

Değiştirilebilir Sanal Enstrümanlar Teknolojisi ile Kalibrasyon Otomasyonu Uygulamalarında Performans Artırılması

Osman Torunoğlu¹ Nurhan Karaboğa²

Öz

Bu çalışmada elektronik ölçüm cihazlarının periyodik kalibrasyonunun daha hızlı ve yüksek doğruluklu yapılabilmesi amacıyla, yeni nesil yazılım ve donanım destekleri kullanılarak kalibrasyon prosedürlerinin otomasyonu gerçekleştirilmiştir. Ölçüm cihazlarının kontrolünde son yıllarda ön plana çıkan Değiştirilebilir Sanal Cihazlar (Interchangeable Virtual Instruments, IVI) teknolojisini kalibrasyon otomasyonu yazılımında kullanılarak, otomasyon sisteminin performansının ve esnekliğinin artırılması hedeflenmiştir. IVI standartlarının sunduğu cihazların değiştirilebilirliği (interchangeability), cihaz benzetimi (simulation) ve durum önbellekleme (state-caching) özellikleri ile sistemin uzun süreçte idame ettirilebilirliğinin sağlanması örnek bir uygulamada gösterilmiştir. Ayrıca bir kalibrasyon otomasyonunun asıl hedefi olan zaman ve insan gücü tasarrufuna yazılım parametrelerinin etkisi örnek uygulama üzerinden ele alınmıştır.

Anahtar Kelimeler: Kalibrasyon, Otomasyon, IVI standartları.

Performance Enhancement in Calibration Automation Applications Using Interchangeable Virtual Instruments

Abstract

In this study, calibration procedures are automated using novel software and hardware support in order to obtain high accuracy and faster process in the periodical calibration activities of the electronic measurement instruments. The performance and flexibility of the automation system are intended to be increased by the employment of the Interchangeable Virtual Instruments (IVI) technology which has emerged recently in the control of the measuring instruments. The long-term maintainability of the system is demonstrated in a sample of application using the interchangeability, the simulation and the state-caching properties of the IVI standards. Also, the effect of the software parameters on the time and workforce savings which are the real target of the calibration automation is discussed in the sample of application.

Keywords: Calibration, Automation, IVI standards.

¹ Yazışma adresi: Müh.Tğm., Kara Harp Okulu, Dekanlık, Elektronik Mühendisliği Bölümü, Bakanlıklar, Ankara, otorunoglu@kho.edu.tr

² Doç.Dr., Erciyes Üniversitesi, Mühendislik Fakültesi, Elektrik Elektronik Mühendisliği Bölümü, Kayseri, nurhan_k@erciyes.edu.tr

Giriş

Ölçüm cihazlarının veya sistemlerin doğruluğunun sürekliliği periyodik olarak yapılan kalibrasyon işlemleri ile sağlanmaktadır. Gelişen teknoloji ile birlikte, ölçüm değerlerinin güvenilirliğini belirleyen kalibrasyon işlemleri, teknolojinin vazgeçilmez bir parçası hâline gelmiştir. Son yıllarda endüstriyel uygulamalarda, yapılan ölçümlerden istenen hassasiyetler de artmakta, yüksek doğrulukta ölçüm sonuçları veren ve güvenilirliği kabul edilebilir sınırlar içerisinde olan cihazlar daha fazla talep görmektedir.

Kalibrasyon, ölçüm cihazlarının ölçtüğü değerlerin doğruluklarının tespiti, sapmalarının belirlenmesi ve bu değerlerin ölçümlere etki eden belirsizlik aralıklarının da verildiği, standartlara uygun bir sertifika ile cihaz kullanıcılarına sunulmasını kapsayan işlemler dizisidir. Kalibrasyon ile doğruluğu tescillenen veya hata değerleri belirlenen cihazlarla çalışarak üretim, arge, eğitim gibi faaliyetlerin güvenilirliğini arttırmak, cihaz kullanıcıları için oldukça büyük bir önem arz etmektedir. Ancak bir başka açıdan bakıldığında, kalibrasyon, cihazların güvenilirliğini tespit etmek ve sürdürmek için önemli olmakla birlikte, hem kullanıcı hem de kalibrasyon laboratuvarı cihazların kalibrasyon işlemlerinin en kısa sürede en doğru şekilde yapılmasını istemektedirler. Özellikle ölçüm cihazlarının eksikliğinin büyük zafiyetlere neden olacağı askerî uygulamalar, arge çalışmaları ve üretim süreçlerinde cihazın kalibrasyon süresini en aza indirmek gerekmektedir.

Savunma teknolojilerine yönelik uygulamalarda, ölçüm cihazlarının önemli bir yere sahip olduğu kesindir. Atış kontrol sistemlerinin hedef üzerinde yüksek isabet sağlamasında, iletişim ağında güvenilir ve yüksek kaliteli haberleşmenin sağlanmasında, hava araçlarının doğru rotada ilerlemesinde ve kamera görüntülerinin yüksek çözünürlüklü ve kaliteli olmasında ölçüm cihazlarının kullanılması kaçınılmazdır. Ayrıca, nükleer serpinti ölçümü, biyolojik ve kimyasal madde tespiti doğrudan ölçüm cihazları ile yapılmaktadır. Sağlık hizmetleri açısından incelendiğinde, günümüzde ölçüm cihazlarından alınan veriler ışığında yapılan teşhis ve tedaviler ölçüm cihazlarının bulunduğu konumu göstermektedir. Sonuç olarak savunma teknolojileri ve askerî uygulamalar açısından bakıldığında ölçüm cihazlarının konumu ve önemi tartışılmazdır. Bu kapsamda ölçüm cihazlarının güvenilirliği askerî uygulamalarda başarıyı getirecektir. Cihazların güvenilirliği ise ancak cihazların kalibrasyon işlemlerine tabii tutulması ile sağlanmaktadır. Cihazın kalibrasyon süresince

kullanılmamasından kaynaklanabilecek zafiyetler ise kalibrasyon süresini en aza indirmekle giderilmektedir.

Askerî uygulamalardaki elektronik ölçüm cihazlarının kalibrasyonuna bakıldığında, işlemleri yazılım ve donanım destekleri kullanarak otomasyon hâline getirmek, kalibrasyon sürecinde zaman ve insan gücü tasarrufu sağlamaktadır.

Kalibrasyon işlemlerini, bir kontrol birimi üzerinden yöneterek, arayüz donanımları ile cihazlara gerekli komutları yollamak, komutlara verilen cevapları çekmek ve elde edilen verileri belirli algoritmalarla işlemek suretiyle insan gücünü en aza indirerek gerçekleştirme faaliyeti, kalibrasyon otomasyonu olarak tanımlanmaktadır. Bir kalibrasyon işleminde otomasyona uygun aktiviteler, ölçüm cihazları ve sistemlerinin kalibrasyonu, sonuçların sertifika edilmesi, verilerin işlenmesi, izlenmesi ve depolanmasıdır (Lili, 2007; Creasman ve Pheanis, 2006).

Kalibrasyon otomasyonunun temel yapısı kontrol birimi, donanım ve yazılım üzerine kurgulanır. Yapılan ilk çalışmalarda kalibrasyon otomasyonu sürecini gerçekleştirmek ön plana çıksa da daha sonraki uygulamalarda otomasyon süreçlerinin daha esnek ve optimal nasıl tasarlanacağı üzerinde durulmuştur (Kennedy, 2004). Kalibrasyon otomasyonunun esnekliğinde amaç, bir sistem ile birkaç cihazın kalibrasyonunu gerçekleştirmek ya da ileri süreçte yeni bir cihazın kalibrasyonunu, sistemde küçük değişiklikler yaparak gerçekleştirebilmektir.

Diğer yandan test ve ölçüm cihazlarının otomasyonunda optimal yaklaşım, otomasyon sistemlerinin uzun süreçte bakım ve onarım maliyetlerini en aza indirmektir (Cheij, 2001). Bu açıdan bakıldığında kalibrasyon otomasyonu uygulamalarının da bakım ve onarım maliyetleri düşük olacak şekilde yapılacak tasarımların tercih edileceği açıktır.

Kalibrasyon otomasyonu sürecinin esnekliğini arttırmak amacıyla yapılan çalışmalar incelendiğinde ilk olarak kontrol birimi seçiminin incelendiği (Randle ve Kerth, 1978) ya da özel tasarım kontrol birimlerinin kullanıldığı bazı uygulamalar (Drnovsek vd, 1998) ön plana çıkmaktadır. Süreci daha kullanışlı ve optimal hâle getirmek amacıyla arayüz donanımı ve cihaz bileşenleri seçimine önem veren uygulamalar da literatürde mevcuttur (Filipski ve Rinfet, 2000).

Ancak otomasyona esnekliği veren asıl bileşenin yazılım olduğu değerlendirilmektedir. Özellikle yeni gelişen kod standartlarının desteğiyle

yapılan çalışmalarda özgün farklar ön plana çıkmaktadır (Lili, 2007). Yazılım dili tercihi de otomasyonun esnekliğini doğrudan etkileyen bir parametredir. Gelişen yazılım teknolojileri, ölçüm cihazlarının uzaktan kontrolünde önemli avantajlar sağlamaktadır. Özellikle, geliştirilen yeni nesil komut aileleri, kullanıcıya standartlaşmayı sağlamaktadır. IEEE 488.2, Programlanabilir Cihazlar İçin Standart Komutları (Standard Commands for Programmable Instruments, SCPI) ve IVI kod standartları ile geliştirilen yazılımlar daha kolay uygulanabilir ve esnek yapıya sahip olmaktadır. IVI kuruluşu tarafından oluşturulan IVI standartları, yazılımda herhangi bir değişim yapmadan, aynı işlevli cihazların fiziksel değiştirilebilirliği felsefesi ile ön plana çıkmaktadır (Bode, 2003).

Gelişen cihaz teknolojileri ile birlikte, kalibrasyon laboratuvarlarına gelen farklı üreticilerin aynı işlemleri yapan cihazlarının, geliştirilecek olan tek bir otomasyon programı üzerinden kalibrasyonlarının yapılması olasılığı önem kazanmaktadır. Yeni nesil komut standartları kullanılarak yapılacak olan kalibrasyon otomasyonu yazılımları ile program sayısı en aza indirilebilmekte ve kalibrasyon otomasyonu yazılımlarının idamesi, bakım ve onarımı daha kolay ve etkin hâle getirilebilmektedir.

Bu çalışmada, kalibrasyon otomasyonu yazılımında IVI standartlarının kullanılması ile otomasyon sisteminin performansının ve esnekliğinin artacağı ve uzun süreçte bakım ve onarım faaliyetlerinin düşeceği üzerine durulmuştur. Örnek uygulamada, IVI standartları ve sürücülerinin kullanıldığı bir kalibrasyon otomasyonu programı tasarlanarak verimliliği incelenmiştir.

IVI Standartları ve Sürücüler

IVI Kuruluşu

IVI standartlarının önemini kavramak için cihaz komut standartları teknolojisinin gelişimini gözden geçirmek gerekmektedir. Elektronik test ve ölçüm cihazlarına herhangi bir işlem yaptırmak için, cihazlara anlam ifade eden komutların yollanması gereklidir.

İlk oluşturulan komutlar daha çok cihazların ana çalışma hatları ile ilgilenmiştir. Komut standardı adına tanımlanan ilk standart denilebilecek IEEE 488.2 standardı ile bazı genel kodlar bir standart haline getirilmiştir. Cihazların kendine has fonksiyonlarını yerine getiren komutlarında herhangi bir standartlaşma olmamıştır. Standartlaşmanın olmaması her cihaz üreticisinin kendi komut sistemini oluşturmasına neden olmuş, bu durum da

bir komut karmaşasını beraberinde getirmiştir. Farklı üreticilerin aynı işlemi yapan cihazlarının farklı kodlarla çalışmasının oluşturduğu olumsuzluklara çözüm olarak 1990'lı yılların başında SCPI komut sistemi tanımlanmıştır. SCPI ile aynı fonksiyonları icra eden cihazlar ortak bir komut kümesi ile çalışabilir hâle gelmiştir. Böylece cihaz komut standartlarında ilk kez fonksiyon temelli bir standartlaşma oluşturulmuştur. Yazılım teknolojisindeki gelişmeler, 1993 yılında cihaz sürücüsü (instrument drivers) kavramını ortaya çıkarmıştır. İlk cihaz sürücüleri SCPI komutlarının taban olarak kullanıldığı Enstrümantasyon için VME (Versa Module Europa) Uzantıları ile Tak Çalıştır (VME eXtensions for Instrumentation, VXIplug&play) teknolojisi ile oluşturulmaya başlanmıştır. Sürücülerin sisteme tanıtılması ile cihazların kontrolü bu sürücüler üzerinden gerçekleştirilebilir olmuştur. Ancak VXIplug&play tarafından oluşturulan sürücüler cihazların fonksiyonlarını tam olarak ifade edememişlerdir. VXIplug&play daha çok cihazlarla haberleşme üzerine durmuş, Genel Amaçlı Arayüz Yolu (General Purpose Interface Bus, GPIB), VXI yolu (VXIbus), RS-232 gibi giriş çıkış (I/O) arayüzlerini kullanabilmek amacıyla ortak yazılım mimarilerini içermiştir. 1998 yılında IVI kuruluşu oluşturulmuş, SCPI ve VXIplug&play'de tanımlı standartlar geliştirilerek cihaz sınıflarına yönelik özgün sürücüler oluşturulmaya başlanmıştır (Cheij, 2001). IVI kuruluşu, cihaz kontrol komutlarında oluşturduğu standartlar ile cihaz üreticilerinden son kullanıcıya kadar uzanan süreçte geniş destek görmüştür. Bu geniş destek, cihaz standartlaşmasının o ana kadar yapılmaması, ancak bir o kadar gerekli olmasından kaynaklanmaktadır.

IVI kuruluşunun farklı üreticiler ve kullanıcılar tarafından desteklenmesi ve oluşturulan standart ve sürücülerin ücretsiz olarak temin edilebilmesi kuruluşun ticari düşüncelerden uzak, bilimsel çalışmalar yapmasını da sağlamaktadır. Böylece IVI standartları ve sürücüleri, yazılım teknolojisindeki gelişme ve yenilikleri cihaz kontrol teknolojisine aktarmaktadır.

Özellikle yeni nesil nesneye yönelik programlama dilleriyle uyumlu çalışması IVI standartları ve sürücülerinin en büyük avantajlarından biridir. Nesne tabanlı dillerin kod yazım kolaylığı getiren uygulamaları ile SCPI komut standartlarında karşılaşılan kod yazım hataları en aza inmektedir. Ayrıca sınıf mantığı ile oluşturulan programlar daha sonraki geliştirme süreçlerinde kod geliştiricilere büyük kolaylıklar sağlamaktadır (Lind vd, 2011).

IVI Sürücü Mimarisi ve Cihaz Sınıfları

IVI standartları ile SCPI ortak komutlarının dil seviyesi arttırılmıştır. Böylece SCPI standartlarının en büyük sorunu olan kodlama hataları azaltılmıştır. IVI'nin IVI-C (C programlama dili tabanlı IVI sürücüleri) ve IVI-COM (Ortak Nesne Modeli tabanlı IVI sürücüleri) olmak üzere iki ayrı sürücü mimarisi vardır. IVI-C programlama dili olarak ANCI-C tabanlı dillere hitap etmektedir. IVI-COM ise yeni nesil nesneye yönelik programlama dillerinin IVI standardı ile uyumunu sağlamaktadır. Nesneye yönelik programlama dillerinin sınıflandırma felsefesi IVI-COM'da da kullanılmıştır (Niu ve Xu, 2008).

IVI, cihazları fonksiyonel özelliklerine göre sınıflara ayırmış ve bu cihaz sınıfları için ortak komutların oluşturulduğu sürücüler tanımlamıştır. Örneğin bir dijital multimetre (Dijital Multimeter, DMM) sınıfında ortak olarak bütün cihazlar DC voltaj ölçümü yapmaktadır. IVI standardı bu şekildeki ortak komutlara yönelik yazılım tabanlı ortak sürücüler oluşturmaktadır. Günümüzde IVI kuruluşunun oluşturduğu dokuz adet sınıf mevcuttur; Güç Kaynağı, DMM, Fonksiyon Üretici, Osiloskop, Güç Ölçer, RF Sinyal Üretici, Spektrum Analizörü, Anahtarlama, Sayıcı/Zamanlayıcı (Cheij, 2001).

IVI Standartlarının Faydaları

IVI standartlarının test ve ölçüm sistemlerine kattığı faydalar üç ana başlık altında, cihazların değiştirilebilirliği, cihaz benzetimi ve durum önbellekleme ile performans artırımı olarak sıralanabilir.

Cihaz Değiştirilebilirliği: Şüphesiz ki IVI standartlarının oluşturduğu en büyük avantaj cihazların değiştirilebilirliği özelliğidir. IVI standartları test ve ölçüm yazılımında çok az değişiklikle ya da hiçbir değişiklik yapmadan donanımsal olarak cihazların değiştirilebilmesini amaçlamaktadır (Fertitta, 2010). Değişimler aynı fonksiyonları işleten cihaz sınıfları içerisinde mümkün olmaktadır. Farklı üreticilerin IVI standartlarını destekleyen aynı sınıf cihazları, herhangi bir arıza, performans artırımı ya da cihaz yenileme durumlarında yazılımsal olarak değişiklik yapmadan değiştirilebilir kılınmaktadır. Bu özellik ile yazılım geliştiricileri bir cihaz değişiminde yazılıma herhangi bir müdahalede bulunmamaktadır.

Cihazların takas edilme özelliği yazılım geliştirme grubuna da ortak bir avantaj sunmaktadır. Cihaz geliştiricilerinin sistemde kod değişimine gitmeden eşzamanlı olarak cihaz ekleme çıkarma durumunu sağlamaktadır. Bir test sistemini birden fazla geliştiricinin oluşturduğu düşünüldüğünde ve

her birinin aynı cihazlara kodlarla müdahale ettiği bir durumda, bu avantaj ile yazılım geliştirici diğer çalışanların oluşturduğu kodlara herhangi bir müdahaleye gerek duymadan cihaz değişimine uygun kodlar kullanabilmektedir. Böylece çalışma grubunun esnekliği ve hızı artmaktadır.

IVI standardının cihaz değiştirilebilirliği özelliği ile uzun süreç bakım onarım maliyeti en aza indirilmektedir. Cihazların arıza veya eskimeden dolayı kullanım dışı olması durumunda yazılım kodlarına hiçbir müdahale etmeden ya da çok az bir değişiklik yaparak donanımsal olarak aynı standartları destekleyen cihazlar kolaylıkla yer değiştirebilir. Böylece, yazılım tabanlı maliyetler ve iş gücü en aza indirilir (Fertitta, 2010).

Cihaz Benzetimi: IVI standartlarının oluşturduğu bir başka avantaj ise cihaz benzetimidir. IVI cihaz sürücüleri ile uygun programlar kullanılarak, cihaz gerektirmeden benzetim yapılabilir. Cihaz benzetimi kullanılarak kodların testi ve geliştirilmesi cihazsız olarak gerçekleştirilmektedir. Yazılım konfigürasyon adımları sürücüler ile hazırlanmakta ve komutlar sonucunda cihazın vereceği cevaplar da üretilmektedir. Sonuç olarak IVI'nin cihaz benzetimi ile kod geliştirme ve test sürecinde donanımsal maliyet etkin bir şekilde azaltılmaktadır (Cheij, 2001).

Durum Önbelleklemesi: IVI standartlarının oluşturduğu diğer bir avantaj ise, durum önbelleklemesi yaparak kontrol birimi ile cihaz arasındaki gereksiz I/O işlemleri en aza indirmektir. Durum önbelleklemesi ile gereksiz işlemlerden ve beklemelerden arındırılmış sistemin performansı artmaktadır. Sistemin performans artırımında, kanal güvenliği sağlanarak bir anda birden çok kanalın iş yapabilmesi durumu da söz konusudur (Cheij, 2001).

IVI Tabanlı Kalibrasyon Otomasyonu Uygulaması

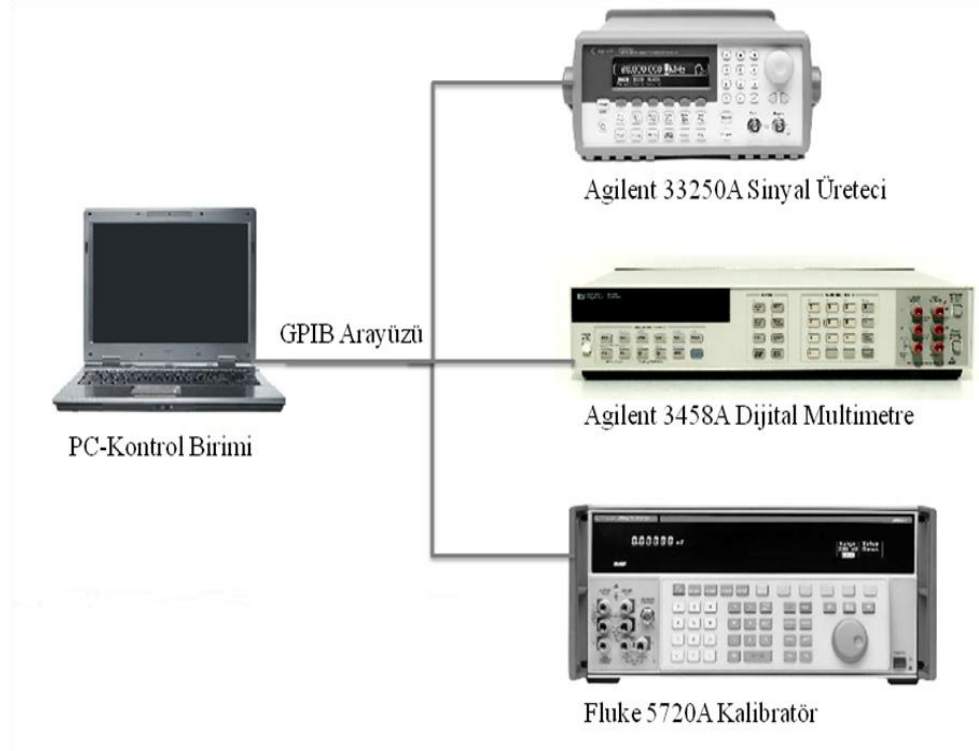
Bu çalışmada 8½ dijital Agilent 3458A model DMM'in kalibrasyon otomasyonu gerçekleştirilmiştir. Agilent 3458A ve aynı tip 8½ dijital hassasiyete sahip multimetreler kalibrasyon laboratuvarlarında ölçüm standardı olarak kullanılmaktadır. Bu özelliğinden dolayı cihazdan yüksek doğruluklu ve düşük belirsizlik değerlerine sahip kalibrasyon sonuçları talep edilmektedir. Diğer yandan cihazın manuel kalibrasyon sürecinin uzun sürmesi ve sürecin yüksek hata olasılığına sahip olması, bu cihaz üzerine çalışma yapılmasının sebepleri arasında gelmektedir.

Cihazın kalibrasyonunda prosedürlerin önerdiği referans cihazlar olarak Fluke 5720A model çok fonksiyonlu kalibratör ile Agilent 33250A

model sinyal üretici kullanılmıştır. Şekil 1’de cihazın kurulum konfigürasyonu verilmiştir.

Hızlı ve güvenilir bir kalibrasyon için cihazın kalibrasyon prosedürleri ISO 17025 standardının belirlediği çerçevede otomasyon uygulaması haline getirilmiştir. Kalibrasyon laboratuvarları için gerekli standartların belirtildiği ISO 17025 standardı, kalibrasyon prosedürlerin yazılım destekli uygulamalara dökülmesini kabul etmektedir. Ancak hazırlanan otomasyon sisteminin sonuçları manuel sonuçlarla doğrulanmalıdır (ISO, 2005).

Uygulamanın ana hattını kontrol birimi ve yazılımının seçimi, donanımsal bileşenlerin entegrasyonu ve yazılım desteği ile kalibrasyon prosedürlerinin gerçekleştirilmesi başlıkları oluşturmaktadır.



Şekil 1. Uygulamanın Cihaz Kurulum Konfigürasyonu

Kontrol Birimi ve Yazılımının Seçimi

Otomasyon sisteminde cihaz, haberleşme arayüzü ve yazılım konfigürasyonunun kontrol edilmesi amacıyla bir kontrol birimi kullanılmaktadır. Genellikle uygulamalarda kontrol birimleri olarak kişisel bilgisayarlar (Personal Computer, PC) kullanılmaktadır. Cihazlardan gelen verilerin depolanması ve işlenmesi de kontrol biriminde yapıldığı için kontrol biriminin doğru seçimi verilerin elde edilmesi, işlenmesi ve saklanmasına doğrudan etki etmektedir. Çalışmanın uygulamasında standart donanımlara sahip bir PC, kontrol birimi olarak kullanılmıştır.

Kontrol birimi üzerinden cihazlara komutların gönderilmesi, verilerin çekilmesi ve işlenmesi herhangi bir yazılım dili kullanılarak gerçekleştirilmektedir. Yazılımla kalibrasyon prosedüründe mevcut sıralı işlemler koda dökülerek arayüz ile kullanıcının kullanımına sunulmaktadır. Yazılım dili seçiminde kullanım alanı ve programı oluşturan personelin yazılım bilgisine göre tercihler yapılmaktadır.

Literatürde kullanılan yazılımlar incelendiğinde, farklı uygulamalar için farklı yazılımların tercih edildiği gözlemlenmektedir. Sistem üzerinde yazılım hâkimiyetinin sağlanması istendiği durumlarda alt seviye programlama dillerinin kullanılması (Niu ve Xu, 2008) söz konusu olmaktadır. Program geliştirici personelin kod bilgisinin az gerektiği, benzetim tabanlı yazılımlar ile yapılan uygulamalar da mevcuttur (Drnovsek vd, 1998). Cihaz üreticisi firmaların oluşturduğu, daha özelleşmiş programlama dilleri kullanarak uygulama geliştirmek de mümkündür (Ahmad vd, 2007). Bununla birlikte cihazdan edinilen verilerin yoğun matematiksel işlemlerde kullanılacağı uygulamalarda, matematiksel işlem kolaylığı sunan yazılımlar tercih edilmektedir (Yang vd, 2010).

Uygulama geliştiricisinin yazılım ve kod bilgisi ile bağlantılı olarak, nesneye yönelik programlama dilleri, özellikle yeni nesil standartları desteklediği ve kullanıcı dostu arayüzler oluşturduğu için tercih edilmektedir (Li vd, 2008; Chen vd, 2009). Bu tip yazılım platformlarının kendine has sınıf, miras gibi özelliklerini kullanarak sistemlerin daha etkili kullanıldığı uygulamalar mevcuttur. Bu çalışmanın uygulamasında özellikle IVI standartlarının kullanımının kolaylığından dolayı, nesneye yönelik programlama kullanılmıştır.

Kalibrasyon prosedürlerinin koda dökümü, cihazlarla haberleşmenin sağlanması, verilerin edinimi ve işlenmesi, gerekli matematiksel işlemlerin yapılması ve cihaz kullanıcılarına sunulacak sertifikanın oluşturulması gibi

işlemler yazılım dili üzerinden gerçekleştirileceği için, bu işlemlerin yaptırılabilceği bir yazılım dili tercihi oldukça önemli olmaktadır.

Diğer yandan otomasyonun kontrol edileceği arayüzün de aynı yazılım dili ile oluşturulacağı düşünüldüğünde, tasarlanan arayüzün kullanıcı açısından kolay anlaşılabilir olması, otomasyon yazılımını oluşturan programcı açısından ise kolay tasarlanabilir ve idame ettirilebilir olması gerekmektedir. Yukarıda anlatılan tüm sebepler bir araya getirildiğinde şu sonuç çıkmaktadır; ana yazılım programının seçimi otomasyon sisteminin performansını etkileyen önemli bir parametredir.

Arayüz Donanımı Seçimi

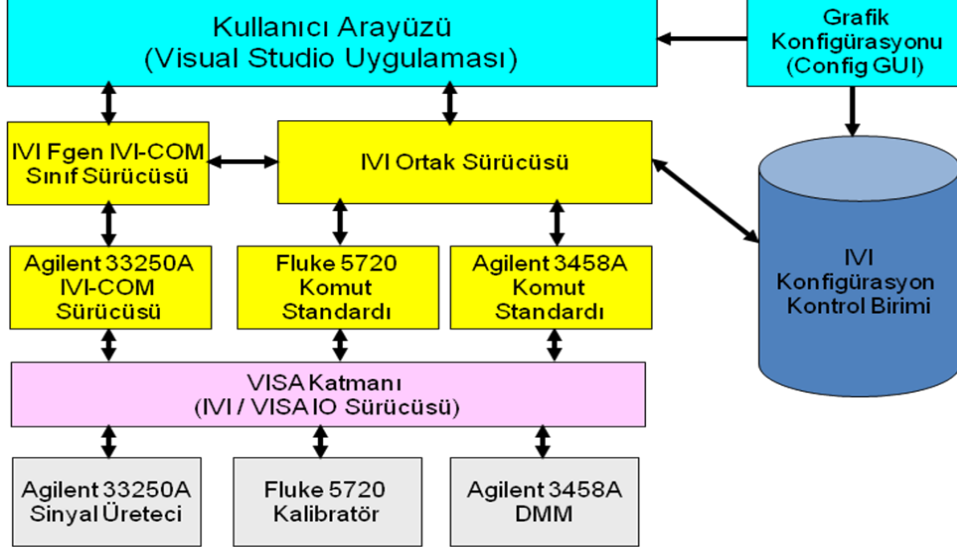
Kalibrasyon işleminde kullanılan cihazlar ile kontrol birimi arasında veri ve işlem akışı arayüz donanımları ile sağlanmaktadır. Arayüz donanımının seçiminde, test edilecek veya veri alınacak cihazın hangi donanımları desteklediği ve sistemin çalışma ortamı etkili parametrelerdir. Yine cihaz sayısı, cihazların kontrol mekanizmasına olan uzaklığı ve çevre koşulları da göz önünde tutulması gereken konulardandır.

Çeşitli uygulamalarda cihazlarla haberleşmeyi sağlayan farklı arayüz donanımları kullanıldığı görülmektedir. Kalibrasyon işleminin laboratuvar ortamında yapıldığı göz önüne alındığında, laboratuvar şartlarına uygun, cihazlarla kolayca adapte olabilen arayüz donanımlarının tercihi ön plana çıkmaktadır. RS-232, Evrensel Seri Yol (Universal Serial Bus, USB), Yerel Alan Ağı (Local Area Network, LAN) ve GPIB cihazlara bağlanmada tercih edilen arayüz donanımlarıdır (Güven ve Uzun, 2007).

Ancak kalibrasyon ve diğer ölçüm uygulamalarında, birden fazla cihaza aynı anda bağlanabilmesi, farklı hızlardaki cihazların aynı sistem içerisinde kullanılmasına izin vermesi, cihaz üreticilerinin geniş desteği, elde edilebilirliği gibi avantajlarından dolayı GPIB standardının diğer standartlara göre daha fazla kullanıldığı gözlemlenmektedir. Bu çalışmanın örnek uygulamasında da arayüz donanımı olarak GPIB kullanılmıştır.

Yazılım Tabanının Oluşturulması

Agilent 3458A DMM'inin kalibrasyonu amacıyla oluşturulacak programın alt yapısı IVI standartları kullanılarak hazırlanmıştır. Kalibrasyonu yapılacak cihaz ve referans değerler üreten cihazlar için oluşturulan IVI altyapısı ile oluşturulan yazılım mimarisi Şekil 2'de verilmiştir.



Şekil 2. Uygulamanın Yazılım Mimarisi

Kullanıcı arayüzünün de oluşturulduğu ana program üzerinden cihazlarla haberleşmenin sağlanması amacıyla, cihazların desteklediği IVI sürücülerini tanımlanmıştır. Fluke 5720 kalibratör ve Agilent 3458A DMM'in haberleşmesi IVI ortak sürücüsü üzerinden cihazlarının kendine has komutlarının yollanması ile gerçekleştirilmektedir. Agilent 33250A sinyal üreticinin haberleşmesi ise, IVI'nin desteklediği IVI COM tabanlı sinyal üretici sınıfı (Fgen) üzerinden gerçekleştirilmektedir. Cihazlarla haberleşmeyi sağlayacak önemli bir yapı da arayüz donanımı üzerinde sanal yolu oluşturacak Sanal Cihaz Yazılım Mimarisi (Virtual Instrument Software Architecture, VISA) katmanıdır. Bu katman da IVI'nin standartlarında bulunan VISA sürücüsü ile oluşturulmuştur.

IVI sürücülerinin sisteme yüklenmesi ile cihazlara işlem yaptıracak kodlar geliştirilebilir hale gelmiştir. Bundan sonraki adımda, sistemin donanımındaki cihazların yazılımsal alanda temsil edilebilmeleri amacıyla, sanal enstrümanlar oluşturulur. İşlemler oluşturulan bu sanal enstrümanlar ile yazılımdan donanıma aktarılır. İşlemlerin aktarımında, daha önceden sisteme tanımlanmış olan IVI sürücülerinin komutları kullanılmaktadır.

Diğer yandan cihaz sürücülerinin sistemle konfigürasyonunu sağlayan konfigürasyon kontrol birimi ile kullanıcı arayüzünü oluşturan grafiksel konfigürasyon birimi yazılım sisteminin destek ünitelerini oluşturmaktadır.

Sistemde kullanılan cihazların tamamının IVI sürücüleri ile kontrol edilmesi, ileriye yönelik yapılacak değişimlerde önemli rol oynayacağı düşünülmektedir. Sistemde bir cihaz arızası olması durumunda, örneğin sinyal üreticinin arıza oluşturması durumunda yazılımsal bileşenlerde çok az değişiklik yaparak yerine benzer özelliklerdeki herhangi bir marka cihaz kullanılabilir.

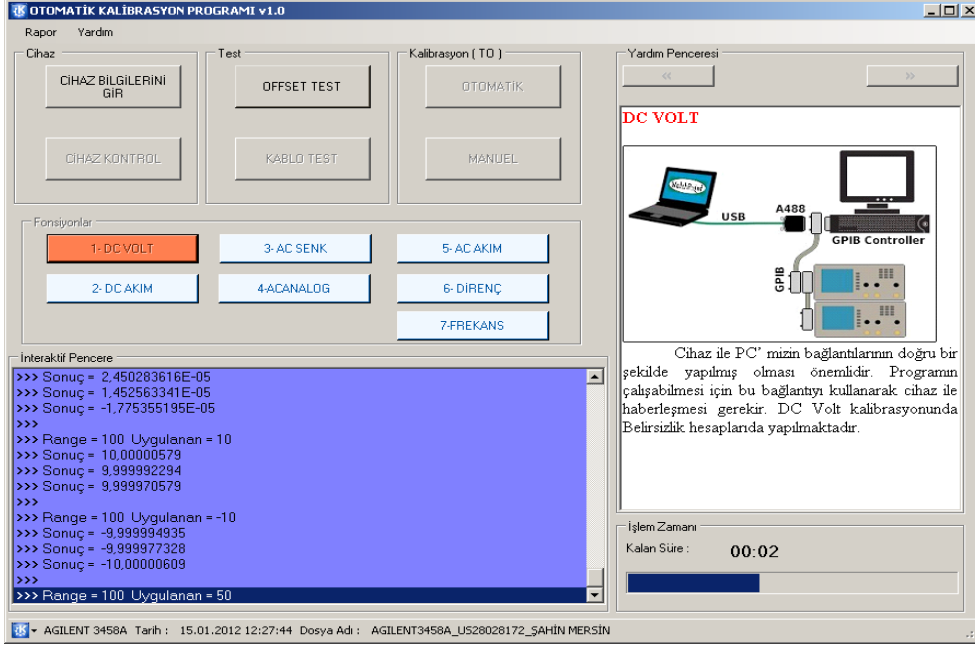
Gerekli IVI sürücülerinin sisteme tanımlanması ile kalibrasyon işlemleri ve bu işlemleri gerçeklemede destek olacak diğer fonksiyonlar oluşturulmuştur. DMM'in kalibrasyonu, üretici değerleri ve EURAMET/cg-15 dokümanına göre belirlenmiş noktalarda referans kalibratörden ve sinyal kaynağından uygulanan değerlerin doğrudan cihaz tarafından ölçülmesi suretiyle gerçekleştirilmiştir. Ayrıca yapılan kalibrasyonun askerî standartlara uygunluğunu sağlamak için, Amerikan Hava Kuvvetleri Metroloji ve Kalibrasyon Programının (Air Force Metrology and Calibration Program Office, AFMETCAL) T.O.33K8-4-551-1 numaralı teknik dokümanı referans alınmıştır (AFMETCAL, 2010).

Uygulamanın Kullanıcı Arayüzü

Kullanıcı arayüzü tasarımında sadelik ve kolay kullanılabilirlik dikkate alınmış, sistemin amacı dışındaki ayrıntılardan uzak durulmuştur. Nesnel özelliklerine ve işlem sırasına göre gruplandırılmış, böylece görsel olarak işlemleri sınıflandırma imkânı sağlanmıştır. Şekil 3'te bütün kontrollerin yapılmasının ardından, DC fonksiyon kalibrasyon işlemi gerçekleştiren program arayüzünün ekranı verilmiştir.

Cihazın kalibrasyonu gerçekleştirilen alt fonksiyonları şunlardır: Ofset, DC Gerilim, AC Gerilim (Senkron, Analog), DC Akım, AC Akım, Direnç (4 kablolu, 2 kablolu), Frekans.

Fonksiyonların her birinde, kalibrasyonu yapılacak cihaza prosedürde belirlenmiş değerler referans kaynaktan uygulanmaktadır. Kalibrasyonu yapılan cihazın uygulanan değerleri algılaması için yeterli zaman verilmekte ve istenilen örnekleme sayısı kadar veri cihazdan çekilerek işlenmek üzere veritabanına kaydedilmektedir.



Şekil 3. DC Volt Kalibrasyon İşlemi Arayüz Ekranı

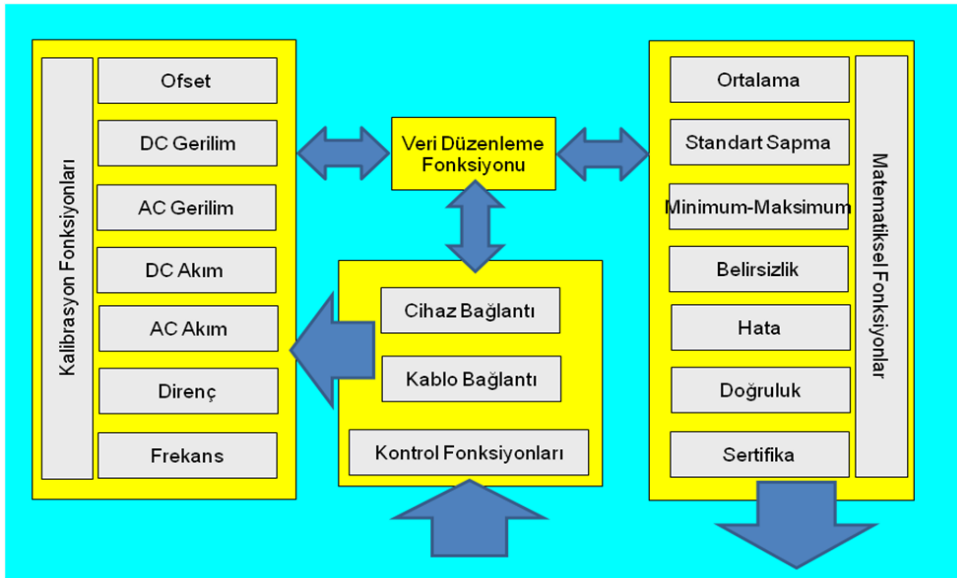
Kalibrasyon işlemlerinin gerçekleştirilmesinde kullanılan yardımcı fonksiyonlar ise aşağıda sıralanmıştır.

Cihaz Kontrol Fonksiyonu: Cihazlar ile haberleşmenin sağlandığı ilgili sanal yolları oluşturan ve doğru olup olmadığını kontrol eden fonksiyondur. Fonksiyon bir yandan sanal yolları kontrol ederken, diğer yandan GPIB arayüzü üzerinde tanımlanan adreslerde belirtilen cihazların olup olmadığını kontrol etmekte ve kullanıcıya bilgi vermektedir.

Kablo Bağlantı Kontrol Fonksiyonu: Her bir kalibrasyon adımının gerçekleştirilmesi anında gerekli olan kablo bağlantılarını kontrol eden fonksiyondur. Bu fonksiyon kullanıcının yaptığı bağlantıların doğruluğu kontrol edilmekte ve hata ihtimali en aza indirilmektedir. Fonksiyon çıktı olarak kullanıcıya bağlantıların doğruluğu hakkında bilgi vermektedir.

Veri Düzenleme Fonksiyonu: Agilent 3458A'dan gelen veriler metinsel ifade türünde olduğu için matematiksel işlemlerde kullanılmamaktadır. Bu fonksiyon ile cihazdan gelen veriler kontrol birimi ve yazılım tabanı için anlamlı hale getirilmektedir. Sertifika çıktısının tamamen bu veriler üzerinden yapılan matematiksel işlemlere dayalı olduğu düşünüldüğünde fonksiyonun kalibrasyon işlemlerinin bütün basamaklarını etkilediği göze çarpmaktadır.

Matematiksel Fonksiyonlar: Kalibrasyon işlemlerinin çıktısı olan sertifikanın hazırlanmasında kullanılacak verileri oluşturan fonksiyonlar bütünüdür. Ortalama, standart sapma, minimum ve maksimum değerler ve hata, doğruluk değerlerinin bulunması matematiksel fonksiyonların altındadır. Bu veriler ile bir kalibrasyon işleminin güvenilirlik seviyesini gösteren, belirsizlik ifadesi hesaplanmaktadır. Şekil 4’te uygulamanın yazılım fonksiyonlarının mimarisi verilmiştir.



Şekil 4. Uygulamanın Yazılım Fonksiyonları Mimarisi

Bulgular

Kalibrasyon otomasyonu uygulamasında yazılım sürecini IVI standartları üzerinden geliştirmeyi amaçlayan çalışma hem kalibrasyon işlemi hem de otomasyon yazılımı açısından çeşitli veriler sunmaktadır. Bir kalibrasyon otomasyonundan istenen getirileri çalışmada görmek mümkündür. Diğer yandan IVI tabanlı yazılım altyapısı ile otomasyonun optimizasyonu ve idame ettirilebilirliği söz konusu olmaktadır.

Kalibrasyon Süresi

Kalibrasyon süresinin en aza indirilmesi kalibrasyon otomasyonu sürecinin en önemli hedefidir. Çalışmanın örnek uygulamasında kalibrasyonu gerçekleştirilen cihaz, kalibrasyon laboratuvarlarında referans

standart olarak sıkça kullanılmaktadır. Kalibrasyon laboratuvarının referans cihaz eksikliği önemli bir sorun olacağı için cihazın kalibrasyon süresinin azaltılması amaçlanmıştır. Kalibrasyon süresinin azaltılması hedeflenirken, cihazın ölçüm hassasiyeti ve doğruluğundan herhangi bir taviz verilmemesi gereklidir.

8½ dijital DMM'ler en hassas ölçüm sonuçlarına ulaşabilmek için ölçme işlemindeki yanıt alma süreçlerini uzun tutmaktadır. Yanıt alma sürecine yapılacak bir müdahale, ölçüm sonuçlarının hassasiyetinin düşmesine neden olduğundan, ölçüm cevabı bekleme sürecinin dikkatle oluşturulması gerekmektedir. Uygulamada kullanılan cihazlarla veri ve komut iletiminin sağlanması için oluşturulan sanal yolların, veri yolu cevap bekleme süreleri yazılım destekli olarak kontrol edilmiş, gereksiz bekleme süreleri ile oluşan zaman kayıpları en aza indirilmiştir.

Kalibrasyon süresine etki eden diğer parametreler de ele alındığında kullanılacak olan bazı temel çözüm önerilerini gösteren kod dizini Tablo 1'de verilmiştir.

Tablo 1. Zaman Sorunlarına Kod Çözümleri

Durum	Kod
Cihaza bir işlemi başlattıktan (OPER) sonra cihazdan işlemin bittiğine dair durum beklenmesi (*OPC?)	<pre>fluke5720.WriteString("OPER", true); fluke5720.WriteString("*OPC?", true); fluke5720.ReadString();</pre>
Sistemde oluşturulan sanal arayüz yollarına cevap bekleme süresi (tarama süresi) tanımlanması	<pre>hp_3458.IO.Timeout = 50000; fluke5720.IO.Timeout = 50000;</pre>
Sisteme gecikme (bekleme) verilmesi	<pre>System.Threading.Thread.Sleep(1000);</pre>

Sistemin zaman performansında etkin rol oynayan komutlardan biri “*OPC?” komutudur. Cihaza “*OPC?” komutu gönderildiğinde cihaz işlemin bitmesini takip etmekte, işlemin bitmesine müteakip “1” cevabı vermektedir. Cihazların işlemlerini bitirip bitirmediğini kontrol eden “*OPC?” komutunun yazılım alt yapısına eklenmesi ile sistemde oluşan performans artırımını Tablo 2’de verilmiştir.

Tablo 2. “*OPC?” komutunun Sistem Performansına Etkisi

Fonksiyonlar	Kontrolsüz (dakika)	*OPC? (dakika)	Kazanç (%)
1. Ofset	21	17	19,0
2. DC Gerilim	59	51	13,6
3. AC Gerilim	150	132	12,0
4. DC Akım	47	41	12,8
5. AC Akım	55	43	21,8
6. Direnç	63	55	12,7
7. Frekans	34	34	0,0

Kontrol birimi ile cihazlar arasında veri alışverişini sağlayan arayüz donanım yollarının IVI-VISA tabanlı sanal yollarının cevap tarama sürelerinin işlemin durumuna göre düzenlenmesi sistem zaman performansının artımında önemli bir parametre olmaktadır. Agilent 3458A DMM’in farklı fonksiyonlar için farklı sonuç verme süreleri bulunmaktadır. Bu durumda, her bir ölçüm fonksiyonu için, hatta fonksiyonlardaki alt ölçümler (dirençte 2 veya 4 kablolu, AC gerilimde senkron veya asenkron gibi) için, sanal arayüzlerin cevap tarama süreleri farklı olmalıdır. IVI-VISA tabanlı oluşturulan kodlarda tarama süresi (timeout) bilgisi ile cihazların arayüz cevap tarama süreleri verilebilmektedir. Her bir fonksiyon için algoritmaların işlenmesine başlamadan tarama süresi değerlerinin belirlenmesi sistemdeki gereksiz bekleme süreleri en aza indirmiş ve sistemin performansını arttırmıştır. Tablo 3’te eski yöntemle tek bir cevap tarama süresinin verilmesi ile IVI-VISA altyapısını kullanarak ayrı ayrı verilmesi durumunda oluşan değişim gösterilmiştir.

Tablo 3. Tarama Süresinin (timeout) Sistem Performansına Etkisi

Fonksiyonlar	Sabit tarama süresi (dakika)	Değişken tarama süresi (dakika)	Kazanç (%)
1. Ofset	17	13	23,5
2. DC Gerilim	51	47	7,8
3. AC Gerilim	132	116	12,1
4. DC Akım	41	36	12,2
5. AC Akım	43	38	11,6
6. Direnç	55	49	10,9
7. Frekans	34	28	17,6

Yazılım tabanlı kod düzenlemeleri ile sistem performansı artmakla birlikte asıl performans IVI standartlarının getirdiği durum önbellekleme ile sağlanmaktadır. Durum önbellekleme, kontrol birimi ile cihaz arasındaki gereksiz I/O işlemleri en aza indirilerek sistem performansının

artırılmasıdır. Sistemin performans artırımında, kanal güvenliği sağlanarak bir anda birden çok kanalın iş yapabilmesi durumu da söz konusu olmakta böylece aynı anda birden fazla cihaza hakim olunabilmektedir.

Bu çalışmada ele alınan zaman sürecini optimum olarak yönetmeye yönelik önemli birkaç parametre ile sistemin performansının ne kadar değiştiği gözlemlenmiştir. Kalibrasyon sürecini en optimum şekilde en az sürede bitirmeyi hedefleyen çalışmalarda yazılım tabanlı düzenlemeler büyük rol oynamaktadır. Kalibrasyon fonksiyonlarının işlem süresi performansına yazılım tabanında yapılan tüm düzenleme ve kontrollerin etkisi Tablo 4'te verilmiştir.

Tablo 4. Zaman Kontrolünün Sistem Performansına Etkisi

Fonksiyonlar	Zaman Kontrolü Yapılmamış (Dakika)	Zaman Kontrolü Yapılmış (Dakika)	Kazanç (%)
1. Ofset	21	12	42,9
2. DC Gerilim	59	42	28,8
3. AC Gerilim	150	105	30,0
4. DC Akım	47	30	36,2
5. AC Akım	55	35	36,4
6. Direnç	63	45	28,6
7. Frekans	34	20	41,2

Diğer yandan kalibrasyon işleminin geneline bakıldığında, çalışmanın en önemli çıktısı olan manuel ve otomatik kalibrasyon arasındaki zaman farkı oluşturulan sistemin performansını verecektir. Manuel kalibrasyon ile kalibrasyon otomasyonu uygulaması karşılaştırıldığında Tablo 5'teki sonuçlar elde edilmektedir.

Tablo 5. Manuel ve Otomatik Kalibrasyonun Performans Karşılaştırması

Fonksiyonlar	Manuel Kalibrasyon (Dakika)	Otomatik Kalibrasyon (Dakika)	Kazanç (%)
1. Ofset	30	12	60,0
2. DC Gerilim	220	42	80,9
3. AC Gerilim	340	105	69,1
4. DC Akım	105	30	71,4
5. AC Akım	90	35	61,1
6. Direnç	90	45	50,0
7. Frekans	60	20	66,7

Manuel kalibrasyonda harcanan süre boyunca teknisyenin işlem başında hazır olması gerektiği gerçeği, kalibrasyon otomasyonunun getirdiği işgücü kazancını daha açık ve net bir şekilde ortaya koymaktadır.

Kalibrasyon İşlemleri

Cihazların hassasiyeti arttıkça kalibrasyon süreçleri de uzamaktadır. Normal koşullarda 8½ dijital bir DMM'in kalibrasyonu da oldukça zor ve uzun bir süreçtir. Tekrarlı ölçüm işlemleri ise süreci içinden çıkılmaz hale getirmektedir. Manuel kalibrasyon işleminde yaklaşık 200 adet ölçüm, 400 adet matematiksel işlem, 3000 adet tuşa basma işlemi yapılmakta iken kalibrasyon otomasyonu ile işlemlerin birkaç tuşa dokunma ve bağlantı kurma rahatlığına indirgenmesi oldukça önemlidir. Böylece insan gücü en aza indirilmektedir.

Yapılan işlemlerin çok az bir kısmının manuel olması ve bu işlemlerin yazılım ile kontrol edilmesi ile kullanıcı kaynaklı hatalar en aza indirilmektedir. Cihaz kontrol ve kablo kontrol fonksiyonları ile kullanıcının yaptığı bağlantılar kontrol edilmektedir.

Kalibrasyon süreci başlı başına bir ölçüm sürecidir. Ancak 8½ dijital bir DMM'in ekranında ölçüm sonucu olarak virgülden sonra sekiz basamak bulunmaktadır. Hassasiyetin yüksek olması önemlidir. Ancak gözle edinilen ölçümlerde ölçüm hassasiyetinin düşük, sonuçların hatalı olma olasılığı çok yüksektir. Diğer yandan ortalama, standart sapma gibi matematiksel hesaplarda çoklu sayıda örnek gerekmektedir. Kalibrasyon otomasyonu ile kullanıcı istediği örnek sayısını girmekte ve ölçüm sonuçlarının edinimi sürecinde olaya müdahil olmamaktadır. Böylece sistem gözlem hatalarını ortadan kaldırmaktadır.

Ölçüm sonuçlarının elde edilmesi ile matematiksel hesap sürecine geçilmektedir. Yine matematiksel hesapların bir yazılım desteği alınmadan yapılması kalibrasyon verilerini yorumlama sürecini oldukça yavaşlatmaktadır. Bu durumda devreye yazılım tabanlı matematiksel fonksiyonlar girmektedir. Oluşturulan matematiksel fonksiyonlar kullanılarak, alınan çoklu sayıda veri ile standart sapma, ortalama vs. işlemler yaptırılmış, böylece matematiksel işlem hataları en aza indirilmiştir.

Kalibrasyon işleminin çıktısı olan kalibrasyon sertifikası için ISO 17025 standardında istenilen belirsizlik, hata vs. hesapları doğru ve güvenli bir şekilde yapmıştır. Böylece, bilimsel ve akredite bir kalibrasyon işlemi

gerçekleştirilmiş ve standartlara uygun olarak hazırlanan sertifika kullanıcıya sunulmuştur.

Yazılım Sürecinin İdame Ettirilebilirliği

Elde edilen bütün kazanımlarda yazılım desteğinin önemli bir etkisi vardır. Bazı parametrelerin belirlenmesinde ise yalnızca yazılım ve komut standardı seçimi etkilidir. Kullanılan IVI standartlarının getirileri ile sistemin performansı yükseltilmiştir. Özellikle IVI standartlarının sunduğu cihazların değiştirilebilirliği, cihaz benzetimi ve durum önbellekleme özellikleri ile otomasyon sistemi daha etkin hale getirilmeye çalışılmıştır.

Bir otomasyon sisteminden beklenen önemli parametrelerden birisi de uzun süreçte bakım ve onarım maliyetinin düşük olmasıdır. Genellikle yapılan çalışmalarda otomasyon sistemlerinin idamesini cihazlar ve diğer donanımsal bileşenler üzerinden sağlanması hedeflenmiştir. Yazılım tabanına yönelik bir hedef ise, yazılım üzerinde yapılacak değişikliklerin yazılım bilgisi gerektirmesinden dolayı talep görmemiştir. Sistem kullanıcıları ile program geliştiricilerinin farklı personeller olması, geliştirilen programların kendine has dil özellikleri olması, sistem kullanıcıları yazılım tabanından uzak tutmuştur.

Ancak, yeni nesil yazılım teknolojileri ile geliştirilen programların uzun süreçte idamesi oldukça kolay hale gelmiştir. Nesneye yönelik programlama dillerinin sınıf felsefesi ile hazırlanan programlara, IVI standartlarının avantajları da eklendiğinde karşımıza oldukça verimli sonuçlar çıkmaktadır. Sistemin yazılım tabanlı uzun süreç idame ettirilebilirlik avantajlarından bazıları aşağıda sıralanmıştır.

Uzun süreçte bir cihazın arıza ya da bakım durumuna girerek devre dışı kalması ya da cihazın kullanım süresinin dolması ile daha yeni teknoloji bir cihazla değiştirilmesi durumlarında sistem yazılım tabanında yapılacak küçük değişimlerle çalışmasına devam edebilmektedir. Sistemde kullanılan sinyal üreticinin arızalandığı ve kalibrasyon laboratuvarında eşdeğer ancak farklı bir firmanın IVI Fgen sınıfını destekleyen bir sinyal üretici olduğu varsayılın. Bu durumda sistemde yapılacak tek şey, arızalı cihazı yerine diğer cihazın kullanılması olacaktır. Sisteme yeni cihazı tanıtmak ve adres yolunu cihaza belirtmek yeterli olacaktır.

Kalibrasyonu yapılan cihaz yerine başka bir cihazın kalibrasyonun yapılması durumunda, standartlara göre düzenlenmiş ölçüm noktaları ve parametreleri değiştirilmeden yalnızca yeni cihazın bilgileri girilerek sistem adapte edilebilir. Çoğu kez benzer fonksiyonları işleten ancak farklı cihazlar

kalibrasyon laboratuvarına gelmekte ve her biri için aynı kalibrasyon prosedürleri farklı programlarda işlenmektedir. Ancak geliştirilen uygulama ile Agilent 3458A yerine, bir başka DMM'in gelmesi durumunda yazılımda yapılacak tanımlama işlemleri ve cihaza özgü komutların eklenmesi ile sistemi yeni cihaza adapte edebilmektedir.

Diğer cihazların sisteme adaptasyonunda, standartlara uygun ölçüm noktalarının seçimi ve IVI tabanlı sanal yolları oluşturan IVI VISA sürücüsü önemli yol oynamaktadır. Standartlarda belirlenen ölçüm noktaları her bir DMM için yaklaşık aynıdır. Böylece değiştirilmesine gerek kalmamaktadır. Diğer yandan gönderilecek komutlar da IVI standartları için aynı olduğundan, değişiklik yapmadan sistem idame ettirilebilmektedir.

Sistem yazılımında görülen değişimler uzun süreç bakım onarım durumlarında kolaylıkla yapılabilmektedir. Nesneye yönelik programlama dillerinde var olan sınıf oluşturma ve sınıflara yeni nesnelere ekleme özellikleri uygulamanın yazılım tabanında mevcuttur. Oluşturulan IVI nesnelere parametreleri kolaylıkla değiştirilebilmekte yeni IVI nesnelere tanımlanabilmektedir.

Sunulan bulgular ile kalibrasyon otomasyonu sisteminin optimizasyonu, idame ettirilebilirliği ve diğer cihazlara adaptasyonu açıklanmıştır. Böylece yapılan çalışmanın ileriye yönelik geliştirmelere açık olduğu anlaşılmaktadır.

Sonuç ve Tartışma

Kalibrasyon, ölçüm cihazlarının doğruluğunun belirlenmesinde ve izlenebilirliğinin sağlanmasında oynadığı rol ile teknolojinin vazgeçilmez unsurlarından biridir. Kalibrasyon, ölçüm güvenilirliğini sağlamakta gerekli olmakla birlikte, cihaz kullanıcıları cihazın kalibrasyonda geçirdiği süreyi verimsiz süre olarak değerlendirmekte ve bu süreyi azaltmak istemektedirler. Diğer yandan kalibrasyon laboratuvarları da işgücü bakımından kalibrasyon süresini azaltmaya çalışmaktadır.

Bu çalışmada Agilent 3458A model 8½ DMM'in kalibrasyon sürecinin daha hızlı ve yüksek doğruluklu gerçekleşmesi için bir kalibrasyon otomasyonu sistemi tasarlanmıştır. Uygulama, bir kalibrasyon otomasyonundan istenilen doğruluk ve hız kazanımlarını sağlamanın yanı sıra, kullanılan yazılım altyapısı ile daha esnek ve idame ettirilebilir bir yapı sunmaktadır.

Tasarlanan otomasyon sisteminde, cihazların yönetiminde kullanılan yeni nesil IVI standartlarının sunduğu faydalardan yararlanılmıştır. IVI standartları ile gelen durum önbellekleme özelliği ile gereksiz bekleme en aza indirilmiştir. IVI'nin önerdiği diğer bir özellik olan cihazların değiştirilebilirliği felsefesi ile kalibrasyon otomasyonunun uzun süreçte bakım, onarım, güncelleme ve geliştirme faaliyetleri daha esnek hale getirilmiştir.

Kaynakça

- AFMETCAL. (2010). *T.O.33K8-4-551-1, Technical manual, Calibration procedure for digital voltmeter 3458A*, USA.
- Ahmad, S., Rustagi, V.K., Govil, A.K., Aggarwal, R., Pal, B. ve Kothari, P.C. (2007). *Automation of direct comparison technique and study of long term performance on the calibration. Asia-Pacific Microwave Conference*, Bangkok, Tayland.
- Bode, F. (2003). IVI comes of age: An overview of IVI specifications with current status. *Aerospace and Electronic Systems Magazine*, 18(6), 31-34.
- Cheij, D. (2001). Using interchangeable virtual instrument (IVI) drivers to increase test system performance. *Aerospace and Electronic Systems Magazine*, 16(7), 8-11.
- Chen S., Yuan X., Yu C., Zhao Q. (2009). *Development of the Automated impedance measurement system based on the slotted line. International Measuring Technology and Mechatronics Automation Conference*, Zhangjiajie, China.
- Creasman, S.M. ve Pheanis, D.C. (2006). *Automated system for calibration and validation of automotive test instrumentation. World Automation Congress, WAC '06*, Budapest.
- Drnovsek, J., Bojkovski, J., Pusnik, I. ve Tasic, T. (1998). *Automation of a calibration laboratory. Precision Electromagnetic Measurements Conference*, Washington, USA.
- Fertitta, K.G. (2010). *Understanding the benefits of IVI*. <http://www.ivifoundation.org/resources/default.aspx> adresinden alınmıştır.
- Filipski, P.S. ve Rinfet, R. (2000). An automated AC-DC transfer calibration system. *IEEE Transactions on Instrumentation and Measurement*, 49(2), 279-284.
- Güven, B. ve Uzun, T. (2007). *Kontrol sistemlerinde kullanılan veri haberleşmesi teknolojileri. 1. Haberleşme Teknolojileri Ve Uygulamaları Sempozyumu*, İstanbul.
- ISO/IEC. (2005). *17025 Deney ve kalibrasyon laboratuvarlarının yeterliliği için genel şartlar standardı*.

- Kennedy, C.R. (2004). Software replacement of test module adapters for depot instrumentation due to obsolescence. *IEEE Aerospace and Electronic Systems Magazine*, 19(7), 25-30.
- Li N., Liu Q., Wang L. ve Liu X. (2008). *A novel approach for automation of precision calibration process. 4th International Information and Automation for Sustainability Conference*, Beijing, China.
- Lili, X. (2007). *A programming model for instrument calibration system. 8th International Electronic Measurement and Instruments Conference*, China.
- Lind, D., Haney, M. ve Healey, R. (2011). *Polymorphic analog instrument environment. The Systems Readiness Technology Conference*, Baltimore, USA.
- Niu S. ve Xu A. (2008). *A COM-based instrument interchangeable framework and its realization approach. The Systems Readiness Technology Conference*, Utah, USA.
- Randle, W.C. ve Kerth, N. (1978). Microprocessors in instrumentation. *Proceedings of the IEEE*, 66(2), 172-181.
- Yang S., Ling L. ve Kang G. (2010). *Research of an Automatic RF Devices Characteristic Test Bench for DC and Scattering Parameters. 2nd International Information Engineering and Computer Science Conference (ICIECS)*, Wuhan, China.