

Üç Fazlı Makine Sürücülerini İçin Alan Yönelimli Kontrolün Donanım Tabanlı Gerçeklenmesi

Hardware Implementation of Field Oriented Control for Three Phase Machine Drives

Burak TÜFEKÇİ^{1,3}, Bugra ÖNAL^{2,3}, Hamza DERE^{2,3} ve H. Fatih UĞURDAĞ²

¹CS Bölümü, Özyeğin Üniversitesi

²EE Bölümü, Özyeğin Üniversitesi

³Semimobility Teknoloji Ltd.

İstanbul, Türkiye

burak.tufekci.17309@ozu.edu.tr

Özetçe —Bu bildiri, üç fazlı makine sürücülerini için “Alan Yönelimli Kontrol” (FOC) yönteminin algılayıcısız uygulaması olan “Birim Amper Başına Maksimum Tork” (MTPA) ve “Akı Zayıflatma” tekniklerinin FPGA üzerinde yüksek anahtarlama frekanslı gerçeklemelerini sunmaktadır. Hem “Fırçasız DC” (BLDC) motorlar hem de “Sabit Miknatıslı Senkron Motor” (PMSM) için ortak bir mimari oluşturulmuştur. İlgili kontrolcü, “Uzay Vektör Modülasyonu” (SVM) kullanılarak gerçekleştirildi. Kullanıcı arayüzü, gerçek zamanlı tork-zaman, hız-zaman ve akım-zaman grafiklerini kullanıcıya sunabilecek şekilde tasarlandı. Bu arayüz, FPGA'nın PS kısmı üzerinde çalışmakta ve kullanıcıya UART üzerinden bağlanmaktadır. Tüm sistem, benzetim ile doğrulanmıştır.

Anahtar Kelimeler—FPGA, Algılayıcısız FOC, MTPA, Akı Zayıflatması, SVM

Abstract—This paper presents a high switching frequency FPGA implementation of Maximum Torque Per Ampere (MTPA) and Flux Weakening which are branch of Field Oriented Control (FOC) method for 3-phase machine drives. A common architecture has been constructed for both BrushLess DC motors (BLDC) and Permanent Magnet Synchronous Motors (PMSM). For this purpose, the controller module was implemented using Space Vector Modulation (SVM) technique. The user interface module was designed to provide real-time torque-time, speed-time, and current-time plots for the user. This interface runs on the PS part of the FPGA and interacts with the user through a UART. The entire system has been verified through simulation.

Keywords—FPGA, Sensorless FOC, MTPA, Flux Weakening, SVM

I. GİRİŞ

Yarı iletken teknolojisindeki gelişmeleri takiben, yüksek frekanslı ve daha az enerji kayıplı güç anahtarları çözümleri ortaya konuldu [1]. Bu gelişmeler, bize güç sistemleri tasarımı konusunda yüksek frekanslı çözümler bulmak için motivasyon verdi. Böylelikle, daha yüksek frekans yaklaşımına dayalı üç fazlı makine denetleyicisi tasarımı sunmak istedik. Bu tasarım için, enerji verimliliği, yüksek performans, paralel işleme ve

yeniden yapılandırılabilir bir ortam sunması nedeniyle FPGA tercih edildi.

Makine denetleyicisi tasarımında makinenin sargı yapısına ve kullanım alanlarına göre farklı kontrol yöntemleri vardır [2]. Bu kontrol methodlarından “Alan Yönelimli Kontrol” (FOC), her ne kadar yüksek algoritma karmaşıklığına sahip olsa da, yüksek ve düşük hızlarda sinüsoidal ve trapezoidal kontrol ile karşılaştırıldığında daha kesin sonuç verdiği için dolayı hız ve tork kontrolü için tercih edildi.

Literatürde Kung ve ark., FOC ile hız kontrolünün FPGA tabanlı tasarımını önermişlerdir [3]. Çalışmalarında, sensörsüz FOC ve faz kilitli döngü kullanarak kayan mod gözlemcili bir tasarım gerçekleştirilmiştir. Hız bilgileri kullanıcı tarafından NIOS II işlemci kullanılarak üretilmiş ve diğer tüm topolojiler FPGA üzerinde gerçekleştirilmiştir. Yazarların elde ettiği sonuçlara göre, durma durumundan hızlanma durumuna geçerkenki back-emf grafiklerini kendi yaklaşımlarını kullanarak pürüzsüz hale getirilebileceğini iddia etmektedirler. Suneeta ve ark., üç fazlı “Fırçasız DC” (BLDC) motorların kontrolünü FPGA tabanlı sunarlar [4]. BLDC motorların kontrolünde, FPGA veya mikrodenetleyici kullanımının sistem gereksinimlerine bağlı olduğu sonucuna varılmıştır. FPGA tabanlı motor kontrolün sağladığı yüksek tasarım özgürlükleri ve mikrodenetleyici tabanlı kontrole göre daha güvenli olması nedeniyle mikrodenetleyici tabanlı kontrolden daha verimli olduğunu gösterilmiştir. Mikrodenetleyici tabanlı motor kontrolü, FPGA tabanlı denetleyiciye göre daha hızlı tasarım geliştirilmesi ve daha ucuz olması nedeniyle daha verimlidir. Bu yüzden sistem gereksinimlerine göre motor kontrolörünün seçilmesi gereklidir.

Babu ve Athul, PI denetleyici perspektifini kullanarak bir asenkron motorun alan yönelimli kontrolünü gerçekleştirilmiştir [5]. Xilinx Virtex-5 üzerinde “Xilinx System Generator” (XSG) araç kutusunu kullanarak mimarilerini geliştirmişlerdir. Bu kontrol mekanizmasını FPGA üzerinde uygulayabilmiş olmaları, yapmış oldukları mimarinin, bellek alanı ve maksimum saat hızı açısından verimli olduğunu gerektirmemektedir. Joakim Eriksson ve ark., üç fazlı makine sistemleri için hızlı bir prototipleme sistemi üzerine araştırma yapmışlardır [6].

FPGA kullanılarak FOC ile çok eksenli bir sistemin yapılabileceği sonucuna varmışlardır. Çalışmalarında nominal torklarını, nominal güçlerini ve nominal hız değerlerini doğrulamışlardır. PWM frekans testleri yapılmış ve bunların sonucunda daha yüksek PWM frekansının simülasyon çıkışında daha düşük akımın dalgalandığını göstermişlerdir. PWM frekansını arttırdıkça MOSFET transistörlerinin anahtarlama darbeleri arasında tamamen açılıp kapanması için yeterli zamana sahip olmadıkları için motor akımlarının bozulduklarını iddia etmişlerdir. Yeni nesil “Silicon Carbide” (SiC) ve “Gallium Nitride” (GaN) tabanlı güç çevirici anahtarlarında bu sorunlar büyük oranda aşılmıştır [7].

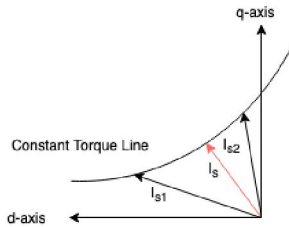
Marufuzzaman ve ark., dq akım tabanlı PI denetleyici önermişlerdir [8]. Çalışmalarında, bir d-q akım PI denetleyicinin, genel sistem performansını artırmanın anahtar unsuru olduğunu iddia etmişlerdir ve dq akım kontrolü üzerine yoğunlaşmışlardır. Gerçekten makalede bahsedildiği gibi dq akım tabanlı PI denetleyici performansı arttıran bir kriter olmasına rağmen bunun yanında birçok performans kriterleri vardır ve bu kriterlerinde geçiştirilmemesi gerekir. Akın ve Alan, FPGA tabanlı bir endüksiyon makinesinin dolaylı kontrolünü incelemişlerdir [9]. Vektör kontrol yöntemleri araştırılmış ve motor parametrelerinden bağımsız olarak “Doğrudan Tork Kontrolü” (DTC) yöntemi yapılmış olmasına rağmen, DTC ile düşük hızlarda verimli geri bildirim sağlanamayacağını iddia etmişlerdir. Bu nedenle, FOC yönteminin DTC’den daha verimli olacağını düşünmüşler, Xilinx Spartan-3’teki XSG araç kutusunu kullanarak FOC modelini hazırlamışlardır.

Bölüm II’de uygulayacağımız metodolojilerimizin teorik bilgilerine, bölüm III’de teorik bilgisi verilen metodolojilerin FPGA üzerinde nasıl gerçekleştirildiğine, bölüm IV’de FPGA üzerinde gerçekleştirmiş olduğumuz yapının sonucuna ve son olarak da bölüm V’de çıkarmış olduğumuz sonuca ulaşabilirsiniz.

II. METODOLOJİ

A. MTPA

“Birim Amper Başına Maksimum Tork” (MTPA) kontrol metodolojisi, gerekli tork seviyesine ulaşmak için stator akımını en aza indirmeye çalışır. Böylelikle bakır kayıpları minimuma indirilir ve genel verimlilik artar. Gerekli torku üretmek için I_{s1} ve I_{s2} gibi çok sayıda stator akımı olasılığı vardır ancak minimum olan, Şekil 1’de gösterildiği gibi I_s ’dir.



Şekil 1: Verilen Tork Değerine Göre Minimum Stator Akımı

PMSM elektrik motorunun tork denklemi aşağıdaki gibidir:

$$T_e = \frac{3}{2} \cdot p_b \cdot (\Psi_m \cdot I_{sq}) + \frac{3}{2} \cdot p_b \cdot (L_d - L_q) \cdot I_{sd} \cdot I_{sq} \quad (1)$$

Ayrıca, inverterin sınırlı güç kaynağına sahip olması nedeniyle I_s sınırlaması vardır:

$$I_{s_{max}}^2 = I_{sd}^2 + I_{sq}^2 \quad (2)$$

$I_{s_{max}}$ invertör tarafından sağlanan maksimum stator akım genliğidir, p_b kutup çifti sayısıdır, Ψ_m kalıcı mıknatısların akı bağlantısıdır, L_d d eksen endüktansı, L_q q eksen endüktansı, I_{sd} , d eksen akımı ve I_{sq} q eksen akımıdır. Denklem (1)’e denklem (2) uygulanırsa aşağıdaki olur:

$$T_e = \frac{3}{2} \cdot p_b \cdot (\Psi_m \cdot \sqrt{I_{s_{max}}^2 - I_{sd}^2}) + \frac{3}{2} \cdot p_b \cdot (L_d - L_q) \cdot I_{sd} \cdot \sqrt{I_{s_{max}}^2 - I_{sd}^2} \quad (3)$$

I_{sd} minimum değerini bulmak için, T_e ’nin türevi I_{sd} ’e göre alınır ve sıfıra eşitlenir.

$$\frac{dT_e}{dI_{sd}} = \frac{3}{2} \cdot p_b \cdot \frac{-I_{sd} \Psi_m + (L_d - L_q)(I_{s_{max}} - 2I_{sd}^2)}{\sqrt{I_{s_{max}}^2 - I_{sd}^2}} = 0 \quad (4)$$

Denklem (4) aşağıdaki gibi basitleştirilebilir:

$$2I_{sd}^2 + \frac{\Psi_m}{L_d - L_q} \cdot I_{sd} - I_{s_{max}}^2 = 0 \quad (5)$$

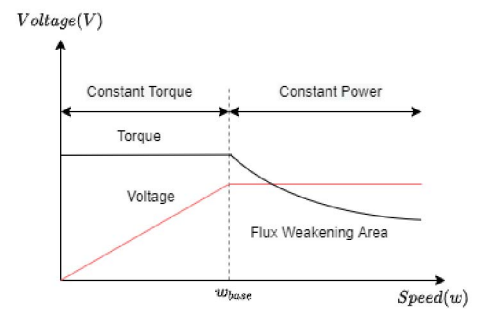
Denklem (5) çözüldükten sonra şu sonuca ulaşabiliriz:

$$I_{sd} = \frac{-\Psi_m + \sqrt{\Psi_m^2 + 8(L_d - L_q)^2 I_{s_{max}}^2}}{4(L_d - L_q)} \quad (6)$$

$$I_{sq} = \sqrt{I_{s_{max}}^2 - I_{sd}^2}$$

B. Akı Zayıflatması

PMSM’nin gerilim seviyesi artırılırken, belirli bir noktadan sonra w_{base} , devrenin gerilimi sabit kalır. Devrenin sınırını ne kadar zorlarsak zorlayalım, gerilimi hala sabit olduğundan sabit güç bölgesinde kalacaktır. Hız artışı sırasında gerilim sabit kalacağından, akı, Şekil. 2’de gösterildiği gibi devre tarafından zayıflatılacaktır.



Şekil 2: Akı Zayıflama Alanı

PMSM'nin çıkıntılı kutup özelliğinin düşük hava boşluğu oluşturmamasından ötürü, geniş ölçekli bir hız uygulaması kolaylıkla gerçekleştirilebilmektedir.

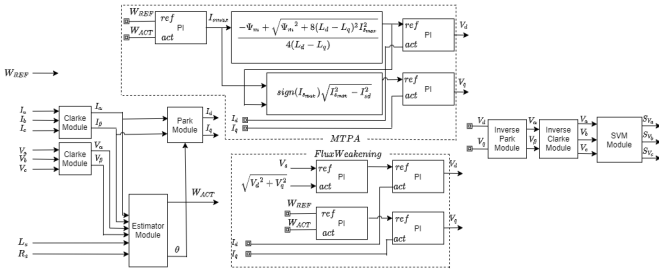
Sabit güç bölgesinde, hem d eksenini hem de q eksenini voltaj seviyeleri 3- ϕ iki seviyeli invertör ile sınırlıdır. Maksimum gerilim V_{smax} değeri, 3- ϕ iki seviyeli invertör tarafından çalıştırılan DC bara gerilimine bağlıdır. Genellikle bu gerilim (V_{smax}) $V_{dc}/\sqrt{3}$ eşittir.

$$V_d^2 + V_q^2 \leq V_{smax}^2 \quad (7)$$

Ayrıca, maksimum akım değeri hem 3- ϕ iki seviyeli invertör hem de akımın maksimum tepe değeri tarafından belirlenir.

$$I_d^2 + I_q^2 \leq I_{smax}^2 \quad (8)$$

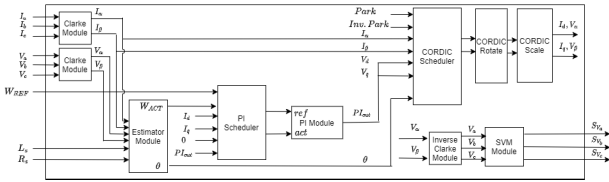
İki kontrol algoritmasının da teorik olarak uygulanış biçimi Şekil 3'deki gibidir.



Şekil 3: MTPA ve Akı Zayıflatması

III.

FPGA, birçok yapılandırılabilir mantık bloğuna (CLB) sahip yarı iletken bir cihazdır. Doğası gereği, sistem gereksinimleri değişirse bize yeniden programlanabilir bir ortam sağlar ve ayrıca kaynaklar yeterli olduğu sürece paralel işleme olanağı da sağlar. Bu tasarımın geliştiriciye sağladığı esneklik, genel bir yapı hazırlanılmasına yardımcı olmakla beraber boru hattı özeliğinden faydalanmasına da olanak sunar. Bu nedenlerle, denetleyici sistemimiz Şekil 3'de görülen yapı baz alınarak FPGA üzerinde uygulanmıştır.



Şekil 4: Donanım Blok Şeması

A. FOC Gerçekleşmesi

Clarke modülünün gerçekleştirilmesi için basit bir çarpma ve toplayıcı hazırlandı. Park modülü ve ters park modülü, COordinate Rotation Digtal Computer (CORDIC) algoritmasının donanımsal olarak üç farklı yapısı (yinelemeli, paralel

ve pipelined) incelenerek, aralarından kaynak kullanımını en aza indireyen yapı olan yinelemeli yapı tercih edilerek gerçekleştirilmiştir. Yinelemeli yapı 15 iterasyon gerçekleştirilerek yapıldı. Bu iterasyon sayısı bizim ne kadar çözünürlük istediğimize göre değiştirilebilmektedir. Her bir döngüde sinüs ve kosinüs açısı değerini hesaplamak yerine Look Up Table (LUT), çarpma ve toplayıcı modülü kullanılarak fixed point işlemleri gerçekleştirildi. Bu nedenle, sadece hesaplamının karmaşıklığından değil, aynı zamanda alan büyümesinden de kurtulduk. Ters clark modülü clark modülü ile aynı şekilde basit bir çarpma ve toplayıcı kullanılarak yapıldı.

MTPA ve Akı Zayıflatması tekniklerinin yapılarında bulunan lineer olmayan aritmetik işlemler, CORDIC modüllerinden birisi olan hyperbolic CORDIC yapısının gerçekleştirilmesi ile oluşturulmuştur. Ayrıca, denetleyici sistemimizin zamanlama hesapları yapıldı ve buna göre bir zaman çizelgesi hazırlandı. Bu yapı, bize boru hattı kullanımında kaynak paylaşım sayımızı arttırmamızı sağladı.

B. SVM Gerçekleşmesi

Uzay Vektörü Modülasyonu, sinüs dalgası oluşturan ve PWM'nu besleyen bir tekniktir. SVM'yi uygulamak için birçok yöntem vardır. SVM'i gerçekleştirmek için biz, minimum yöntemini kullandık. Minimum değere sahip örneklenen gerilimlere (V_{min}) ve maksimum değere sahip örneklenen gerilimlere (V_{max}) densin. Ortak voltaj (V_{offset}) değerini hesaplamak için,

$$V_{offset} = -\frac{V_{max} - V_{min}}{2} \quad (9)$$

0-0-0 ve 1-1-1 durumları THD'yi arttırdığından dolayı bu durumları atlamak için faz voltajı değerinden ortak voltaj değerini çıkarırız. Ortak voltaj değerini faz voltaj değerinden çıkararak, faz voltajının üçüncü harmonik değerini ortadan kaldırmış oluruz. Teoride çok zor olmayan bu modülün inşası, basit bir durum makinesi kullanılarak gerçekleştirildi. Denetleyici sistemimiz üç fazlı makineler için genel bir yapıya sahip olduğundan, geri bildirim döngüsünde sinüsoidal tipte bir komütasyon kullanılıyorsa, SVM modülü kullanımı gerekmez.

C. PI Kontrol Gerçekleşmesi

PI denetleyicinin gerçekleştirilmesinde ilk olarak ayrılmaz indirgeme hazırlanılarak başlandı. Bu özellik PI denetleyicisinin çıktısında doğruluğunun artmasını sağlar. Özellikle motor kontrol uygulamalarında PI denetleyicisinin parametreleri motor çalışmadan önce belirlenip modüle verilmelidir. Bu nedenle, PI denetleyici modülü ayrılmaz indirgeme göz önüne alınarak basit bir durum makinesi ile beraber oluşturulmuştur. Bu oluşturduğumuz PI denetleyicisini genel olarak hazırladığımız için alt modüllerde, d-akımı, q-akımı ve hız değerlerinin ayarlanması için kullandık. Ayrıca PI denetleyicisinin minimum ve maksimum parametrelerini ayarlayabilmesi için kullanıcıya esneklik tanıdık.

D. Kullanıcı Arayüzü Gerçekleşmesi

Kullanıcı arayüzü, kullanıcının hız değerini ayarlayabilmesi, motoru başlatabilmesi ve durdurabilmesi, motorun dönüş yönünü değiştirebilmesi amacıyla oluşturuldu. Hız-zaman,

tork-zaman ve akım-zaman grafikleri kullanıcıya gönderilerek kullanıcının bilgilendirilmesi amaçlandı. Zynq üzerinde yapı tasarlama kolaylığı nedeniyle, kullanıcı arayüzü iletişim modülü Zynq işlemci kullanılarak yapıldı. Kullanıcı ile PC arasındaki iletişim protokolü UART olarak tercih edildi.

IV. SONUÇLAR VE DEĞERLENDİRMELER

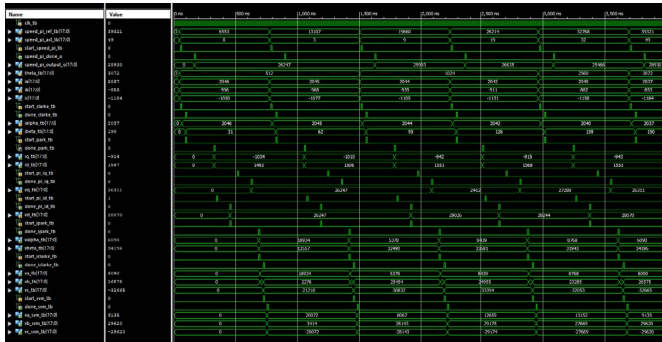
Bütün bir yapıyı oluşturmadan önce, bu yapının 4 farklı yerinde kullanacağımız, üç farklı CORDIC yapısı incelendi ve gerçekleştirildi. Bu gerçekleştirilmelerin donanım üzerinde kullanacağı kaynaklar Tablo I'de gösterilmiştir.

TABLE I: CORDIC Yapıları Transistör Alanı Karşılaştırılması

Yapı	FFs	AddSub-19	Mux-19	Mux-1	# Transistör
Yinelemeli	242	3	13	80	12128
Pipelined	956	45	9	1	62606
Paralel	115	45	9	0	34006

Yapılan çalışmada, Şekil 5'de görüldüğü gibi clark ve ters clark modülünün gerçekleşme zamanı ayrı ayrı 4'er saat darbesi, park ve ters park modülünün gerçekleşme zamanı ayrı ayrı 23'er saat darbesi, d ve q akımları için pi kontrol modülü ayrı ayrı 11'er saat darbesi, svm modülü ise 3 saat darbesi sürmektedir. Ia, Ib ve Ic akımları Zynq'in Analog-Digital-Converter (ADC) modülü 12 bit giriş kabul ettiği için test ortamında 12 biti geçmeyecek şekilde verildi. 12 bitlik modülden 18 bitlik modüle basit bir ölçekleme modülü yapıldı fakat incelemenin sağlığı açısından test ortamına konulmadı.

PI denetleyicisindeki Kp ve Ki değerleri sabit uygulandı ve simülasyon süresince değiştirilmedi ve yine PI denetleyicisindeki id-iq referans ve gerçek değerleri matlab simulik tarafından üretilen gerçek sonuçlara yakın sonuçlar baz alınarak uygulandı. Gerçekleştirilmedeki toplam gecikme 75 saat darbesi ve minimum giriş verme süremiz 63 saat darbesi kadardır. Minimum periodumuz 5.9 ns (maksimum frekansımız 169 MHz)'dir. Makalelerimizde, FPGA üzerinde üç fazlı makine kontrolü için FOC tercih eden kişilerle yapılan karşılaştırma, bir önceki çalışmamız [10]'daki gibiydi. Kaynak kullanımı sonucuna bakarak kaynaklarımızı en verimli şekilde kullandığımızı belirtmiştik.



Şekil 5: MTPA Gecikme Grafiği

Tablo II'de görüldüğü gibi, MTPA çalışma frekansı 169 MHz, gecikme saat döngüsü 75, minimum giriş verme süremiz 63 saat döngüsünde bir ve 55 bit çıkış aldığımız için veri hızımız her bir saniye için 148 Mb'dir.

TABLE II: MTPA ve Akı Zayıflatması Zaman Karşılaştırması

Yapı	Min. Periyot	Max. Frek.	Gecikme	Baş. Aralığı	Veri Hızı
MTPA	5.9 ns	169 MHz	75	1/63	148 Mbps
Akı Zay.	5.9 ns	169 MHz	84	1/72	129 Mbps

V. SONUÇ

Bu yazıda, kullanıcı arayüzü ve genel bir tasarım olan FOC metodolojisinin algılayıcı uygulamaları, MTPA ve Akı Zayıflatmasını öneriyoruz. Bu öneri ile birlikte, FOC algoritması yürütülürken, kullanıcı, gerçek zamanlı tork-zaman, hız-zaman ve akım-zaman grafiklerini okuyabilecektir. Kullanıcı, ayrıca motorun hızını ve dönüş yönünü gerçek zamanlı olarak değiştirebilecektir. Uygulamamız, Xilinx Zynq-7020'de Verilog HDL'yi manuel olarak kullanarak yapıldı. Ayrıca, uygulamanın test senaryoları, ISim test ortamı kullanılarak hazırlandı ve test edildi. Sonuç olarak, donanım uygulamamız, bellek alanı ve maksimum anahtarlama frekansı açısından "Yüksek Düzeyli Sintez" (HLS) araçları kullanılarak oluşturulan ve FOC algoritması'nı gerçekleyen literatürdeki diğer yapılara göre üstünlük sağlamış, bunların yanına MTPA ve Akı Zayıflatması eklenerek zenginleştirilmiştir.

BİLGİLENDİRME

Bu çalışma TÜBİTAK BIGG-2180770 kapsamında desteklenmiş olup, yazarlar bu bağlamda TÜBİTAK ve Özyeğin Üniversitesi'ne teşekkürlerini sunarlar.

KAYNAKLAR

- [1] J. Xu, L. Gu, Z. Ye, S. Kargarrazi, and J. Rivas-Davila, "Cascode GaN/SiC Power Device for MHz Switching," in *Proc. Applied Power Electronics Conference and Exposition (APEC)*, s. 2780-2785, 2019.
- [2] M. Agrawal, P. Nema, and R. Lodhi, "A Review of Speed Control Methods of Induction Motor," *IOSR Journal of Electrical and Electronics Engineering (JEEE)*, s. 9-14, vol. 13, 2018.
- [3] Y. Kung, Hoang Than, Y. Lin, and L. Huang, "FPGA Based Speed Controller Design for a Ceiling Fan Motor," in *Proc. International Future Energy Electronics Conference (IFEEC) and ECCE Asia*, s. 30-34, 2017.
- [4] S. Suneeta, R. Srinivasan, and R. Sagar, "FPGA Based Control Method for Three Phase BLDC Motor," *International Journal of Electrical and Computer Engineering*, s. 1434-1440, vol. 6, 2016.
- [5] N. R. Babu and K. S. Athul, "The Field Oriented Control of Induction Motor Using FPGA," *Global Journal of Pure and Applied Mathematics*, s. 1157-1170, vol. 11, 2015.
- [6] J. Eriksson and L. Hermansen, "Rapid Prototyping: Development and Evaluation of Field Oriented Control Using LabView FPGA," Master's thesis, Mälardalen Universitesi, Isvec, 2011.
- [7] K. Li, P. Evans, and M. Johnson, "SiC/GaN Power Semiconductor Devices: A Theoretical Comparison and Experimental Evaluation Under Different Switching Conditions," *IET Electrical Systems in Transportation*, s. 3-11, vol. 8, 2018.
- [8] M. Marufuzzaman, M. Reaz, and M. Ali, "FPGA Implementation of an Intelligent Current dq PI Controller for FOC PMSM Drive," in *Proc. International Conference on Computer Applications and Industrial Electronics (ICCAIE)*, s. 602-605, 2010.
- [9] O. Akin and I. Alan, "The Use of FPGA in Field Oriented Control of an Induction Machine," *Turkish Journal of Electrical Engineering and Computer Science*, s. 943-962, vol. 18, 2010.
- [10] B. Tufekci, B. Onal, H. Dere, and H. F. Ugurdag, "Efficient FPGA Implementation of Field Oriented Control for 3-Phase Machine Drives," in *Proc. IEEE East-West Design & Test Symposium (EWDTS)*, 2020.