Gazi University Journal of Science
GU J Sci Part C
4(2):71-82 (2016)

Mikrodenetleyici Temelli Sistemlerde Dspic Kullanımı ve Örnek Uygulama Geliştirme Süreci

Naki GÜLER[1,♠], Erdal IRMAK[2]

[1] *Gazi Üniversitesi Teknik Bilimler Meslek Yüksekokulu, Yenimahalle, Ankara*

[2] *Gazi Üniversitesi Teknoloji Fakültesi Elektrik Elektronik Mühendisliği Bölümü, Teknikokullar, Ankara*

Başvuru: 18/06/2014 Kabul: 23/11/2014

ÖZ

Günümüzde birçok endüstriyel uygulamada mikrodenetleyici tabanlı kontrol ve izleme sistemleri kullanılmaktadır. Bu çalışmada, son zamanlarda kullanımı gittikçe yaygınlaşmakta olan dsPIC (Digital Signal Controller) mikrodenetleyicisi ele alınmıştır. dsPIC, özellikle yüksek hız ve tepki zamanı gerektiren uygulamalarda düşük maliyetli ve etkili çözümler sunmaktadır. Ülkemizde yüksek hızlarda çalışabilen ve analog çözünürlüğü yüksek dsPIC denetleyicilerinin kullanımının yaygınlaşması yüksek hassasiyetli sistemlerin geliştirilmesini sağlayacaktır. Bu amaçla çalışmada dsPIC tabanlı uygulama geliştirme süreci ayrıntılı bir şekilde ele alınmış ve örnek uygulamalarla dsPIC'in nasıl kullanılacağı ortaya konulmuştur. Bu amaçla öncelikle, bilgisayar ile iletişime olanak sağlayan RS-232 protokolü, darbe genişlik modülasyonu ve analog/dijital dönüşüme ait kütüphanelerin C dili ile nasıl oluşturulacağı bilgisayar kontrollü bir DA kapalı döngü hız kontrol sistemi üzerinden örneklendirilmiştir. Aynı uygulama, çok sık kullanılan PIC (Peripheral Interface Controller) mikrodenetleyicisi ile de gerçekleştirilerek karşılaştırmalı performans analizi yapılmıştır. Sunulan bu çalışmanın özellikle uygulama çözümü geliştirme sürecinde akademisyen ve mühendislere önemli katkılar sağlayacağı düşünülmektedir. Ayrıca çalışma eğitim amaçlı olarak lisans ve ön lisans düzeyindeki ilgili derslere bir kaynak teşkil edecektir.

Anahtar Kelimeler: *dsPIC, kontrol ve izleme, program geliştirme*

♠Corresponding author, *e-mail: nakiguler@gazi.edu.tr*

Use of Dspic in Mıcrocontroller Based Systems and Sample Application Development Process

ABSTRACT

Nowadays, microcontroller based control and monitoring systems are being used in several industrial applications. This study deals with dsPIC (Digital Signal Controller) microcontroller, usage of which is becoming more widespread. As compared to similar ones, dsPIC offers more efficient solutions especially in the applications which require high speed and high response time. However, it is observed that use of dsPIC has not been studied sufficiently in Turkey, whereas many researchers have used it in many kinds of industrial solutions as seen in international literature. For this reason, this study addresses dsPIC based application development process in detail, and its usage is put forward with sample experimental applications. Firstly, some general examples like creating a serial communication with a PC using RS-232 protocol, generating pulse width modulation signals and realizing analog/digital conversions are exemplified on a sample experimental set which focuses closed-loop speed control of a DC motor over the computer. Dynamic libraries of these sample applications are produced in C programming language. In order to make a comparative performance analysis, the same applications are also developed by PIC (Peripheral Interface Controller) microcontroller which is used very commonly in industrial applications. It is considered that the study presented in this paper will provide useful contribution to academicians and engineers especially in the process of producing solutions for industrial applications. Furthermore, the system presented can be used for educational purposes in graduate and undergraduate courses of engineering colleges.

Keywords: *dsPIC, control and monitoring, program development*

1. GİRİŞ (INTRODUCTION)

Üretim ve imalat alanındaki teknolojik gelişmeler hızlı, kararlı ve doğru çalışan sistemlerin önemini artırmaktadır. Özellikle elektrik ve elektronik alanlarında mikrodenetleyiciler yaygın olarak kullanılmaktadır. Bu çalışmada hızlı, kararlı ve yüksek çözünürlüklü algılama kapasitesine sahip olan dsPIC mikrodenetleyicisine ait genel bilgiler verilerek yazılım ve donanım örnekleri oluşturulmuştur.

dsPIC'lerin motor kontrolü için özel darbe genişlik modülasyonu (DGM) kanalları bulunduğundan literatürdeki çalışmalar bu alanda yoğunlaşmıştır. Özellikle elektrikli araçlarda kullanılan motorların geri bildirimli olarak kontrolü için dsPIC tercih edilmiştir [1-3]. Farklı kontrol teknikleri kullanılarak motorların sürülmesi gerçekleştirilmiş ve motor parametreleri geri bildirimli olarak alınmıştır. Elde edilen veriler bulanık mantık, oransal integral kontrol gibi kontrol teknikleri ile değerlendirilerek sistemlerin kararlı çalışmaları sağlanmıştır. Bu noktada dsPIC'in yüksek çalışma hızı ve diğer kontrolörlere göre işlem çözme kapasitelerinin yüksek olması sistemlerin daha hızlı ve kararlı çalışmalarını sağlamıştır. Ayrıca elektrikli araçların enerji hattı üzerinden haberleşmesi gibi sistemlerde de dsPIC kullanılmaktadır[4].

Günümüzde enerji kaynaklarının azalması yenilenebilir enerji kaynaklarının önemini artırmıştır. Yenilenebilir enerjide üretilen enerjinin kullanılabilir hale gelmesi için güç elektroniği sistemleri ile şebekeye bağlanması veya depolanması gerekmektedir. Bu aşamada güç elektroniği çeviricilerinin kontrol sinyallerinin üretilmesi, özellikle DGM ve analog birimler kullanılarak güç elektroniği çeviricilerinin kontrolü ve üretilen enerjinin şebeke üzerinde bozucu etkiler yaratmaması gibi olumsuz durumları engellemek için hızlı çalışan bir kontrolöre ihtiyaç duyulmuştur. Bu ihtiyacın giderilmesi amacı ile birçok yenilenebilir enerji sisteminde kontrolör olarak dsPIC' tercih edilmiştir [5-8]. Çünkü dsPIC denetleyicileri, analog işlemlerde yüksek performans gösterecek şekilde tasarlanmış ve yüksek hızlarda çalışan kontrolörlerdir. Bu nedenle ses tanıma, ses kayıt cihazları, yüksek çözünürlüklü ses yakalama ve modemler gibi yüksek hızlarda analog dijital çevrim gerektiren uygulamalarda sıkça kullanılmaktadır [9 - 12].

Bu çalışmanın amacı, ülkemizde elektrik-elektronik devre ve kontrol sistemi içeren endüstriyel uygulamalar ve eğitim alanında yaygın olarak kullanılmakta olan birçok mikrodenetleyiciden daha üstün özelliklere sahip olan dsPIC'e ait genel bilgiler vermek ve kullanıcıların yüksek performanslı sistemler oluşturmasını sağlamaktır. dsPIC için oluşturulan kütüphanelerin kullanımını açıklamak amacı ile dsPIC kontrollü örnek uygulamalar geliştirilerek kullanıcıların özgün kütüphane kullanımı ve dsPIC hakkında bilgi edinmeleri sağlanmıştır. Yapılan uygulama çalışması yaygın olarak kullanılmakta olan ve dsPIC ile yaklaşık olarak aynı maliyet bandında olan PIC18F452 mikrodenetleyicisi ile karşılaştırılarak dsPIC'in yüksek performanslı olarak çalıştığı kanıtlanmıştır.

2. UYGULAMA GELİŞTİRME SÜRECİNDE MİKRODENETLEYİCİ SEÇİMİ (SELECTION OF MICROCONTROLLER IN APPLICATION DEVELOPMENT PROCESS)

Uygulama geliştirme aşamasında farklı üstünlüklere sahip birçok mikrodenetleyici ve kontrolör kullanılmaktadır. Denetleyiciler geliştirilecek sisteme göre seçilerek maliyet,

hassasiyet, doğruluk ve zaman açısından avantajlar sağlanmaktadır.

Kontrolörlerin, elektronik tasarımların performansını değiştirecek özellikleri karşılaştırılarak avantaj ve dezavantajları Tablo 1'de gösterilmiştir. Bu tablodan açıkça görüldüğü gibi çalışma performansı, işlemci (CPU) hızı, analog kanal sayısı, DGM (PWM) kanal sayısı, soket tipi ve maliyet açısından dsPIC'in üstünlükleri bulunmaktadır. İşlem yapma hızının yüksek olması kontrol ve karar mekanizmalarının yönteminde optimum kararlılık ve doğruluk sağlamaktadır. Soket tipinin DIP (Dual in-line package) olması uygulama geliştirme aşamasında maliyeti azaltmakta ve kolaylık sağlamaktadır. Tablodan görüldüğü gibi, yaklaşık olarak aynı giriş çıkış portu sayısına ait PIC ve dsPIC işlemcileri karşılaştırıldığında, işlemci hızı, çalışma hızı analog kanal sayısı, DGM kanal sayısı gibi birçok özellik açısında dsPIC'in sistem geliştirme sürecinde daha etkin çözümler sunacağı görülecektir. dsPIC'ten yüksek hızlarda çalışabilen ve giriş/çıkış pin sayısı fazla olan DSP (Dijital Sinyal İşlemci) soket tipinin SMD (Yüzeye Montajlı) olması ve maliyetinin yüksek olması uygulama sistemleri ve eğitimde kullanılmasını kısıtlamaktadır.

Tablo1. dsPIC, PIC ve DSP kontrolörlerinin karşılaştırılması[13-18]

(Comparison of dsPIC, PIC, and DSP controllers [13-18])

	dsPIC30F4011	**PIC18F452**	**TMS320F2812 DSP**
Maksimum Çalışma Hızı	8.33-ns Çevrim Süresi	25-ns Çevrim Süresi	6.67-ns Çevrim Süresi
İşlemci Hızı	30 MIPS	10 MIPS	120 MIPS'e kadar artırılabilir
Giriş/Çıkış Pinleri	30 Pin	28 Pin	56 Pin
Analog Giriş	9 Kanal	5 Kanal	16 Kanal
Analog Çözünürlük	10 Bit	10 Bit	12 Bit
DGM Kanal Sayısı	6 Kanal	2 Kanal	16 Kanal
Soket Tipi	DIP/SMD	DIP/SMD	SMD
Mimari	16 Bit	8 Bit	32 Bit
Maliyet	€7.1	€7.39	€33

Açıklamalardan görüldüğü gibi sıklıkla kullanılan birçok kontrolör elemandan daha yüksek performanslı olarak çalışabilen ve maliyeti düşük bir mikrodenetleyici olan dsPIC birçok uygulama için ideal bir elemandır. Fakat simulasyon programlarında dsPIC mikrodenetleyicilerinden birçoğunun bulunmaması, bazı derleyicilerde kütüphanelerinin bulunmaması ve örnek uygulamaların az olması dsPIC'in kullanımını kısıtlamaktadır [13]. Bu amaçla çalışmada, yukarıda belirtilen eksikliklerin giderilmesi için dsPIC temelli uygulama geliştirme süreci detaylı olarak ele alınmıştır.

3. dsPIC İLE UYGULAMA GELİŞTİRMEDE MİKRODENETLEYİCİ KÜTÜPHANELERİNİN OLUŞTURULMASI (CREATING THE MICROCONTROLLER LIBRARIES FOR dsPIC BASED APPLICATION DEVELOPMENT)

Uygulama geliştirme sürecinde, mikrodenetleyicilerde kullanılacak olan birimlerin önceden yapılandırılması ve kullanıma hazır hale getirilmesi gerekmektedir. Bu amaçla kullanılacak birimlerin yapılandırılması için programlama dillerinde kütüphaneler oluşturulmaktadır. Esnek çalışma yapısı, kütüphanelere kolay müdahale edilmesi ve gelişmiş arayüzü nedeni ile bu çalışmada C programlama dili ve HI-TECH C derleyicisi tercih edilmiştir. HI-TECH C dili ile dsPIC'lerde uygulama geliştirmek amacı ile kütüphaneler bulunmamaktadır. Çalışmada, HI-TECH C DSPICC 9.60 deneme sürümü derleyicisi için kütüphaneler oluşturulmuştur. Oluşturulan kütüphaneler *.c dosya uzantısı ile *C:\ProgramFiles\HI-TECHSoftware\DSPICC\std\9.60\include* klasörüne kaydedilerek kullanıma hazırlanmıştır.

3.1 RS-232 İletişim Protokolü Kütüphanesi (RS-232 Communication Protocol Library)

Günümüzde birçok uygulamada denetleyiciler bilgisayar ile haberleşerek çalışmaktadır [19 - 21]. dsPIC30F4011 içerisinde 2 adet RS-232 iletişim kanalı bulunmaktadır. Bu kanallar aracılığı ile aynı işlemci iki bilgisayar ile aynı anda yönetilebilmektedir. Aşağıdaki program komutları dsPIC30F4011' in 1. seri iletişim kanalını yapılandırmak amacı ile hazırlanmıştır.

Tablo 2. *UART1* haberleşme protokolü kontrolörleri [14]

(*UART1* communication protocol controllers [14])

	U1MODE	**U1STA**	**IEC0**
Bit 0	STSEL	URXDA	INT0IE
Bit 1	PDSEL(0)	OERR	IC1IE
Bit 2	PDSEL(1)	FERR	OC1IE
Bit 3	-	PERR	T1IE
Bit 4	-	RIDLE	IC2IE
Bit 5	ABAUD	ADDEN	OC2IE
Bit 6	LPBACK	URXISEL(0)	T2IE
Bit 7	WAKE	URXISEL(1)	T3IE
Bit 8	Reserved	TRMT	SPI1IE
Bit 9	Reserved	UTXBF	U1RXIE
Bit 10	ALTIO	UTXEN	U1TXIE
Bit 11	Reserved	UTXBRK	ADIE
Bit 12	-	-	NVMIE
Bit 13	USIDL	-	SI2CIE
Bit 14	-	-	MIWCIE
Bit 15	UARTEN	UTXISEL	CNIE

Tablo 2'de *UART1* haberleşme protokolü kontrolörleri görülmektedir. *U1MODE*, 1. haberleşme bölgesinin genel ayarlarının yapıldığı kısımdır. *U1MODE*'nin 0. ve 15. bitleri lojik 1 yapılarak haberleşme durdurma biti (*STSEL*) seçilmiş ve haberleşme protokolü aktif (*UARTEN*) edilmiştir. *U1STA*, 1. haberleşme kanalının kesme vb. ayarlarının kontrol edildiği bölümdür. Kontrolörün 5. ve 10. bitleri aktif duruma getirilerek *ADDEN* ve *UTXEN* özellikleri kullanıma hazırlanmıştır. *ADDEN* ile karakter algılama özelliği, *UTXEN* ile veri okuma aktif hale getirilmiştir. Tek yönlü veri gönderme sistemlerinde *UTXEN* biti 0 yapılarak mikrodenetleyicinin veri göndermesi engellenebilmektedir.

U1BRG seri iletişim hızının belirlendiği kontrol bölgesidir. Kontrolör ve bilgisayarın iletişim hızının aynı olması gerekmektedir aksi halde iletişim sırasında verilerde bozulma ve yanlış algılama sorunları yaşanacaktır. dsPIC30F4011'in *U1BRG* kontrol bölgesine atanan değer kontrolörün iletişim hızını belirlemektedir. Denklem 1' de mikrodenetleyicinin komut işleme hızı hesaplanmıştır. Denklem 2' de komut işleme hızı kullanılarak kontrol bölgesine atanacak sayısal değerin hesabı yapılmıştır[14].

```
void seri1_init( float Fosc, long baudrate )
{
        float Fcy; U1MODE=0x8001;
         U1STA=0X0420;  Fcy=Fosc/4;
         U1BRG = ((Fcy)/(16*baudrate))-
        1;
         IEC0bits.U1RXIE=1;
}
void putch1(unsigned char byte)
{
        while(!IFS0bits.U1TXIF);
        U1TXREG = byte;
}
unsigned char getch1(void)
{
        while(!IFS0bits.U1RXIF);
        return U1RXREG;
}
```

$$f_{CY} = \frac{f_{OSC}}{4} \quad (1)$$

$$BRG = \frac{f_{CY}}{16 \times BaudRate} - 1 \quad (2)$$

Burada; f_{cy} komut işleme hızını (Hz), f_{osc} osilatör frekansını (Hz), *BRG* 16 bit kontrol bölgesine atanacak sayısal değeri, *Baud Rate* seri iletişim bilgi akış hızını ifade etmektedir. Denklem 2,*U1BRG* kontrolörüne yazılarak kullanıcının bant genişliği ve mikrodenetleyicinin çalışma hızını girmesi ile çalışma hızı için gerekli ayarlar yapılmaktadır. *IEC0* kontrol bölgesi ile mikrodenetleyicinin kesme ayarları yapılandırılmaktadır. Bu bölgede, *U1RXIE* konumu lojik 1 yapılarak UART haberleşmede veri okuma kesmesi aktif duruma getirilmiştir.

Oluşturulan kütüphane içerisine *putch1* alt programı ile veri gönderme için gerekli kodlar yapılandırılmıştır. *Getch1* alt programı ile veri okuma için gerekli kodlar yapılandırılmıştır. Oluşturulan kod yazılımı "seri.c" isimli olarak "*C:\ProgramFiles\HI-TECHSoftware\DSPICC\std\9.60\include*" klasörüne kaydedilerek kullanıma hazır hale getirilmiştir.

3.2 DGM Kütüphanesinin Oluşturulması (Creating PWM Libraries)

DGM, güç elektroniği, motor kontrol uygulamaları gibi alanlarda anahtarlama elemanlarının kontrolü için kullanılan yüksek frekanslı anahtarlama sinyalleridir. Endüstriyel sistemlerde DGM sinyallerinin üretilmesi için çeşitli kontrolörler kullanılmaktadır[2-8]. Kullanılan kontrolör elemanın ürettiği anahtarlama sinyalinin kalitesi, anahtarların iletim ve kesime gitmesi sırasında doğabilecek kararsız durumları engellemektedir. Bu nedenle anahtarlama elemanlarının kontrolünde kararlı, doğru ve dalga şekli düzgün anahtarlama sinyalleri tercih edilmektedir. dsPIC kararlı çalışması ve yüksek çalışma frekansı sebebi ile birçok motor kontrol ve güç elektroniği sisteminde tercih edilmiştir[1-3, 5-8]. Kullanılacak anahtarlama sinyalinin frekansının hesaplanması için denklem 3 ve denklem 4 kullanılmıştır[14].

$$T_{CY} = \frac{1}{f_{CY}} \quad (3)$$

$$PTPER = \frac{1}{T_{CY} \times f_{PWM}} - 1 \quad (4)$$

Burada; T_{CY} komut işleme periyot süresini (s), PTPER kontrol bölgesine atanacak sayısal değeri, f_{PWM} üretilecek darbe genişlik modülasyonu sinyalinin hızını (Hz) ifade etmektedir. dsPIC30F4011'in DGM kanallarını kullanmak için yazılan kütüphane komutları aşağıda verilmiştir. *PTPER* kontrol bölgesi, üretilecek olan DGM sinyalinin frekansının belirlendiği kısımdır. Sinyalin frekansı denklem 4'te görüldüğü gibi, kullanılan osilatöre bağımlı olarak değişmektedir. Kullanım kolaylığı açısından formül kütüphaneye eklenerek yazılımcının üretmek istediği anahtarlama frekansı ve osilatörün frekansını girerek DGM sinyalinin frekansını belirlemesi sağlanmıştır.

Tablo 3'de DGM kanallarının kontrolörleri görülmektedir. dsPIC30F4011 motor kontrol uygulamaları ve eviricilerin kontrolünde kolaylık sağlamak için DGM sinyallerinin tersini de üretmektedir. *PWMCON1* kontrol bölgesinde kullanılacak olan DGM kanallarının seçilme işlemi gerçekleştirilmektedir. Kütüphane içerisine yazılan komutlarda bütün kanalların aktif duruma getirilmesi için *PENxL* ve *PENxH* bitleri 1 olarak belirlenmiştir. *PTMODx* kontrol bölgeleri üretilecek DGM sinyallerine ait kesmelerin yapılandırıldığı bölgelerdir. Yapılan uygulamalarda kesme kullanılmayacağı için bu bölüm lojik 0 olarak belirlenmiştir. Oluşturulan kod yazılımı "*pwm.c*" isimli olarak "*C:\ProgramFiles\HI-TECHSoftware\DSPICC\std\9.60\include*" klasörüne kaydedilerek kullanıma hazır hale getirilmiştir.

Tablo 3. DGM kontrolörleri [14]

(PWM controllers [14])

	PWMCON1	**PTCON**
Bit 0	PEN1L	PTMOD (0)
Bit 1	PEN2L	PTMOD (1)
Bit 2	PEN3L	PTCKPS (0)
Bit 3	-	PTCKPS (1)
Bit 4	PEN1H	PTOSP (0)
Bit 5	PEN2H	PTOSP (1)
Bit 6	PEN3H	PTOSP (2)
Bit 7	-	PTOSP (3)
Bit 8	PTMOD1	-
Bit 9	PTMOD2	-
Bit 10	PTMOD3	-
Bit 11	-	-
Bit 12	-	-
Bit 13	-	PTSIDL
Bit 14	-	-
Bit 15	-	PTEN

```
void pwm_init( float Fosc, float Fpwm)
{
        float Fcy,Tcy;
        Fcy=Fosc/4;
        Tcy=1/Fcy;
        PTPER=(1/(Tcy*Fpwm))-1;
        PTCON = 0x8000;
        PWMCON1 = 0x0077;
}
```

3.3 Analog – Dijital Veri Dönüşüm Kütüphanesinin Oluşturulması

(Creating Analog/Digital Data Conversion Library)

Kapalı döngü çalışan ölçüm sistemlerinde analog veri okuma yaygın olarak kullanılmaktadır [2-12]. Analog veri dönüşümleri çeşitli kontrolörlerin içerisinde bulunan alt mimariler ile sayısal verilere dönüştürülerek algılanmaktadır.

dsPIC30F4011'in 9 adet 10 bit çözünürlüklü 500 ksps örnekleme hızına sahipanalog girişi bulunmaktadır. Tablo 4'te analog-dijital veri dönüşümü kontrolörleri görülmektedir. Analog girişlerin kullanılması için gerekli yapılandırma kodları aşağıda *adc_init* alt programı içerisinde görülmektedir. *ADCON1* kontrol bölgesi analog girişlerinin aktif edilmesi vb. genel ayarların yapıldığı bölgedir. *ADCON1* kontrol bölgesini *ADON* altyapısı lojik 1 yapılarak analog-dijital çevrim aktif hale getirilmiştir.

10 bit analog-dijital veri dönüşümü kullanmak için *ADCON2* bölgesinin *FORM* altyapısının 2. biti lojik 1 yapılmıştır. *ADCON3* kontrol bölgesi analog dijital dönüşüm hızının yapılandırıldığı bölgedir. Analog veri okuma ayarlarında *ADCON3* kontrol bölgesinin *SAMC* ve *ADCS* altyapıları ile veri dönüştürme ve örnekleme hızı belirlenmiştir. "*read_adc*" alt programı ile analog verilerin sayısal verilere dönüştürülme işlemi gerçekleştirilmiştir. Program komutları içerisindeki "*kanal*" değişkeni analog veri dönüşümü yapılacak kanal numarasını ifade etmektedir. Oluşturulan kod yazılımı "*adc.c*" isimli olarak "*C:\ProgramFiles\HI-TECHSoftware\ DSPICC\ std\9.60\include*" klasörüne kaydedilerek kullanıma hazır hale getirilmiştir.

Tablo 4. Analog-dijital veri dönüşümü kontrolörleri [14]

(Analog/digital data conversion controllers [14])

	ADCON1	ADCON2	ADCON3
Bit 0	DONE	ALTS	ADCS (0)
Bit 1	SAMP	BUFM	ADCS (1)
Bit 2	ASAM	SMPI (0)	ADCS (2)
Bit 3	SIMSAM	SMPI (1)	ADCS (3)
Bit 4	-	SMPI (2)	ADCS (4)
Bit 5	SSRC (0)	SMPI (3)	ADCS (5)
Bit 6	SSRC (1)	-	-
Bit 7	SSRC (2)	BUFS	ADRC
Bit 8	FORM (0)	CHPS (0)	SAMC (0)
Bit 9	FORM (1)	CHPS (1)	SAMC (1)
Bit 10	-	CSCNA	SAMC (2)
Bit 11	-	-	SAMC (3)
Bit 12	-	-	SAMC (4)
Bit 13	ADSIDL	VCFG (0)	-
Bit 14	-	VCFG (1)	-
Bit 15	ADON	VCFG (2)	-

```
void adc_init()
{
        ADCON1=0X00FF;
        ADCON2=0x0100;
        ADCON3=0xF019;
        ADCHS=0x0001;
        ADCON1bits.ADON=1;
}

int read_adc(int kanal)
{
        ADCHS=kanal;
        IFS0bits.ADIF=0;
        ADCON1bits.ASAM=1;
        while(!IFS0bits.ADIF);
        ADCON1bits.ASAM=0;
        return(ADCBUF0);
}
```

4. DENEYSEL ÇALIŞMALAR (EXPERIMENTAL STUDIES)

Bu bölümde, oluşturulan kütüphanelerin kullanımının açıklanması için örnek uygulamalar geliştirilmiş ve dsPIC ve PIC mikrodenetleyicileri için aynı uygulama gerçekleştirilerek karşılaştırmalı analiz yapılmıştır.

4.1 RS-232 İletişim Örneği (RS-232 Communication Example)

Oluşturulan seri iletişim kütüphanesi kullanılarak dsPIC ile bilgisayar arasında bilgi alışverişi yapmak için bir örnek yazılım geliştirilmiştir. Örnek uygulamada bilgisayar üzerinden dsPIC'in B portundaki çıkışların lojik olarak kontrolü gerçekleştirilmiştir. Bilgisayar tarafında, *Windows* işletim sistemlerinde bulunan *HyperTerminal* kullanılarak sayısal değerler gönderilmiştir. Gönderilen sayısal değerler B portuna atanmıştır. Gelen sayısal bilgilere göre B portunun lojik

değerleri değiştirilmiştir. Seri iletişim örneğine ilişkin yazılım kodları aşağıda verilmiştir.

Geliştirilen yazılımda *"#include <seri.c>"* komutu ile seri iletişim kütüphanesi eklenmiştir. Ana programda *seri1_init* komutu ile 1. seri port kanalının yapılandırılması 9600 baud/rate oranı ile gerçekleştirilmiştir. Gelen verileri okumak amacı ile seri olarak isimlendirilen *UART1* veri okuma kesmesi kullanılmıştır. Alınan sayısal veriler *data* isimli değişkene kaydedilmiştir. Gelen veriler *ASCII* kodlama sistemi ile kodlandığından sayısal değere dönüşümü yapılmıştır. B portuna *data* değişkeninin sayısal değeri girilerek B portunun lojik değerleri değiştirilmiştir.

```
#include<htc.h>
#include<seri.c>

chardata;

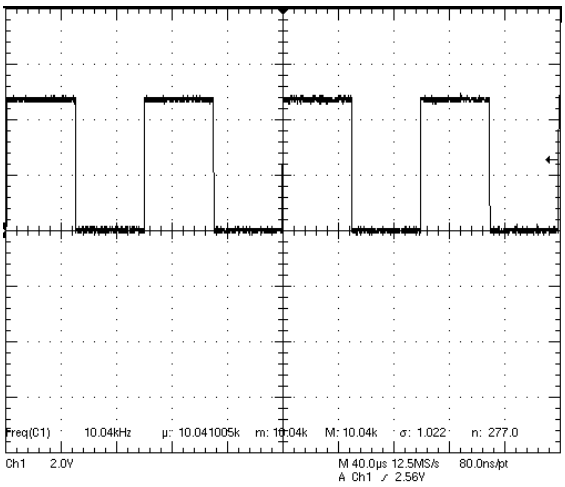
void interrupt seri(void) @U1RX_VCTR

{
        data=getch1();
        data=data-48; //ASCII Kod Dönüşümü
}

void
main(void)
{
        seri1_init(10000000,9600);
        TRISF=0xff;
        TRISB=0x00;
        for(;;)
        {
                PORTB=data;
        }
}
```

4.2. DGM Örneği (PWM Example)

Bu örnek uygulamada, oluşturulan kütüphanenin kullanımı ve DGM sinyallerinin üretilmesi gerçekleştirilmiştir. DGM sinyallerinin kullanılması için gerekli ayarlar "*pwm.c*" kütüphanesi altında gerçekleştirilmiştir. DGM ayarlarının yapıldığı *pwm.c* kütüphanesi "*#include<pwm.c>*" komutu ile programa dahil edilmiştir.



(a)

```
#include<htc.h>
#include<pwm.c>
void main(void)
{
        TRISE=0X00;
        pwm_init(10000000,10000);
        PDC1=100;
}
```

(b)

Şekil 1. a: Üretilen DGM sinyali, b: DGM kodları (a:The PWM signal generated, b: PWM codes)

Ana programda "*pwm_init(10000000,10000)*" komutu ile kütüphane içerisindeki alt program kullanılarak üretilecek DGM sinyalinin ayarları gerçekleştirilmiştir. Komut içerisindeki sayısal değerler kontrolörün çalışma hızını ve üretilecek olan DGM sinyalinin frekansını belirtmektedir. Ana program içerisinde kullanılan "*PDC1=100*" komutu, üretilen DGM sinyalinin genliğini belirtmektedir. Üretilen DGM sinyali Şekil 1 (a)'da gösterilmiştir. Girilen frekans ve genlik doğrultusunda DGM sinyalinin başarılı olarak üretildiği şekilden açıkça görülmektedir.

4.3. Analog – Dijital Veri Dönüşüm Örneği (Analog-Digital Data Conversion Example)

Analog veri dönüşüm işlemi için gerekli ayarlar "*adc.c*" isimli kütüphane içerisinde gerçekleştirilmiştir. Uygulamada analog-dijital veri dönüşümü yapılması için öncelikle kütüphane "*#include<adc.c>*" komutu ile yazılıma dahil edilmiştir.

Gerçekleştirilen uygulamada dsPIC30F4011'in *AN0* kanalından analog dijital veri dönüşümü yapılmıştır. Ana program içerisinde öncelikle "*adc_init*" komutu ile analog-dijital veri dönüşümünün kullanılması için gerekli ayarlar yapılmıştır. "*veri=read_adc(0)*" komutu ile *AN0* kanalından okunan analog verilerin sayısal verilere çevrilip "*veri*" değişkenine atanması sağlanmıştır. Sürekli olarak veri okuma işleminin gerçekleşmesi amacı ile sistem kısır döngüye sokulmuştur. Böylece kontrolör çalıştığı sürece dsPIC30F4011'in *AN0* kanalından analog veri okuma işlemi gerçekleştirilmiştir.

```
#include<htc.h>
#include<adc.c>
long veri=0;
void main(void)
{
        TRISB=0XFF;
        adc_init();
        while (1){
                veri=read_adc(0);}
}
```

4.4. Örnek Deneysel Çalışma: DA Motoru Hız Kontrol Sistemi

(Sample Experimental Study: DC Motor Speed Control System)

Önceki uygulamalarda oluşturulan kütüphaneler ayrı ayrı kullanılarak seri iletişim, DGM ve analog birimlerin kullanımı açıklanmıştır. Bu uygulamada, kütüphaneler tümleşik olarak kullanılmış ve bir kapalı döngü motor hız kontrol sistemi uygulaması gerçekleştirilmiştir. Ayrıca PIC18F452 mikrodenetleyicisi için de aynı hız kontrol sistemi yazılımı gerçekleştirilerek karşılaştırılmalı performans analizi yapılmıştır.

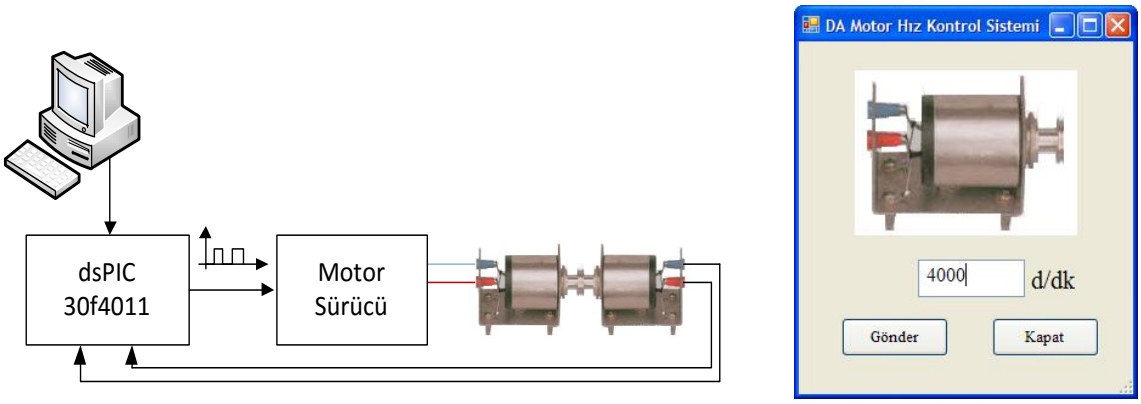
Uygulamada birbirine akuple yapıda sabit mıknatıslı bir DA motor ve sabit mıknatıslı bir DA jeneratör kullanılmıştır. Sabit mıknatıslı DA motorun hızının hesaplanması için gerekli matematiksel ifade denklem 5' te verilmiştir [22].

$$\omega_m = \frac{1}{k_E}\left(V_t - \left(\frac{R_a}{k_T} \times T_{em}\right)\right) \quad (5)$$

Burada; ω motor hızını (rad/s), k_E motorun gerilim sabitini, V_t terminal gerilimini (V), R_a endüvi direncini (Ω), k_T motorun moment katsayısını, T_{em} elektromekanik torku (Nm) ifade etmektedir. Denklemde görüldüğü gibi, sabit mıknatıslı DA motorunun hızı terminal gerilimine (V_t) bağlı olarak değişmektedir. Sistemde motorun hızı, motor sürücüsünün çıkış gerilimi ile doğru orantılı olarak değişmektedir. Takojeneratörde üretilen gerilime ait matematiksel ifade denklem 6'da görülmektedir[23]. Görüldüğü gibi, takojeneratörde üretilen gerilim takojeneratörün devir sayısı ile doğru orantılı olarak değişmektedir.

$$E = \frac{Z \times n \times \emptyset}{60} \quad (6)$$

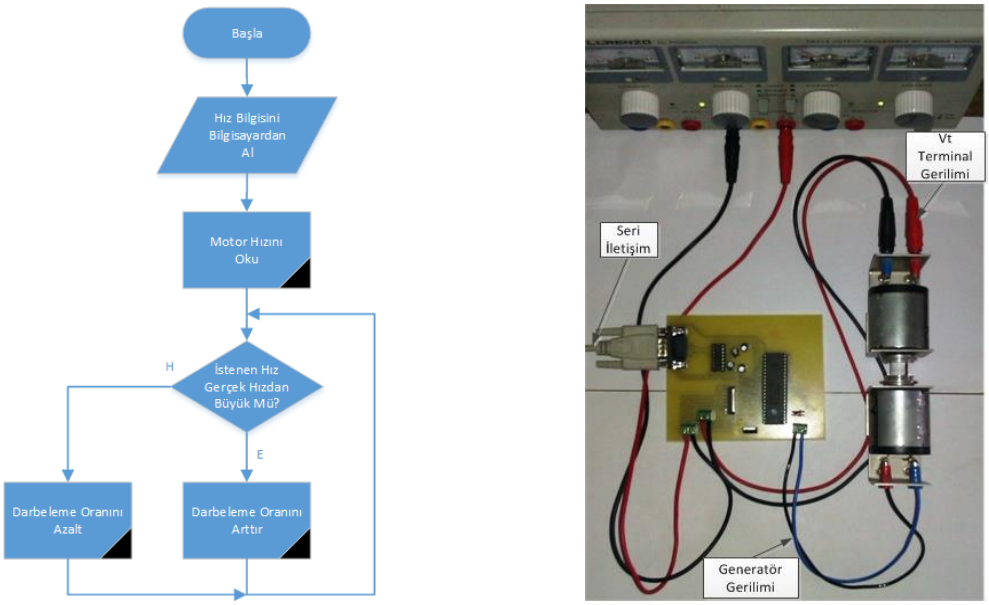
Burada; E endüvide endüklenen gerilimi (V), Z endüvi üzerindeki toplam iletken sayısını, n devir sayısını (n/min), $\emptyset$ bir kutup akısını (Wb) ifade etmektedir. Teorik açıklamalar doğrultusunda gerçekleştirilen uygulamanın blok diyagramı Şekil 2' de gösterilmiştir.



Şekil 2. a: Kapalı döngü motor hız kontrol sistemi, b: Referans hız bilgisinin gönderilmesi için tasarlanan arayüz (a: Closed-loop speed control system of DC motor, b: The computer interface designed for submitting reference speed)

Sistemde motorun hız bilgisinin bilgisayardan gönderilmesi için C# programa dili kullanılarak oldukça basit bir arayüz geliştirilmiştir. Geliştirilen arayüzde motorun hız bilgisi kullanıcı tarafından girilerek seri iletişim kanalı aracılığı ile dsPIC'e gönderilmektedir. Geliştirilen arayüz Şekil.3'te görülmektedir. Arayüzden girilen hız bilgisi, kontrolörde referans hız olarak alınmaktadır.

Şekil 4 (a)'da PIC ve dsPIC için gerçekleştirilen motor hız kontrol sistemi yazılımının blok diyagramı görülmektedir. Hazırlanan yazılımda öncelikle motorun dönmesi istenen hız değeri (referans hız) bilgisayar arayüzü üzerinden alınmıştır. Motorun gerçek hızı takojeneratörde üretilen gerilim ölçülerek hesaplanmıştır. Gerçek hız ile referans hız karşılaştırılarak, referans hızın gerçek hızdan büyük olması durumunda DGM sinyalinin darbeleme oranı artırılmaktadır. Böylece motora uygulanan gerilim artırılarak motorun hızı arttırılmaktadır. Öte yandan referans hız değerinin gerçek hızdan küçük olması durumunda kontrolör DGM sinyalinin darbeleme oranını azaltarak motorun yavaşlamasını sağlamaktadır. Sistem motor hızı istenilen değere sabitlenene kadar hızı denetlemektedir.



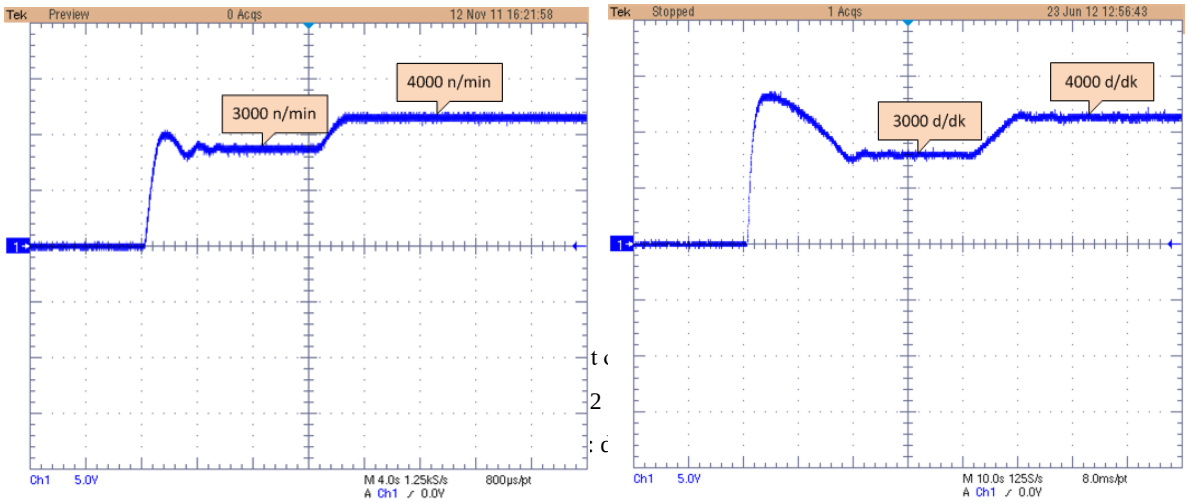
(a) (b)

Şekil 3. a: Yazılıma ait akış diyagramı, b: dsPIC kontrollü DA motor hız kontrol devresi

(a: Software flowchart, b: dsPIC controlled DC motor speed control circuit)

DA motor kontrolü için oluşturulan uygulama devresinin görüntüsü Şekil 4 (b)'de görülmektedir. Devrede motor kontrolü için L298 motor kontrol entegresi kullanılmıştır. L298 içerisinde, transistörlerden oluşan bir motor hız kontrol ünitesi bulunmaktadır. Entegre, uygulanan DGM'ye göre çıkış gerilimini değiştirmektedir. Örnek uygulamada arayüzden devir sayısı 3000 d/dk girilmiştir. Sistemin ilk çalışma anında motor durduğu için referans hız gerçek hızdan büyüktür. Mikrodenetleyici, DGM darbeleme oranını artırarak motora uygulanan gerilimi artırmıştır. Motorun ilk kalkınma anında hızı istenen değerin üzerine çıkmıştır. Sistem motorun hızını takojeneratörden üretilen gerilime göre hesaplayarak DGM sinyalinin darbeleme oranını azaltmıştır. Bu çalışma süreci, motor hızı referans hıza sabitlenene değin devam etmiştir. Şekil 5 (a)'da dsPIC kontrollü deneysel çalışma süresince takojeneratörde üretilen gerilim görülmektedir. Şekil 5 (b)'de ise aynı sistemde PIC kontrollü deneysel çalışmada takojeneratörde üretilen gerilim görülmektedir.

Şekil 5 (a) ve Şekil 5 (b)karşılaştırıldığında, kalkınma anında dsPIC'in işlem yapma hızı yüksek olduğundan motor devrinde düşük bir miktar sıçrama olmuş ve motor hızı yaklaşık 5 s'de referans değere sabitlenmiştir. Kalkınma anında PIC mikrodenetleyicisinin analog çevrim hızı ve işlem hızı düşük olduğundan motor istenilen devir sayısına 20 s'de ulaşabilmiştir. Motor 3000 devirde dönerken arayüzden referans hız değeri 4000'e çıkarılmıştır. Bu durumda kontrolörler DGM sinyallerinin genliğini artırarak motor hızının artmasını sağlamıştır. Hızlanma durumu incelendiğinde dsPIC 2 s gibi kısa bir sürede motor devrini 4000 d/dk'ya getirirken PIC mikrodenetleyicisinde bu işlem yaklaşık 8s sürmüştür. Kapalı döngü çalışan sistem motor devri 4000'e geldiğinde motor gerilimini sabitlemiştir.



(a) (b)

Şekil 5. Takojeneratörde üretilen gerilime ait osiloskop görüntüleri a: dsPIC kontrollü, b: PIC18F452 kontrollü

(Voltage graphs of tachogenerator a: dsPIC controlled b: PIC18F452 controlled)

Uygulamada sıkça kullanılmakta olan motor hız kontrol sistemlerinin yüksek hızlı ve minimum hata ile çalışması istenmektedir. Aksi durumda malzemeler ve sistemler üzerinden fiziksel arızalar oluşmaktadır. Gerçekleştirilen bu uygulama çalışması ile dsPIC ve PIC mikrodenetleyicilerinin karşılaştırılmalı performans analizleri yapılmış ve dsPIC'in işlem yapma hızı ve çözünürlük açısından üstün özelliklere sahip olduğu gösterilmiştir.

5. SONUÇ (CONCLUSION)

Günümüzde, endüstriyel ortamlarda hızlı, kararlı ve yüksek çözünürlüklü çalışan sistemler sıklıkla kullanılmaktadır. Bu çalışmada, kontrol ve algılama mekanizmasındaki üstünlükleri nedeni ile endüstriyel ortamlarda kullanımı gittikçe yaygınlaşmakta olan dsPIC mikrodenetleyicisi hakkında genel bilgiler verilmiştir. dsPIC ve diğer kontrolörler karşılaştırılarak ele alınan dsPIC denetleyicisi ile yaklaşık özelliklere sahip denetleyicilere göre maliyet, hız, çözünürlük ve donanımsal yapı bakımından üstünlükleri bulunduğu görülmüştür.

Çalışmada, dsPIC kontrolörü kullanılarak uygulama geliştirme sürecinin örneklenmesi amacıyla C dilinde kütüphaneler oluşturulmuş ve kütüphanelerin açıklanması örnek uygulamalarla desteklenmiştir. Böylece genel

olarak mikrodenetleyici kütüphanelerinin alt yapısı anlatılmış ve kullanıcıların kendilerine özgün kütüphaneler oluşturmasında fikir sahibi olmaları sağlanmıştır. Oluşturulan özgün kütüphanelerin uygulama yazılımlarında kullanılması ve alt programların çalışması gösterilmiştir.

Kontrol ve geribildirim aşamasında karşılaştırma yapılması açısından uygulamada sıklıkla kullanılan DA motor hız kontrol sistemi uygulaması gerçekleştirilmiştir. Yapılan çalışmada endüstriyel ortamlarda ve eğitimde yaygın olarak kullanılan PIC18F452 mikrodenetleyicisi ve dsPIC30F4011 karşılaştırılmıştır. Deneysel çalışmada DA motorun kapalı döngü hız kontrol sistemini gerçekleştirecek mikrodenetleyici yazılımları her iki kontrolör için yazılmış ve uygulama sonuçları analiz edilmiştir. Deneysel analiz sonucunda, dsPIC'in işlem ve algılama-dönüştürme hızı nedeni ile sistem performansını artırdığı ve motor hızının istenilen değere getirilmesi için geçen süreyi azalttığı görülmüştür. Çalışma sonucunda, diğer mikrodenetleyicilere göre daha hızlı ve kararlı çalışan dsPIC mikrodenetleyicisinin kullanımı açıklanarak kullanıcıların daha kararlı ve hızlı çalışan sistemler geliştirmesi hedeflenmiştir.

Sunulan bu çalışmanın özellikle uygulama çözümü geliştirme sürecinde akademisyen ve mühendislere önemli katkılar sağlayacağı düşünülmektedir. Ayrıca çalışma eğitim amaçlı olarak lisans ve ön lisans düzeyindeki ilgili derslere bir kaynak teşkil edecektir.

KAYNAKLAR (REFERENCES)

[1] Kanokvate, T., Seubsuang, K., Prapon, J. ve Pakasit, S., "An Improvement on Position Estimation and Start up Operation for Switched Reluctance Motor Drives", IEEE Power Engineering Society General Meeting, Florida, USA, 1-6, 24-28 Haziran 2007.

[2] Mboup, A. B., Ndiaye, P. A., Guerin, F. ve Lefebvre D., "Control Design for Hybrid Electrical Energy Systems Based on DC/DC Converters Duty Cycle Value", Ever Monaco 2009, Monaco, 99-108, 26-29 Mart 2009.

[3] Miasa, S., Al-Mjali, M. ve Al-Haj Ibrahim, A., "Fuzzy Control of a Two-Wheel Balancing Robot Using DSPIC", 7th International Multi-Conference on Systems Signals and Devices (SSD), Amman, 1-6, 27-30 Haziran 2010.

[4] Bassi, E., Benzi, F., Almeida, L. ve Nolte, T., "Powerline Communication in Electric Vehicles", IEEE International Electric Machines and Drives Conference (IEMDC), Miami, 1749-1753, 3-6 Mayıs 2009.

[5] Ramí, Rez, F.A., Arjona, M.A. ve Hernandez, C., "Emulation of a Single Phase dsPIC Based Grid-Connected Wind Energy Conversion System", International Conference on Electrical Machines XIX (ICEM), Rome, 1-6, 6-8 Eylül 2010.

[6] Nanakos, A.Ch., Polyzos, N.P., Tatakıs, E.C. ve Karnavas, Y.L., "A Grid Connected P/V System With Energy Storage Capability", International Conference on Renewable Energy Sources (RES'08), Corfu, 257-262, 26-28 Ekim 2008.

[7] Yuvarajan, S. ve Kaderbhai, M., "Hybrid Renewable Energy System with Wind Turbine and PV Panels", IEEE Green Technologies Conference, Grapevine, 1-4, 15-16 Nisan 2010.

[8] Kamil, M., "Grid-Connected Solar Microinverter Reference Design Using a dsPIC® Digital Signal Controller", Microchip, 2011.

[9] Siddiqui, R.A., Amer, W., Ahsan, Q., Grosvenor, R.I. ve Prickett, P.W., "Multi-Band Infinite Impulse Response Filtering Using Microcontrollers for e-Monitoring Applications", Microprocessors and Microsystems, Cilt 31, No 6, 370-380, 2007.

[10] Tian X. ve Zhao Z, "Heart Sound Acquisition Based on PDA and Bluetooth", 4th International Conference on Biomedical Engineering and Informatics (BMEI), Shanghai, China, 773-776, 15-17 Ekim. 2011.

[11] Premaratne, P., Ajaz, S., Monaragala, R., Bandara, N. ve Premaratne, M., "Design and İmplementation of Edge Detection Algorithm in Dspic Embedded Processor", 5th International Conference on Information and Automation for Sustainability (ICIAFs), Sri Lanka, 8-13, 17-19 Aralık 2010.

[12] Negishi, Y.ve Kawaguchi, N., "Instant Learning Sound Sensor: Flexible Environmental Sound Recognition System", Networked Sensing Systems, 2007. INSS '07. Fourth International Conference on , Braunschweig, Germany, 305, 6-8 Haziran 2007.

[13] Irmak E. ve Guler N., "Application of a High Efficient Voltage Regulation System with MPPT Algorithm", In J Electr Power Energy Syst, Cilt 44, No 1, 703-712, 2013.

[14] dsPIC30F4011/4012 Data Sheet, Microchip, 2010.

[15] dsPIC30F4011,http://tr.farnell.com/microchip/dsPIC30F4011-30i-p/mcu-dsp-16bit-30mips-48k-flsh-dip/dp/1439598, Feb 15th 2011.

[16] TMS320F2812DSP,http://tr.farnell.com/texas-instruments/tms320f2812zhhs/dsc-32bit-128k-flash-179bga/dp/1962187, Feb 15th 2011.

[17] PIC18F452,http://tr.farnell.com/jsp/search/browse.jsp?N=2030+203063&Ntk=gensearch&Ntt=18f452&Ntx=mode+matchallpartial, Feb 15th 2011.

[18] Siemens S7 400 PLC - CPU 416F-2DP, http://www.megaelektrikshop.net/urunler_4208_kategori_6517.html, Feb 15th 2011.

[19] Çolak, İ., Irmak, E., Güler, N. ve Issı, F., "Remote Monitoring and Reporting the Parameters of A Hybrid Power System via Internet", *EVER'12*, Monaco (2012).

[20] Irmak E. ve Vadi S., "Computer Based Implementation of Speed Control Experiment Depending on Frequency Variation for Induction Motors", Journal of The Faculty of Engineering and Architecture of Gazi University, Cilt: 26, No 1, 57-62, 2011.

[21] Colak İ., Irmak E., Kabalcı E. ve Issı, F., "Design and Implementation of a Remote Laboratory Platform Using MATLAB Builder for NE." Comput. Appl. Eng. Educ., 2011, doi: 10.1002/cae.21553.

[22] N. Mohan, W. P. Robbin, ve T. Undeland, "Motor Drive Application", Power Electronics, 2. Baskı, New York: Wiley, 404-426, 1995.

[23] Bal G., Doğru Akım Makinaları ve Sürücüleri, 3.Baskı, Ankara: Seçkin Yayıncılık, 2008.