**MICROCHIP - AN1078**

***Sensorless Field Oriented Control of a PMSM***

INTRODUCTION

Tasarımcılar, çevresel taleplerin enerji tasarruflu klimalar, çamaşır makineleri ve diğer ev aletleri üreten gelişmiş motor kontrol tekniklerine olan ihtiyacı artırmaya devam etmesini bekleyebilirler. Şimdiye kadar, sofistike motor kontrol çözümleri yalnızca özel kaynaklardan temin edilebilmekteydi. Ancak, yeni nesil Dijital Sinyal Kontrolörleri (DSC'ler) sayesinde gelişmiş, uygun maliyetli motor kontrol algoritmalarının uygulanması artık bir gerçek.

Örneğin bir klima, motordaki hız değişiklikleri için hızlı tepki gerektirir. Daha sessiz ve daha enerji verimli üniteler üretmek için gelişmiş motor kontrol algoritmalarına ihtiyaç vardır. Saha Odaklı Kontrol (FOC), bu çevresel talepleri karşılamak için önde gelen bir yöntem olarak ortaya çıkmıştır.

Bu uygulama notunda Microchip dsPIC® DSC ailesi kullanılarak Kalıcı Mıknatıslı Senkron Motor (PMSM) için sensörsüz bir FOC algoritmasının uygulanması anlatılmaktadır.

Why Use the FOC Algorithm?

BLDC motorlar için geleneksel kontrol yöntemi, statoru altı adımlı bir süreçte çalıştırır ve bu da üretilen tork üzerinde salınımlar oluşturur. Altı adımlı kontrolde, rotor bir sonraki konuma ulaşana kadar bir çift sargıya enerji verilir ve ardından motor bir sonraki adıma komütasyona tabi tutulur. Hall sensörleri, motoru elektronik olarak komütasyona sokmak için rotor konumunu belirler. Gelişmiş sensörsüz algoritmalar, rotor konumunu belirlemek için stator sargısında üretilen geri-EMF'yi kullanır.

Altı adımlı kontrolün (trapezoidal kontrol olarak da adlandırılır) dinamik yanıtı çamaşır makineleri için uygun değildir çünkü yük bir yıkama döngüsü içinde dinamik olarak değişir ve farklı yüklere ve seçilen yıkama döngüsüne göre değişir. Ayrıca, önden yüklemeli bir çamaşır makinesinde, yük tamburun üst tarafındayken yerçekimi gücü motor yüküne karşı çalışır. Yalnızca FOC gibi gelişmiş algoritmalar bu dinamik yük değişikliklerini idare edebilir.

Bu uygulama notu cihazların PMSM tabanlı sensörsüz FOC kontrolüne odaklanmaktadır çünkü bu kontrol tekniği cihaz motor kontrolünde en büyük maliyet avantajını sunmaktadır. Sensörsüz FOC tekniği ayrıca, motorun su altında kalması veya kablo demeti yerleştirme kısıtlamaları nedeniyle konum veya hız sensörlerinin kullanılamadığı bazı uygulamalara getirilen kısıtlamaların da üstesinden gelir. Rotor üzerindeki sabit bir mıknatıs tarafından üretilen sabit bir rotor manyetik alanı ile PMSM bir cihazda kullanıldığında çok verimlidir. Buna ek olarak, stator manyetik alanı sargıların sinüzoidal dağılımı ile üretilir. Asenkron motorlarla karşılaştırıldığında, PMSM boyutuna göre güçlüdür. Ayrıca fırça kullanılmadığı için elektriksel olarak bir DC motordan daha az gürültülüdür.

Why Use Digital Signal Controllers for Motor Control?

dsPIC DSC'ler çamaşır makineleri ve klima kompresörleri gibi cihazlar için uygundur, çünkü motor kontrolü için ideal olan çevre birimlerini içerirler:

* Pulse-Width Modulation (PWM)
* Analog-to-Digital Converter (ADC)
* Quadrature Encoder Interface (QEI)

Denetleyici rutinlerini gerçekleştirirken ve dijital filtreleri uygularken, dsPIC DSC'ler tasarımcıların kodu optimize etmesini sağlar çünkü MAC komutları ve kesirli işlemler tek bir döngüde yürütülebilir. Ayrıca, doygunluk yetenekleri gerektiren işlemler için, dsPIC DSC'ler donanım doygunluk koruması sunarak taşmaları önlemeye yardımcı olur.

dsPIC DSC'ler, motor kontrolünde çok önemli bir işlev olan akım algılama için hızlı ve esnek Analog-Dijital (A/D) dönüşüme ihtiyaç duyar. dsPIC DSC'ler, giriş örneklerini 1 Msps hızlarında dönüştürebilen ve aynı anda dört adede kadar girişi işleyebilen ADC'lere sahiptir. ADC'lerdeki çoklu tetikleme seçenekleri, sargı akımlarını ölçmek için ucuz akım algılama dirençlerinin kullanılmasını sağlar. Örneğin, PWM modülü ile A/D dönüşümlerini tetikleme özelliği, ucuz akım algılama devresinin belirli zamanlarda girişleri algılamasını sağlar (anahtarlama transistörleri, akımın algılama dirençlerinden akmasını sağlar).

MOTOR CONTROL WITH DIGITAL SIGNAL CONTROLLERS

dsPIC DSC Motor Kontrol ailesi, aşağıdakiler de dahil olmak üzere en popüler motor türlerini kontrol etmek için özel olarak tasarlanmıştır:

* AC Induction Motor (ACIM)
* Brushed DC Motor (BDC)
* Brushless DC Motor (BLDC)
* Permanent Magnet Synchronous Motor (PMSM)

dsPIC DSC motor kontrol ailesini temel alan çeşitli uygulama notları yayınlanmıştır (bkz. "Referanslar" bölümü). Bu uygulama notları Microchip web sitesinde (www.microchip.com) mevcuttur.

Bu uygulama notu, dsPIC DSC'nin bir PMSM'nin sensörsüz alan odaklı kontrolünü gerçekleştirmek için motor kontrolü için özel olarak uygun çevre birimlerinden (motor kontrolü PWM ve yüksek hızlı ADC) nasıl yararlandığını göstermektedir. dsPIC DSC'nin DSP motoru gerekli hızlı matematiksel işlemleri destekler.

Data Monitoring and Control Interface

Veri İzleme ve Kontrol Arayüzü (DMCI), uygulamanın operasyonel kısıtlamalarının aralık değerlerinin, açık/kapalı durumlarının veya ayrık değerlerin değişken kontrolüne bağlı olduğu projeler için MPLAB® IDE ile hızlı dinamik entegrasyon sağlar. Gerekirse, uygulama geri bildirimi grafiksel olarak gösterilebilir. Örnekler arasında motor kontrolü ve ses işleme uygulamaları yer alır.

DMCI şunları sağlar:

* Nine slider controls and nine boolean (on/off) controls (see Figure 1)
* 35 input controls (see Figure 2)
* Four graphs (see Figure 3)

Arayüz, kaydırıcı, doğrudan giriş veya boolean kontrollerinin herhangi bir kombinasyonuna dinamik olarak atanabilen program sembollerinin (değişkenler) projeye duyarlı navigasyonunu sağlar. Kontroller daha sonra MPLAB IDE içindeki program değişkenlerinin değerlerini değiştirmek için etkileşimli olarak kullanılabilir. Grafikler, program tarafından oluşturulan verileri görüntülemek için dinamik olarak yapılandırılabilir.

Application Highlights

Bu uygulama notunun amacı, Microchip dijital sinyal denetleyicileri kullanarak PMSM için sensörsüz, alan odaklı kontrolün yazılım tabanlı bir uygulamasını göstermektir.

Kontrol yazılımı şu özellikleri sunar:

* Bir PMSM'nin vektör kontrolünü uygular.
* Konum ve hız tahmin algoritması. Konum sensörlerine olan ihtiyacı ortadan kaldırır.
* Hız aralığı 500 ila 17000 RPM arasında test edilmiştir.
* 50 µs'lik bir kontrol döngüsü periyodu ile yazılım yaklaşık 21 MIPS CPU ek yükü gerektirir (mevcut toplam CPU'nun yaklaşık 2/3'ü).
* Uygulama 450 bayt veri belleği depolama alanı gerektirir. Kullanıcı arayüzü ile birlikte yaklaşık 6 Kbyte program belleği gerekmektedir. Uygulamanın bellek gereksinimleri, bu yazının yazıldığı sırada en küçük ve en uygun maliyetli dsPIC33F cihazı olan dsPIC33FJ12MC202 üzerinde çalışmasına izin vermektedir.
* Dahili program değişkenlerinin bir osiloskop üzerinde gerçek zamanlı olarak gözlemlenmesini sağlamak için isteğe bağlı bir tanılama modu etkinleştirilebilir. Bu özellik kontrol döngüsü ayarlamasını kolaylaştırır.

SYSTEM OVERVIEW

Şekil 4'te gösterildiği gibi, motor miline bağlı herhangi bir konum sensörü yoktur. Bunun yerine, motor üzerindeki akım ölçümleri için invertörün bir parçası olan düşük endüktanslı şönt dirençler kullanılır. Motor sargılarını sürmek için güç aşaması olarak 3 fazlı bir invertör kullanılır. Güç invertörüne yerleştirilen akım algılama ve hata üretme devresi, tüm sistemi aşırı akımlara karşı korur.

Şekil 5, 3 fazlı topolojinin yanı sıra akım algılama ve arıza üretim devresinin nasıl uygulandığını göstermektedir. Eviricinin sol tarafında gösterilen ilk transistör, bu uygulama notunun bir parçası olmayan Güç Faktörü Düzeltmesi (PFC) için kullanılır.

Bu uygulama notunda atıfta bulunulan donanım, Microchip web sitesinden (www.microchip.com) temin edilebilen dsPICDEM™ MCLV Geliştirme Kartı (DM330021) (50 VDC'ye kadar) ve dsPICDEM™ MCHV Geliştirme Kartıdır (DM330023) (400 VDC'ye kadar).

FIELD ORIENTED CONTROL

A Matter of Perspective

FOC'nin (bazen vektör kontrolü olarak da adlandırılır) nasıl çalıştığını anlamanın bir yolu, koordinat referans dönüşümü sürecinin zihinsel bir görüntüsünü oluşturmaktır. Bir AC motorun çalışmasını stator perspektifinden hayal ederseniz, statora uygulanan sinüzoidal bir giriş akımı görürsünüz. Bu zaman değişkenli sinyal dönen bir manyetik akı oluşturur. Rotorun hızı, dönen akı vektörünün bir fonksiyonudur. Sabit bir perspektiften bakıldığında, stator akımları ve dönen akı vektörü AC büyüklükleri gibi görünür.

Şimdi, motorun içinde olduğunuzu ve stator akımları tarafından üretilen dönen akı vektörü ile aynı hızda dönen rotorun yanında koştuğunuzu hayal edin. Kararlı durum koşulları sırasında motora bu perspektiften bakarsanız, stator akımları sabit değerler gibi görünür ve dönen akı vektörü sabittir.

Nihayetinde, istenen rotor akımlarını (doğrudan ölçülemeyen) elde etmek için stator akımlarını kontrol etmek istersiniz. Koordinat referans dönüşümü ile stator akımları standart kontrol döngüleri kullanılarak DC değerleri gibi kontrol edilebilir.

Vector Control Summary

Dolaylı vektör kontrol süreci aşağıdaki gibi özetlenebilir:

1. 3 fazlı stator akımları ölçülür. Bu ölçümler ia ve ib değerlerini sağlar. Ic aşağıdaki denklemle hesaplanır: ia + ib + ic = 0.
2. 3 fazlı akımlar iki eksenli bir sisteme dönüştürülür. Bu dönüşüm, ölçülen ia ve ib ile hesaplanan ic değerlerinden iα ve iβ değişkenlerini sağlar. iα ve iβ, stator perspektifinden bakıldığında zamanla değişen dörtlü akım değerleridir.
3. İki eksenli koordinat sistemi, kontrol döngüsünün son iterasyonunda hesaplanan bir dönüşüm açısı kullanılarak rotor akısı ile hizalanacak şekilde döndürülür. Bu dönüşüm iα ve iβ'dan Id ve Iq değişkenlerini sağlar. Id ve Iq, dönen koordinat sistemine dönüştürülen karesel akımlardır. Kararlı durum koşulları için Id ve Iq sabittir.
4. Hata sinyalleri Id, Iq ve her biri için referans değerleri kullanılarak oluşturulur.
   * Id referansı, rotor mıknatıslanma akısını kontrol eder.
   * Iq referansı, motorun tork çıkışını kontrol eder.
   * Hata sinyalleri PI kontrolörlerine girilir.
   * Kontrolörlerin çıkışı, motora gönderilecek voltaj vektörü olan Vd ve Vq'yu sağlar.
5. vα, vβ, iα ve iβ'nın girişler olduğu yeni bir dönüşüm açısı tahmin edilir. Yeni açı, bir sonraki gerilim vektörünün nereye yerleştirileceği konusunda FOC algoritmasına rehberlik eder.
6. PI kontrolörlerinden gelen Vd ve Vq çıkış değerleri, yeni açı kullanılarak sabit referans çerçevesine geri döndürülür. Bu hesaplama bir sonraki vα ve vβ karesel gerilim değerlerini sağlar.
7. vα ve vβ değerleri tekrar 3 fazlı va, vb ve vc değerlerine dönüştürülür. 3 fazlı gerilim değerleri, istenen gerilim vektörünü oluşturan yeni PWM görev döngüsü değerlerini hesaplamak için kullanılır. Dönüştürme, PI iterasyonu, geri dönüştürme ve PWM üretme işlemlerinin tamamı Şekil 6'da gösterilmiştir.

Bu uygulama notunun sonraki bölümlerinde bu adımlar daha ayrıntılı olarak açıklanmaktadır.

COORDINATE TRANSFORMS

Bir dizi koordinat dönüşümü aracılığıyla, klasik PI kontrol döngüleriyle tork ve akının zamanla değişmeyen değerlerini dolaylı olarak belirleyebilir ve kontrol edebilirsiniz. İşlem, 3 fazlı motor akımlarının ölçülmesiyle başlar. Pratikte, üç akım değerinin anlık toplamı sıfırdır. Bu nedenle, üç akımdan sadece ikisini ölçerek üçüncüsünü belirleyebilirsiniz. Bu nedenle, donanım maliyeti üçüncü akım sensörünün masrafıyla azaltılabilir.

dsPIC DSC ile 3 fazlı akım ölçümü için tek şönt uygulaması da mümkündür. Tek şönt algoritmasının ayrıntılı açıklaması için AN1299, "Bir PMSM'nin Sensörsüz FOC'si için Tek Şönt Üç Fazlı Akım Yeniden Yapılandırma Algoritması" (DS01299) bölümüne bakın.

Clarke Transform

Clarke Dönüşümü olarak adlandırılan ilk koordinat dönüşümü, statora referanslı üç eksenli, iki boyutlu bir koordinat sistemini aynı referansı koruyarak iki eksenli bir sisteme taşır (bkz. Şekil 7, burada ia, ib ve ic bireysel faz akımlarıdır).

Park Transform

Bu noktada, α-β olarak adlandırılan eksenle iki eksenli ortogonal bir sistem üzerinde temsil edilen stator akımına sahip olursunuz. Bir sonraki adım, rotor akısıyla birlikte dönen başka bir iki eksenli sisteme dönüştürmektir. Bu dönüşüm, Şekil 8'de gösterildiği gibi Park Dönüşümünü kullanır. Bu iki eksenli dönen koordinat sistemi d-q ekseni olarak adlandırılır. θ rotor açısını temsil eder.

PI Control

Üç etkileşimli değişkeni bağımsız olarak kontrol etmek için üç PI döngüsü kullanılır. Rotor hızı, rotor akısı ve rotor torkunun her biri ayrı bir PI modülü tarafından kontrol edilir. Uygulama gelenekseldir ve Şekil 'de gösterildiği gibi integral sargısını sınırlamak için terim (Kc.Excess) içerir. Fazlalık, sınırsız çıkış (U) ve sınırlı çıkışın (Out) çıkarılmasıyla hesaplanır. Kc terimi Fazlalığı çarpar ve birikmiş integral kısmını (Sum) sınırlar.

PID CONTROLLER BACKGROUND

Oransal İntegral Türev (PID) kontrolörlerinin tam bir tartışması bu uygulama notunun kapsamı dışındadır; ancak bu bölüm size PID işleminin bazı temellerini sağlar.

Bir PID kontrolörü kapalı bir kontrol döngüsünde bir hata sinyaline yanıt verir ve istenen sistem yanıtını elde etmek için kontrol edilen miktarı ayarlamaya çalışır. Kontrol edilen parametre hız, tork veya akı gibi ölçülebilir herhangi bir sistem büyüklüğü olabilir. PID kontrolörün avantajı, bir veya daha fazla kazanç değerini değiştirerek ve sistem yanıtındaki değişikliği gözlemleyerek deneysel olarak ayarlanabilmesidir.

Dijital bir PID kontrolörü periyodik bir örnekleme aralığında çalıştırılır. Sistemin düzgün bir şekilde kontrol edilebilmesi için kontrolörün yeterince sık çalıştırıldığı varsayılır. Hata sinyali, kontrol edilecek parametrenin istenen ayarının o parametrenin gerçek ölçülen değerinden çıkarılmasıyla oluşturulur. Hatanın işareti, kontrol girişinin gerektirdiği değişimin yönünü gösterir.

Kontrolörün Oransal (P) terimi, hata sinyalinin bir P kazancı ile çarpılmasıyla oluşturulur ve PID kontrolörünün hata büyüklüğünün bir fonksiyonu olan bir kontrol yanıtı üretmesine neden olur. Hata sinyali büyüdükçe, kontrolörün P terimi daha fazla düzeltme sağlamak için büyür.

P teriminin etkisi, zaman geçtikçe toplam hatayı azaltma eğilimindedir. Ancak, hata sıfıra yaklaştıkça P teriminin etkisi azalır. Çoğu sistemde, kontrol edilen parametrenin hatası sıfıra çok yaklaşır ancak yakınsamaz. Sonuç, kalan küçük bir kararlı durum hatasıdır.

Kontrolörün İntegral (I) terimi küçük kararlı durum hatalarını ortadan kaldırmak için kullanılır. I terimi hata sinyalinin sürekli çalışan bir toplamını hesaplar. Bu nedenle, küçük bir kararlı durum hatası zaman içinde büyük bir hata değerine birikir. Biriken bu hata sinyali bir I kazanç faktörü ile çarpılır ve PID kontrolörün I çıkış terimi haline gelir.

PID kontrolörünün Diferansiyel (D) terimi kontrolörün hızını artırmak için kullanılır ve hata sinyalinin değişim oranına yanıt verir. D terimi girişi, mevcut hata değerinin bir önceki değerden çıkarılmasıyla hesaplanır. Bu delta hata değeri, PID kontrolörün D çıkış terimi haline gelen bir D kazanç faktörü ile çarpılır.

Kontrolörün D terimi, sistem hatası daha hızlı değiştikçe daha fazla kontrol çıktısı üretir. Tüm PID kontrolörleri D veya daha az yaygın olarak I terimlerini uygulamayacaktır. Örneğin, bu uygulama motor hızı değişikliklerinin nispeten yavaş tepki süresi nedeniyle D terimlerini kullanmaz. Bu durumda, D terimi PWM görev döngüsünde algoritmaların çalışmasını etkileyebilecek ve aşırı akım hatalarına neden olabilecek aşırı değişikliklere neden olabilir.

Adjusting the PID Gains

Bir PID kontrolörünün P kazancı genel sistem yanıtını belirler. Bir kontrolörü ilk ayarladığınızda, I ve D kazançlarını sıfıra ayarlayın. Daha sonra sistem aşırı aşım veya salınım olmadan ayar noktası değişikliklerine iyi yanıt verene kadar P kazancını artırabilirsiniz. P kazancının düşük değerlerini kullanmak sistemi 'gevşek' bir şekilde kontrol ederken, yüksek değerler 'daha sıkı' kontrol sağlayacaktır. Bu noktada, sistem muhtemelen ayar noktasına yakınsamayacaktır.

Makul bir P kazancı seçtikten sonra, sistem hatasını sıfıra zorlamak için I kazancını yavaşça artırabilirsiniz. Çoğu sistemde yalnızca az miktarda I kazancı gereklidir. I kazancının etkisi, yeterince büyükse, P teriminin etkisinin üstesinden gelebilir, genel kontrol yanıtını yavaşlatabilir ve sistemin ayar noktası etrafında salınmasına neden olabilir. Salınım meydana gelirse, I kazancını azaltmak ve P kazancını artırmak genellikle sorunu çözecektir.

Bu uygulama, entegre hatanın çıkış parametresini doyurması durumunda ortaya çıkan integral sarmasını sınırlamak için bir terim içerir. Entegre hatadaki herhangi bir artış çıkışı etkilemez. Biriken hata azaldığında, çıkışın doyuma ulaşmasına neden olan değerin altına düşmesi (veya çözülmesi) gerekecektir. Kc katsayısı bu istenmeyen birikimi sınırlar. Çoğu durum için bu katsayı Ki değerine eşit olarak ayarlanabilir.

Her üç kontrol ünitesinin de çıkış parametresi için bir maksimum değeri vardır. Bu değerler UserParms.h dosyasında bulunabilir ve SVGen() rutininde doygunluğu önlemek için varsayılan olarak ayarlanmıştır.

Control Loop Dependencies

Bu uygulamada birbirine bağlı üç PI kontrol döngüsü vardır. Dış döngü motor hızını kontrol eder. İki iç döngü dönüştürülmüş motor akımlarını, Id ve Iq'yu kontrol eder. Daha önce de belirtildiği gibi Id döngüsü akının kontrolünden, Iq değeri ise motor torkunun kontrolünden sorumludur.

Inverse Park

PI iterasyonundan sonra, dönen d-q ekseninde iki gerilim bileşeni vektörüne sahip olursunuz. Tekrar 3 fazlı motor gerilimine dönmek için tamamlayıcı ters dönüşümlerden geçmeniz gerekecektir. İlk olarak, iki eksenli dönen d-q çerçevesinden iki eksenli sabit α-β çerçevesine dönüşüm yaparsınız. Bu dönüşüm, Şekil 10'da gösterildiği gibi Ters Park Dönüşümünü kullanır.

Inverse Clarke

Bir sonraki adım, sabit iki eksenli α-β çerçevesinden statorun sabit üç eksenli, 3 fazlı referans çerçevesine dönüştürmektir. Matematiksel olarak bu dönüşüm, Şekil 11'de gösterildiği gibi Ters Clark Dönüşümü ile gerçekleştirilir.

Space Vector Modulation (SVM)

Vektör kontrol sürecindeki son adım, 3 fazlı motor gerilim sinyalleri için darbe genişliği modülasyon sinyalleri üretmektir. Uzay Vektör Modülasyonu (SVM) tekniklerini kullanırsanız, üç fazın her biri için darbe genişliği üretme işlemi birkaç basit denkleme indirgenir. Bu uygulamada, Ters Clarke Dönüşümü, hesaplamaları daha da basitleştiren SVM rutinine katlanmıştır.

Üç invertör çıkışının her biri iki durumdan birinde olabilir. İnvertör çıkışı ya artı (+) bara rayına ya da eksi (-) bara rayına bağlanabilir, bu da Tablo 1'de gösterildiği gibi çıkışın 23 = 8 olası durumuna izin verir.

Her üç çıkışın da artı (+) bara veya eksi (-) bara ile bağlantılı olduğu iki durum boş durum olarak kabul edilir çünkü fazların hiçbirinde hattan hatta gerilim yoktur. Bunlar SVM yıldızının orijininde çizilir. Kalan altı durum, Şekil 12'de gösterildiği gibi, her durum arasında 60 derecelik dönüşe sahip vektörler olarak temsil edilir.

DVM süreci, herhangi bir sonuç vektörünün iki komşu vektörün bileşenlerinin toplamı ile temsil edilmesine izin verir. Şekil 13'te, UOUT istenen sonuçtur. U60 ve U0 arasındaki sektörde yer alır. Belirli bir T PWM periyodu sırasında, U0 T1/T için ve U60 T2/T için çıkarsa, periyodun ortalaması UOUT olacaktır.

T0, sargılara hiçbir etkin voltajın uygulanmadığı, yani boş vektörün uygulandığı bir zamanı temsil eder. T1 ve T2 değerleri, değiştirilmiş bir Ters Clark dönüşümü kullanılarak ekstra hesaplama yapılmadan elde edilebilir. Vα ve Vβ'yı ters çevirirseniz, SVM yıldızından 30 derece kaydırılmış bir referans ekseni oluşturulur. Sonuç olarak, altı segmentin her biri için bir eksen o segmentin tam karşısında yer alır ve diğer iki eksen segmenti simetrik olarak sınırlar. Bu iki sınırlayıcı eksen boyunca vektör bileşenlerinin değerleri T1 ve T2'ye eşittir. Hesaplamaların ayrıntıları için kaynak koddaki CalcRef.s ve SVGen.s dosyalarına bakın.

Şekil 14'ten PWM periyodu T için T1 vektörünün T1/T için ve T2 vektörünün T2/T için çıkış verdiğini görebilirsiniz. Kalan süre boyunca boş vektörler çıkar. dsPIC DSC, periyodun merkezi etrafında simetriyi zorlayan merkez hizalı PWM için yapılandırılmıştır. Bu yapılandırma, her periyot sırasında hattan hatta iki darbe üretir. Etkin anahtarlama frekansı iki katına çıkar ve güç cihazlarındaki anahtarlama kayıplarını artırmazken dalgalanma akımını azaltır.

SENSORLESS FOC FOR PMSM

Algoritmanın önemli bir kısmı, FOC için gereken komütasyon açısının nasıl hesaplanacağıdır. Uygulama notunun bu bölümü komütasyon açısını (θ) ve motor hızını (ω ) tahmin etme sürecini açıklamaktadır.

Sensörsüz kontrol tekniği, konum sensörleri kullanmadan motorun konumunu tahmin ederek FOC algoritmasını uygular. Şekil 16, konum tahmin edici işlevinin basitleştirilmiş bir blok diyagramını göstermektedir.

Motor konumu ve hızı, ölçülen akımlara ve hesaplanan gerilimlere göre tahmin edilir.

Motor Model

Şekil 15'te gösterildiği gibi sargı direnci, sargı endüktansı ve geri-EMF ile temsil edilebilen bir DC Motor modeli kullanarak PMSM konumunu tahmin edebilirsiniz.

Calculating F and G Parameters

Motor modelinde belirli bir motor için değiştirilmesi gereken iki parametre vardır. Bu iki parametre F ve G kazançlarıdır, burada:

EQUATION 2:

---

R ve L sabitleri basit bir multimetre kullanılarak ölçülür. Örneğin, hattan hatta bir direnç ölçülürse, F ve G kazançları için kullanılan R, faz direncine ihtiyaç duyulduğu için ölçümün ikiye bölünmesiyle elde edilir. Aynı prosedür endüktans hesaplaması L için de geçerlidir. Örneğin, Hurst motoru 20 kHz'de algoritma ile çalıştırıldığında, ölçülen hattan hatta direnç 5,34Ω ve hattan hatta endüktans 3,84 mH olarak ölçüldüğünde, motor modeli parametreleri şunlardır:

EQUATION 3:

---

Current Observer

Konum ve hız tahmincisi bir akım gözlemcisine dayanmaktadır. Bu gözlemci, Denklem 1 ile gösterildiği gibi motorun sayısallaştırılmış bir modelidir. Değişkenler ve sabitler şunları içerir:

* Motor Phase Current (is)
* Input voltage (vs)
* Back-EMF (es)
* Winding resistance (R)
* Winding inductance (L)
* Control period (Ts)
* Output Correction Factor Voltage (z)

Sayısallaştırılmış model, donanımın yazılımsal bir temsilini sağlar. Ancak, ölçülen akım ile tahmin edilen akımı eşleştirmek için, Şekil 17'de gösterildiği gibi, sayısallaştırılmış motor modelinin kapalı döngü kullanılarak düzeltilmesi gerekir.

Biri donanımda (gölgeli alan) diğeri yazılımda olmak üzere, her iki sisteme de aynı giriş (vs) beslenen iki motor temsilini göz önünde bulundurarak ve ölçülen akımı (is) modelden tahmin edilen akımla (is\*) eşleştirerek, sayısallaştırılmış modelimizden gelen geri-EMF'nin (es\*) motordan gelen geri-EMF (es) ile aynı olduğunu varsayabiliriz.

Not: \* tahmin edilen değişkeni ifade eder

Hata değeri MaxSMCError değerinden küçük olduğunda, kayan modlu kontrolörün çıkışı Şekil 17'deki PMSM bloğunun altındaki denklemde verildiği gibi doğrusal aralıkta çalışır.

Doğrusal aralığın dışındaki bir hata değeri için, kayan mod kontrolörünün çıkışı hatanın işaretine bağlı olarak (+Kslide)/(-Kslide) olur.

Sayısallaştırılmış motor modelini telafi etmek için bir kayar mod kontrolörü veya SMC kullanılır. Bir SMC, motordan ölçülen akım ile sayısallaştırılmış motor modelinden tahmin edilen akım arasındaki hatanın işaretini hesaplayan bir toplama noktasından oluşur. Hesaplanan hata işareti (+1 veya -1) bir SMC kazancı (K) ile çarpılır. SMC kontrolörünün çıkışı düzeltme faktörüdür (Z). Bu kazanç, sayısallaştırılmış modelden gelen voltaj terimine eklenir ve ölçülen akım (is) ile tahmin edilen akım (is\*) arasındaki hata sıfır olana kadar (yani ölçülen akım ile tahmin edilen akım eşleşene kadar) işlem her kontrol döngüsünde tekrarlanır.

Back-EMF Estimation

Sayısallaştırılmış modeli kompanse ettikten sonra, giriş gerilimi (Vs) ve akım (is\*) için aynı değişken değerlere sahip bir motor modeliniz olur. Sayısallaştırılmış model kompanse edildikten sonra, bir sonraki adım Şekil 18'de gösterildiği gibi düzeltme faktörünü (Z) filtreleyerek geri-EMF'yi (es\*) tahmin etmektir. Geri-EMF tahmini (es\*), her kontrol döngüsünden sonra es\* değişkenini güncellemek için modele geri beslenir. Tahmini Teta hesaplaması için eα ve eβ değerleri (es'in vektör bileşenleri) kullanılır.

Back-EMF Filtering

Filtrelemeyi sağlamak için, Denklem 4 ile birinci dereceden, dijital bir alçak geçiren filtre kullanılır.

EQUATION 4: FIRST-ORDER DIGITAL LOW-PASS FILTER:

---

Kesme frekansı değeri, sürücü akımı ve motor voltajının frekansına, yani saniyedeki elektrik devrine eşit olacak şekilde ayarlanır. Uyarlanabilir filtrenin uygulanması nedeniyle, tüm hız aralıklarında Teta kompanzasyonu için -45°'lik (filtre başına) sabit bir faz gecikmesi vardır, çünkü motor hız kazandıkça kesme frekansı değişir.

İlk filtrenin çıktısı iki blokta kullanılır. İlk blok modelin kendisidir, bir sonraki tahmini akımı (is\*) hesaplamak ve ayrıca tahmini Teta'yı (θ∗) hesaplamak için kullanılır. Motor modelinden çıkan daha düzgün bir sinyali hesaplamak için ikinci bir birinci dereceden filtre kullanılır.

Relationship Between Back-EMF and Rotor Position

Geri-EMF ikinci kez filtrelendikten sonra Teta hesaplanır. es ve θ arasındaki ilişki Şekil 19'da gösterilen grafiğe dayanarak açıklanabilir.

FIGURE 19: BACK-EMF AND THETA RELATIONSHIP

Çizim, geri-EMF'nin vektör bileşenlerini (eα ve eβ) ve rotor açısını (θ) ilişkilendiren bir trigonometrik fonksiyonu göstermektedir. Arctangent, Teta'yı hesaplamak için geri-EMF vektör bileşenleri üzerinde hesaplanır. Denklem 5, fonksiyonun yazılımda nasıl uygulandığını göstermektedir:

EQUATION 5: THETA CALCULATION

--

Gerçek uygulama, hızlı olan ve kayan nokta uygulamasından daha az bellek kullanan Dijital Bilgisayarla Koordinat Rotasyonu (CORDIC) adlı sayısal ve yinelemeli bir algoritma kullanır. CORDIC algoritmasının tartışılması bu uygulama notunun kapsamı dışındadır.

Speed Calculation

Teta hesaplaması sırasında uygulanan filtreleme işlevi nedeniyle, hesaplanan açı motor sargılarına enerji vermek için kullanılmadan önce bir miktar faz kompanzasyonu gerekir. Teta kompanzasyonunun miktarı Teta'nın değişim oranına veya motorun hızına bağlıdır. Teta kompanzasyonu iki adımda gerçekleştirilir:

1. Motorun hızı, telafi edilmemiş Teta hesaplamasına göre hesaplanır.
2. Hesaplanan hız filtrelenir ve Şekil 20'de gösterildiği gibi telafi miktarını hesaplamak için kullanılır.

Hız, m örnek üzerinden Teta değerlerinin biriktirilmesi ve ardından biriktirilen Teta'nın bir sabitle çarpılmasıyla hesaplanır. Bu uygulama notunda hız hesaplaması için kullanılan formül Denklem 6'da gösterilmiştir.

EQUATION 6: SPEED CALCULATION

---

Hız hesaplamasında daha düzgün bir sinyal elde etmek için Omega'ya (ω\*) birinci dereceden bir filtre uygulanarak FilteredOmega (ω\*filtered) elde edilir. Birinci dereceden filtre topolojisi, geri-EMF filtrelemesi için kullanılanla aynıdır.

Adaptive Filter

Uyarlanabilir filtre aşağıdaki iki işlevi yerine getirir:

* Kayar modlu kontrolör için alçak geçiren filtre kazancının hesaplanması
* Tüm hız aralığı için Teta'yı dinamik olarak telafi edin

Konum tahmini için iki alçak geçiren filtre uygulanmıştır. İlk filtre, kayan mod kontrolörünün çıkışını (düzeltme faktörü (Z)) tahmini geri-EMF'ye (es\*) filtreler ve ikinci filtre, tahmini geri-EMF'yi (es\*) filtrelenmiş tahmini geri-EMF'ye (efiltered\*s) filtreler. Alçak geçiren filtre için kazancın hesaplanmasına yönelik türetme Denklem 7'de gösterilmektedir.

EQUATION 7:

---

Uyarlanabilir filtrenin tasarımı, kesme frekansı artan motor hızıyla birlikte değişmeye devam ettiğinden, tüm hız aralıklarında Teta telafisi için sabit bir faz gecikmesi tutar. İki uyarlanabilir filtrenin uygulanması nedeniyle, 90°'lik sabit faz gecikmesi yalnızca hesaplanan Teta'ya eklenen sabit bir ofsettir.

FIELD WEAKENING

PMSM için alan zayıflaması, dönen çerçevenin d eksenindeki stator akımı için negatif bir değer empoze edilmesi anlamına gelir ve bu da hava aralığı akı bağlantısını zayıflatma rolüne sahiptir.

Bir invertör durumunda, voltaj çıkışı statorun direnci ve endüktif reaktansı üzerinde düşer ve kalan voltaj BEMF'ye karşı koymak için kullanılır. BEMF, motorun hızı ve motorun gerilim sabiti ΚΦ ile orantılıdır. Sürücünün maksimum çıkış gerilimi sınırlaması göz önüne alındığında, hava aralığı akı bağlantısı ile orantılı olan gerilim sabitini (ΚΦ) azaltarak hızda (nominal hızın üzerinde) bir artış elde edilebilir. Bununla birlikte, hava aralığı akı bağlantısındaki bir azalma, torktaki azalma ile eş anlamlıdır. Bununla birlikte, bazı uygulamalar için motorun nominal hızlardan daha yüksek hızlarda çalışması gerekir ve bu nedenle, motorun hız aralığını nominal hız değerinin ötesine yükselten alan zayıflatma özelliği kullanışlıdır.

PMSM'de alan zayıflatma Id değerinin '0'dan negatif bir değere düşürülmesiyle elde edilir. Alan zayıflatma tablosu UserParms.h'de tanımlanan Id değerlerinden oluşur (dqKFw0'dan dqKFw15'e kadar) ve bunlar kullanıcı gereksinimine göre değiştirilebilir. Kullanıcı alan zayıflatma özelliğini kullanmak istemiyorsa UserParms.h'de NOMINALSPEEDINRPM değerinin FIELDWEAKSPEEDRPM değerine eşit olmasına izin verin.

FieldWeakening fonksiyonu CtrlParm.qVelRef hız referansı giriş parametresini alır ve değerin UserParms.h içinde tanımlanan NOMINALSPEEDINRPM değerinden küçük veya büyük olup olmadığını kontrol eder.

* Hız referansı daha küçükse, alan zayıflaması gerçekleştirilmez
* Hız referansı daha büyükse, uygun değer doğrusal enterpolasyon yoluyla FdWeakParm.qFwCurve'den döndürülür

Şekil 21'de alan zayıflatmanın blok diyagramı gösterilmektedir.

Şekil 22, test sırasında kullanılan bir alan zayıflama eğrisini göstermektedir.

PERFORMANCE MODE

Speed Mode

Hız modunda, ölçülen motor hızı bir PI döngüsü kullanılarak potansiyometreden gelen referans ile karşılaştırılır. PI çıkışı, akımın Q-bileşeninin PI'sine giriş olarak uygulanır.

Motor çalışması sırasında, yükteki değişimden bağımsız olarak mil devri sabit tutulur. Bu mod, UserParms.h dosyasındaki //#define TORQUEMODE tanımına yorum yapılarak seçilebilir. Şekil 23'te Hız modunun blok diyagramı gösterilmektedir.

Torque Mode

Tork modunda, hız PI döngüsü bypass edilir ve potansiyometreden gelen referans doğrudan akımın Q bileşeninin PI döngüsüne giriş olarak beslenir.

Motorun çalışması sırasında, motor tarafından üretilen tork ve akım tüketimi sabit tutulur (potansiyometre tarafından ayarlandığı gibi). Bu nedenle, daha ağır bir yük altında mil devri düşebilir. Bu mod, UserParms.h dosyasındaki //#define TORQUEMODE tanımının yorumu kaldırılarak seçilebilir.

Şekil 24'te Tork modunun blok diyagramı gösterilmektedir.

Voltage Ripple Compensation

Dalgalanma telafisi, DC bara dalgalanmasına bağlı olarak Vd ve Vq'yu (ters park dönüşümü bloğuna girişler) telafi etmek için kullanılır. Kullanıcı, UserParms.h dosyasında #define ENVOLTRIPPLE ile kodun dalgalanma telafisi bölümünü etkinleştirebilir. Bu özellik kodda etkinleştirilirse, yazılım DC veri yolundaki voltaj dalgalanmasını telafi eder. Bu da DC veri yolu üzerindeki tampon kapasitörün boyutunu azaltarak donanım tasarımını ekonomik hale getirir.

Denklem 8, D bileşeni için dalgalanma kompanzasyonunun uygulanmasını açıklamaktadır. Şekil 25 ve Şekil 26 sırasıyla D ekseni ve Q ekseni için bara gerilimi dalgalanma kompanzasyonunun blok diyagramını göstermektedir

EQUATION 8:

---

FLOW CHARTS

FOC algoritması PWM ile aynı hızda yürütülür. PWM, iki şönt direnç ve motorun referans hızını ayarlayan bir potansiyometre kullanarak iki sargı için A/D dönüşümlerini tetikleyecek şekilde yapılandırılmıştır. Algoritmayı gerçekleştirmek için A/D'nin kesmeleri etkinleştirilir. Şekil 27, A/D kesme alt rutininin genel yürütülmesini göstermektedir.

Şekil 28, motorun konumunu ve hızını tahmin etmek için SlideMode Kontrolörünü kullanma sürecini göstermektedir

MOTOR START-UP

Sensörsüz FOC algoritması geri-EMF tahminine dayandığından, tahmini geri-EMF değerini elde etmek için minimum bir hıza ihtiyaç vardır. Bu nedenle, motor sargılarına uygun tahmini açı ile enerji verilmelidir. Bunun üstesinden gelmek için bir motor başlatma alt rutini (bkz. Şekil 29) geliştirilmiştir.

Motor durduğunda ve Başlat/Durdur düğmesine basıldığında, dsPIC DSC motoru döndürmeye başlamak için bir dizi sinüzoidal gerilim üretir. Motor sabit bir hızlanma oranında döner ve FOC algoritması Id ve Iq akımlarını kontrol eder. Teta açısı (komütasyon açısı) hızlanma oranına bağlı olarak artar.

Diyagramda gösterildiği gibi, motorda sabit bir ivme elde etmek için faz açısı karesel bir oranda artırılır. Theta açık döngü durum makinesi tarafından üretiliyor olsa bile, FOC blokları hala yürütülmekte ve tork bileşeni akımını ve akı bileşeni akımını kontrol etmektedir. Motoru çalıştırmak için gereken istenen torku ayarlamak için harici bir potansiyometre kullanılır. Bu potansiyometre mekanik yük özelliklerine bağlı olarak deneysel olarak ayarlanır. Bu başlatma alt rutini motoru başlatmak için sabit bir tork sağlar. Başlatma rampasının sonunda yazılım, Şekil 6'da gösterildiği gibi konum ve hız tahmincisinden Theta alarak kapalı döngü, sensörsüz kontrole geçer.

MAIN SOFTWARE STATE MACHINE

FOC algoritmasını bir yazılım durum makinesi olarak görselleştirmek faydalı olacaktır (bkz. Şekil 30). İlk olarak, motor sargılarının enerjisi kesilir ve sistem kullanıcının Başlat/Durdur düğmesine (dsPICDEM MCLV Geliştirme Kartı üzerindeki S2) basmasını bekler. Kullanıcı Başlat/Durdur düğmesine bastığında, sistem tüm değişkenlerin ilk değerlerine ayarlandığı ve kesmelerin etkinleştirildiği başlatma durumuna girer. Ardından, tork (Iq) ve akı üretimi (Id) için akım bileşenlerinin kontrol edildiği ve motorun dönmesini sağlamak için komütasyon açısının (Theta) bir rampa şeklinde oluşturulduğu başlatma alt rutini yürütülür.

Başlangıç alt rutininden geçtikten sonra, sistem sensörsüz FOC kontrolüne geçer, burada hız kontrolörü yürütme iş parçacığına eklenir ve Kayma Modu Kontrolörü (SMC) daha önce açıklandığı gibi Teta'yı tahmin etmeye başlar. Motor sensörsüz FOC kontrol durumuna girdiğinde, referans hız harici bir potansiyometreden sürekli olarak okunur ve motoru durdurmak için başlat/durdur düğmesi izlenir.

Sistemdeki herhangi bir hata motorun durmasına ve S2'ye tekrar basılana kadar Motor Durdu durumuna dönmesine neden olur. Durum diyagramı yazılımın tüm farklı durumlarını ve sistemin farklı bir duruma geçmesini sağlayan koşulları gösterir.

BENEFITS OF DSC-BASED FOC CONTROL

Motor kontrolünde DSC'lerin kullanılmasının önemli bir avantajı, cihazların üretimini daha verimli hale getiren ortak bir tasarım platformunun pratikliğidir. Bu, cihaz üreticilerinin artık PMSM veya sensörsüz FOC algoritma kontrollü diğer motor türlerini kullanan bir dizi cihaz modeli sunmanın ekonomik bir yoluna sahip olduğu anlamına gelir.

Bu yazılım tabanlı motor kontrol tasarımları, yalnızca kontrol parametrelerini değiştirerek birden fazla pazara hitap etmek için hızlı özelleştirme sağlar.

Ürün yazılımı Fikri Mülkiyetinin (IP) korunması, birçok coğrafyada işbirliği yapan cihaz tasarım ekiplerini sıklıkla kullanan üreticiler için bir diğer önemli konudur. Bir cihaz için FOC uygulamasının A yerinden, kullanıcı arayüzü kartının B yerinden ve nihai sistem entegrasyonunun C yerinden gelmiş olabileceği bir senaryoyu hayal etmek kolaydır.

CONCLUSION

Bu uygulama notu, tasarımcıların cihaz uygulamalarında Sensörsüz FOC algoritması gibi gelişmiş motor kontrol tekniklerini uygulamak için DSC'lerden nasıl yararlanabileceklerini göstermektedir. Bir dsPIC DSC'yi programlamak bir MCU'yu programlamaya benzediğinden, cihaz tasarımcıları motor kontrol algoritmalarını hızlı bir şekilde tasarlayabilir ve prototiplerini test edebilirler.

Tasarımcının algoritmalarını PMSM, BLDC, BDC ve ACIM gibi çeşitli motor platformlarına kolayca taşımasına olanak tanıyan DMCI gibi güçlü IDE tabanlı araçlar sayesinde motor kontrolünde ince ayar yapmak kolaylaşır.































































