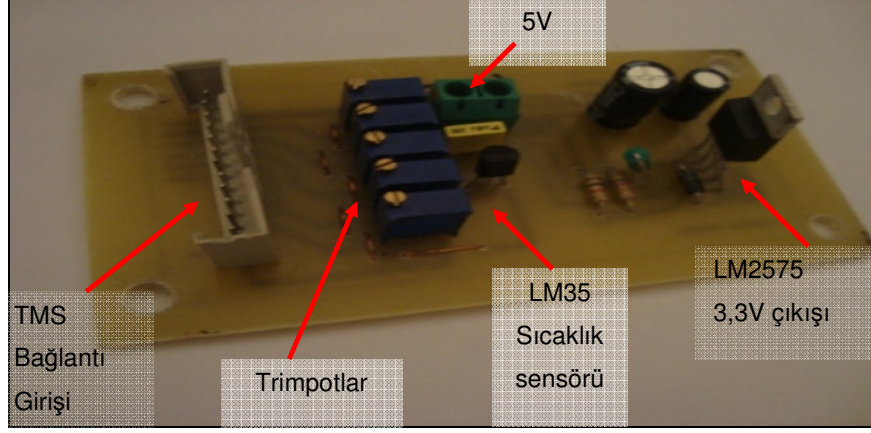


Şekil 4.12. Röle denetim devre kartı açık şeması.



Şekil 4.13. Röle denetim devre kartı görünüşü.

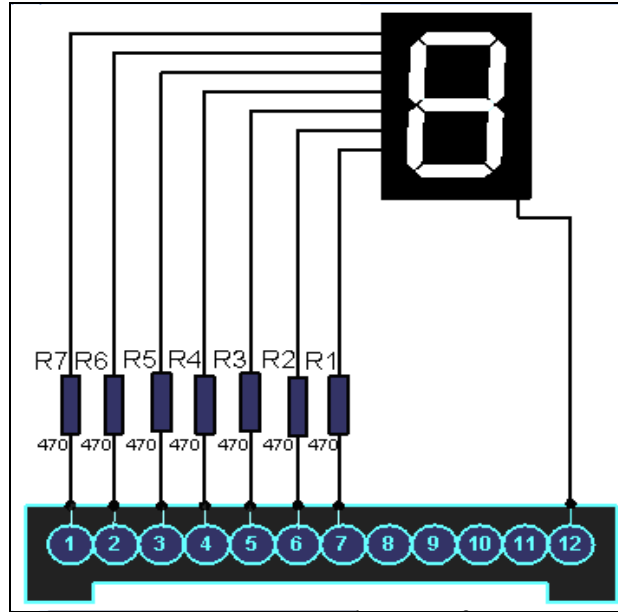
Tasarlanan bu devre aynı zamanda bir sayısal çıkış devresidir. Burada amaç, Sİİ'den alınan 3.3V'luk sayısal bir gerilimle yüksek akımda çalışan bir anahtarlama elemanı



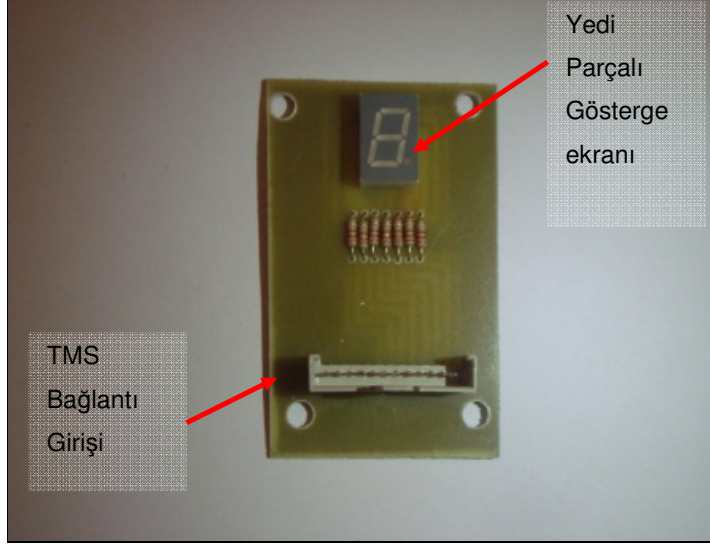
Şekil 4.15. Analog giriş devre kartı görünüşü.

4.3.7. Yedi Parçalı Gösterge Devre Kartı

Bu devre üzerinde 7 adet led bulunan bir ekranın direçle birlikte Sİİ'nin sayısal çıkış portuna bağlanan bir devredir. Ekran, ortak katod bağlantılı olduğundan içerisinde bulunan ledlerin ortak uçları devrenin (-)'sine bağlanırken, denetim (+) besleme ile sağlanmaktadır. Şekil 4.16'da açık şeması verilen devre kartının görüntüsü Şekil 4.17'deki gibidir.



Şekil 4.16. Yedi parçalı gösterge devre kartı açık şeması.



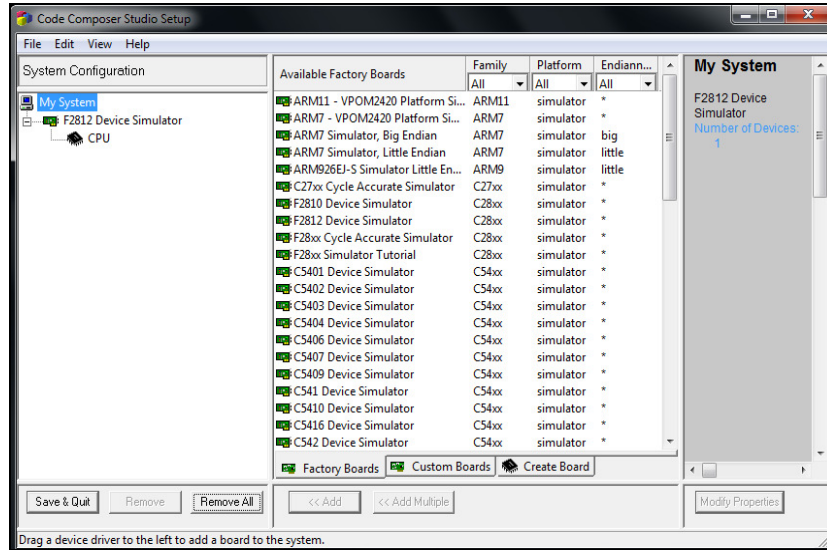
Şekil 4.17. Yedi göstergeli ekran devre kartı görünüşü.

BÖLÜM 5

eZdspF2812 DENEY SETİNDE GERÇEKLEŞTİRİLEN UYGULAMALAR

Gerçekleştirilen deney seti üzerinde basitten ileri seviyeye kadar birçok uygulamanın gerçekleştirilebilmesi mümkündür. Bu uygulamalara başlanmadan önce yapılması gereken ayarlar ve dikkat edilmesi gereken hususlar şu şekildedir:

- 1- Matlab R20007a kurulumu gerçekleştirilir.
- 2- CCS 3.1'in kurulumu gerçekleştirilir.
- 3- CCS programı açılarak Sİİ tanıtılır. Böylece program ile işlemcinin bağlantısı kurulur (Şekil 5.1).



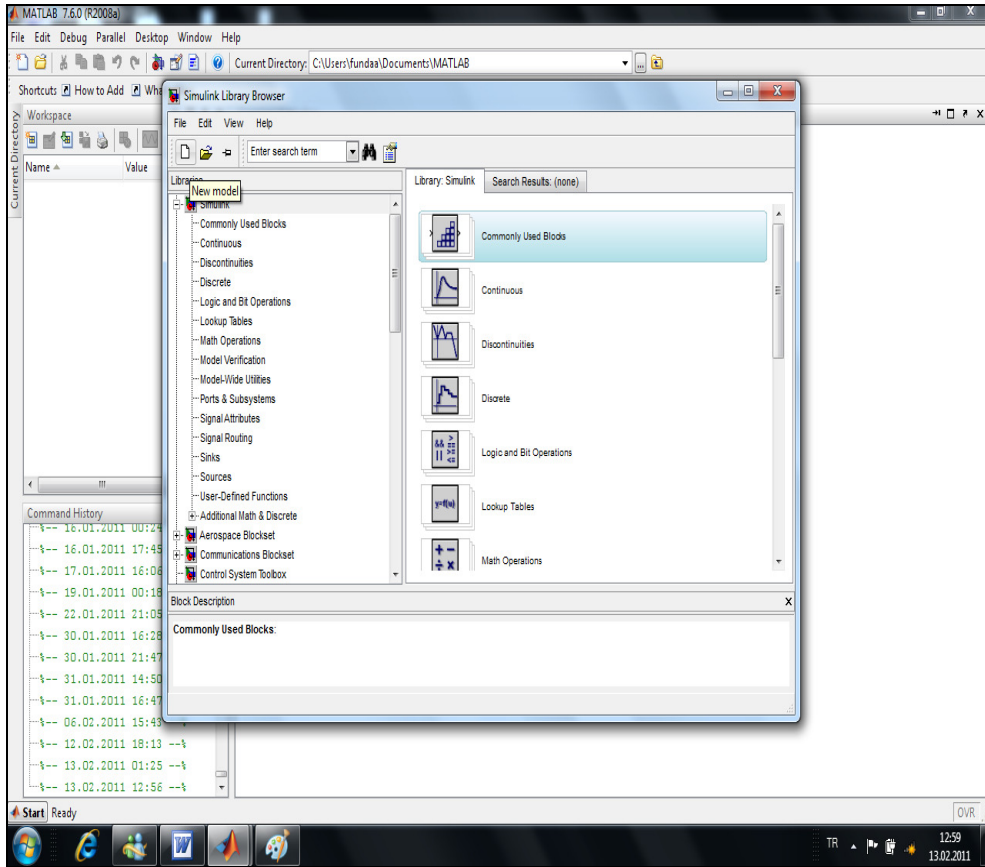
Şekil 5.1. eZdspF2812 genişletme kartının CCS programına tanıtılması.

- 4- Deney seti üzerindeki devre kartlarının akım ve gerilim değerleri kontrol edilir.
- 5- Set üzerinde bulunan adım motoru ve DA motorunun işlemciyle mesafesine dikkat edilir. Birbirlerini etkilemeyecek şekilde konumlandırılır.

5.1. KAYAN LED UYGULAMASI

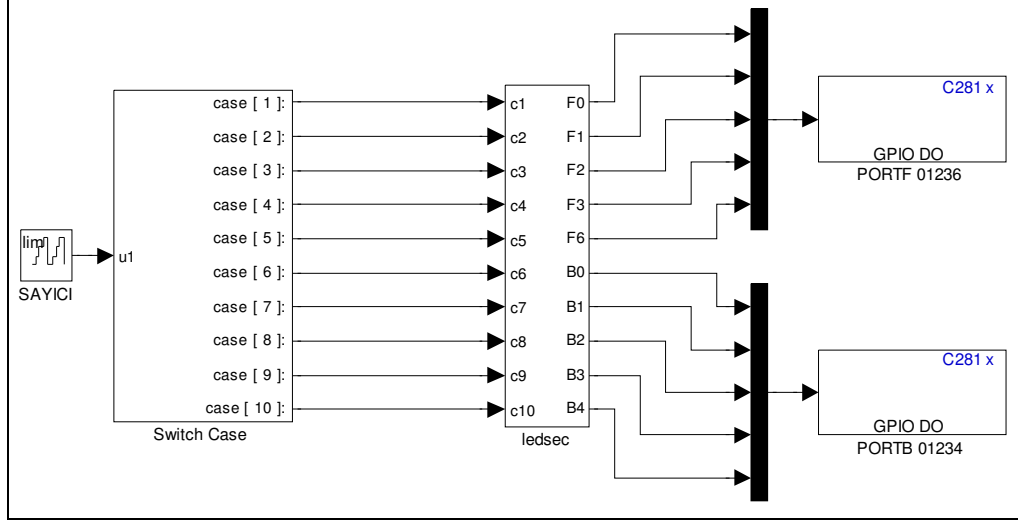
Kayan led uygulamasında yapılmak istenen, Şekil 4.12’de gösterilen sayısal çıkış kartındaki ledleri, sırasıyla 1 konumuna getirerek yanmalarını sağlamaktır. Uygulamaya ait işlem basamakları şu şekilde gerçekleştirilir:

- 1- Matlab Simulink ortamında yeni bir simulink model penceresi açılır (Şekil 5.2).



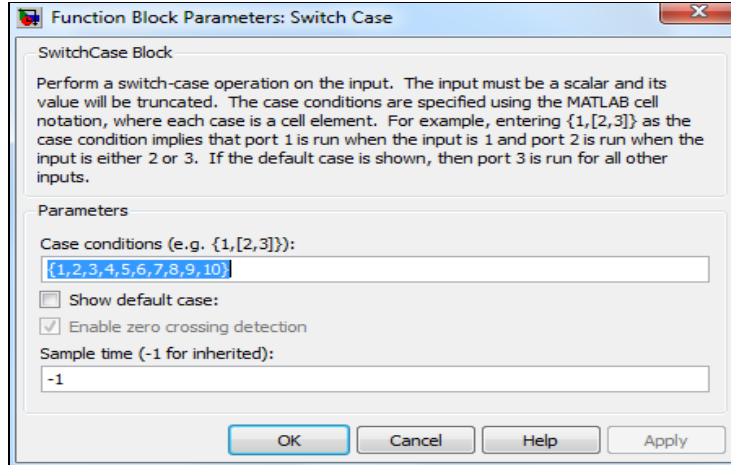
Şekil 5.2. Yeni bir simulink model penceresinin açılması.

- 2- Açılan pencere içerisine Şekil 5.3’te görülen devre çizilir ve *.mdl. uzantılı olarak kaydedilir. Devrede farklı durum değişiklikleri meydana geleceğinden simulink kütüphanesinde bulunan “Ports & Subsystems” başlığındaki “Switch Case” ve “Switch Case Action Subsystem” bloklarından yararlanılır.



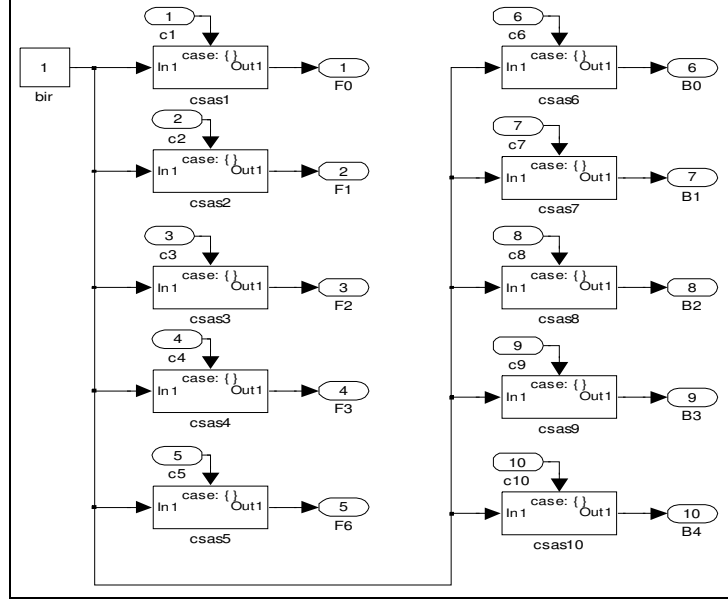
Şekil 5.3. Kayan led uygulaması.

- 3- Switch case, her durum için farklı çıktılar oluşturan bir yapıdır. Oluşturulmak istenen durum sayısı Şekil 5.4'deki fonksiyon parametre özelliklerinden ayarlanır.



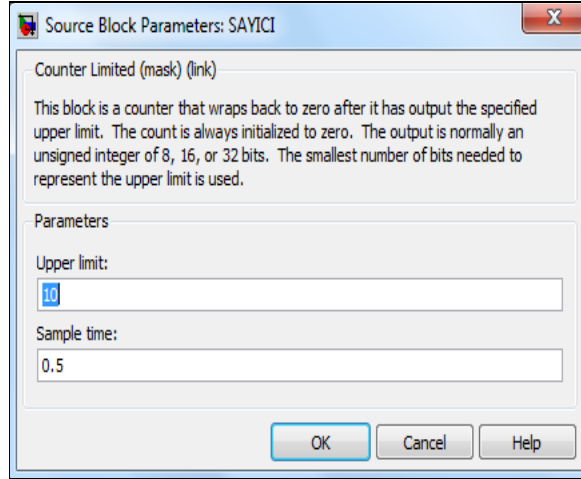
Şekil 5.4. Switch case fonksiyon parametreleri.

- 4- “Switch case action subsystem” bloğu modelde gösterilen “Ledsec” bloğuna altistem olarak gömülerek switch case bloğu ile ledler arasındaki bağlantı sağlanır. Switch case action subsystem bloklarından oluşan Ledsec blok yapısı Şekil 5.5'te görüldüğü gibidir.



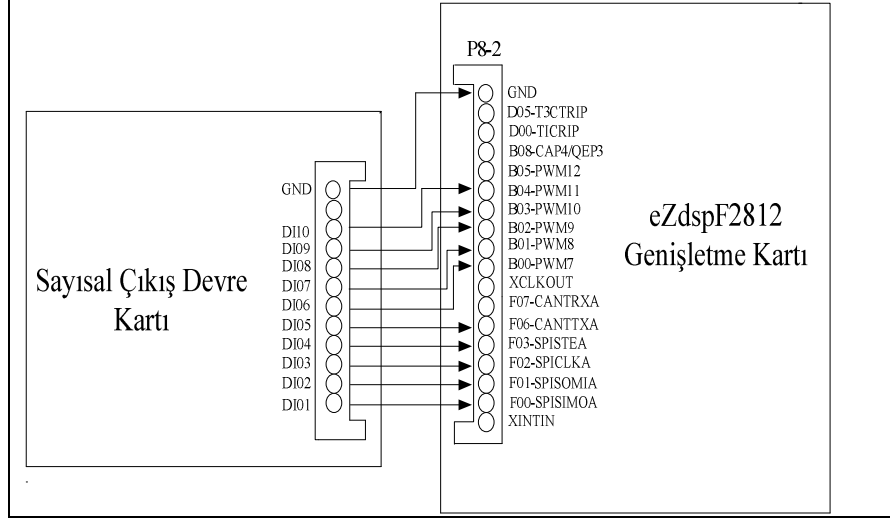
Şekil 5.5. Ledsec blok yapısı.

- 5- Modeldeki limitli sayıcı da oluşturulan durum sayısına göre Şekil 5.6’da gösterildiği gibi ayarlanarak sistemi aktifleştirecek sinyal oluşturulurken, elde edilen durum bilgileri Sİİ’nin sayısal çıkış bloklarına gönderilir.



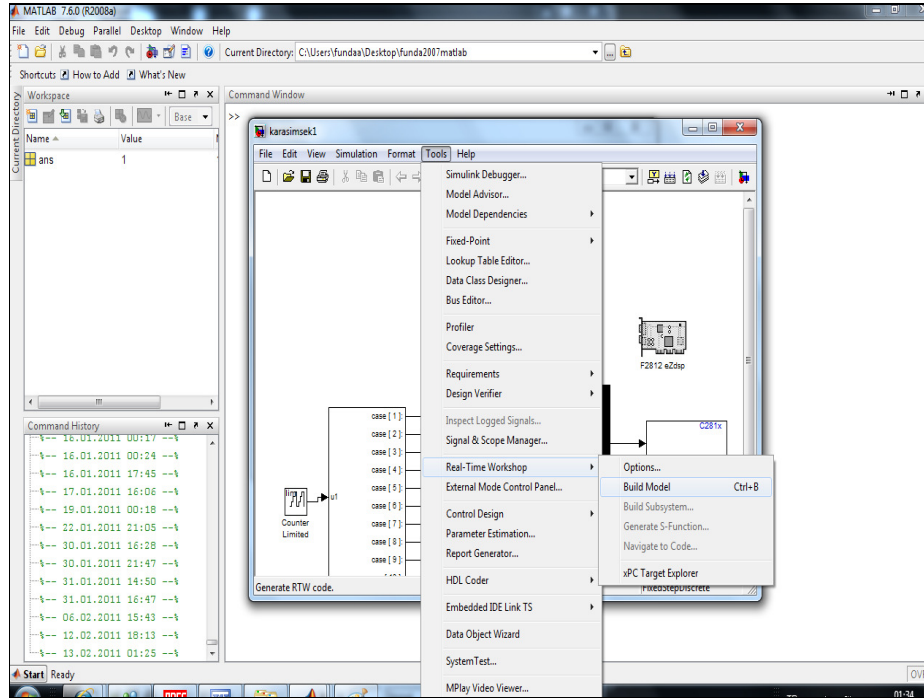
Şekil 5.6. Limitli sayıcı blok parametreleri.

- 6- Devrenin simulink modeli tamamlandıktan sonra Şekil 4.11’de gösterilen devre kartı Şekil 4.3’teki eZdspF2812 genişletme kartının P8-2 portuna Şekil 5.7’deki gibi bağlanır.



Şekil 5.7. Sayısal çıkış devre kartının bağlantısı.

7- Her iki devrenin de beslemesi verilir. Sayısal çıkış kartı +5V ile beslenir.

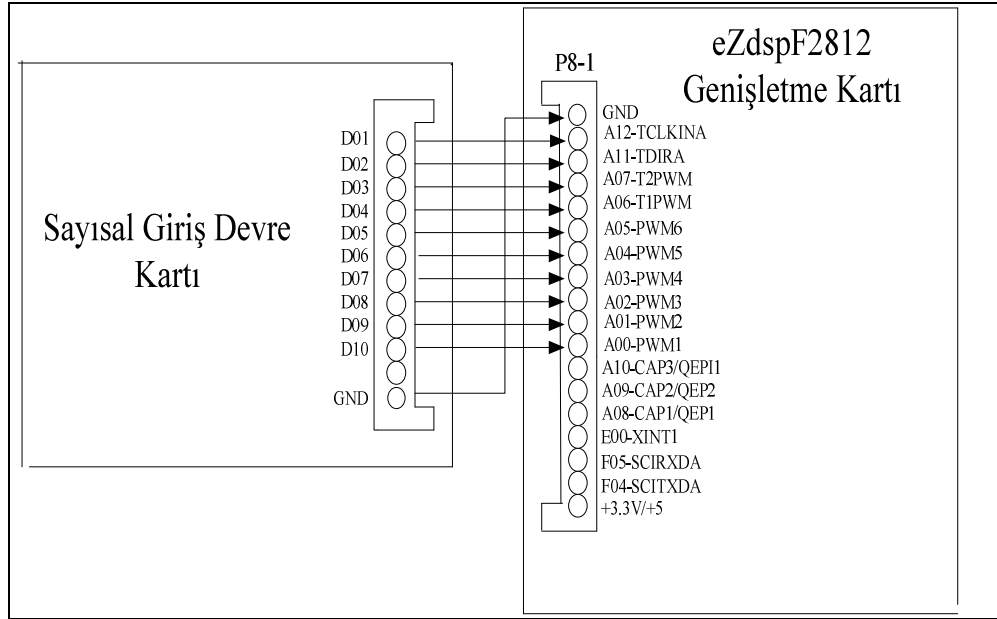


Şekil 5.8. Simulink modelinin CCS programına gönderilmesi.

8- Matlab simulink penceresi açıkken, “Tools” menüsünden “Real Time Workshop” ve “Build Model” seçenekleri takip edilerek, CCS programının

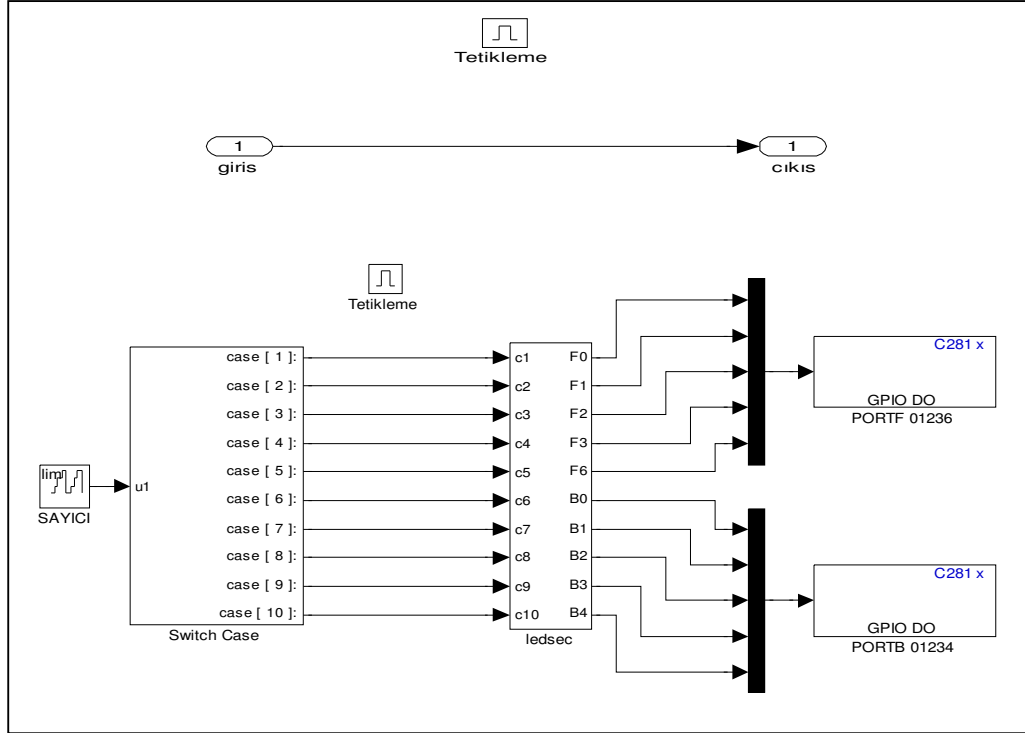
otomatik olarak açılması ve çizilen modelin makine diline çevrilerek işlemciye yüklenmesi sağlanır (Şekil 5.8).

- 9- Yapılan uygulama deney seti üzerinde bulunan “Sayısal çıkış devre kartı” üzerinden gözlemlenir. Böylece program çalıştığı sürece gerçekleştirilmek istenen değişiklikler deney kartı üzerinde görülür.
- 10- Sistem modeli tek başına çalışabildiği gibi deney seti üzerindeki “Sayısal giriş devre kartı”nda bulunan (Şekil 4.10) butonların denetiminde de gerçekleştirilebilir. Devre kartı Şekil 5.9’daki gibi eZdspF2812 genişletme kartının P8-1 portuna bağlanır.

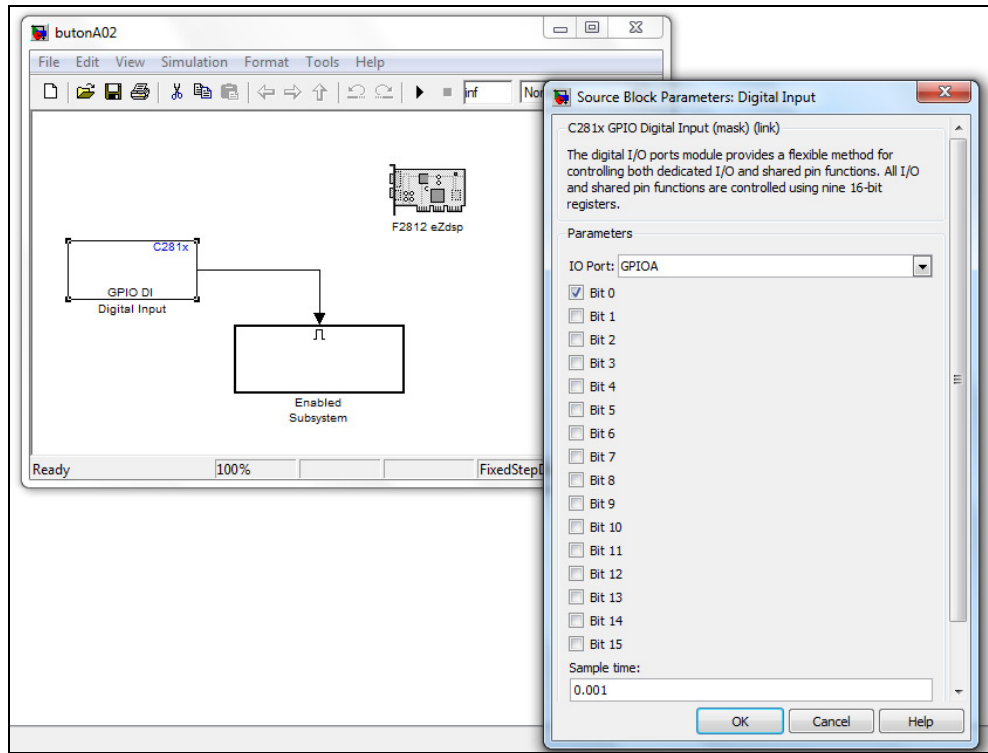


Şekil 5.9. Sayısal giriş devre kartının bağlantısı.

- 11- Simulink kütüphanesinden “Enabled subsystem” seçilir ve içyapısındaki giriş çıkış pinleri yerine kayan led uygulaması yerleştirilir (Şekil 5.10). “Enabled subsystem” bu modelde “tetiklemeli alt sistem” olarak değiştirilmiştir.
- 12- Denetimi yapılacak buton “sayısal giriş bloğu” içerisinden Şekil 5.11’deki gibi seçilir. Böylece yapılan uygulama A0 butonundan tetikleme geldiği sürece gerçekleştirilecektir.
- 13- Program işlemciye yüklendiğinde ledlerin çalışması için A0 butonuyla tetiklenmesi gerekmektedir.



Şekil 5.10. Tetiklemeli altsistem oluşturulması.

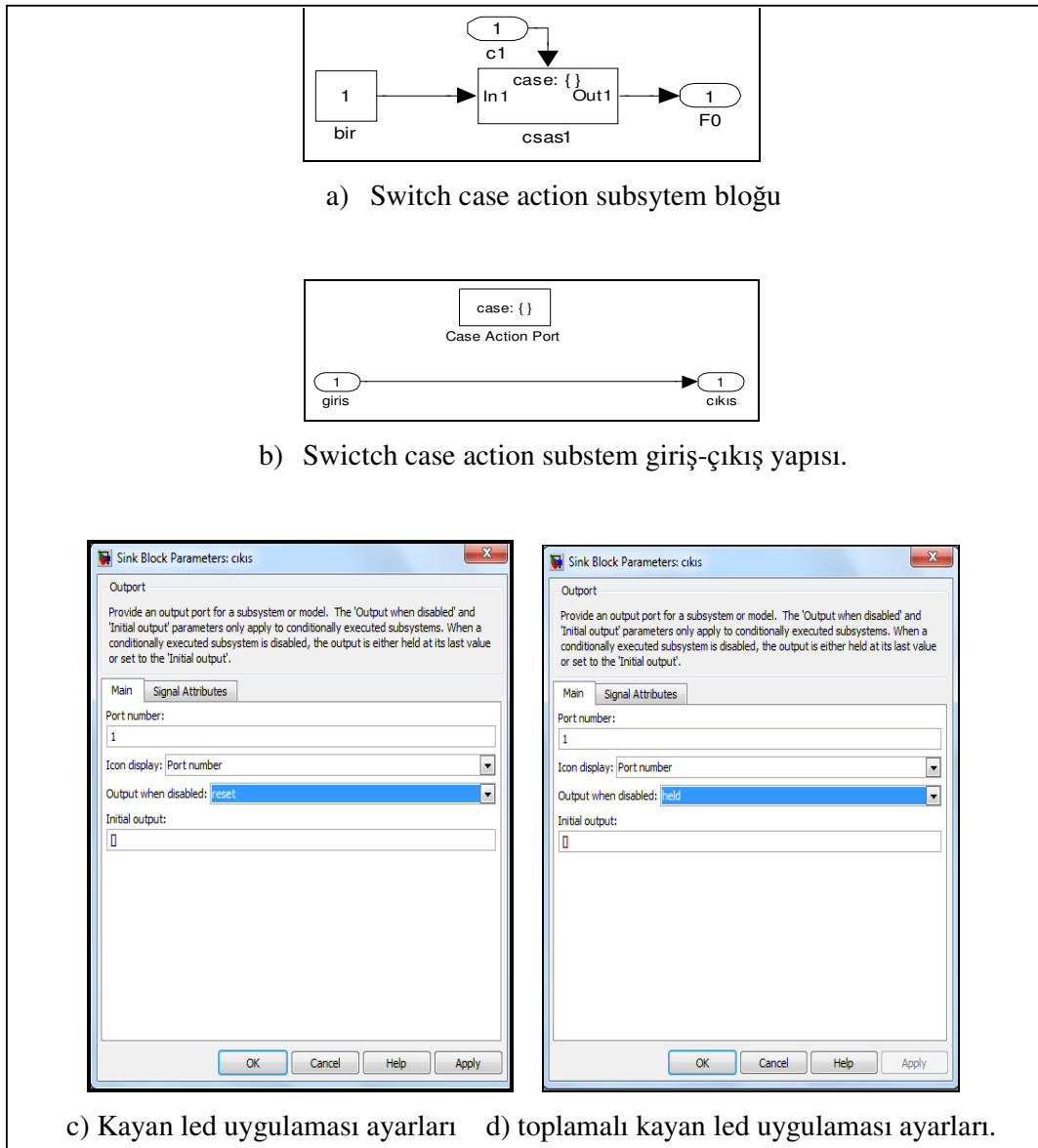


Şekil 5.11. Giriş butonunun ayarlanması.

5.2. TOPLAMALI KAYAN LED UYGULAMASI

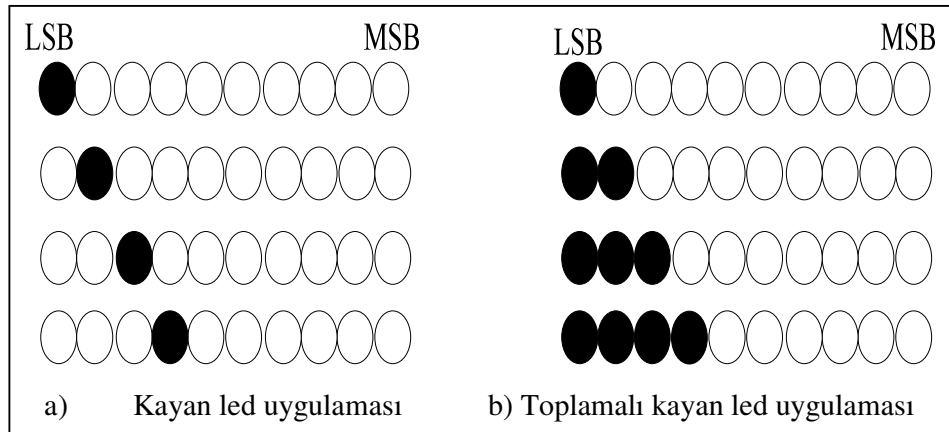
Bu uygulamada, ledlerin yine sırasıyla yanması gerçekleştirilir ancak; diğer uygulamadan farklı olarak bir led yanarken bir önceki ledin yanma durumunu koruması istenir. Uygulama şu şekilde gerçekleştirilir:

1. Matlab Simulink ortamında yeni bir simulink model penceresi açılır (Şekil 5.2).



Şekil 5.12. Switch case action subsystem bloğu ayarları.

2. Açılan pencereye Şekil 5.3'te görülen devre çizilir ve *.mdl. uzantılı olarak kaydedilir. Devre çiziminde bir farklılık yoktur.
3. Uygulamadaki farklılık "Ledsec" bloğunun alt sistemi olan "Case" bloğundaki çıkış özelliklerinin değiştirilmesiyle gerçekleştirilmiştir. Şekil 5.12.'de kayan led uygulaması ve toplamalı kayan led uygulamalarının çıkış özelliklerinin ayarlanması görülmektedir.
4. Devrenin simulink modeli tamamlandıktan sonra Şekil 4.11'de görülen devre kartı Şekil 4.3'teki eZdspF2812 genişletme kartının P8-2 portuna Şekil 5.7'deki gibi bağlanır.
5. Her iki devrenin de beslemesi verilir. Sayısal çıkış devre kartı +5V ile beslenir.
6. Matlab simulink penceresi açıkken, "Tools" menüsünden "Real Time Workshop" seçenekleri takip edilerek, CCS programının otomatik olarak açılması ve çizilen modelin makine diline çevrilerek işlemciye yüklenmesi sağlanır (Şekil 5.8).
7. Böylece program çalıştığı sürece gerçekleştirilmek istenen değişiklikler deney kartı üzerinde görülür.
8. Kayan led uygulaması ve toplamalı kayan led uygulamasında, ledlerin yanma şekilleri dört durum için Şekil 5.13'te gösterilmektedir.

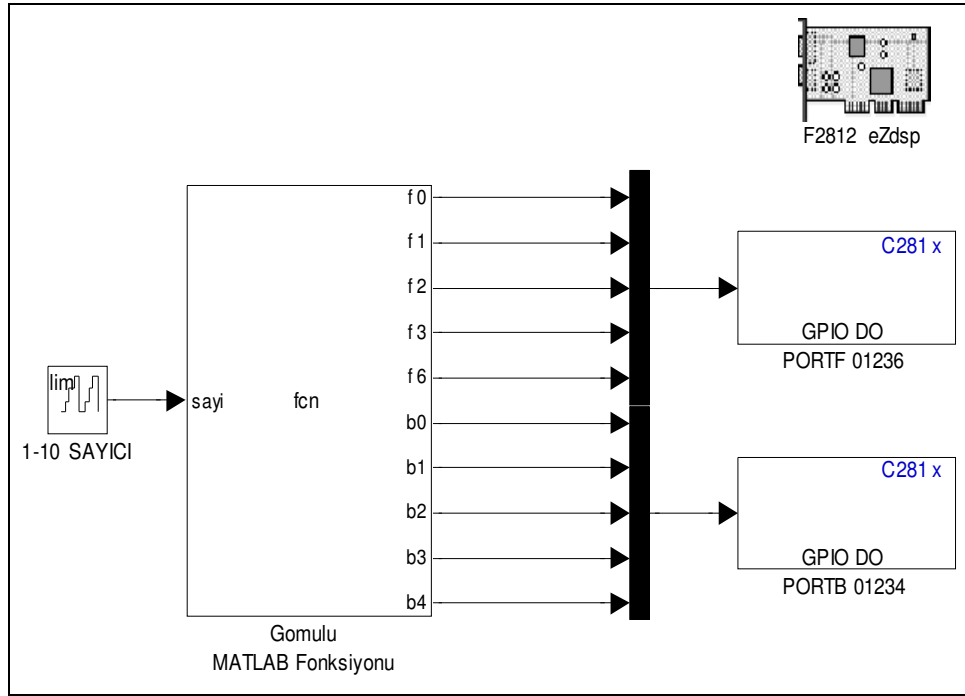


Şekil 5.13. Ledlerin yanma düzenlerinin karşılaştırılması.

5.3. GÖMÜLÜ MATLAB FONKSİYONU UYGULAMASI

Gömülü matlab fonksiyonu ile kayan led uygulamasının devre bağlantıları ve ledlerin çalışma şekilleri kayan led uygulaması ile aynıdır. Bu uygulamanın farkı “Switch Case” bloğu yerine simulink kütüphanesinde ve “User Defined Functions” başlığı altındaki “Embeded Matlab Function” bloğunun kullanılmasıdır. Uygulamaya ait işlem basamakları şu şekildedir:

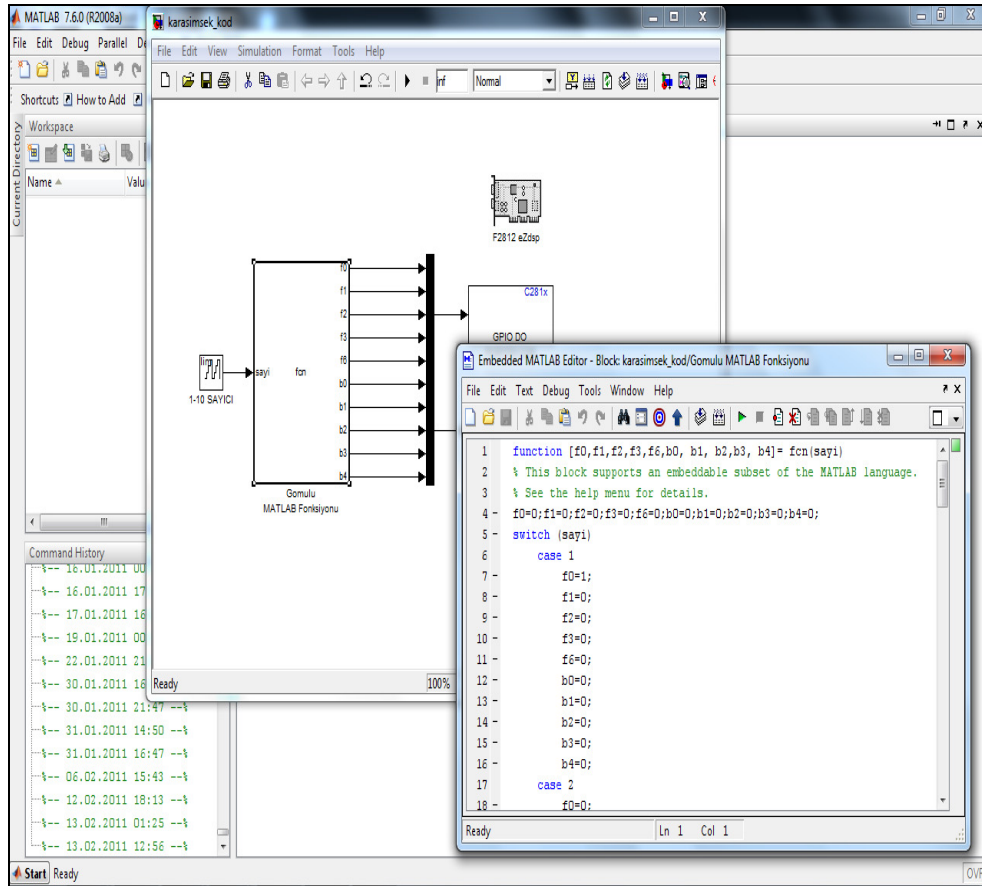
- 1- Matlab Simulink ortamında yeni bir simulink model penceresi açılır (Şekil 5.2).
- 2- Açılan pencereye Şekil 5.14’te görülen devre çizilir ve *.mdl. uzantılı olarak kaydedilir.



Şekil 5.14. Gömülü matlab fonksiyonu ile kayan led uygulaması.

- 3- Bu uygulamada “Switch Case” bloğu yerine simulink kütüphanesinde “ User Defined Functions” başlığı altındaki “Embeded Matlab Function” bloğu kullanılır. Bu bloğun adı gerçekleştirilen uygulamada “Gömülü Matlab Fonksiyonu” olarak değiştirilir.

- 4- Matlab fonksiyon bloğu, istenilen işlemi simulink blokları yerine yazılımsal olarak gerçekleştirebilme imkanı veren bir bloktur. Burada da farklı durumlar için case yapısı kullanılır ama; “Case” bloğu yerine fonksiyon bloğu içerisine gömülmüş kodlarla aynı işlem gerçekleştirilir (Şekil 5.15.).
- 5- Programda öncelikle $y=fcn(u)$ şeklinde fonksiyon oluşturulur. Fonksiyonda çıkış birimi ledler iken; giriş birimi, değişken durum değeri “sayı” olarak tutulur. Çıkış değerlerinin başlangıç değerleri sıfırlanarak case yapısı oluşturulur.
- 6- Devrenin simulink modeli tamamlandıktan sonra Şekil 4.11’de görülen devre kartı Şekil 4.3’teki “eZdspF2812 genişletme kartı”ın P8-2 portuna Şekil 5.7’deki gibi bağlanır.



Şekil 5.15. Gömülü matlab fonksiyonu içerisine case yapısının yazılımı.

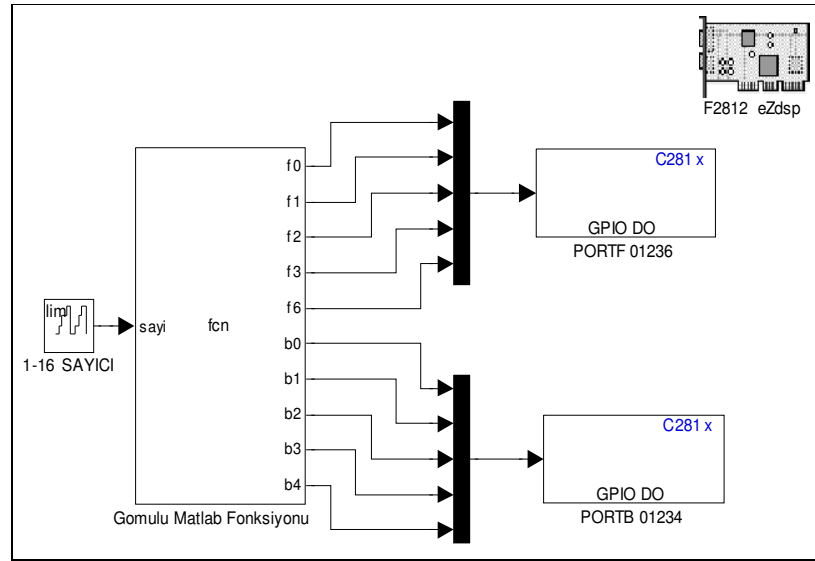
- 7- Her iki devrenin de beslemesi verilir. Sayısal çıkış kartı +5V ile beslenir.

- 8- Matlab simulink penceresi açırken, “Tools” menüsünden “Real Time Workshop” ve “Build Model” seçenekleri takip edilerek, CCS programının otomatik olarak açılması ve çizilen modelin makine diline çevrilerek işlemciye yüklenmesi sağlanır (Şekil 5.8.). Böylece program çalıştığı sürece gerçekleştirilmek istenen değişiklikler deney kartı üzerinde görülür.

5.4. BİNARY SAYICI UYGULAMASI

Binary sayıcı uygulaması da ledlerle yapılan diğer uygulamalar gibidir. Burada yapılmak istenen, ledlerin binary sayılar şeklinde artarak 1-16 arası saymalarını sağlamaktır. Uygulamaya ait işlem basamakları şu şekildedir:

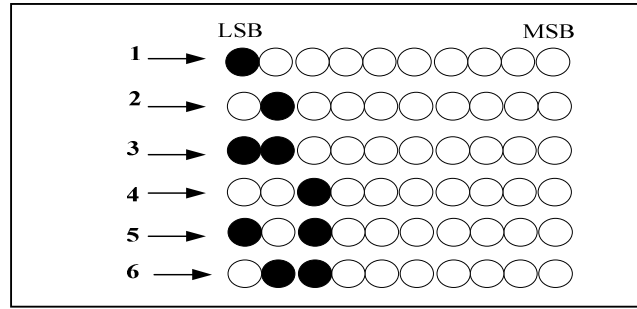
1. Matlab Simulink ortamında yeni bir simulink model penceresi açılır (Şekil 5.2).
2. Açılan pencereye Şekil 5.16’da görülen devre çizilir ve *.mdl. uzantılı olarak kaydedilir.



Şekil 5.16. Binary sayıcı uygulaması.

3. Farklı durum değişiklikleri, “Gömülü matlab fonksiyonu” içerisindeki çıkış durumlarının değiştirilmesiyle gerçekleştirilir. Burada toplam 16 farklı durum olduğundan limitli sayıcının üst limiti 16’ya çıkarılmıştır.

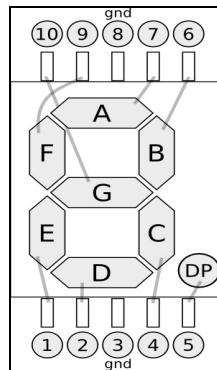
4. Devrenin simulink modeli tamamlandıktan sonra Şekil 4.11’de görülen devre kartı Şekil 4.3’teki “eZdspF2812 genişletme kartı”nın P8-2 portuna Şekil 5.7’deki gibi bağlanır.
5. Her iki devrenin de beslemesi verilir. Sayısal çıkış kartı +5V ile beslenir.
6. Matlab simulink penceresi açıkken, “Tools” menüsünden “Real Time Workshop” ve “Build Model” seçenekleri takip edilerek, CCS programının otomatik olarak açılması ve çizilen modelin makine diline çevrilerek işleme yüklenmesi sağlanır (Şekil 5.8).
7. Deney seti üzerinde bulunan “sayısal çıkış kartı” üzerindeki ledlerin çıkışları 6 durum için Şekil 5.17’de gösterildiği gibidir.



Şekil 5.17. Binary sayıcı uygulamasında ledlerin yanma düzeni.

5.5. YEDİ PARÇALI GÖSTERGE UYGULAMASI

Yedi parçalı gösterge, 7 adet ledden oluşturulmuş bir yapıdır Bu yapı elektronik devrelerde numaraları ve bazı karakterleri göstermek için kullanılır.

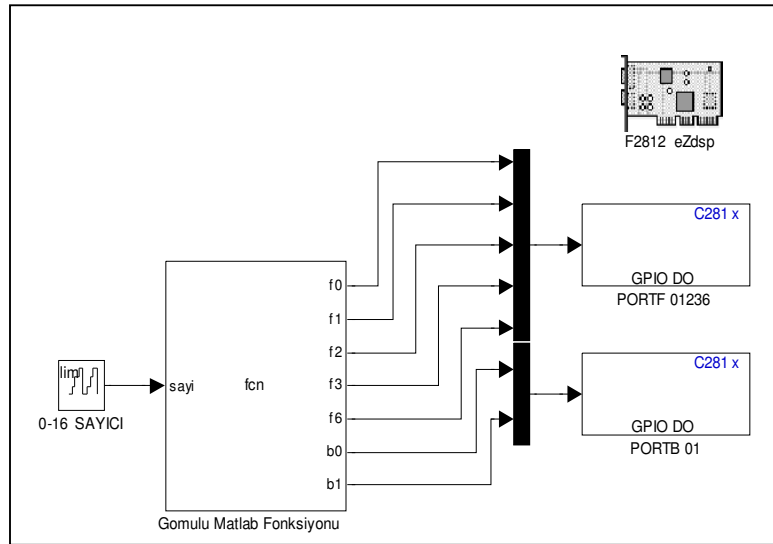


Şekil 5.18. Yedi parçalı gösterge üzerindeki ledlerin harflerle ifadesi.

Ekran, ortak anot ve ortak katot bağlantılı olmak üzere iki çeşittir. Gerçekleştirilen deney setinde kullanılan gösterge ortak katotludur ve devre içerisindeki tüm ledlerin katodları birbirine bağlıdır. Eleman çalıştırılırken ortak uca (-) verilirken, diğer uçlara verilen (+) beslemelere göre istenen rakam ve bazı harflerin değerleri görüntülenir. Şekil 5.18’de gösterge üzerindeki ledlerin konumları harflerle ifade edilmektedir.

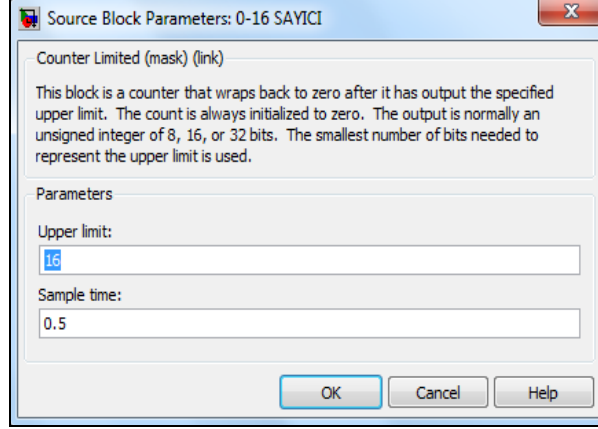
Uygulamaya ait işlem basamakları şu şekildedir:

- 1- Matlab Simulink ortamında yeni bir simulink model penceresi açılır (Şekil 5.2).

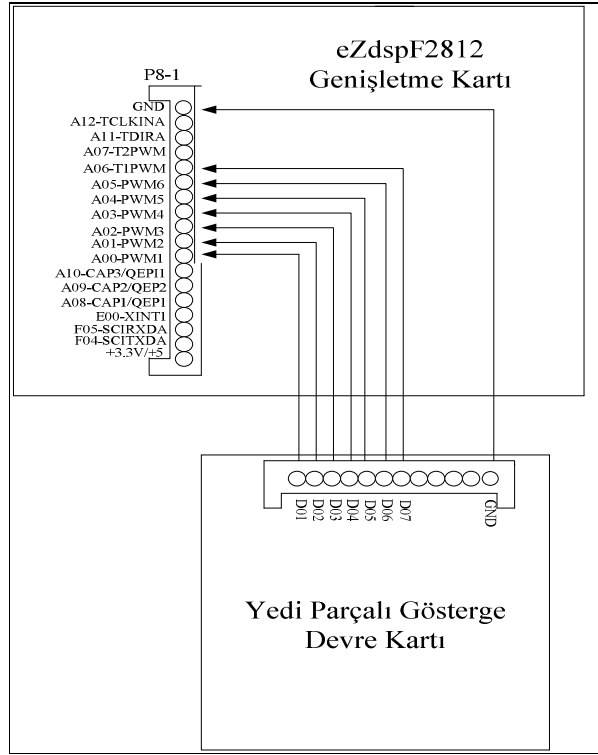


Şekil 5.19. Yedi parçalı gösterge uygulaması.

- 2- Açılan pencereye Şekil 5.19’da görülen devre çizilir ve *.mdl. uzantılı olarak kaydedilir.
- 3- Sistemin matlab simulink modelinde yine “Gömülü matlab fonksiyonu” kullanılmıştır. Bu uygulamada ekran üzerinde 0-9 arası rakamların ve a, b, c, d, e ve f harflerinin sayısal olarak görüntülenmesi istenmiştir.
- 4- Fonksiyon içerisinde oluşturulacak durum sayısı (case) onaltıdır. Bu durumda limitli sayıcının da üst değeri Şekil 5.20’de görüldüğü gibi 16 olarak ayarlanır.



Şekil 5.20. Sayıcı bloğunun üst limitinin ayarlanması.



Şekil 5.21. Yedi parçalı gösterge devre kartının bağlantısı

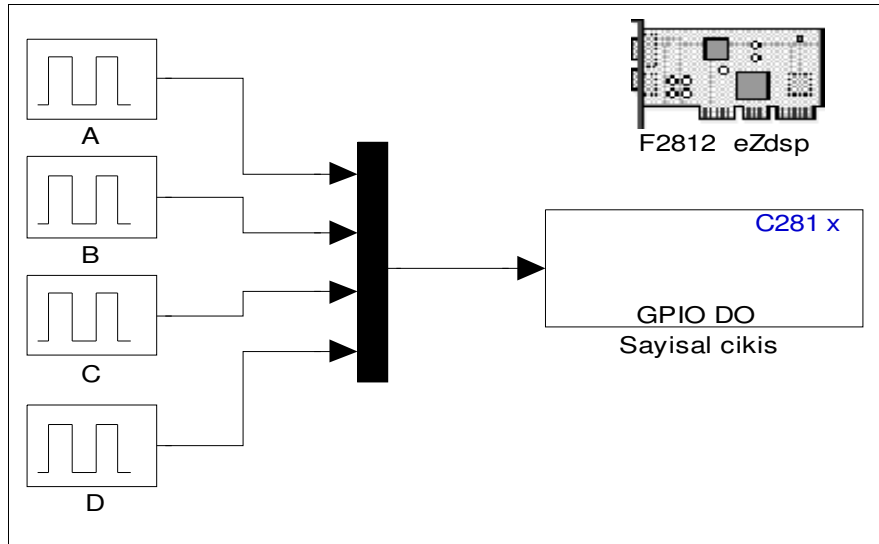
- 5- Devrenin simulink modeli tamamlandıktan sonra Şekil 4.17’de görülen devre kartı Şekil 4.3’teki eZdspF2812 genişletme kartının P8-1 portuna Şekil 5.21’deki gibi bağlanır.
- 6- Her iki devrenin de beslemesi verilir. Yedi parçalı gösterge devre kartı, +5V ile beslenir.

- 7- Matlab simulink penceresi açırken, “Tools” menüsünden “Real Time Workshop” ve “Build Model” seçenekleri takip edilerek, CCS programının otomatik olarak açılması ve çizilen modelin makine diline çevrilerek işlemciye yüklenmesi sağlanır (Şekil 5.8).
- 8- Deney seti üzerinde bulunan “Yedi parçalı gösterge devre kartı” üzerinde yapılan uygulama gözlenir.

5.6. RÖLE DENETİM UYGULAMASI

Röle denetim uygulaması için Şekil 4.13’te açık şeması gösterilen “röle denetim devre kartı” tasarlanmıştır. Devre üzerinde bulunan anahtarın birinci konumunun aktif hale getirilmesiyle kart üzerindeki istenen röle aktif olur. Böylece röle çıkışına bağlı led ile fan çalışmaya başlar. Uygulamaya ait işlem basamakları şu şekildedir:

- 1- Matlab Simulink ortamında yeni bir simulink model penceresi açılır (Şekil 5.2).

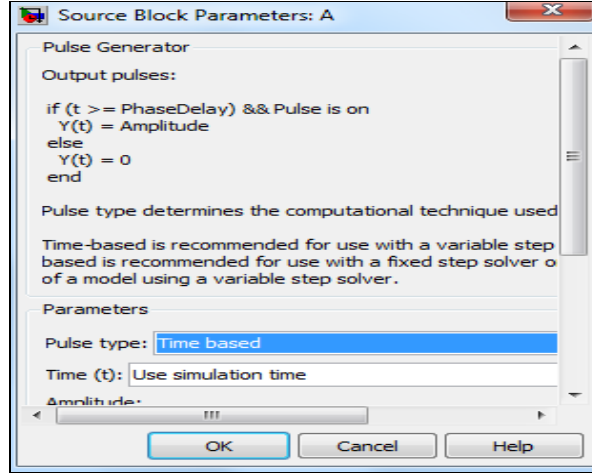


Şekil 5.22. Röle denetim uygulaması.

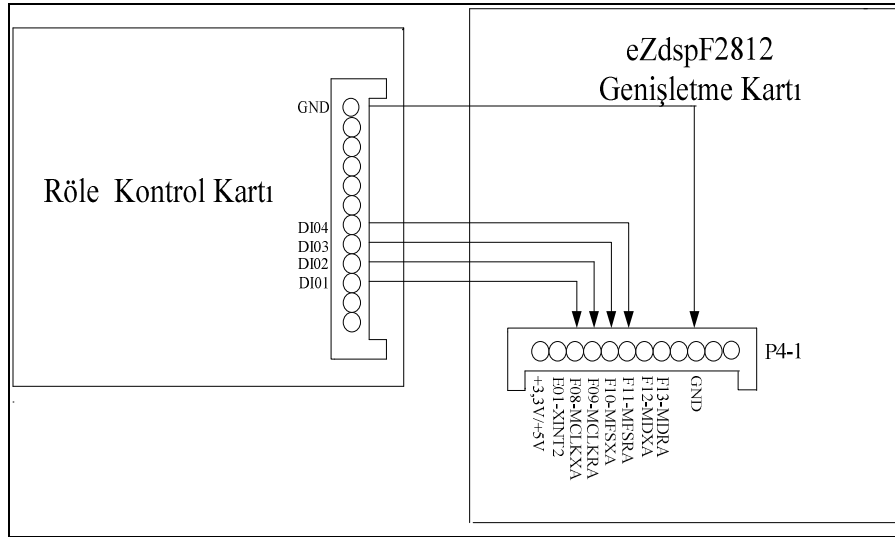
- 2- Açılan pencereye Şekil 5.22’de görülen devre çizilir ve *.mdl. uzantılı olarak kaydedilir.
- 3- Rölelere gidecek bilgi sayısal çıkış bloğuna aktarılırken, bu bloğu

aktifleştiren sinyal, simulink kütüphanesindeki “Sources başlığı” altındaki A, B, C ve D olarak isimlendirilen “Pulse generator” bloğundan alınır.

- 4- Darbe üretici blok parametreleri ayarları da Şekil 5.23'te görüldüğü gibi yapılır.



Şekil 5.23. Darbe üretici blok parametreleri.



Şekil 5.24. Röle denetim devre kartının bağlantısı

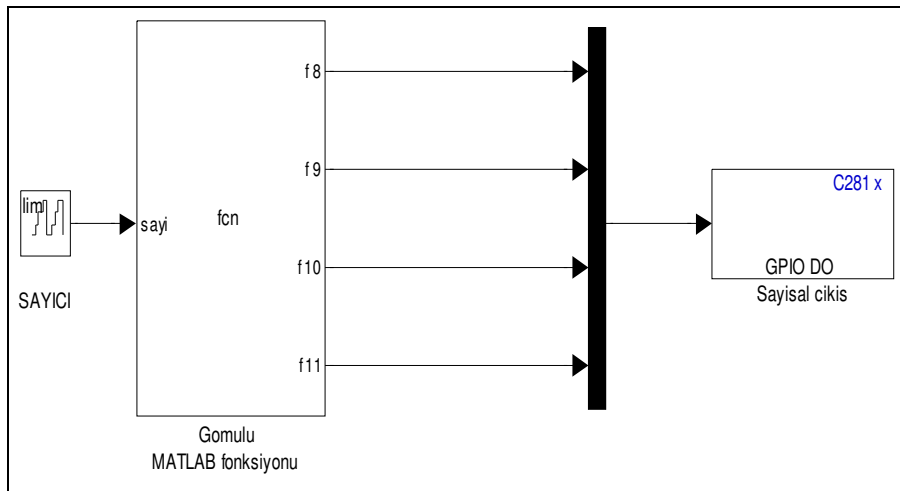
- 5- Devrenin simulink modeli tamamlandıktan sonra Şekil 4.12'de açık şeması bulunan devre kartı Şekil 4.3'teki eZdspF2812 genişletme kartının P4-1 portuna Şekil 5.24'teki gibi bağlanır.

- 6- Her iki devrenin de beslemesi verilir. Röle denetim devre kartı, +5V ile beslenir.
- 7- Matlab simulink penceresi açıkken, “Tools” menüsünden “Real Time Workshop” ve “Build Model” ” seçenekleri takip edilerek, CCS programının otomatik olarak açılması ve çizilen modelin makine diline çevrilerek işlemciye yüklenmesi sağlanır (Şekil 5.8).
- 8- Deney seti üzerinde bulunan “Röle denetim devre kartı ” üzerinde yapılan uygulama gözlenir.

5.7. ADIM MOTOR DENETİM UYGULAMASI

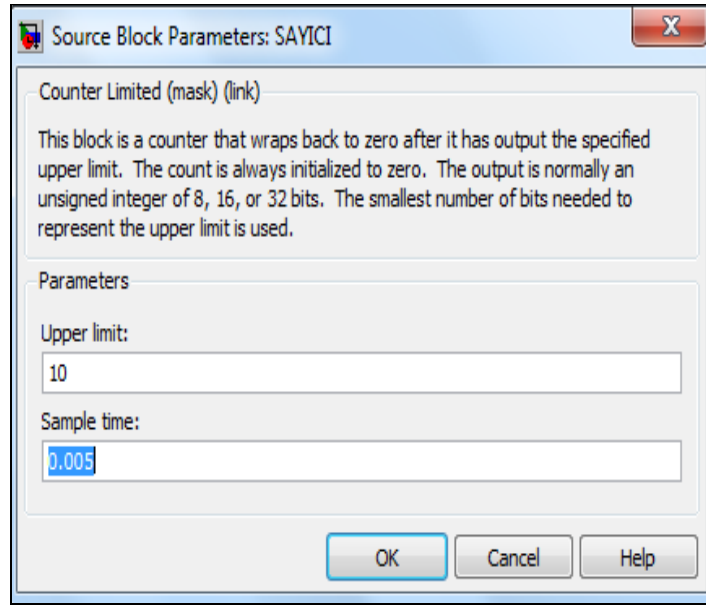
Adım motor bağlantısı, röle denetim kartı üzerine yerleştirilmiştir. Kart üzerinde bulunan anahtar 2 konumuna alındığında röleler devreden çıkar ve devrede bulunan tek kutuplu (unipolar) bağlantı yapısını sahip adım motoru, transistörler aracılığıyla sürülür. Uygulamaya ait işlem basamakları şu şekildedir:

- 1- Matlab Simulink ortamında yeni bir simulink model penceresi açılır (Şekil 5.2).
- 2- Açılan pencereye Şekil 5.25’de görülen devre çizilir ve *.mdl. uzantılı olarak kaydedilir.



Şekil 5.25. Adım motor denetim uygulaması.

- 3- Devrenin matlab simulink modeli, içerisinde 4 durum değişkeni barındıran gömülü matlab fonksiyonundan oluşur. Devrede tetikleme sinyali yine limitli sayıcı tarafından gerçekleştirilirken, motorun çalışma bilgisi “Sayısal çıkış” bloğuyla motora aktarılır.
- 4- Adım motor iki fazlı olduğundan motor saat yönünde döndürülmek istenirse gömülü matlab fonksiyonu içerisine, onluk tabanda 10, 9, 5 ve 6 değerleri; saat yönünün tersine döndürülmek istendiğinde onluk tabanda 6, 5, 9 ve 10 değerlerinin binary sayı karşılıkları motor girişlerine gönderilir.
- 5- Modelde bulunan “Sayıcı” blok parametrelerindeki örnekleme zamanı (sample time) ayarlanarak motorun dönüş hızı değiştirilir (Şekil 5.26).



Şekil 5.26. Sayıcı bloğunda örnekleme zamanı ayarının yapılması.

- 6- Devrenin simulink modeli tamamlandıktan sonra Şekil 4.13’de görülen devre kartı Şekil 4.3’teki eZdspF2812 genişletme kartına Şekil 5.24’deki gibi bağlanır.
- 7- Her iki devrenin de beslemesi verilir. Röle denetim devre kartı, +5V ile beslenir.
- 8- Matlab simulink penceresi açıkken, “Tools” menüsünden “Real Time Workshop” ve “Build Model” seçenekleri takip edilerek, CCS programının

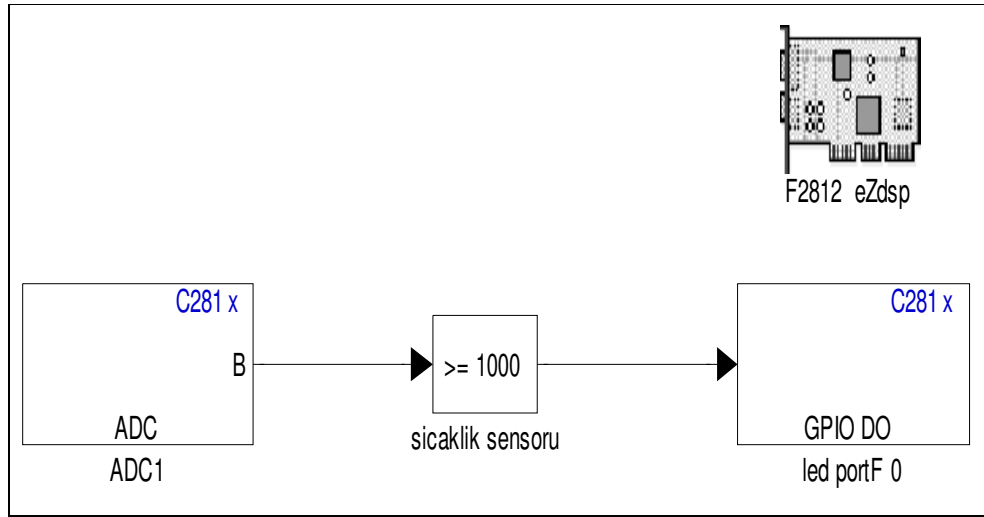
otomatik olarak açılması ve çizilen modelin makine diline çevrilerek işlemciye yüklenmesi sağlanır (Şekil 5.8).

- 9- Deney seti üzerinde bulunan “Röle denetim devre kartı ” üzerinde motorun dönme şekilleri gözlenir.

5.8. ADC SICAKLIK SENSÖRÜ UYGULAMASI

TMS320F2812 işlemcisi, 12-bit çözünürlüklü 16 adet (2x8) 0-3V DA analog girişli bir ADC’ ye sahiptir. Bu ADC özelliklerini tanıtmak için gerçekleştirilen deney seti üzerinde Şekil 4.15’te açık şeması gösterilen “analog giriş devre kartı” tasarlanmıştır. Burada yapılmak istenen sıcaklık sensöründen alınan analog bilgilerin sayısal bilgiye çevrilmesini gözlemlemektir. Uygulamaya ait işlem basamakları şu şekildedir:

- 1- Matlab Simulink ortamında yeni bir simulink model penceresi açılır (Şekil 5.2).
- 2- Açılan pencereye Şekil 5.27’de görülen devre çizilir ve *.mdl. uzantılı olarak kaydedilir.

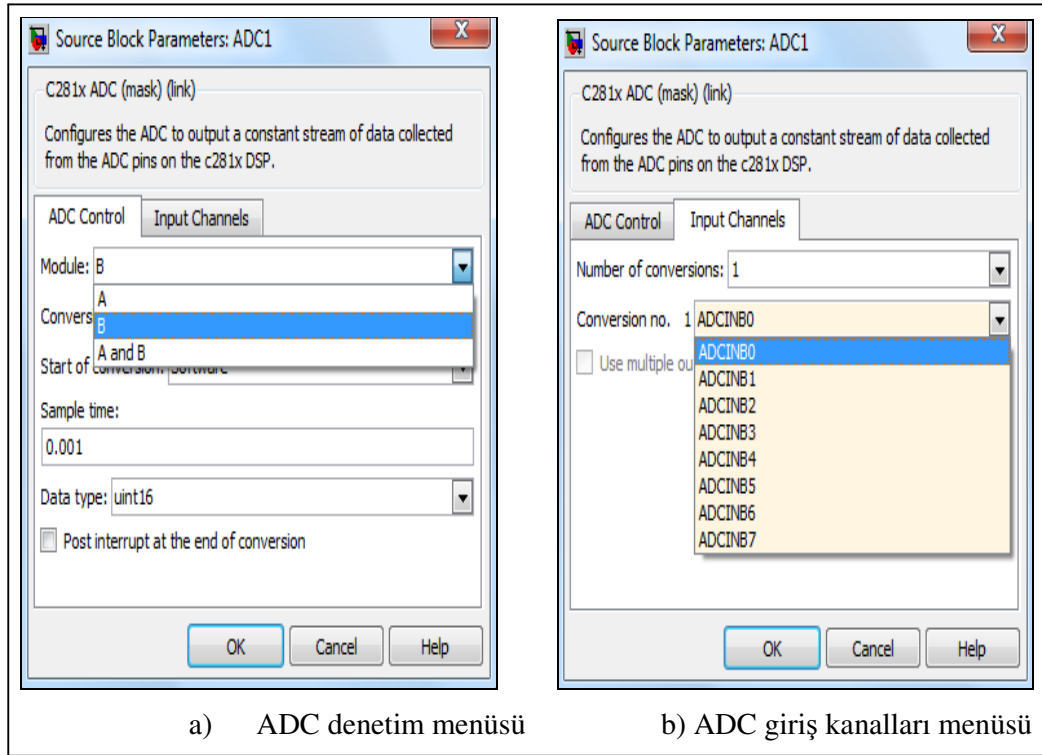


Şekil 5.27. ADC sıcaklık sensörü uygulaması.

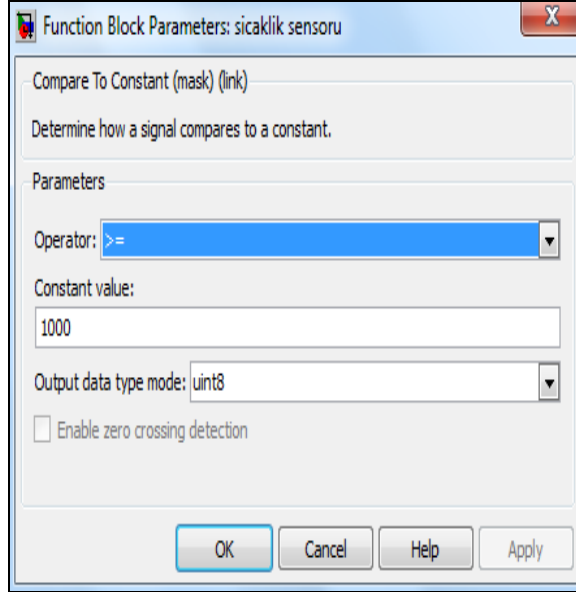
- 3- Modelde bulunan sıcaklık sensörü ADC bloğunun B kanalına bağlanır ve giriş için ADCINB0 kanalı seçilir (Şekil 5.28).
- 4- Sıcaklık sensörü olarak adlandırılan blok ise bir “Karşılaştırmalı işlemci”

bloğudur. Bu blok simulink kütüphanesinde, “Logic&bit operations” başlığı altında bulunan “compare to constant” bloğudur. Bu blok için “>=” işlemini seçilirken sabit değeri 1000 yapılır.

- 5- Sensörden gelen analog sıcaklık bilgisi sabit değer olarak girilen 1000 ($V=3/4095 * 1000= 0.732V$) değerini aştığı anda karşılaştırıcının çıkışı sayısal 1 değeri üretilir ve bu değer sayısal çıkış bloğuyla lede gönderilir. Şekil 5.29’da bu bloğa ait parametreler görülmektedir.

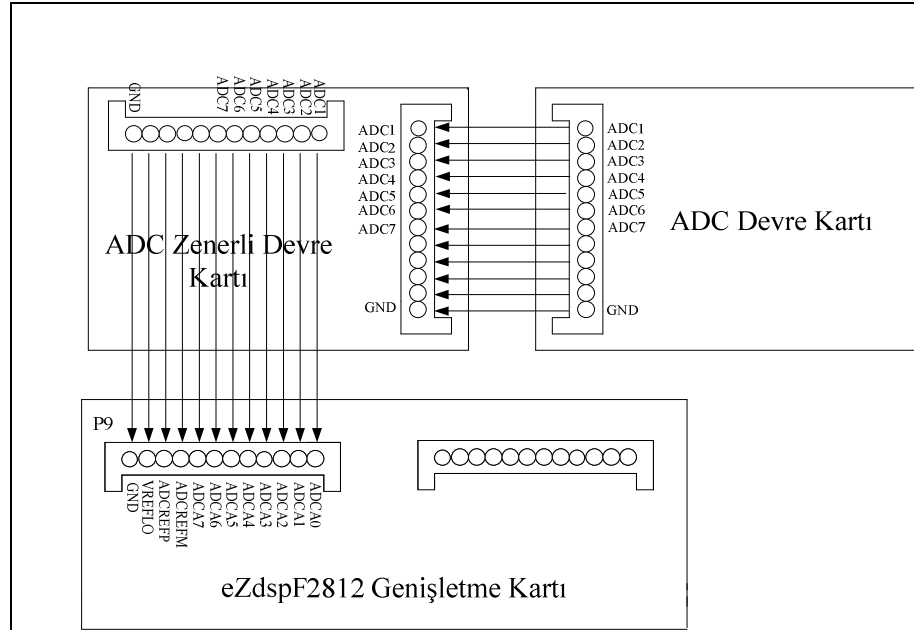


Şekil 5.28. ADC sıcaklık sensörü uygulaması için blok parametreleri ayarları.



Şekil 5.29. ADC sıcaklık sensörü blok parametreleri.

- 6- Devrenin simulink modeli tamamlandıktan sonra Şekil 4.12’de açık şeması bulunan devre kartı Şekil 4.3’teki eZdspF2812 genişletme kartına Şekil 5.30’daki gibi bağlanır.



Şekil 5.30. ADC devre kartının bağlantısı.

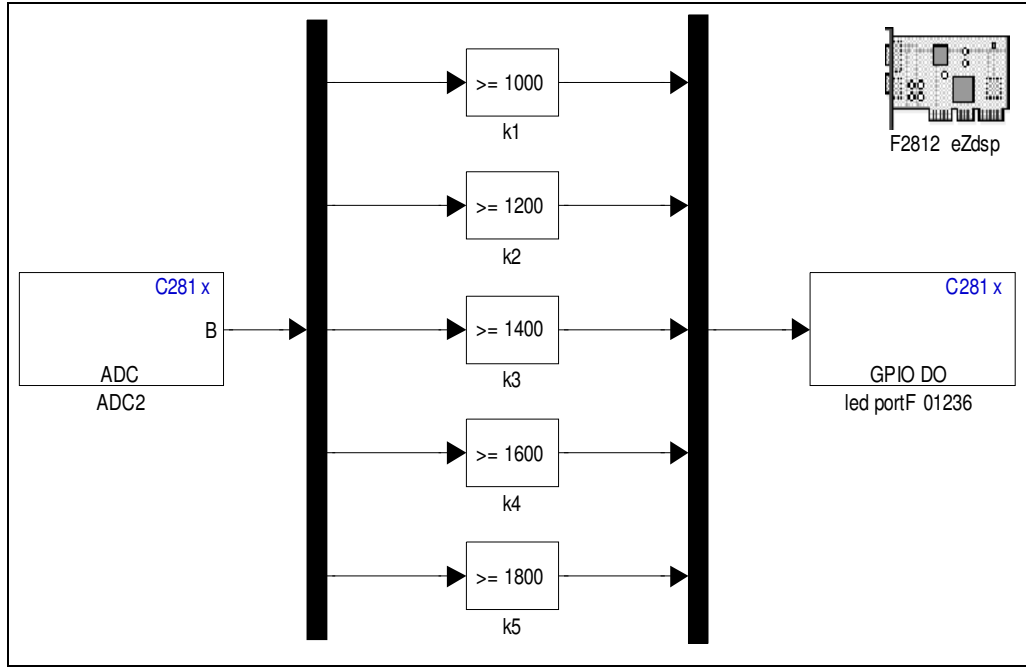
- 7- Buradaki “ADC zenerli devre kartı” olarak adlandırılan kart her kanal için birer tane olmak üzere 3V’luk 7 adet zenerden oluşmaktadır. Burada amaç Sİİ’yi yüksek gerilimlerden korumaktır.
- 8- Sıcaklık sensöründen alınan bilginin dışarıdan gözlenebilmesi için sayısal çıkış kartındaki ledlerden yararlanılabilir. Bunun için Şekil 5.6’daki bağlantı, Şekil 5.30’daki bağlantıya ek olarak gerçekleştirilir.
- 9- Sisteme bağlanan devrelerin ve eZdsp geliştirme kartının beslemeleri verilir.
- 10- Matlab simulink penceresi açıkken, “Tools” menüsünden “Real Time Workshop” ve “Build Model” seçenekleri takip edilerek, CCS programının otomatik olarak açılması ve çizilen modelin makine diline çevrilerek işleme yüklenmesi sağlanır (Şekil 5.8).
- 11- Deney seti üzerinde bulunan sayısal çıkış devre kartı üzerindeki F0 pinine bağlı led üzerinden sensörün aldığı analog değerın sayısal karşılığı gözlemlenir.

5.9. POTANSİYOMETRELİ ADC UYGULAMASI

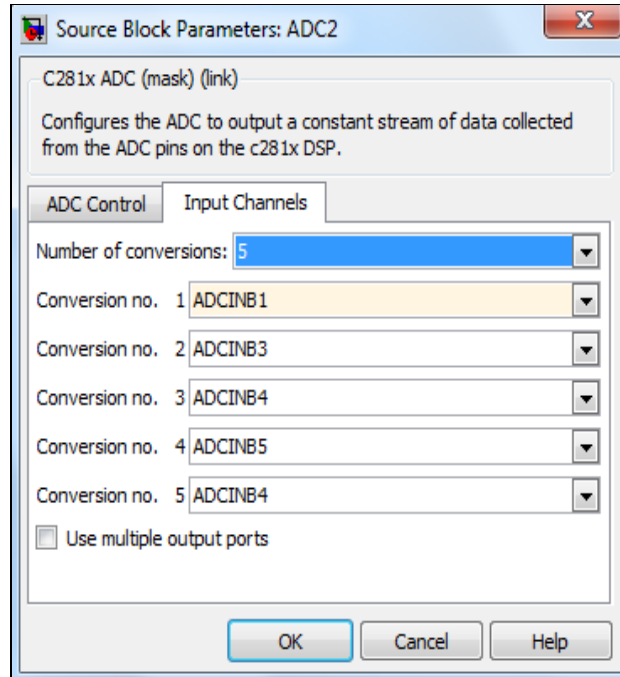
Potansiyometrelı ADC uygulaması, ADC sıcaklık sensörü uygulamasında kullanılan devre kartı üzerinden gerçekleştirilmektedir. Bu uygulamada analog veriler, devre kartı üzerindeki 5 adet trimpottan alınır ve yine çıkış bilgisi sayısal çıkış blođuyla sayısal çıkış kartı üzerindeki ledlere aktarılır. Uygulamaya ait işlem basamakları řu şekildedir:

- 1- Matlab Simulink ortamında yeni bir simulink model penceresi açılır (Şekil 5.2.).
- 2- Açılan pencereye Şekil 5.31’de görülen devre çizilir ve *.mdl. uzantılı olarak kaydedilir.
- 3- ADC blođunun giriş sayısı 5’e çıkarılmış ve giriş kanalları Şekil 5.32’de görüldüđü gibi ayarlanmıştır. Uygulamadaki her bir trimpottan alınan analog deđer 1000, 1200, 1400, 1600, 1800 deđerleri ile kademeli olarak karşılaştırılmaktadır. Her bir karşılaştırıcının çıkışı F portuna bağlanan ledleri sürmektedir.
- 4- Sİİ’ye gönderilmekte ve bu deđer 1500 sabit deđerini geçtiđi anda da sayısal

değere çevrilmektedir.



Şekil 5.31. Potansiyometreli ADC uygulaması.



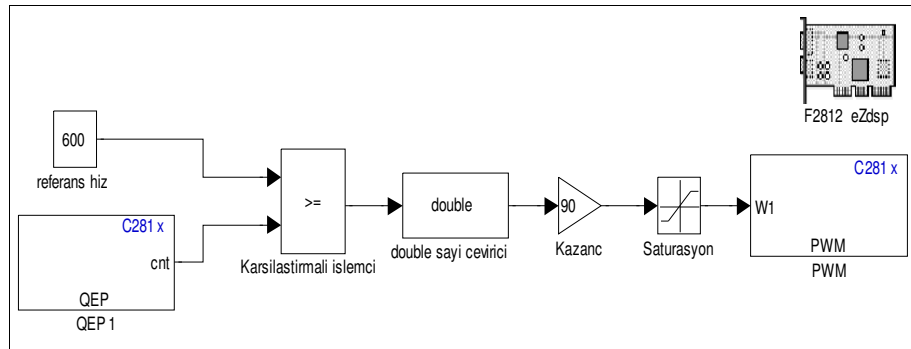
Şekil 5.32. Uygulamasıda ayarlanan ADC blok parametreleri.

- 5- Devrenin simulink modeli tamamlandıktan sonra Şekil 4.14'te açık şeması bulunan devre kartı Şekil 4.3'teki eZdspF2812 genişletme kartına Şekil 5.30'daki gibi bağlanır.
- 6- Devredeki sayısal değişiklikler Sayısal çıkış kartı üzerinde yer alan üzerinde yer alan F0, F1, F1, F2 ve F3 pinlerine bağlı ledler tarafından gözlemlenir. Bunun için Şekil 5.6' Şekil 5.30'daki bağlantılar gerçekleştirilir.
- 7- Sisteme bağlanan devrelerin ve eZdsp geliştirme kartının beslemeleri verilir.
- 8- Matlab simulink penceresi açıkken, "Tools" menüsünden "Real Time Workshop" ve "Build Model" seçenekleri takip edilerek, CCS programının otomatik olarak açılması ve çizilen modelin makine diline çevrilerek işlemciye yüklenmesi sağlanır (Şekil 5.8).
- 9- Deney seti üzerinde bulunan "Sayısal çıkış devre kartı" üzerindeki F0, F1, F2, F3 ve F6 pinlerine bağlı ledler üzerinden, trimpotlardan alınan analog değerın sayısal karşılığı gözlemlenir.

5.10. DA MOTOR HIZININ AÇ-KAPA İLE DENETİM UYGULAMASI

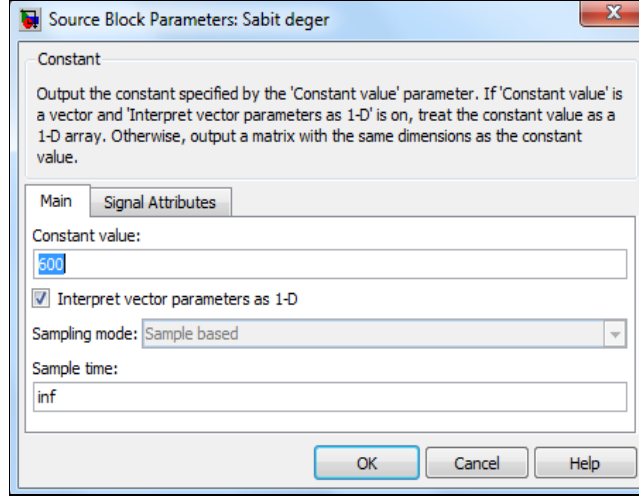
Çalışmada, 12V çalışma gerilimine sahip 2000 devir/dakika özelliğinde bir DA motor kullanılmıştır. Motora ait olan encoder ise 512 devir/pals çözünürlüğe sahip bir encoderdir. Dolayısıyla frekans cevabı da 17,007 kHz' dir. Uygulamaya ait işlem basamakları şu şekildedir:

- 1- Matlab Simulink ortamında yeni bir simulink model penceresi açılır (Şekil 5.2).



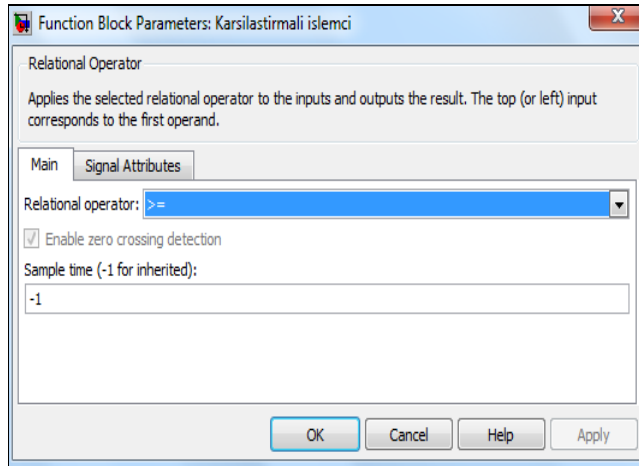
Şekil 5.33. DA motor hızının aç-kapa ile denetim uygulaması.

- 2- Açılan pencereye Şekil 5.33'te görülen devre çizilir ve *.mdl. uzantılı olarak kaydedilir.

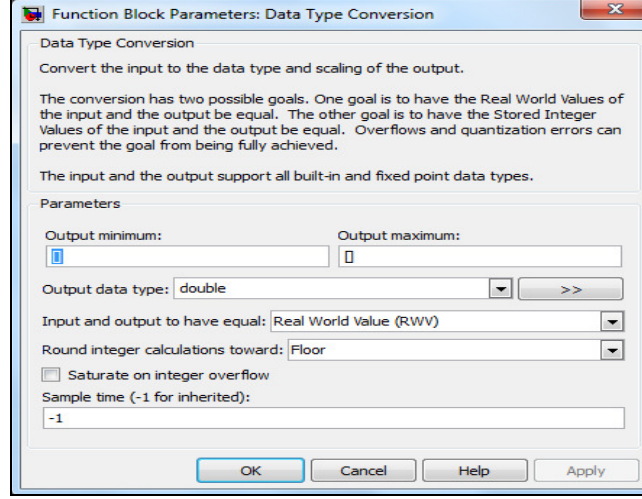


Şekil 5.34. Referans hız blok parametreleri.

- 3- Referans hız olarak değiştirilen “Constant” blok parametresindeki değer motorda sabitlenmesini istediğimiz dev./dak. değeridir. Sabit değer bloğu Şekil 5.34'de görüldüğü gibi ayarlanabilmektedir.
- 4- Karşılaştırmalı işlemcide ilişkisel operatör ve örnekleme zaman ayarları yapıldıktan sonra, karşılaştırma işlemi yapılır (Şekil 5.35).

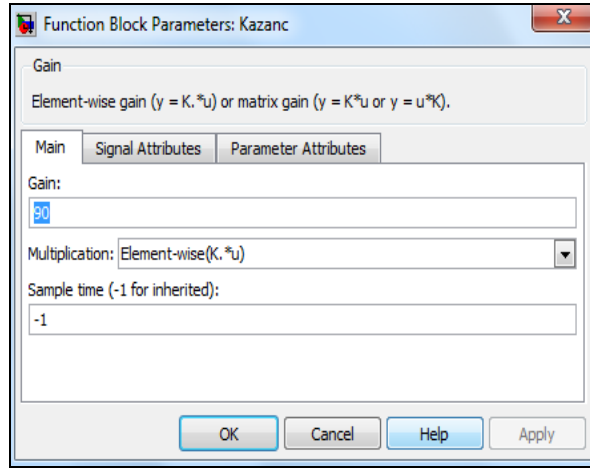


Şekil 5.35. Karşılaştırmalı kaydedici blok parametreleri.



Şekil 5.36. Double sayı çevirici blok parametreleri.

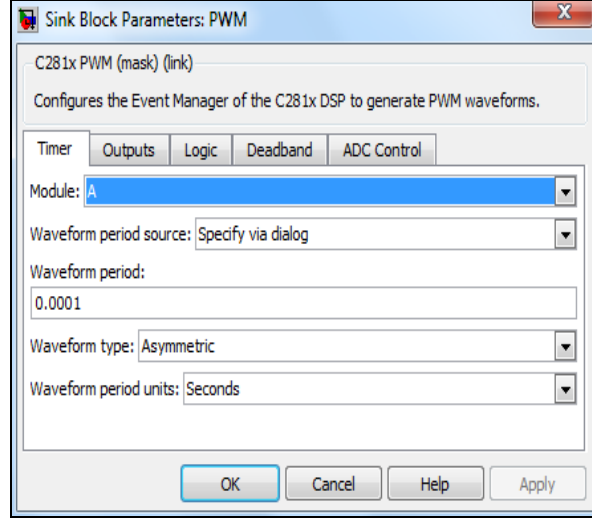
- 5- Bloğa bağlanan ilk değer, bağlanan ikinci değerden büyük ise karşılaştırma işleminin sonucu boolean bir ifade olan “true” şeklindedir. Yani modeldeki sabit değer QEP değerinden büyük olduğunda sonuç “true” olur. Bu değer modelde görülen “double sayı çevirici” diye adlandırılan blok ile Şekil 5.36’daki gibi double olarak ayarlanarak, boolean ifade double sayı karakterine dönüştürülür.



Şekil 5.37. Kazanç bloğu parametreleri

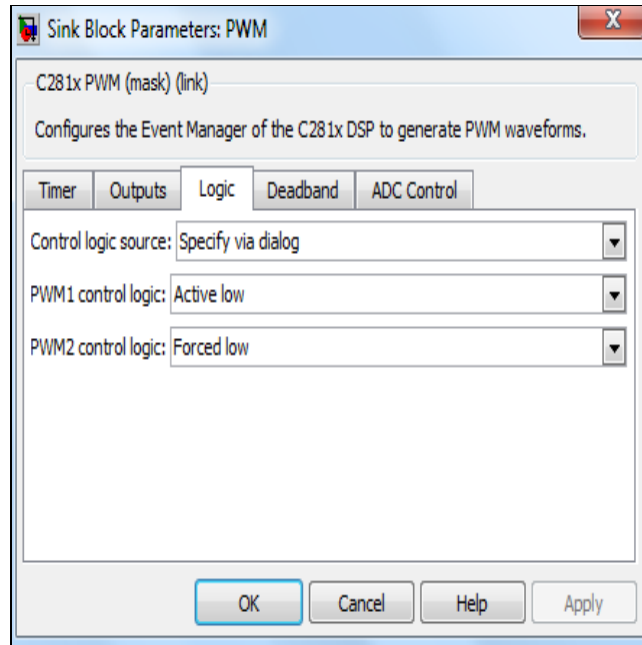
- 6- İstenen şekilde double sayı tipine gönüştürülen değer “kazanç” bloğuna gelir. Burada kazanç değeri ve örnekleme zamanı ayarlanabilen sayı bir sonraki

PWM bloğunda oluşan PWM sinyalinin motora uygulanma oranını vermektedir. Şekil 5.37’de kazanç bloğu parametreleri görülmektedir.

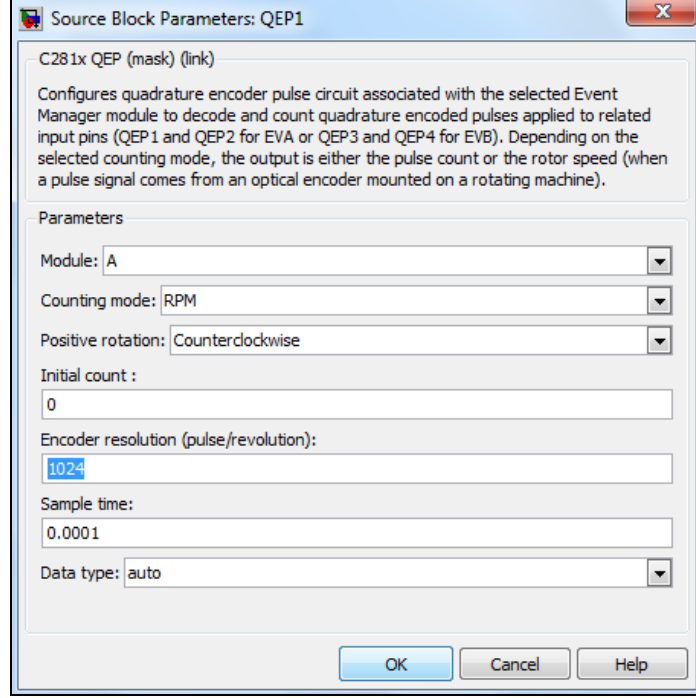


Şekil 5.38. PWM blok parametreleri zamanlayıcı menüsü ayarları.

7- t_{on} ve t_{off} sürelerinin değişimiyle motorun hızını değiştirebilen PWM sinyaline ait blok parametreleri ayarları Şekil 5.38. ve 5.39’da verilmiştir.

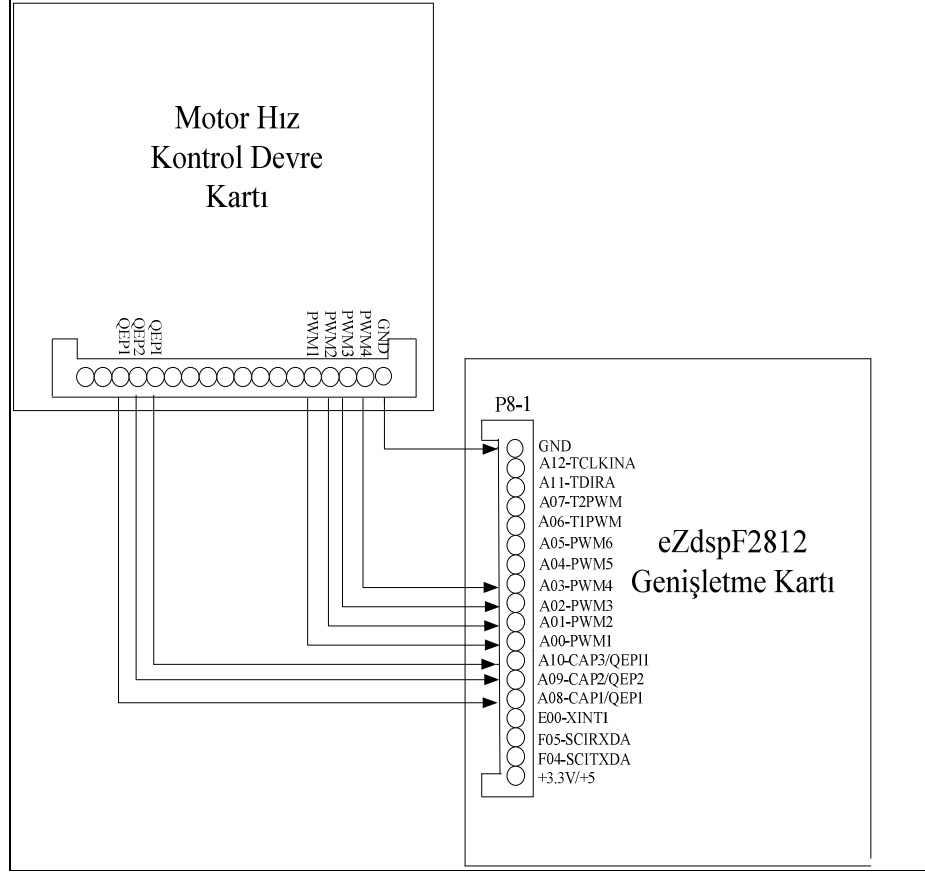


Şekil 5.39. PWM blok parametreleri lojik menüsü ayarları.



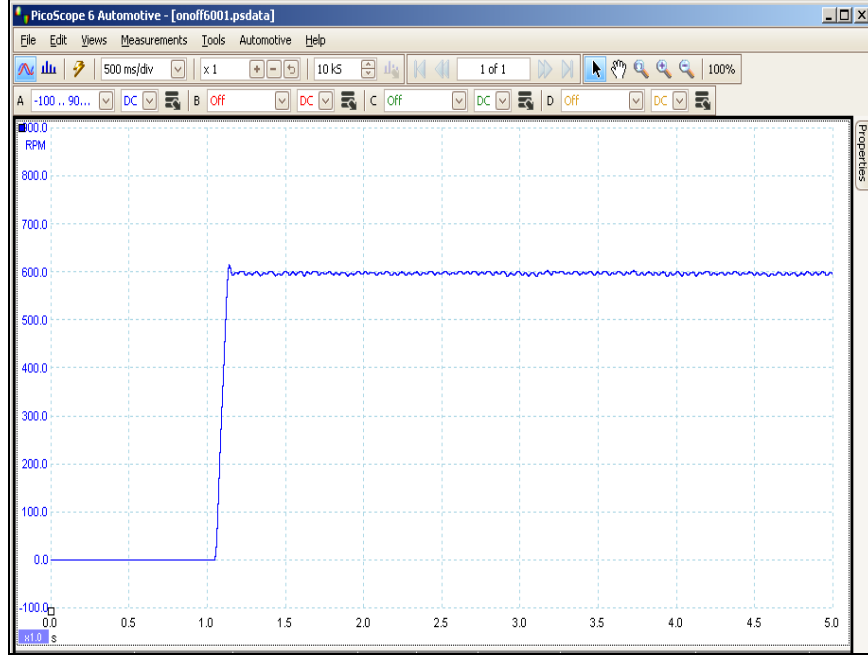
Şekil 5.40. QEP blok parametreleri.

- 8- DA motora bağlı yük değişimleri motorun hızında değişiklik meydana getirir ve bu değişimler QEP tarafından algılanır. QEP blok parametre ayarları ise Şekil 5.40'daki gibi ayarlanır.
- 9- Devrenin simulink modeli tamamlandıktan sonra Şekil 4.4'de görülen devre kartı Şekil 4.3'teki eZdspF2812 genişletme kartına Şekil 5.41'deki gibi bağlanır.
- 10- Sisteme bağlanan devrelerin ve eZdsp geliştirme kartının beslemeleri verilir.
- 11- Matlab simulink penceresi açıkken, "Tools" menüsünden "Real Time Workshop" ve "Build Model" seçenekleri takip edilerek, CCS programının otomatik olarak açılması ve çizilen modelin makine diline çevrilerek işleme yüklenmesi sağlanır (Şekil 5.8).
- 12- Deney seti üzerinde bulunan "motor hız denetim devre kartı" üzerinde bulunan DA motor üzerinden gözlemlenir.



Şekil 5.41. Motor hız denetim devre kartının bağlantısı.

13- DA motorda istenen sabit hız değerinin elde edilip edilmediğini görmek için motor çıkışına bilgisayar tabanlı osilaskop bağlanmıştır. Bilgisayar tabanlı osilaskoptan elde edilen DA motor hızının aç-kapa denetimli gerçekleştirilmesi Şekil 5.42’de görülmektedir. Burada kullanılan“Pico 4323” model picoscope 12 bit çözünürlüklü, 10MHz analog bant genişliğine sahip, 4 kanallı bir picoscope dur [25]. Bu picoscope motor hızındaki değişimi izlemek amacıyla kullanılmıştır. Ancak, deney setiyle birlikte verilmemektedir. Motor hızı, “Extech 461501”.model takometre aracılığıyla da ölçülmektedir. Bu takometre RS232 bağlantısı ile bilgisayara bağlanabilmekte ve 100000 devire kadar ölçüm yapabilmektedir [26].

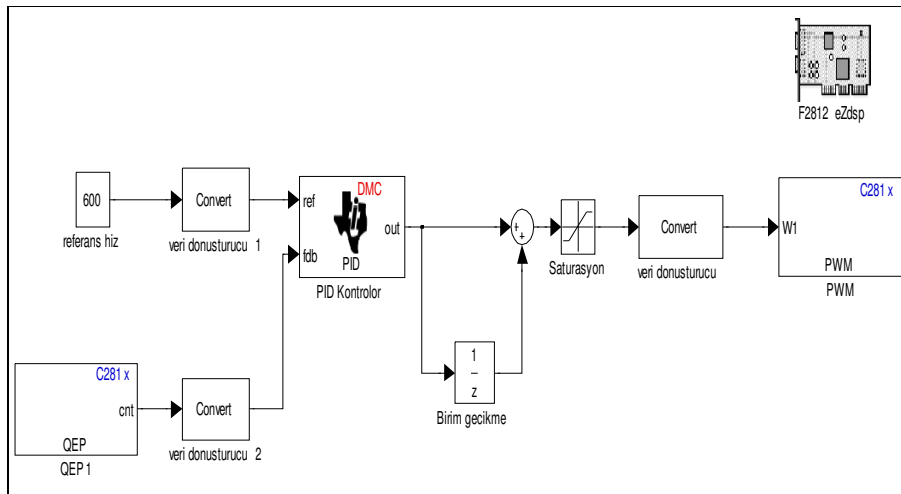


Şekil 5.42. DA motorunun aç-kapa hız denetimi.

5.11. DA MOTOR HIZININ PID İLE DENETİM UYGULAMASI

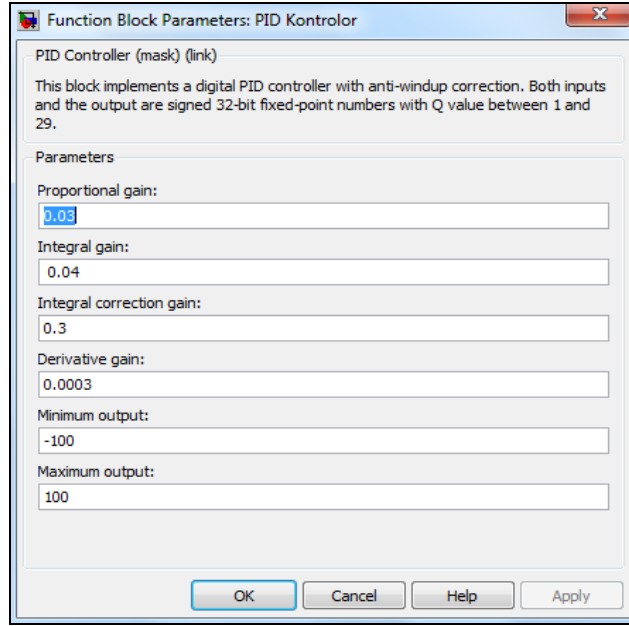
Uygulamaya ait işlem basamakları şu şekildedir:

- 1- Matlab Simulink ortamında yeni bir simulink model penceresi açılır (Şekil 5.2).



Şekil 5.43. Matlab simulink ortamında gerçekleştirilen sistem modeli.

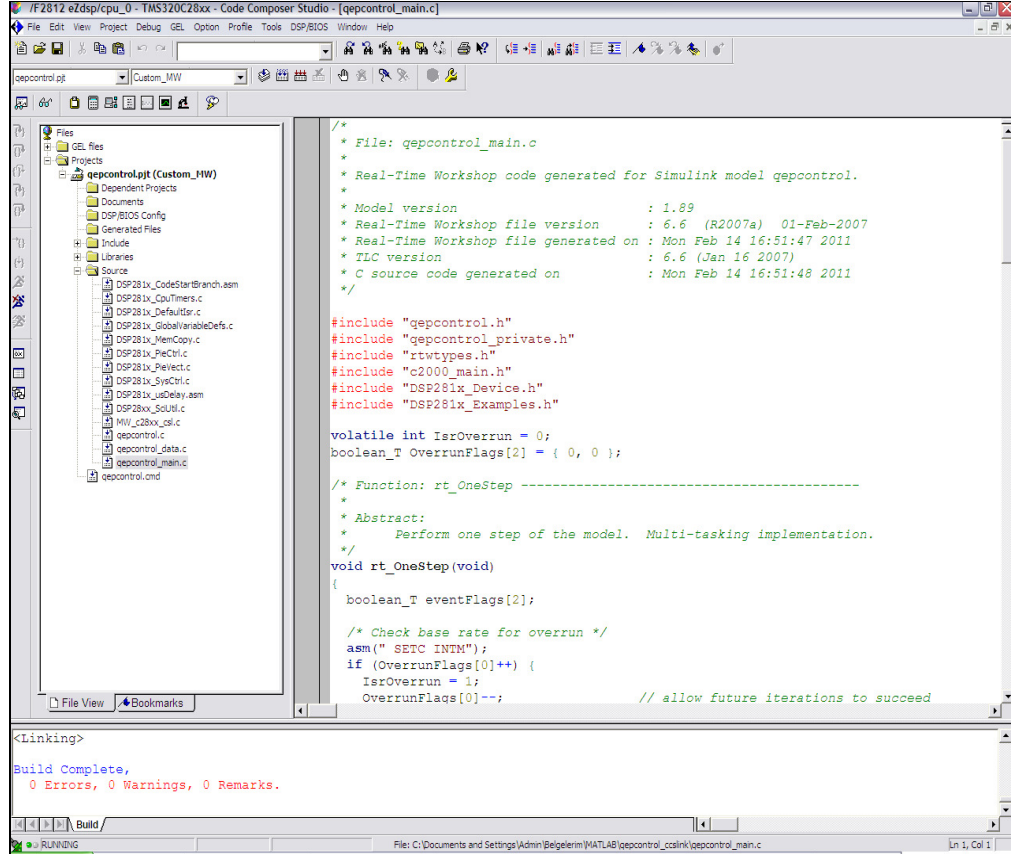
- 2- Açılan pencereye Şekil 5.43’de görülen devre çizilir ve *.mdl. uzantılı olarak kaydedilir.
- 3- DA motor hızının PID denetimiyle gerçekleştirilmesi uygulamasında motordan alınmak istenen sabit hız değeri aç-kapa denetim uygulamasındaki değerle aynıdır. Değer 600 olarak ayarlanır. Böylece her iki denetim yönteminden elde edilen çıktılar alınarak karşılaştırılmaları yapılabilmektedir.
- 4- Uygulamanın matlab simulink modeli oluşturulurken kullanılan tek farklı blok “PID kontrolör” bloğudur. PID blok parametreleri Şekil 5.44’deki gibidir.



Şekil 5.44. PID kontrolör blok parametreleri.

- 5- Yapılan uygulamada, DA motoruna ait performans eğrisi üzerinden yapılan ölçümler sonucunda PID parametreleri 0.001s aralıklarla belirlenen kriterler sağlanıncaya kadar artırılmıştır. Hesaplanan kazanç parametreleri ile sistemin aşısız, oturma ve yükselme zamanı iyi olan bir çıkış elde edildiği görülmüştür. Hesaplanan bu parametre değerleri sisteme uygulanmıştır ve sistem dinamiklerinde meydana gelen değişimlere karşılık kazanç parametrelerinin sistem için etkileri dikkate alınarak parametreler değiştirilmiştir.

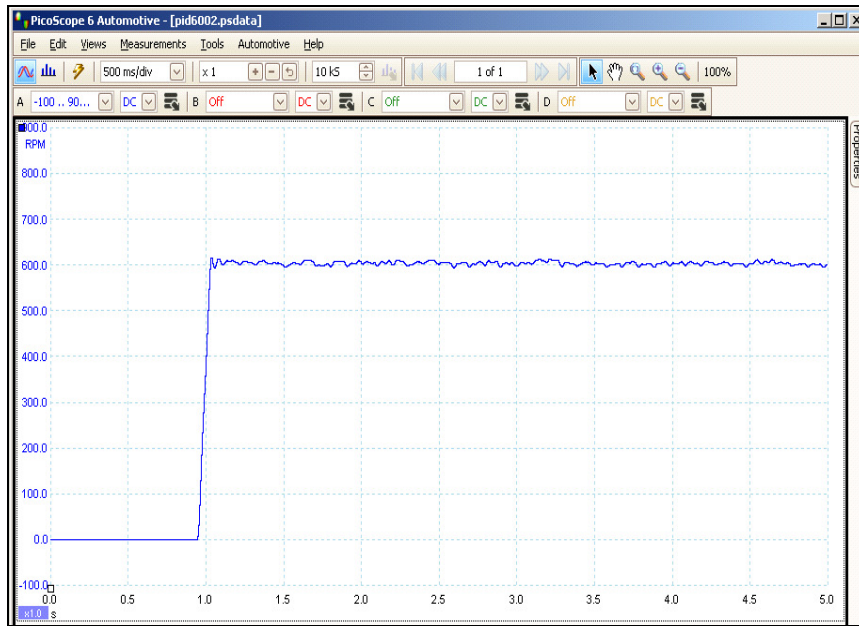
- 6- Devrenin simulink modeli tamamlandıktan sonra Şekil 4.4’de görülen devre kartı Şekil 4.3’teki eZdspF2812 genişletme kartına Şekil 5.41’deki gibi bağlanır.
- 7- Sisteme bağlanan devrelerin ve eZdsp geliştirme kartının beslemeleri verilir.



Şekil 4.45. CCS ekranında program parametrelerinin değiştirilmesi.

- 8- Matlab simulink penceresi açıkken, “Tools” menüsünden “Real Time Workshop” ve “Build Model” seçenekleri takip edilerek, CCS programının otomatik olarak açılması ve çizilen modelin makine diline çevrilerek işlemciye yüklenmesi sağlanır (Şekil 5.8).
- 9- Matlab Simulink ortamındaki blokların yerleştirilmesiyle ve bu bloklarla ilgili parametre ayarlarının değiştirilmesiyle gerçekleştirilen durum değişiklikleri CCS ortamında da yapılabilmektedir. CCS’nin “proje yöneticisi” penceresinde yer alan “Projects” dosyasından değişiklik yapılmak istenen program seçilir.

- 10- *.pjt uzantılı olan bu program içerisindeki dosyalara ulaşılır. Bu dosyalar Dependent projects, Documents, DSP/Bios config, Generated files, Include, Libraries ve Sources adlı dosyalardır.
- 11- “Source” dosyası, programla ilgili değişiklikleri yapabileceğimiz “*_main.c” uzantılı programın bulunduğu dosyadır. Değişikliklerin yapıldığı ekran Şekil 4.5’de görülmektedir.
- 12- DA motor hızının PID denetimi ile gerçekleştirilmesi picoscope ile bilgisayar ekranında da görülebilmektedir (Şekil 5.46).



Şekil 5.46. DA motoru PID hız denetimi.

BÖLÜM 6

SONUÇ VE ÖNERİLER

Sayısal işaret işlemciler yüksek hız ve doğruluklu işlem yapabilme özelliklerinden dolayı kontrol uygulamalarında gün geçtikçe önemi artan elemanlar olmuştur. Ayrıca, etkileşimli bir benzetim ve prototipleme ortamına sahip olan Simulink ve CCS programlarıyla da etkileşimli olduğundan herhangi bir kod yazımına gerek kalmadan hazır simulink bloklarıyla yapılmak istenen uygulama oldukça kolay ve pratik hale gelmiştir.

Bu çalışmada 32 bitlik TMS320F2812 mikroişlemcisi ile sayısal giriş/çıkış, analog giriş, seri iletişim, PWM, QEP, Motor hız denetimi, adım motoru, röle, sıcaklık sensörü ve 7 parçalı ekran uygulamaları, için uygulama geliştirme ortamı tasarlanmış ve gerçekleştirilmiştir.

Çalışmada F2812 eZdsp geliştirme kiti kullanılarak bir deney seti gerçekleştirilmiştir. Deney seti üzerinde geliştirme kartının kullanımını kolaylaştıran bir genişletme devresi bulunmaktadır. Bu devreyle işlemcinin tüm pinlerine rahatlıkla ulaşılabilmektedir. Gerçekleştirilen sistem üzerinde temel ve ileri seviyedeki uygulamalar rahatlıkla gerçekleştirilebilmektedir. Deney seti üzerinde bulunan devrelerle, işlemcinin PWM, QEP, analog giriş, sayısal giriş- çıkış, seri iletişim özelliklerinden yararlanılarak yapılan uygulamalar mevcuttur.

Gerçekleştirilen deney setiyle, teknik ve mühendislik eğitimi veren fakülte ve yükseköğretim kurumlarının müfredatında yer alan sayısal işaret işleme dersinin teorik anlatımına uygulamalı eğitim desteği sağlanmaktadır. Bu çalışmada gerçekleştirilen sistem ile öğretimin kolay ve etkili bir şekilde gerçekleştirilmesinde önemli bir materyal eksikliği giderilmektedir. Deney seti geliştirilebilir yapıda olup istenilmesi durumunda yeni uygulamaların eklenmesi mümkündür.

KAYNAKLAR

1. Avunduk, D., "TMS320c5x işlemcisi ile uygulama geliştirme", Lisans tezi, *Karabük Üniversitesi Teknik Eğitim Fakültesi*, Karabük, 1-47 (2008).
2. Yıldırım, A., "TMS320F2812 DSP işlemcisi ile uygulama geliştirme", Lisans tezi, *Karabük Üniversitesi Teknik Eğitim Fakültesi*, Karabük, 1-38 (2009).
3. İnternet: EMO "Sayısal işaret işleme (DSP), bir DSP mimarisinin incelenmesi ve uygulamaları" [http://www.emo.org.tr/ekler/ c3d69ed781b16bcek.pdf?dergi=201](http://www.emo.org.tr/ekler/c3d69ed781b16bcek.pdf?dergi=201) (2009).
4. İnternet: UBM Company "Electronic Engineering News" <http://www.eetimes.com/electronics-news/4196866/Databeans-releases-2006-DSP-market-study> (2011).
5. Edizkan, R., "Eskişehir Osmangazi Üniversitesi'nde DSP donanım eğitimi", Elektrik Elektronik Bilgisayar Biyomedikal Mühendislikleri Eğitimi IV. Ulusal Sempozyumu, *Eskişehir Osmangazi Üniversitesi Kongre ve Kültür Merkezi*, Eskişehir, 23-34 (2009).
6. Ertürk S., "Kocaeli Üniversitesi, elektronik ve haberleşme müh. bölümünde okutulan işaret işleme dersinin değerlendirmesi", *11. Sinyal İşleme ve İletişim Uygulamaları Kurultayı*, İstanbul,12-17 (2003).
7. Kuntalp, D. ve Kalaycı, G., "Proje tabanlı eğitime bir örnek: sayısal işaret işleme", Elektrik Elektronik Bilgisayar Biyomedikal Mühendislikleri Eğitimi IV. Ulusal Sempozyumu, *Eskişehir Osmangazi Üniversitesi Kongre ve Kültür Merkezi*, Eskişehir, 22-45 (2009).
8. İnternet: YÖK "Yök Ulusal Tez Arama Merkezi" <http://tez2.yok.gov.tr/>(2011).
9. Karamustafaoğlu, O., "AÜ. fen ve teknoloji öğretmenlerinin öğretim materyallerini kullanma düzeyleri: Amasya ili örneği", *Bayburt Eğitim Fakültesi Dergisi*, Bayburt, 1: 90-101 (2006).
10. Nyquist, H., "Certain topics in telegraph transmission theory", *Trans. AIEE*, 47: 617-644 (1982).
11. Kesler, S., "Elektrik makineleri denetim sistemlerinde TMS320F2812 DSP kullanımı", Akademik Bilişim, *Çanakkale Onsekiz Mart Üniversitesi*, Çanakkale, 12-22 (2008).

12. Doğan, M., “TMS320 DSP işlemci ile real time uygulama geliştirme”, Yüksek lisans tezi, *Sakarya Üniversitesi Fen Bilimleri Enstitüsü*, 1-46 (2003).
13. İnternet: MEGEP “Endüstriyel otomasyon teknolojileri dersi, sayısal işaret işleme-1 ders notu” http://megep.meb.gov.tr/mt_prog_modul_pdf /523 EO0382.pdf (2007).
14. İnternet: Texas Instruments Company “TMS320F2812 Digital Signal Processors data manual” <http://datasheet.emcelettronica.com /ti/TMS 320F2812> (2010).
15. İnternet: Texas Instruments Company “F2812 teaching material” http://en.pudn.com/downloads143/ebook/detail621581_en.html (2011).
16. Kesler, S., “Bilezikli asenkron makinelerin bilezikler üzerinden bulanık mantık tabanlı hız denetiminin TMS320F2812 ile gerçekleştirilmesi”, Doktora tezi, *Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Anabilim Dalı*, 3-7 (2006).
17. Kesler, M., Uçar, M. ve Özdemir E., “Paralel aktif güç filtresi için DSP tabanlı denetim algoritmasının otomatik gömülü kod üretimi ile hızlı prototiplendirilmesi”, *Elektrik – Elektronik - Bilgisayar Mühendisliği Sempozyumu (ELECO)*, Bursa, 24-28 (2006).
18. İnternet: Texas Instruments Company “Texas Ins., TMS320F2812 Digital Signal Processor Implementation Tutorial” <http://www.scribd.com/doc/2885 7722/F2 812- DSP-Full-Tutorial> (2004).
19. İnternet: Texas Instruments Company “Texas Ins., eZdsp TMS320F2812 Digital Signal Processor Technical References” http://www.isep.pw. Edu pl/ ZakladNapedu/labskp/ 2812_ezdsp_TechRef_F.pdf (2003).
20. İnternet: Gazi Üniversitesi Bilişim Enstitüsü “Duran F., Gömülü sistemler, uzaktan eğitim yüksek lisans ders notları” <http://www.uegazi.edu.tr/ akademik0/ASPX/Common/index.aspx> (2010).
21. İnternet: Texas Instruments Company “Target Support Package 4 User’s Guide, For use with Texas Instruments C2000TM ” http://www.mathworks.com/help/toolbox/targetsupport/tic2000_product_page.html (2011).
22. İnternet: : Texas Instruments Company “eZdsp F2812 Data Sheet” http:// www.isep.pw.edu.pl/Zaklad/labskp/2812_ezdsp_TechRef_F.pdf (2009).
23. İnternet: Texas Instruments Company “TMS320x280x, 2801x, 2804x enhanced quadrature encoder pulse (eQEP) module” <http://focus.ti. com/ lit/ ug/spru790d/spru790d.pdf> (2008).

24. Coşkun, İ. ve Terzioğlu H., “Hız performans eğrisi kullanılarak kazanç (PID) parametrelerinin belirlenmesi”, *Selçuk Üniversitesi Teknik Bilimler Meslek Yüksekokulu*, 6: 3-7 (2007).
25. İnternet: Pico Technology Company “Discontinued Products: PicoScop3423” http://www.picotech.com/discontinued/PicoScope3223_and_PicoScope3423.html (2011).
26. İnternet: Extech Instruments Company “Digital Tachometer Counter 461501” <http://www.valuetesters.com/Extech-4601-Digital-Tachometer-Counter.php> (2011).

ÖZGEÇMİŞ

Funda DEMİR 1984 yılında Ankara’da doğdu, ilk ve orta öğrenimini aynı şehirde tamamladı. 2003 yılında Gazi Üniversitesi Teknik Eğitim Fakültesinde Elektronik ve Bilgisayar Eğitimi Bölümü Elektronik Öğretmenliği Programında lisans öğrenimine başladı. 2008 yılında mezun oldu. 2009 yılında Karabük Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Bilgisayar Eğitimi Anabilim Dalı’nda yüksek lisans eğitimine başladı. Halen KBÜ Teknik Eğitim Fakültesi Elektronik ve Bilgisayar Eğitimi Bölümü’nde yüksek lisans öğrenimine devam etmektedir.

ADRES BİLGİLERİ

Adres : Karabük Üniversitesi
Teknik Eğitim Fakültesi
Balıklarkayası Mevkii / KARABÜK

Tel : (0507) 2473808

E-posta : funda_yorulmaz@hotmail.com