

ELEKTRİK MAKİNALARI DENETİM SİSTEMLERİNDE TMS320F2812 DSP KULLANIMI

Bildiri ID:120

Dr. Selami KESLER
Pamukkale Üniversitesi

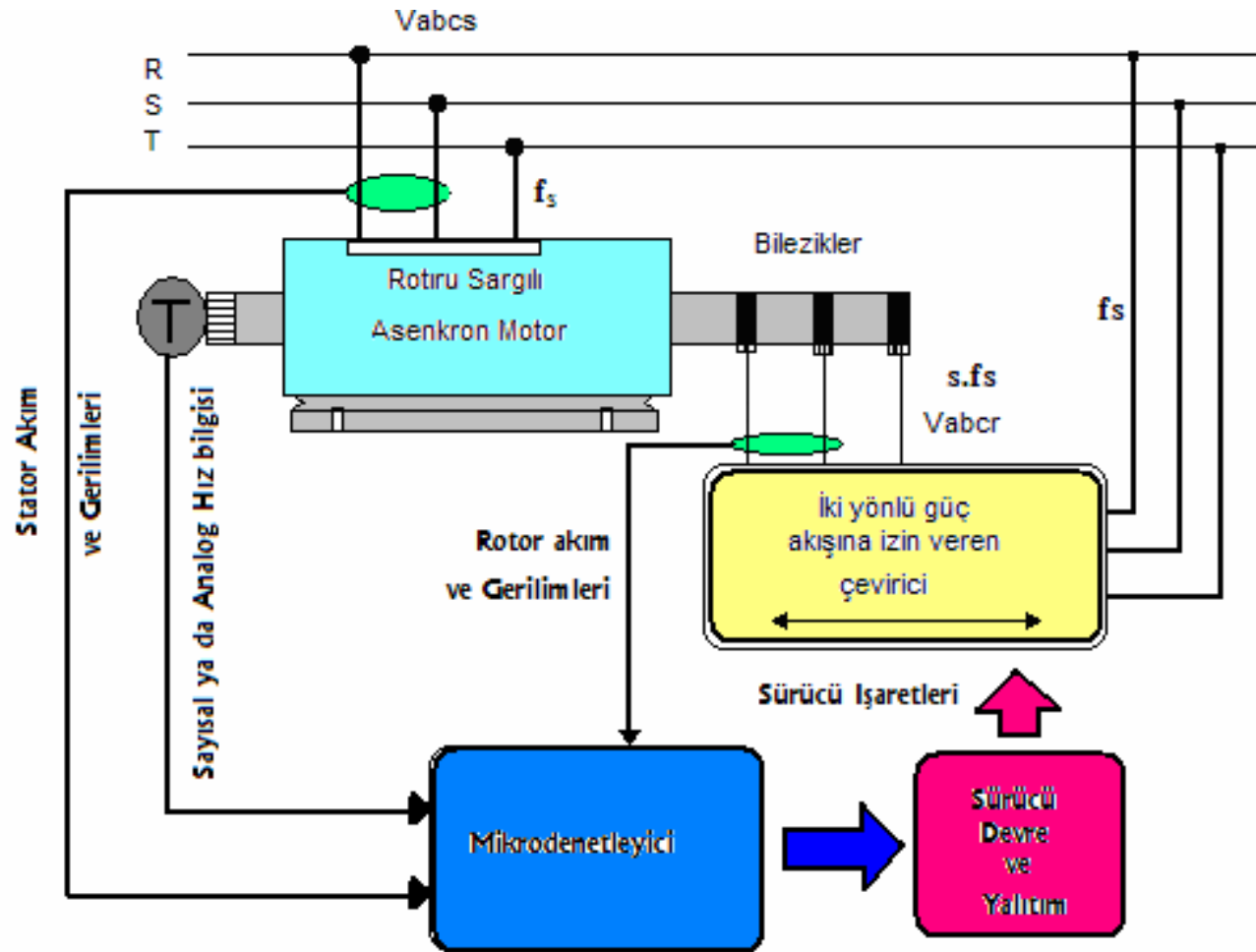
AKADEMİK BİLİŞİM'08

**ÇANAKKALE ONSEKİZ MART
ÜNİVERSİTESİ**

ÖZET

- **ÖRNEK BİR MOTOR KONTROL SİSTEMİ**
- NEDEN DSP?
- **NEDEN TMS320F2812 ?**
 - MODEL TANIMI VE YAPISI
 - GENEL ÖZELLİKLERİ
- **CCS NEDİR?**
- CCS'de KAYITÇI VE DEĞİŞKENLERİNİN GÖZLENMESİ
- **DSP KAYITÇILARININ NESNE TABANLI KULLANIMI**
- PWM İŞARETLERİ VE ÖLÜ ZAMAN AYARI
- **ADC KANAL YÖNETİMİ**
- Alternatif (AC) İŞARETLERİN ÖRNEKLENMESİ
- **SAYISAL HIZ BİLGİSİ OKUNMASI**
- IQmathLib DESTEĞİ VE VIRTUAL FLOATING-POINT işlem
- **ÖRNEK MODEL SONUÇLARI**

Örnek Bir Motor Kontrol Sistemi



Kontrol yöntemi ?

PID, FLC vb.

Kontrol Sistemi Girdileri ?

Hız, Akım, Sıcaklık vb.

Kont.Sis.Çıktıları ?

Denetim işaretleri vb.

Sürücü ve Güç Devresi ?

IGBT inverter vb

Neden DSP ?

- Analog ve sayısal işaretlerin okunması ve bu işaretlerin normalizasyonu
- Kontrol yöntemine bağlı aritmetik, trigonometrik, türevsel işlemler, algoritma döngüleri ve iterasyonlar
- Yönteme bağlı kontrol işaretlerinin üretilmesi
- Kontrol işaretlerinin sürücü devreye uygun üretilmesi
- Kontrol değişkenlerinin gerçek zamanlı gözlenmesi ve grafik çıktılar
- Çevresel birimlere kolay erişim
 - **Gibi işlemlerin hızlı ve yüksek doğrulukta yapılmasını sağlayan ve büyük boyutlu programlamaya imkan tanıyan bir SAYISAL İŞARET İŞLEYİCİ---DSP**

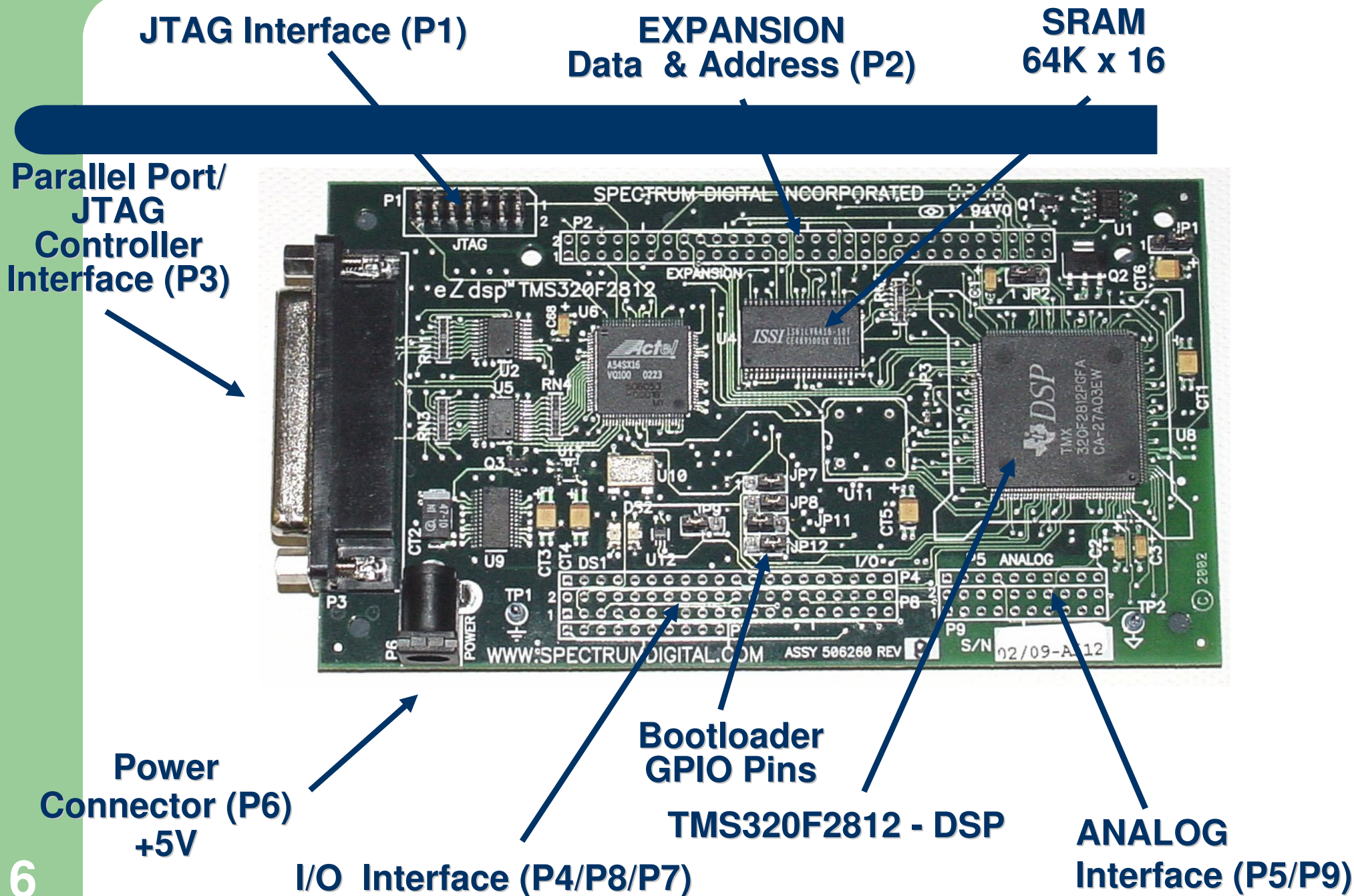
Neden TMS320F2812 DSP ?

- Son yıllarda motor kontrol sistemlerine uygun Texas Instruments tarafından geliştirilmiş özel bir işlemci
- 150MIPS işlem başarımı
- 18K Word RAM, 128K Word Flash EEPROM
- 30MHz Clock
- 6 çift tümleyenli, 4'ü bağımsız toplam 16 PWM
- Programlanabilir ölü zaman mantığı
- 12-bit 16 kanal ADC
- 56-pin Sayısal Giriş/Çıkış
- Gerçek zamanlı analiz, grafik arayüz, değişken ve kayıtçı gözlemi
- 10 adet 16-bit karşılaştırıcı
- 4 adet Genel amaçlı zamanlayıcı/sayıcı
- 2 adet index girişli kare dalga kodlayıcı
- 6 adet programlanabilir yakalama girişi (3 ad EVA ve 3 ad EVB de)
- Yaklaşık 100ns gecikmeli hızlı interrupt yöneticisi
- McBSP (128 kanal, 32-bit T/R), eCAN, Full-duplex haberleşme
- IQmath kütüphane desteği ile floating-point işlem kolaylığı ve duyarlılığı
- CCS, C/C++ , Matlab, VISSIM desteği



eZdsp F2812

Spectrum Digital



Code Composer Studio CCS

Menu ve ikonlar

Proje yöneticisi:

Kaynak ve amaç dosyalar, compiler, linker ve build

Edit alanı, kaynak program

The screenshot displays the Code Composer Studio (CCS) interface for a Texas Instruments CPU-28xx simulator. The main window shows the source code for 'SchedulerLab1.c' with a cursor at line 29, column 1. The code includes a main function that initializes a constant array 'y', loops through it, and calls 'MemTestC' and 'port0 = sine()'. The '28xx Registers' window shows the current state of registers, with PC at 3FD09A and ACC at 000A0074. The 'Port0 Output' window displays a sine wave graph. The 'Status' window at the bottom left shows 'Compile Complete, 0 Errors, 0 Warnings, 0 Remarks.' The 'Watch Locals' window at the bottom center shows variables like 'value', 'length', 'start', 'i', 'rvalue', 'status', and 'MemAddr'. The 'Memory' window at the bottom right shows memory addresses and their corresponding values.

CPU gözlem

Grafik penceresi

Durum penceresi

Gözlem penceresi

Bellek izleme

Kayıtçı ve deęişkenlerin gözlenmesi

The screenshot displays the Code Composer Studio (CCS) interface for debugging an F2812 eZdsp. The main window shows the 'GEL' menu with 'Set PLL Ratio' selected. A 'Port0 Output' graph is visible, showing a sine wave. Below the graph, a 'Watch' window displays the following data:

Name	Value	Type	Rac
TIMER0TPRH	0x0000	int	hex
GPTCONA	8448	int	unsi
Vref	66.355	float	float
ck	-0.5	float	float
Vbeta	-64.72669	float	float
n	5	uns...	unsi

The bottom status bar indicates 'CPU HALTED' and 'Ln 42, Col 1'.

CPU

Bellek

Kaynak
programdaki
deęişkenler

Interruptlar

Zamanlayıcı

Sayıcı

Diziler

Grafik

Gerçek
zamanlı max
100ms

Kayıtçıların Nesne Tabanlı Kullanımı

PeripheralName.RegisterName.all=0x???? //16-bit yada 32-bit erişim

Ör: **EvaRegs.ACTRA.all=0x0666;**

//PWMx ve PWMx+1 in göreceli durumu

{PeripheralName.RegisterName.half.LSW

PeripheralName.RegisterName.half.MSW }

// en anlamlı ve en anlamsız hanelere erişim 8 yada 16-bit

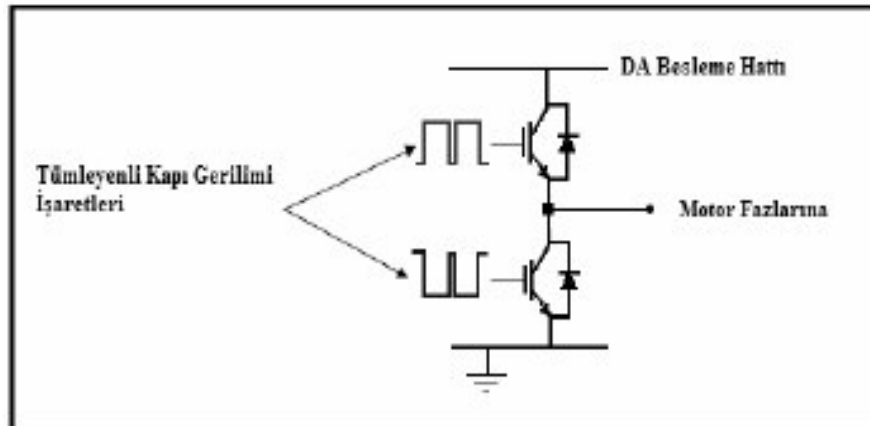
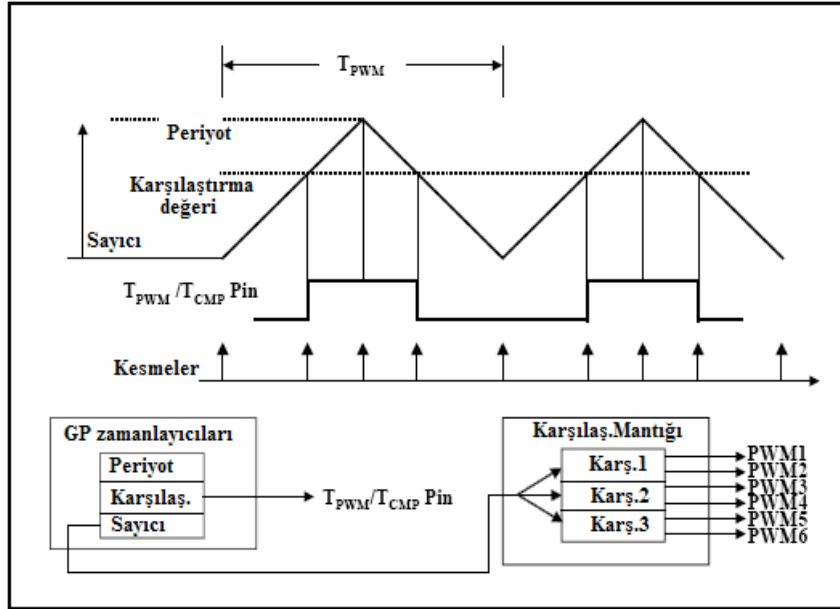
PeripheralName.RegisterName.bit.FieldName=0x????

// bit düzeyinde erişim

Ör: **GpioMuxRegs.GPAMUX.bit.PWM1_GPIOA0=1;**

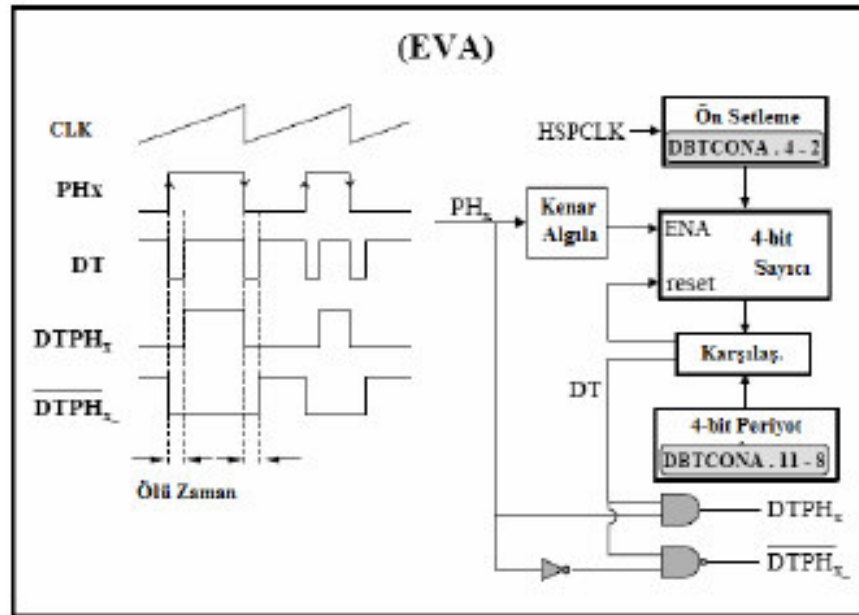
//GPIO Multiplexer'in A Portu İlk biti PWM1 kanalı için yetkili

PWM –EvaRegs & GpioMuxRegs

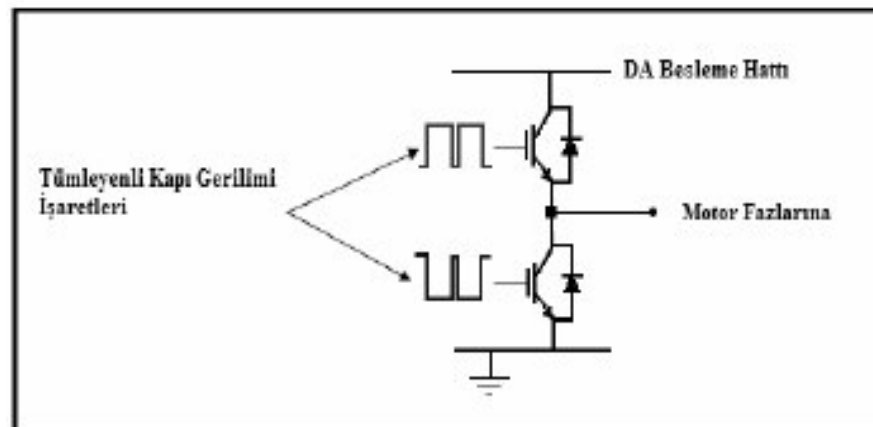


- ✓ Zamanlayıcı periyodu setle-T1PR
- ✓ Sayıcı değeri resetle-T1CNT
- ✓ Karşılaştırma değeri güncellemesi yap-CMPRx
- ✓ Yükselme-düşme biçimi belirle -- ACTRA
- ✓ Ölü zaman lojiğini ayarla
 - DBTCONA ve COMCONA
- ✓ Periyod sonunda kesmeleri yetkilendir
- ✓ PWM kanallarını yetkilendir
 - GpioMuxRegs.GPAMUX

Ölü Zaman Lojigi ve Önemi

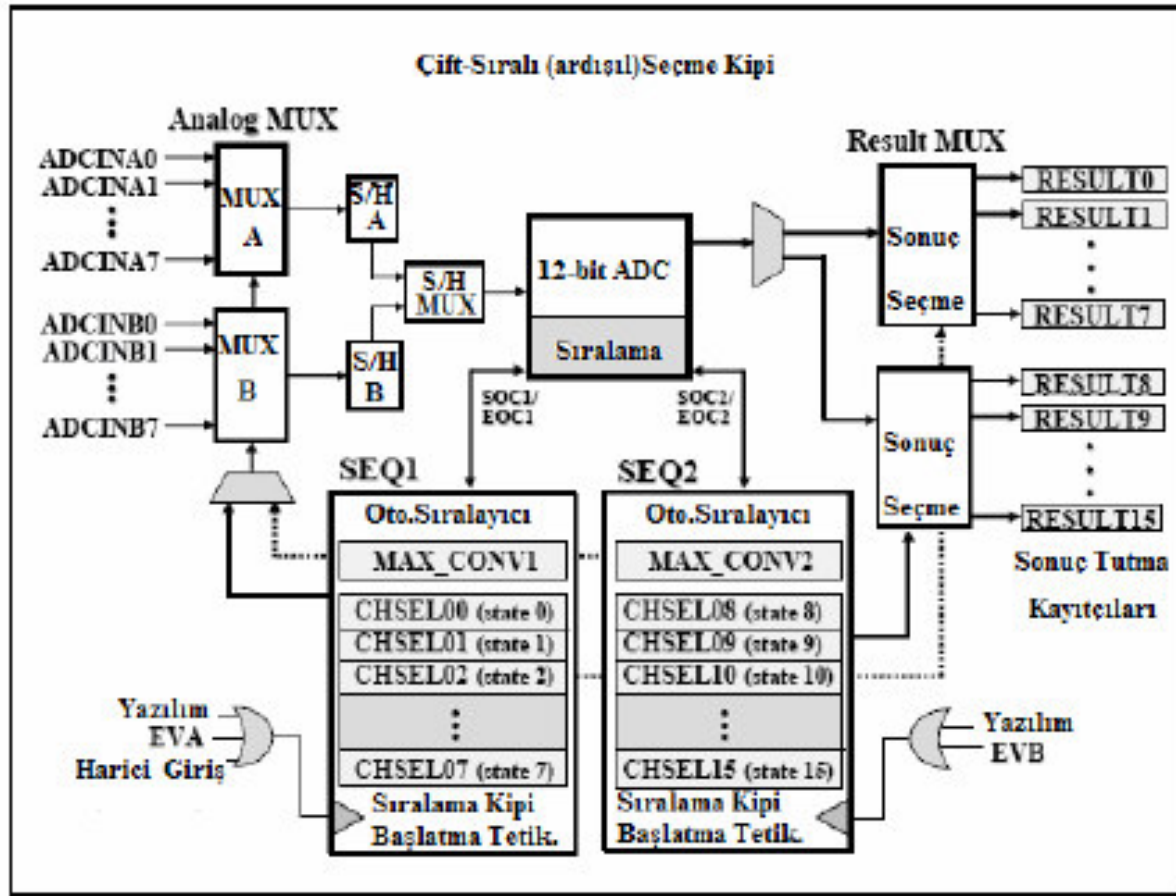


- Aynı faza bağlı anahtar elemanların aynı anda iletimde kalmalarını ve kısa devre olmalarını önler.
- Her bir PWM çifti, sinüs ya da SVPWM üretiminde ölü zamanlar oldukça hassas programlanabilmektedir.
- Bu lojigi olmayan denetleyiciler için ayrıca elektronik bir devre tasarımı gerekir.
- PIC uygulamalarında ise işlemsiz bekleme zamanları oluşturulur.



```
EvaRegs.DBTCONA.bit.EDBT1=1;//PWM1-2  
EvaRegs.DBTCONA.bit.EDBT2=1;// PWM3-4  
EvaRegs.DBTCONA.bit.EDBT3=1;// PWM5-6  
EvaRegs.COMCONA.all=0xA600;//SVPWM  
0xB600  
EvaRegs.DBTCONA.bit.DBTPS=6;//Ölü zaman  
EvaRegs.DBTCONA.bit.DBT=12;//C->3µs
```

ADC kanal yönetimi



25Mhz 0-3vDC=>0-4095 digital

16 kanal sıralı olarak
dönüştürüldüğünde tek
bir kanal 80ns,

Sadece bir kanal dönüşümü
için 200ns zaman harcar.

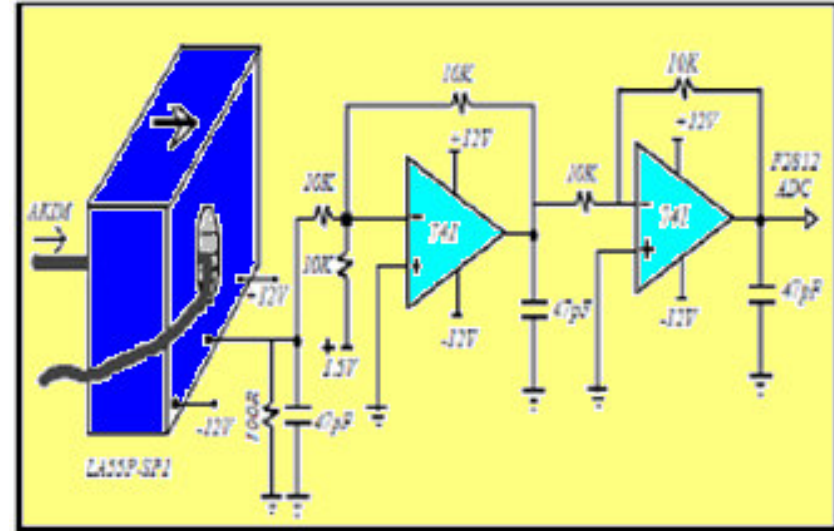
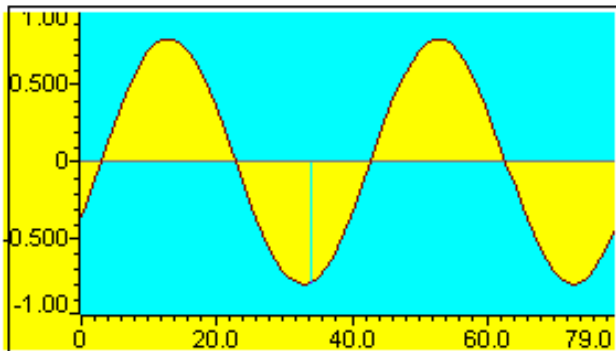
- ✓ Kaç kanal dönüştürülecek?
- ✓ Hangi kanal hangi sonuç kayıtcısında saklanacak?
- ✓ Hangi sırada dönüşüm yapılacak?
- ✓ Dönüştürme için ADC'yi hangi Olay Yöneticisi uyaracak?
- ✓ Interrupt için hangi birim yetkili?

AdcRegs içinde;
ADCMAXCONV, ADCCHSELSEQx ve ADCTRLx

AC İşaretlerin Örneklenmesi

```
RefVoltDigital=(AdcRegs.ADCRESULT2 >>4);  
RefVoltAnalog= RefVoltDigital *3.0/4095;
```

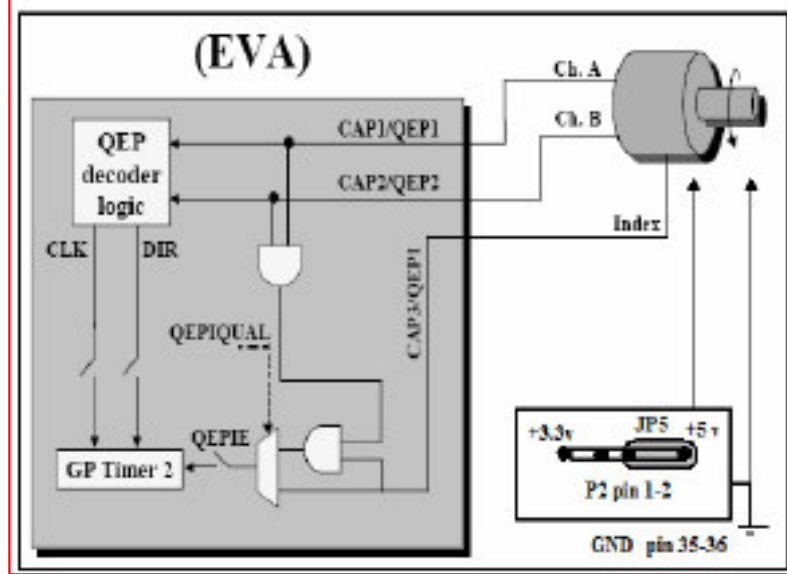
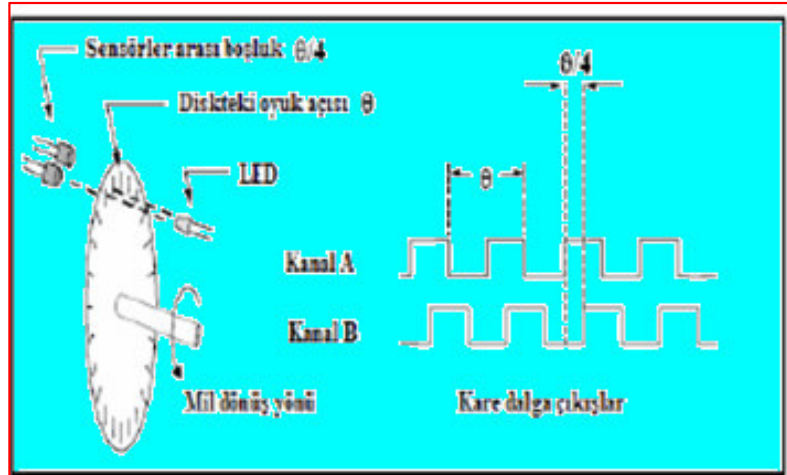
ADC 12-bit olup kayıtlar 16-bit'dir. Bu nedenle normalize edilecek sayısal değerler 4-bit sağa kaydırılarak sonuç kayıtlısından alınır.



0-3v DC analog girişe uygun olması için alternatif işaretler +1.5V üzerine eklenir. Ve normalizasyonda sayısal değerden 2048 sayısı çıkartılır.

AC giriş aralığı -1.5V -- +1.5V

Sayısal Hız/konum Kodlayıcılar&DSP



GND seviye eřitliđi için Encoder beslemesi DSP den alınır.

PWM için Timer1, ve Sayıcısı

Encoder için Timer2 ve sayıcısı setlenir.

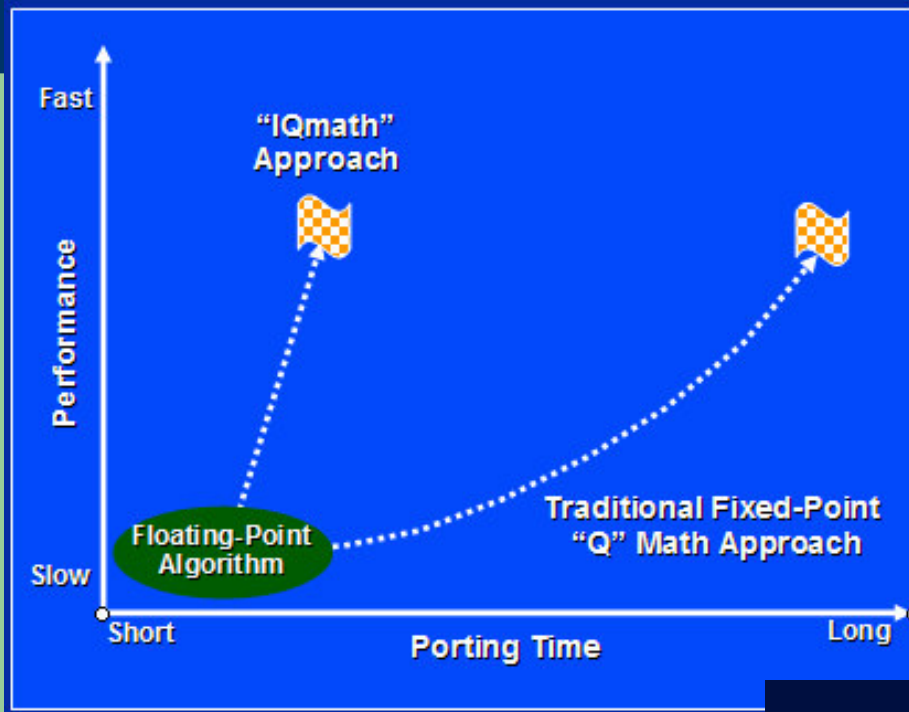
Yakalama lođiđi devre dıřı, kare dalgı kodlayıcı lođiđi etkin.

```
EvaRegs.T2PR=0xFFFF;
EvaRegs.T2CNT=0x0000;
EvaRegs.T2CON.all=0x1870;
```

```
EALLOW;
GpioMuxRegs.GPAMUX.bit.CAP1Q1_GPIOA8=1
GpioMuxRegs.GPAMUX.bit.CAP2Q2_GPIOA9=1
EDIS;
```

```
EvaRegs.CAPCONA.all=0xE000;
```

IQmath yaklaşımı & Virtual Floating-Point



C derleyicisi için

```
#include<IQmathLib.h>
```

C++ için;

```
#include<IQmathCPP.h>
```

I: Tam kısım

f: Kesirli kısım

S:işaret

IQ:integer quotient

I8Q24 gibi



$$-2^I + 2^{I-1} + \dots + 2^1 + 2^0 \cdot 2^{-1} + 2^{-2} + \dots + 2^{-Q}$$

IQmath Library Desteđi

F2812 fixed-point işlemcidir. Ve yazılan float işlemlerin derleyici tarafından dönüştürülmesi zaman alır, işlem adım sayısı artar, işlem doğruluđu azalır, sayı duyarlıđı düşer.

ÇÖZÜM: IQmath

Kayan Noktalı

```
float Y, M, X, B;
```

```
Y = M * X + B;
```

**Geleneksel
Sabit noktalı
Q format**

```
long Y, M, X, B;
```

```
Y = ((i64) M * (i64) X + (i64) B << Q) >> Q;
```

C için IQmath

```
_iq Y, M, X, B;
```

```
Y = _IQmpy(M, X) + B;
```

**C++ için
IQmath**

```
iq Y, M, X, B;
```

```
Y = M * X + B;
```

IQmathLib & Trigonometri

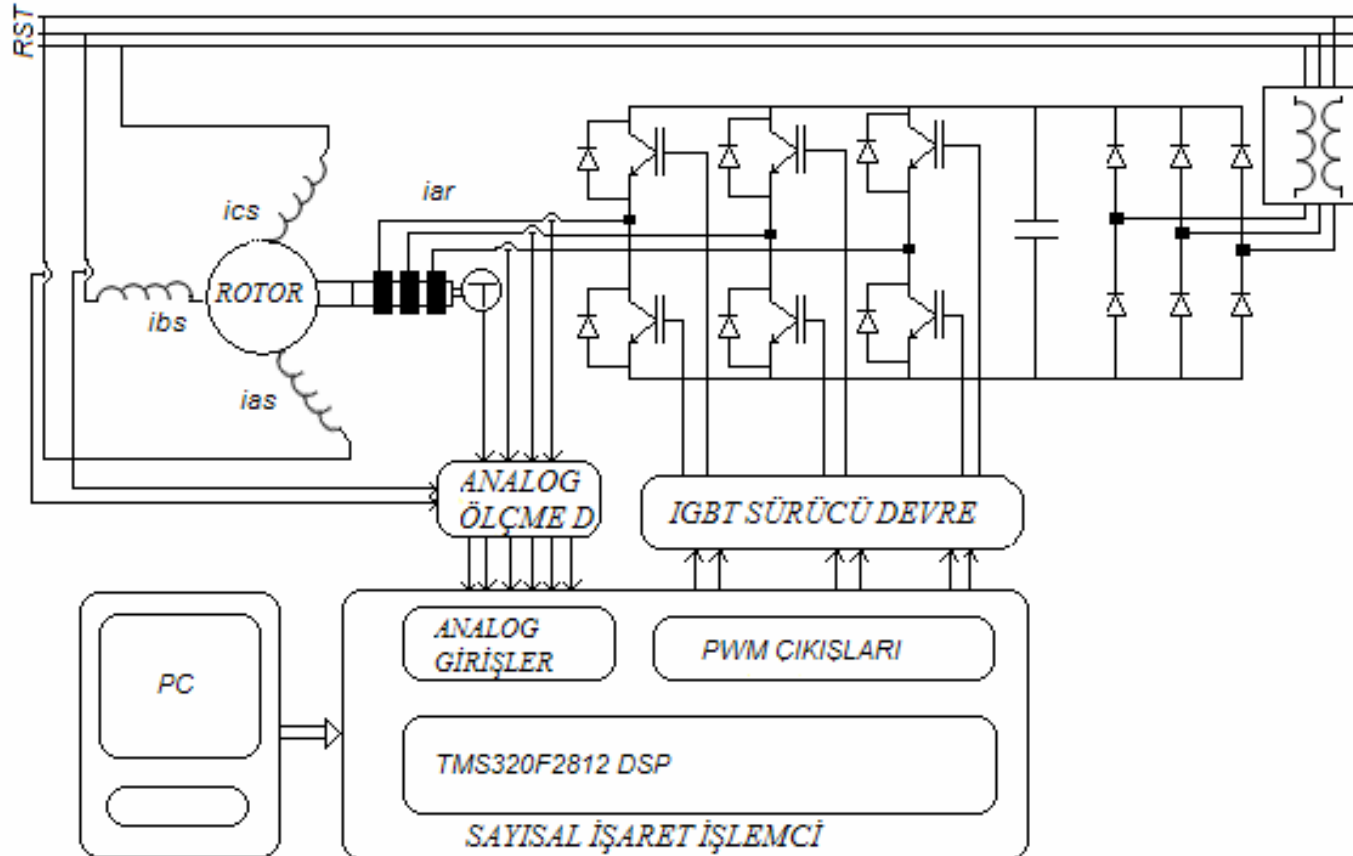
Kayan Nokatalı	C'de IQmath :	C++ da IQmath
float A, B;	_iq A, B;	iq A, B;
A = 1.2345	A = _IQ(1.2345)	A = IQ(1.2345)
A * B	_IQmpy(A, B)	A * B
A / B	_IQdiv(A, B)	A / B
A + B	A + B	A + B
A - B	A - B	A - B
>, >=, <, <=, ==, !=, &&,	>, >=, <, <=, ==, !=, &&,	>, >=, <, <=, ==, !=, &&,
sin(A), cos(A)	_IQsin(A), _IQcos(A)	IQsin(A), IQcos(A)
sin(A*2pi), cos(A*2pi)	_IQsinPU(A), _IQcosPU(A)	IQsinPU(A), IQcosPU(A)
atan(A), atan2(A,B)	_IQatan(A), _IQatan2(A,B)	IQatan(A), IQatan2(A,B)
atan2(A,B)/2pi	_IQatan2PU(A,B)	IQatan2PU(A,B)
sqrt(A), 1/sqrt(A)	_IQsqrt(A), _IQisqrt(A)	IQsqrt(A), IQisqrt(A)
sqrt(A*A + B*B)	_IQmag(A,B)	IQmag(A,B)
if(A > Pos) A = Pos if(A < Neg) A = Neg	_IQsat(A, Pos, Neg)	IQsat(A, Pos, Neg)

Sinüzoid üyelik kümeleri kullanan bir BMD yazılımında fixed-point işlemciler oldukça zaman kaybetmektedirler.

Ancak IQmath desteği bu sorunu ortadan kaldırmıştır.

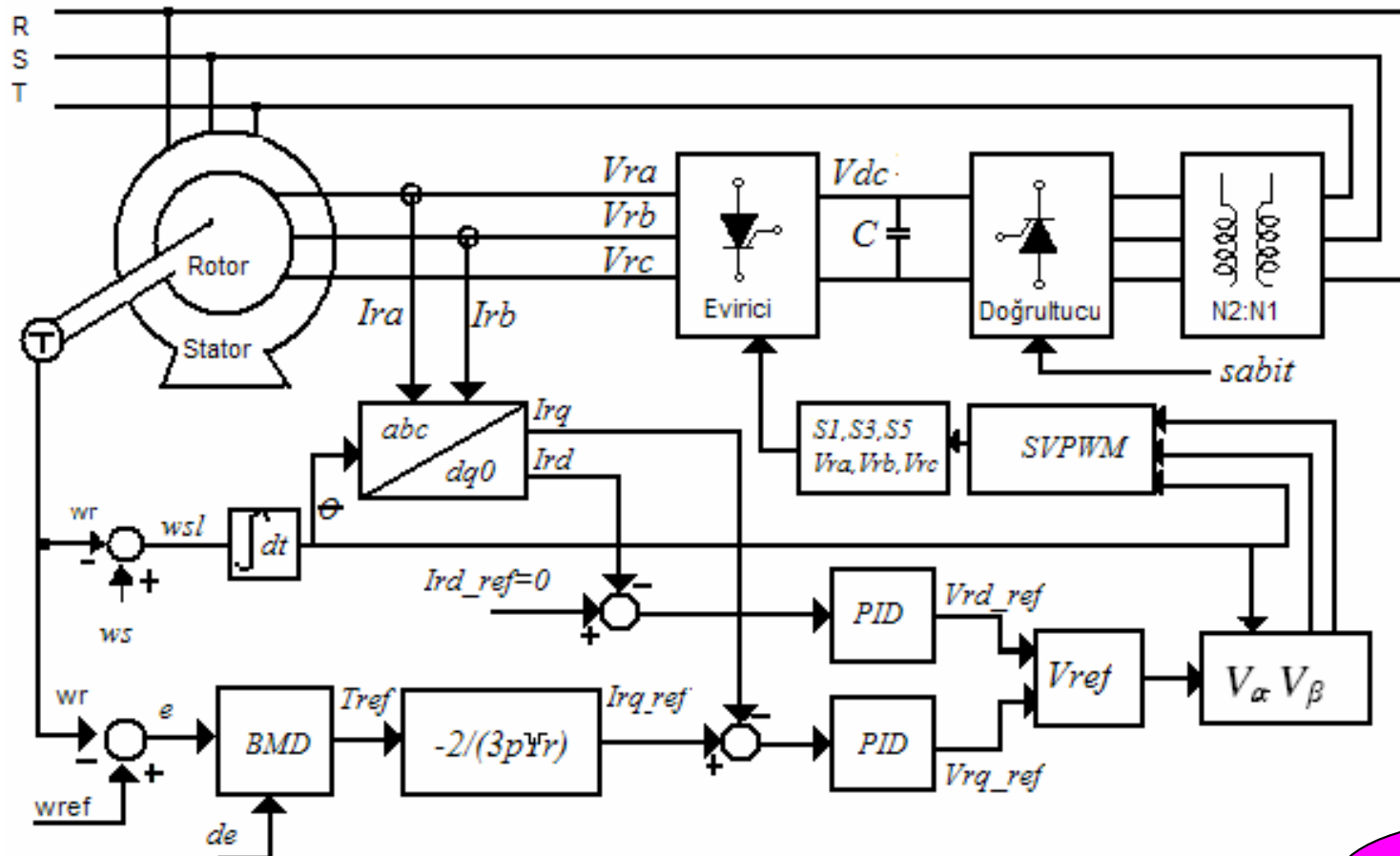
Uygulama

Bilezikler üzerinden rotor sargılarına gerilim enjeksiyonu ile hız ve kayma gücü denetimi



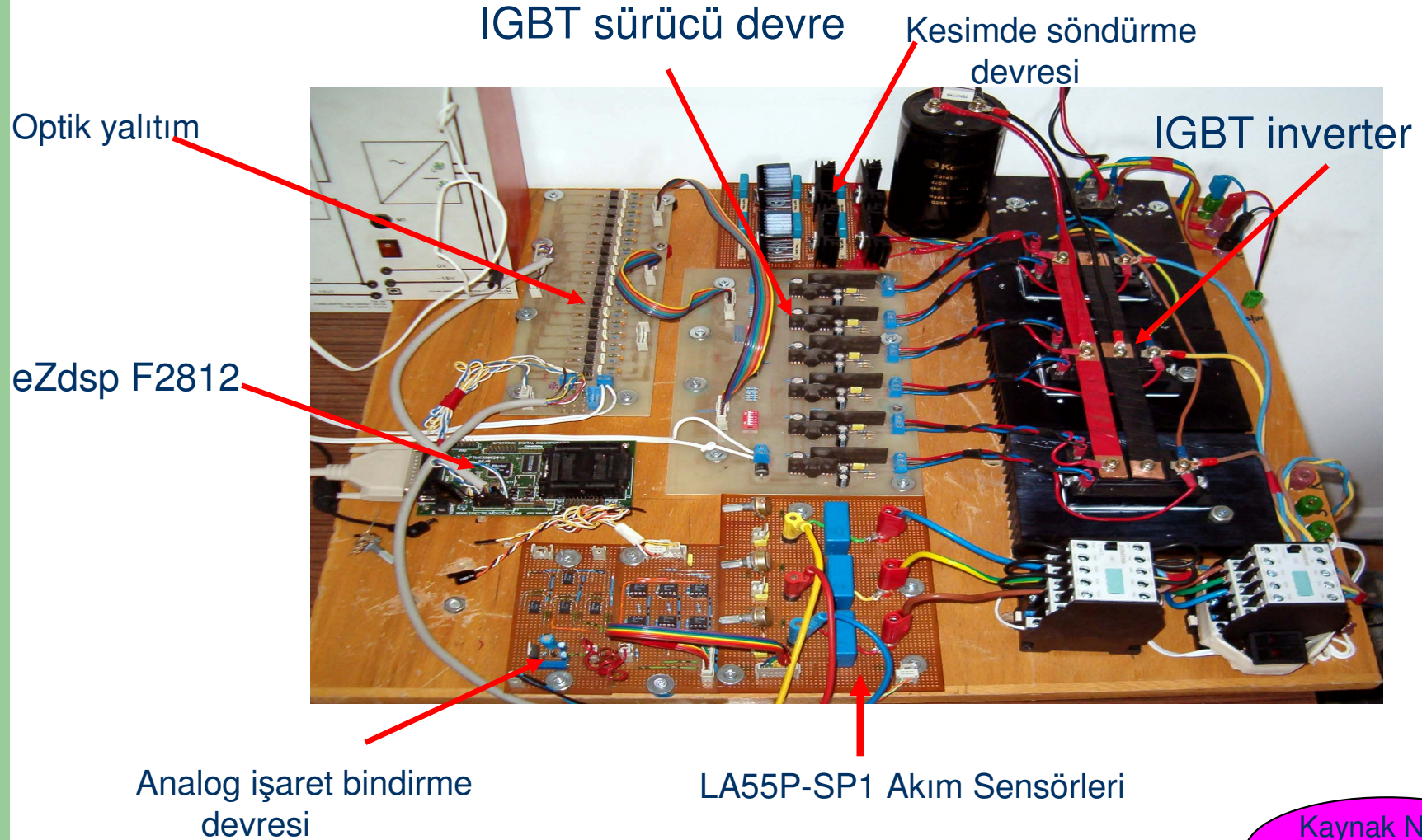
Kontrol Yöntemi

BMD & SVPWM



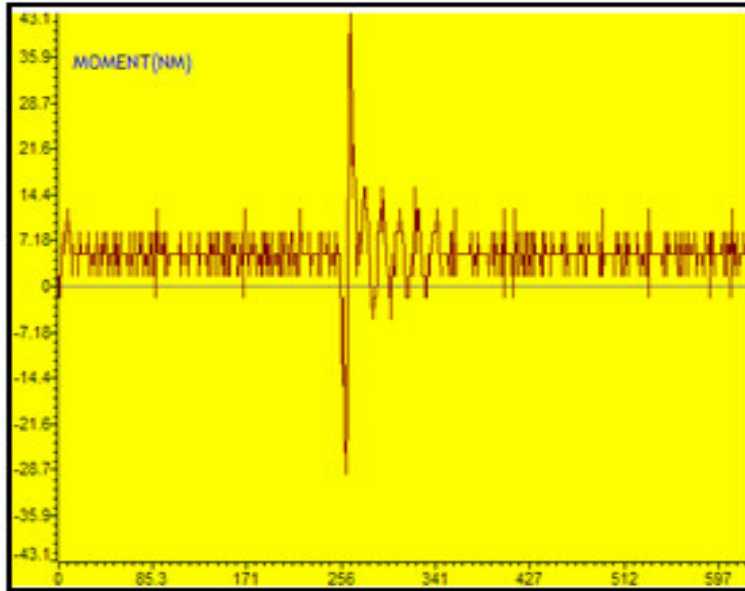
Kaynak No:[6]
Kesler,S. (2006)

Uygulama Devresi

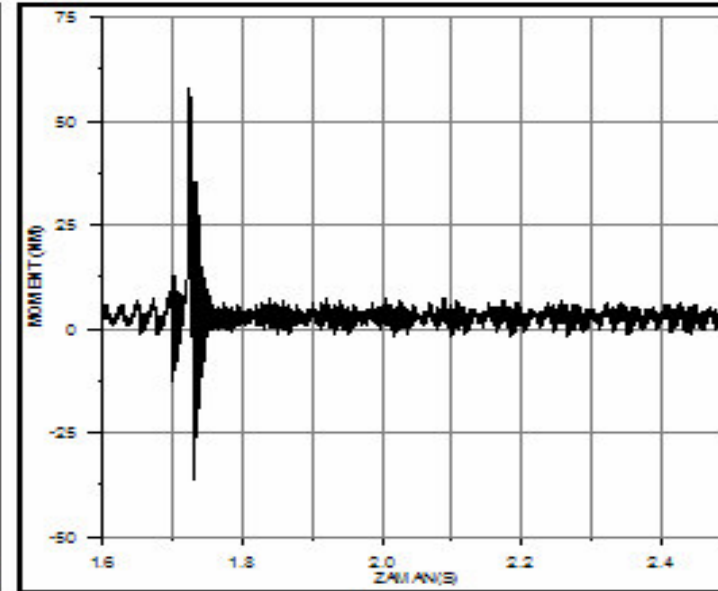


Kaynak No:[6]
Kesler,S. (2006)

Örnek Sonuçlar MOMENT



(a)



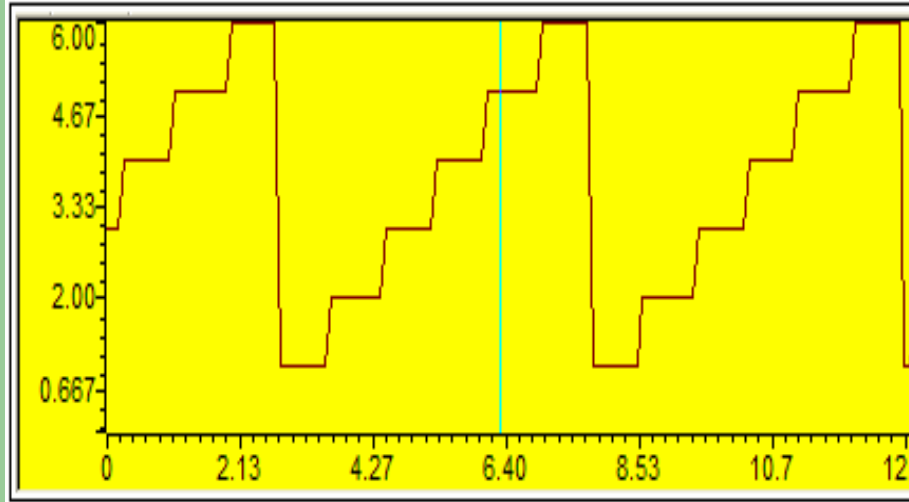
(b)

Rotor sargılarına BM denetimli gerilim enjeksiyonu anında moment deęiřimi

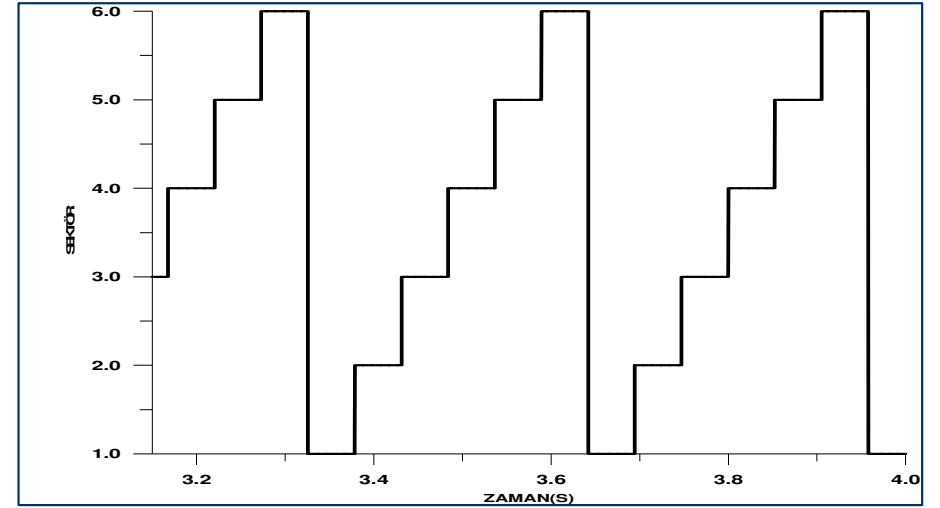
A) DSP ıktısı-deneysel B) C++ benzetimi

Örnek sonuçlar

Anahtarlama



A.



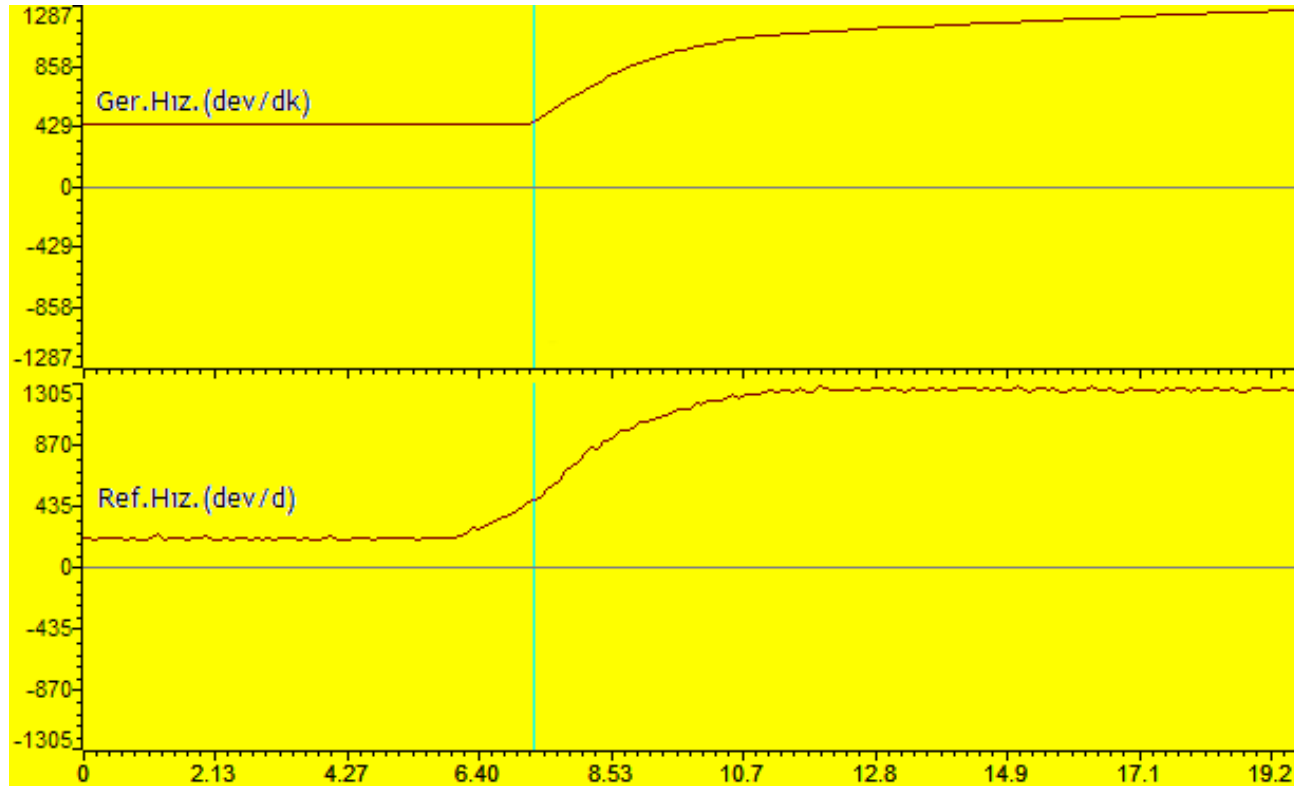
B.

SVPWM anahtarlama bölge geçişleri

A) DSP çıktısı-deneyisel ve B) C++ benzetimi

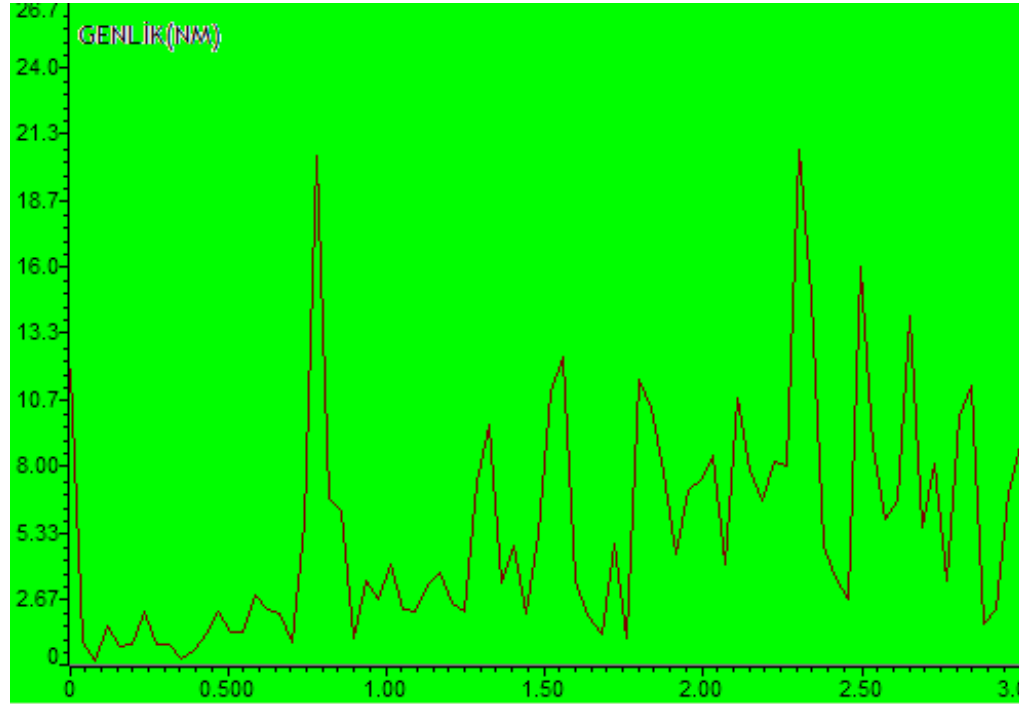
Örnek Sonuçlar

HIZ



400-1300 dev/d aralıklı hız denetimi –deneysel DSP çıktısı, gerçek ve referans hız değişimi

Moment HARMONİKLERİ



Denetimli 600dev/d bořta alıřma moment FFT genlikleri

DSP ıktısı, gerek zamanlı analiz, CCS desteęi

Kaynak No:[6]
Kesler,S. (2006)



- Haberleşme, ses ve görüntü işleme, robot uygulamaları, ölçme tekniği ve gerçek zamanlı sistem analizi gibi alanlarda oldukça yaygın kullanılan Sayısal İşaret İşleyiciler her türlü konverter ve elk.makinası dinamiklerinin kontrolünde de önemli bir yere sahiptir.
- Sözü edilen diğer özelliklerinin yanında Flash EEPROM programlama desteği olan TMS320F2812 DSP düşük güç tüketimi ve düşük maliyeti nedeniyle de kullanımı gittikçe yaygınlaşmaktadır.

Kaynaklar

- [1] Texas Ins., TMS320f2812 Digital Signal Processor Implementation Tutorial, 2004
- [2] Texas Ins., EzDSP TMS320F2812 Digital Signal Processor Technical References, 2004
- [3] Texas Ins. World Wide Web site www.ti.com
- [4] Tang, Y. ve Xu, L., Vector Control and Fuzzy Logic Control of Doubly Fed Variable Speed Drives with DSP Implementation, IEEE Trans. Energy Conversion, 10, 4 (1995) 661-668
- [5] Poddar, G. ve Ranganathan, V.T., Direct Torque and Frequency Control of Doubly Inverter Fed Slip-Ring Induction Motor Drive, IEEE Trans. Industrial Electronics, 51,6 (2004) 1329-1337.
- [6] Kesler, S., Bilezikli Asenkron Makinaların Bulanık mantık Tabanlı Hız Denetiminin TMS3220F2812 DSP ile Gerçekleştirilmesi, Doktora Tezi, Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2006

TEŞEKKÜRLER



Dr.Selami KESLER

Pamukkale Üniversitesi