

Gömülü ve Gerçek Zamanlı Yazılımlarda Arayüz Giriş-Çıkış İzleyici

Murat SENEM¹, Ali KELEŞ²

Aselsan A.Ş. SST-GGZYTM, P.K. 1 06172, Yenimahalle, Ankara

¹ msenem@aselsan.com.tr

² akeles@aselsan.com.tr

Özet. ASELSAN Savunma Sistem Teknolojileri Sektör Başkanlığı Gömülü ve Gerçek Zamanlı Yazılım Tasarım Müdürlüğü bünyesinde geliştirilen atış kontrol yazılımları, zaman ve güvenlik kritik gerçek zamanlı gömülü yazılımlardır. Bu yazılımlar, ASELSAN'da geliştirilen silah sistemlerinde birçok algılayıcıdan gelen verileri işleyerek, atılan mühimmatın hedefi vurma olasılığını artırmak amacıyla en doğru zamanda en doğru atışı yapabilmek için gerekli işlemleri yerine getirmektedir. Yazılım, bu temel işlevlerini yerine getirirken birden fazla alt sistem ile haberleşmektedir. Atış kontrol yazılımı ile alt sistemlerin entegrasyonu aşamasında hatalar oluşmakta ve bu hataların kaynağının tespit edilmesi zaman almaktadır. Bununla beraber ekstra donanım (seri kanal kabloları, CAN-BUS adaptör vb. gibi) ve yazılım (seri kanal izleme programları, CAN-BUS Analizörler vb. gibi) araçları ihtiyacı bulunmaktadır. Ayrıca atış kontrol sistemlerinin doğası gereği sahada gerçekleşen entegrasyon ve test faaliyetlerinde tespit edilen problemlerin hızlı çözümü gerekmektedir. Yazılımın çevre birimleri ile olan haberleşmelerindeki giriş-çıkış yapan tüm verileri izlemek amacıyla geliştirilecek bir aracın, yazılımın hem çevre birimleri ile olan entegrasyonunda hem de arazi testlerinde büyük kolaylık sağlayacağı ve zaman kazandıracağı öngörülmüştür. Bu çalışmada, yazılımın kullandığı fiziksel haberleşme kanalları olan seri kanal, CAN-BUS, Paralel G/Ç hatlarındaki verilerin izlenmesini sağlayacak bileşenin geliştirilmesi ve bu bileşen ile elde edilen kazanımlar anlatılacaktır.

Anahtar Kelimeler: Atış Kontrol Yazılımı, Hata Ayıklama, Yazılım Entegrasyonu, Gerçek Zaman.

Interface I/O Monitoring in Embedded and Real Time Software

Murat SENEM¹, Ali KELEŞ²

Aselsan A.Ş. SST-GGZYTM, P.K. 1 06172, Yenimahalle, Ankara

¹ msenem@aselsan.com.tr

² akeles@aselsan.com.tr

Abstract. Fire control software developed in ASELSAN Defence Systems Technology Sector/Embedded and Real-Time Software Design Department, is time and security critical real-time embedded software. This software performs required processes to fire ammunition at the right time in order to increase the probability of hitting target by processing data coming from many sensors in the weapon systems developed in ASELSAN. The software communicates with many subsystems while performing these main functionalities. Errors may occur during the integration process between subsystems and fire control software, and it is a time-consuming process to determine the source of these errors. Also, additional equipment (such as serial channel cables, CAN-BUS adapter etc.) and software tools (such as serial channel monitoring programs, CAN-BUS analyzers, etc.) are required. In addition, rapid solution of the problems, which are identified in the field integration and testing activities that are nature of fire control systems, are required. It is estimated that a tool, which will be developed to monitor I/O data in communication of the software and system environment units, provides great simplicity and time saving both integration of the software with system environment units and field tests. In this paper, the benefits of this component and the development process of the component that provide the monitoring of data on serial channels, CAN-Bus and parallel I/O lines, which are the physical communication channels used by the software, will be described.

Keywords: Fire Control Software, Debugging, Software Integration, Real-Time.

1 Giriş

Atış kontrol sistemlerinde [1] atış kontrol yazılımları, sistemin etkin çalışması için kritik bir öneme sahiptir. Atış kontrol yazılımlarının temel görevleri arasında; birçok sensör ve eğleyicinin kontrolü, çeşitli kaynaklardan alınan hedeflerin yönetimi, isabetli vuruş için gerekli balistik hesapların yapılması, servo denetleyicileri vasıtasıyla silahın istenilen açılara yönlendirilmesi, atış yapabilmek için silahların kurulması ve ateşlenmesi yer almaktadır.

ASELSAN Savunma Sistem Teknolojileri (SST) Sektör Başkanlığı Gömülü ve Gerçek Zamanlı Yazılım Tasarım (GGZYT) Müdürlüğü bünyesinde çeşitli silah sistemi projeleri [2] için atış kontrol yazılımları yapılmaktadır. Bu yazılımlar gerçek zamanlı olup gömülü sistemlerde çalışmaktadır. Geliştirilen atış kontrol yazılımları birden fazla çevre birimi ile aynı anda haberleşme kurmaktadır. Fiziksel haberleşme kanalları ve haberleşme protokolleri birimden birime seri kanal, CAN-BUS veya Ethernet olarak değişkenlik gösterebilmektedir. Aynı zamanda paralel giriş-çıkış hatlarında okuma yazma fonksiyonları gerçekleştirilmektedir.

Geliştirilen atış kontrol yazılımlarının çevre birimleri ile entegrasyonu sistemlerde yapılmaktadır. Bununla beraber atış kontrol sistemlerinin doğası gereği bazı testler fabrikada değil sahada yapılmak zorundadır. Entegrasyon ve saha testleri sırasında oluşan hatalarda, çıkan problemlerde çevre birimi ile yazılım arasındaki haberleşmeler kontrol edilip yazılıma giren çıkan verinin doğruluğunun araştırılması gerekmektedir. Bu yazılımlar gömülü sistemlerde çalışan yazılımlar olduğu için hata ayıklama yöntemleri de oldukça zorludur. Giriş-çıkış veri kontrolü için her protokole özgü araçlara, kablolar, bağlantılara ve programlara ihtiyaç duyulmaktadır. Örneğin seri kanal kontrolü için ilgili seri kanala kablo ile bağlanıp seri kanal izleme programları ile bu kanal takip edilmelidir veya CAN-BUS hattı dinlenmek istiyorsa bu protokole uygun kablolar ile sistemin ilgili portuna bağlanıp bu protokolün veri akışını takip edebileceğimiz analiz araçları kullanmak gerekmektedir. Bu yöntemleri, sistemin çalışma esnasında uygulamak zaman maliyetli olup aynı zamanda ara kablolar, izleme programları gibi birden fazla dış unsur içerdiği için verimli olmamaktadır.

Yazılımın çevre birimleri ile entegrasyonu sırasında ve arazi testlerinde karşılaşılan hataların ve problemlerin tespit ve çözümünde daha hızlı ve verimli olabilmek amacıyla kablolar ve izleme programları olmadan yazılıma giriş-çıkış yapan verileri izleme ihtiyacı bulunmaktadır. Atış kontrol yazılımlarımızda kullanılmak üzere dış dünya ile TCP aracılığıyla haberleşen bir bileşen geliştirip bu ihtiyacı ortadan kaldırmayı hedefledik. Bu bileşen sistemin normal çalışma sekansını bozmadan çalışmalıdır. Ayrıca yazılımın gerçek zamanlı çalışma yapısını da bozmadan işlevini yerine getirmelidir.

Bu makalede, bileşenin geliştirilmesi aşamasından, getirdiği faydadan ve gelecekte bileşende geliştirilebilecek özelliklerden bahsedilecektir.

2 Benzer Çalışmalar

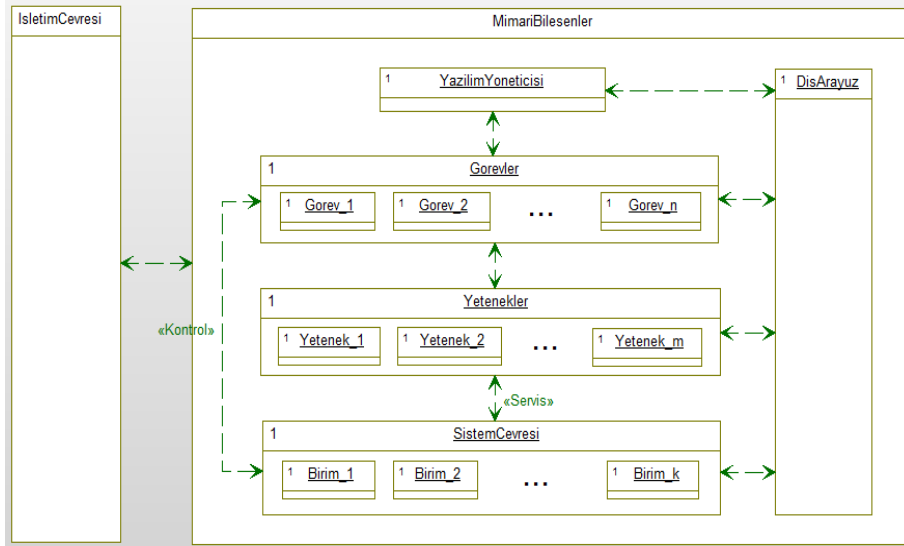
Literatürde yapılan çalışmalar, sistemleri kapalı sistem olarak değerlendirip giren çıkan verileri doğrulamaya yönelik çalışmalardır. Bu yöntem sistem içerisinde çalışan yazılımı da doğrulamaya yönelik bir uygulama olsa da çalışma anında giriş çıkış yapan verileri de gözlemlemenin gerektirdiği durumlar olmaktadır.

Watterson, Conal, and Donal Heffernan [3] gömülü sistemlerde çalışma zamanında yazılıma giriş çıkış yapan verilerin doğrulanması ve izlenmesi ile ilgili çalışma yapmışlardır. Yapılan bu çalışmada gömülü sistemlerde zorlu ve karmaşık olan doğrulama ve izleme yöntemlerinin basitleştirilmesinden bahsedilmiştir.

Chen, Xinming, Danai Chasaki, and Tilman Wolf [4] bir ağ sisteminde benzer bir çalışma yaparak karmaşık sistemlerde giriş çıkış izleyicilerin önemini göstermişlerdir. Bu çalışma ile bu tip sistemlerde dış dünya ile olan veri trafiğini izlemek için bir yöntem geliştirmenin ve bu yöntemin performans olarak nasıl etki edildiğini anlatılmıştır.

3 Yapılan Çalışma

ASELSAN SST GGZYTM bünyesinde atış kontrol yazılımlarında Şekil 1’de görülen silah sistemleri referans mimarisi [5] kullanılmaktadır. İşletim çevresi katmanı donanım ile yazılım arasındaki soyutlama katmanını temsil etmektedir. Sistem çevresi katmanı çevre birimleri ile haberleşmek için geliştirilen bileşenleri ifade etmektedir. Yetenekler katmanı atış kontrol yazılımlarındaki silah yönetimi, hedef yönetimi, platform yönetimi ve araç yönetimi gibi temel yeteneklerin bulunduğu katmandır. Görev seviyesi katmanı ise projeden projeye değişkenlik gösteren sistemi başlatma, atış, yönlendirme vb. görevlerin bulunduğu katmandır.



Şekil 1. Atış Kontrol Yazılımları Silah Sistemleri Referans Mimarisi

3.1 Geliştirme

Geliştirmiş olduğumuz bileşen (DeviceAnalyzer) yazılıma giriş-çıkış yapan verileri izlemeyi amaçlıyordu. Yazılıma giriş-çıkış yapan veriler aslında yazılımın üzerinde çalıştığı donanıma giriş-çıkış yapan veriler olduğu için öncelikle donanım soyutlama katmanını olan, Şekil 1’de görülen Silah Sistemleri Referans Mimarisine göre işletim çevresi katmanında değişiklik yapılması gerekiyordu. Yalnız bu katmanda yapılan değişikliklerin yazılımın gerçek zamanlı çalışma prensibini ve gerçek zamanlı çalışma performansını etkilememesi gerekiyordu. Bu sebeple donanım soyutlama katmanına birden fazla işletim sistemi ve işlemci kartı için seri kanal, CAN_BUS ve paralel IO hatlarını izlemek amacıyla eklentiler yapılmıştır.

Donanım soyutlama katmanında yaptığımız değişikliğin ardından dış dünya ile haberleşecek bileşeni geliştirmek gerekmektedir. Device Analyzer bileşeni Şekil 1’e göre yazılımın görevler seviyesinde geliştirilen bir parçasıdır ve mimari tasarımında bir değişikliğe ihtiyaç bulunmamaktadır. Bu bileşen, dış dünya ile TCP aracılığıyla haberleşecek şekilde geliştirildi. Bileşen bir sunucu olarak TCP bağlantısı açıyor ve bağlanan istemciye veri gönderimini gerçekleştiriyor. Bununla beraber bileşen, tüm haberleşme protokollerinin bütün kanallarını TCP’den yollamak yerine, sadece kullanıcı tarafından izlenmek istenen protokolün ilgili hattının verisini yollamak üzere tasarlandı. Örneğin; 3 numaralı seri kanal dinlenilmek istendiğinde, bileşen kendisinde gelen komut ile beraber bu kanaldaki giriş-çıkış verilerini TCP aracılığıyla dış dünyaya aktarma görevini yerine getiriyor. Bileşen, ayrıca dış dünyaya TCP mesajlarını sürekli olarak değil, bir kuyrukta tutarak belirlenen periyot ile toplu olarak göndermektedir.

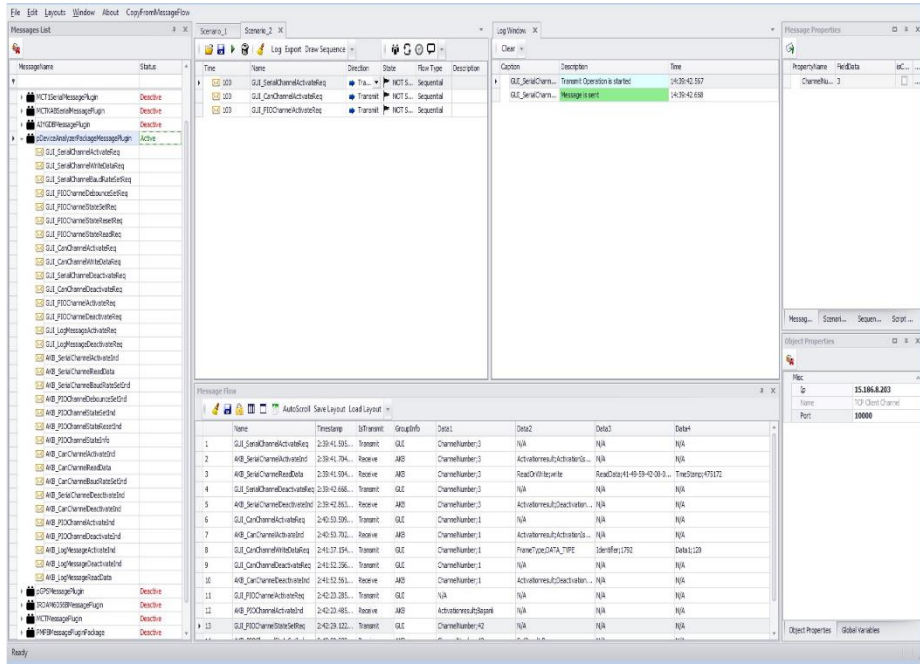
Bileşen, atış kontrol yazılımı giriş-çıkış hatları verilerini izlemek amacıyla geliştirilmeye başlansa da yazılımın çevre bileşenleri ile olan haberleşmelerini test etmek amacıyla başka özellikler de eklenmiştir. Bu özelliklerden ilki bileşen aracılığıyla herhangi bir haberleşme protokolünün istenilen kanalına veri yazabilme özelliğidir. Örneğin; CAN-BUS 4 numaralı kanaldan haberleşme yapılan bir çevre birimine kullanıcının istediği bir komutu gönderebilir. Bir diğer eklenen özellik de bileşenin, atış kontrol yazılımına herhangi bir çevre biriminden veri geliyormuş gibi davranıyor olmasıdır. Örneğin; seri kanal protokolü ile haberleşme kurduğumuz bir çevre bileşenin yerine bu bileşen kullanılarak çevre birimi arayüzündeki tüm mesajlar çevre biriminden geliyormuş gibi yazılıma gönderilebilir. Bu sayede yazılım geliştirme aşamalarında sisteme ve çevre birimine ihtiyaç duymadan yani donanım ihtiyacı olmadan yazılımın istenilen dış arayüzü test edilebilmektedir.

3.2 Kullanıcı Arayüz Programı

Bileşen ile haberleşecek kullanıcı arayüz programı ASELSAN bünyesinde daha önceden geliştirilen Köstebek adlı bir programa eklenti yapılarak geliştirildi. Bu program ve eklenti ile ilgili örnek bir ekran görüntüsü Şekil 2’de gösterilmiştir. Geliştirilen bu arayüz yazılımı eklentisi bileşene ilgili komutları gönderecek ve gelen verileri yazılıma giren ve yazılımdan çıkan veri olarak gruplayacak bir yapıda tasarlandı. Bununla beraber atış kontrol yazılımı verileri periyodik olarak göndereceği için gelen mesajda atış kontrol yazılımının gönderdiği ve aldığı birden fazla mesaj bulunacaktır. Bu me-

sajlar da atış kontrol yazılımı ve kullanıcı arayüz yazılımı arasında belirlenen arayüz kontrol mesajlarına göre ayrıştırılıp kullanıcıya gösterilmelidir.

Kullanıcı atış kontrol yazılımının IP adresini ve ilgili port numarasını girerek bileşen ile bağlantı kurabilecektir. Kullanıcı, izlemek istediği kanal ile ilgili komutları kullanıcı arayüz programından göndererek atış kontrol yazılımının o kanalındaki haberleşme trafiğini izleyebilecektir. Ayrıca atış kontrol yazılımının haberleştiği çevre birimine de ilgili komut yollanarak çevre biriminde istenilen bir senaryo da test edilebilir. Bunların yanında kullanıcı arayüz programında, atış kontrol yazılımının haberleştiği çevre birimleri yerine geçerek mesaj gönderme özelliği bulunduğu için genel amaçlı bir simülör gibi de düşünülebilir.



Şekil 2. Kullanıcı Arayüz Programı Örnek Ekran Görüntüsü

3.3 Performans Etkisi ve Kullanım Faydaları

ASELSAN SST GGZYTMM bünyesinde geliştirilen atış kontrol yazılımları gerçek zamanlı, zaman ve güvenlik kritik yazılımlardır. Geliştirilen bileşenin yazılımın gerçek zamanlılık özelliğini ve yazılımın çalıştığı sistemin normal çalışmasını etkilememesi gerekmektedir. Geliştirilen atış kontrol yazılımları gerçek zamanlı vxWorks işletim sisteminde çalışmaktadır. Performans etkisini görmek amacıyla DeviceAnalyzer bileşenin çalışması esnasında, bileşen ve kullanıcı arayüz programı arasında veri akışı varken, WindRiver firmasının Workbench [6] adlı aracı ile ölçüm alınarak inceleme yapılmıştır. Workbench’de bulunan “System Viewer Configuration” özelliği ile çalışma anında kayıt alınıp bu kayıt incelenerek bileşenin, atış kontrol yazılımı üzerindeki performansa olan etkisi araştırılmıştır.

İncelenen kayıttan Device Analyzer için geliştirilen iş parçacıklarının yazılımda çalıştığı süre 70 mikro saniye olarak belirlenmiştir. Bileşenin en düşük öncelikli görev ve sahip olması ve işlemci yükünün yüzde birini dahi meşgul etmiyor olması sebebiyle bu sürenin yazılımın gerçek zamanlı çalışma performansına olumsuz bir etkisi olmadığı değerlendirilmiştir. Device Analyzer kullanımı sayesinde yazılım ve çevre birimleri entegrasyonu aşamasında karşılaşılan problemlerin çözümünde hızlı sonuç almamız mümkün hale gelmiştir. Bununla beraber daha önceden kullanmak zorunda kalınan ekstra donanım (seri kanal kabloları, CAN-BUS adaptörleri) ve yazılım (seri kanal monitör programları, CAN-BUS analiz programları) araçları ihtiyacını ortadan kaldırarak entegrasyon aşamasında kolaylık sağlamıştır. Ayrıca bileşen aracılığıyla çevre birimine mesaj gönderme özelliği kullanılarak çevre birimi ile ilgili var olan bir hata ya da atış kontrol yazılımından kaynaklı bir hatanın bulunması daha kolay olmuştur. Son olarak da çevre birimi yokken atış kontrol yazılımına çevre biriminden mesaj geliyormuş gibi simüle etme olanağı sağlaması sayesinde yazılım geliştirme aşamalarında sistem ve çevre birimleri olmadan yazılımımızı test etmemizin önünü açmıştır. Böylece yazılımın sisteme ve çevre birimlerine entegrasyonu esnasında minimum hata ve zaman kaybı yaşamamızı sağlamıştır.

4 Değerlendirme

ASELSAN SST GGZYTMM bünyesinde ASELSAN'da yapılmakta olan silah sistemleri projeleri için geliştirilen atış kontrol yazılımları gömülü olarak çalışmakta olup birden fazla çevre birimi ile haberleşme kurmaktadır. Bu çevre birimleri ile olan haberleşmelerinde oluşan problemlerin tespiti zor olmaktadır. Ayrıca gömülü sistemler için geliştirilen yazılımlar oldukları için hata ayıklama yöntemleri de çok yaygın kullanılamamaktadır.

Bu sorunları aşmak ve hatayı belirlerken zaman kazanmak amacıyla Device Analyzer adlı bir bileşen geliştirdik. Sistem entegrasyon aşamasında yazılımın bir birimle entegrasyonu sırasında çıkan bir hatayı belirleyebilmek için kullanılan kablolar, analiz programları ile hatanın belirlenme süresi saatler hatta günler mertebesindeyken Device Analyzer bileşeni ile hata dakikalar içerisinde belirlenebilmektedir. Bundan dolayı bileşenin kullanımının bize zaman kazandırdığı, sağladığı alt yapı ile sistemdeki yazılım entegrasyon aşamasında ve arazi testlerinde büyük kolaylık sağladığı görüldü.

Bileşenin, getirdiği kolaylıklar ile beraber geliştirilebilecek bazı özellikleri bulunmaktadır. Kullanıcı arayüz programında mevcut durumda gelen mesajlar byte byte ayrıştırılarak gösteriliyor ve gelen mesajları anlamlandırma görevi kullanıcıya ait oluyor. Atış kontrol yazılımı ile dinlenmek istenen çevre birimi arasındaki arayüz kontrol mesajlarının kullanıcı arayüzü tarafında bilinmesini sağlayacak bir alt yapı ile kullanım kolaylığı sağlanabilir ve bu sayede kullanıcı mesajları bire bir okuyabilecektir ve mesajlar kullanıcı tarafından direk olarak anlamlı bir şekilde görülebilecektir.

Kaynaklar

1. https://en.wikipedia.org/wiki/Fire-control_system, son erişim tarihi 2017/05/24
2. http://www.aselsan.com.tr/en-us/press-room/Brochures/Command-Control-Comm-Computer-Systems/FIRE_CONTROL_SYSTEMS_ENG.pdf, son erişim tarihi 2017/05/24
3. Watterson, Conal, and Donal Heffernan. "Runtime verification and monitoring of embedded systems." *IET software* 1.5 (2007): 172-179.
4. Chen, Xinming, Danai Chasaki, and Tilman Wolf. "External monitoring of highly parallel network processors." *High Performance Switching and Routing (HPSR), 2013 IEEE 14th International Conference on*. IEEE, 2013.
5. Kahraman, E., İpek, T., İyidir, B., Bazlamaççı, C.F., Bilgen, S.: Bileşen Tabanlı Yazılım Ürün Hattı Geliştirmeye Yönelik Alan Mühendisliği Çalışmaları. In: UYMS'09, pp. 283–287 (2009)
6. https://www.windriver.com/products/product-notes/PN_WB_1110.pdf, son erişim tarihi 2017/05/24