

nos3\fs\apps\to\fs\examples\multi

To_custom.c Dosyası

Bu kod, telemetri paketlerini UDP ve RS422 üzerinden güvenli ve hatasız bir şekilde göndermek, gönderim sırasında hataları kontrol etmek, route'ları yönetmek ve TO uygulamasının telemetri çıkışını başlatıp durdurur.

Veri Akışı

1. Başlatma / Init
TO_CustomInit() fonksiyonu çağrıldığında UDP soketi oluşturulur, kritik mesaj ID'leri belirlenir ve route'lar tanımlanır; Route 0 UDP için, Route 1 RS422 seri port için aktif hâle getirilir ve Route 0 CF kanalına bağlanır.
2. Uygulama Komutlarının İşlenmesi
TO_CustomAppCmds() fonksiyonu ile gelen komutlar alınır, komut kodu kontrol edilir ve özel komutlar işlenir; örneğin test paket gönderimi yapılır, tanımlanmamış komutlar hata ile sonuçlanır.
3. Telemetri Verisinin İşlenmesi ve Gönderimi
TO_CustomProcessData() fonksiyonunda telemetri verileri buffer'a kopyalanır ve route ID'ye göre veri gönderimi gerçekleştirilir; Route 0 için UDP üzerinden, Route 1 için RS422 seri port üzerinden gönderim yapılır.
4. Gönderim Boyutu Kontrolü
TO_CustomProcessSizeSent() fonksiyonu ile gönderilen veri boyutu kontrol edilir ve eğer veri eksik veya hatalı ise hata mesajı loglanır ve ilgili route devre dışı bırakılır.
5. Telemetri Çıkışının Açılması
TO_CustomEnableOutputCmd() fonksiyonu çağrıldığında telemetri çıkışı etkinleştirilir, hedef IP/port veya seri port ayarları yapılır, route'lar konfigüre edilir ve aktif hâle getirilir.
6. Telemetri Çıkışının Kapatılması / Cleanup
TO_CustomDisableOutputCmd() fonksiyonu ile çıkış devre dışı bırakılır ve TO_CustomCleanup() çağrılarak UDP soketi kapatılır, kaynaklar temizlenir.

nos3\fs\apps\to\fs\examples\multi_tf

To_custom.c Dosyası

Bu kod, cFS TO uygulamasının özel bir versiyonu olup, telemetri verilerini iki farklı fiziksel yolla (UDP ve RS422) CCSDS uyumlu frame formatında göndermek için tasarlanmıştır.

Aynı zamanda bağlantı hatalarını tespit edip ilgili kanalı kapatma gibi güvenlik mekanizmaları içeriyor.

Veri Akışı

1. Komut ile çıkış başlatılır (TO_CustomEnableOutputCmd).

Telemetri çıkışı etkinleştirilir, hedef IP/port veya seri port tanımlanır ve kanallar gönderime hazır hale getirilir.

2. Telemetri paketleri TO_CustomProcessData üzerinden alınır.

Uygulama tarafından oluşturulan telemetri paketleri alınır ve frame içine eklenmek üzere işlem sırasına konulur.

3. Paketler CCSDS TM SDLP frame içine eklenir (TO_CustomProcessPacket).

Alınan paketler, TM SDLP kütüphaneleri kullanılarak CCSDS uyumlu frame'in veri alanına eklenir.

4. Frame tamamlanınca Idle Packet eklenir (TO_CustomFrameSend).

Idle Packet: Telemetri frame'inde gönderilecek gerçek veri olmadığında, frame'in boş kalan kısmını doldurmak için kullanılan sabit boyutlu ve genellikle pseudo-random veri içeren paket.

5. Frame, CADU formatında UDP veya RS422 ile gönderilir.

CADU: Fiziksel iletim için hazırlanmış, CCSDS transfer frame'ini OCF ve ASM ekleyerek paketleyen veri birimi.

ASM: CADU'nun başında yer alan ve alıcının senkronizasyon yapmasını sağlayan sabit bit dizisi.

6. Gönderim boyutu hatalıysa route devre dışı bırakılır (TO_CustomProcessSizeSent).

Frame gönderimi eksik veya hatalı ise ilgili iletişim kanalı kapatılır ve veri kaybı önlenir.

nos3\fsf\apps\to\fsf\examples\rs422

To_custom.c Dosyası

Bu kod, cFS TO uygulamasının özel bir versiyonu olup, telemetri verilerini RS422 seri portu üzerinden göndermek için tasarlanmıştır.

Aynı zamanda gönderim hatalarını kontrol ederek telemetri çıkışını devre dışı bırakma mekanizması içerir.

Veri Akışı

1. Çıkış Başlatma (TO_CustomEnableOutputCmd)

Yer istasyonundan veya uygulamadan gelen komutla telemetri çıkışı açılır, hangi seri port kullanılacağı belirlenir ve RS422 kanalı gönderime hazır hale getirilir.

2. Telemetri Paketlerinin Alınması (TO_CustomProcessData)

Uygulama tarafından üretilen telemetri verileri, TO uygulaması aracılığıyla seri port üzerinden gönderilmek üzere buffer'a yazılır.

3. Paket Gönderimi ve Kontrol

Gönderilen byte sayısı "isentsize" ile kontrol edilir ve log yazılır ve gönderim durur.

4. Çıkış Kapatma (TO_CustomDisableOutputCmd / TO_CustomCleanup)

Telemetri çıkışı, yer istasyonundan gelen komut veya uygulamanın kapanması durumunda devre dışı bırakılır; seri portun kapatılması işlemi ise bu dosyada yapılmaz, bunun yerine başka bir modül (CI) tarafından gerçekleştirilir.

nos3\fsw\apps\to\fsw\examples\udp

To_custom.c Dosyası

Bu kod, cFS TO uygulamasının özel bir versiyonu olup, telemetri verilerini UDP soketi üzerinden göndermek için tasarlanmıştır.

Aynı zamanda gönderim hatalarını kontrol ederek telemetri çıkışını devre dışı bırakma mekanizması içerir.

Veri Akışı

1. Çıkış Başlatma (TO_CustomEnableOutputCmd)

Yer istasyonu veya uygulamadan gelen komutla telemetri çıkışı açılır, hedef IP ve port belirlenir, UDP soketi oluşturulup yapılandırılır ve Route 0 üzerinden gönderime hazır hale getirilir.

2. Telemetri Paketlerinin Alınması (TO_CustomProcessData)

Uygulama tarafından oluşturulan telemetri paketleri TO uygulamasına iletilir ve IO_TransUdpSnd() fonksiyonu kullanılarak UDP soketi üzerinden hedef IP adresine gönderilir.

3. Gönderim Kontrolü

Gönderilen veri boyutu kontrol edilir; eğer hata oluşursa telemetri çıkışı devre dışı bırakılır ve log kaydı oluşturulur. Başarılı gönderimde ise paket sorunsuz bir şekilde hedef IP ve porta iletilir.

4. Çıkış Kapatma (TO_CustomDisableOutputCmd / TO_CustomCleanup)

Yer istasyonu komutuyla veya uygulama kapanınca UDP soketi kapatılır ve telemetri çıkışı devre dışı bırakılır.

nos3\fs\apps\to\fs\examples\udp_tf

To_custom.c Dosyası

Bu kod, cFS TO uygulamasının özel bir versiyonu olup, telemetri paketlerini Transfer Frame formatında hazırlayarak UDP soketi üzerinden hedef IP ve porta göndermek için tasarlanmıştır. Ayrıca gönderim hatalarını kontrol ederek telemetri çıkışını devre dışı bırakma mekanizmasını içerir.

Veri Akışı

1. Telemetri Çıkışının Başlatılması

Yer istasyonundan veya uygulamadan gelen “telemetri başlat” komutu ile telemetri çıkışı etkinleştirilir. Bu sırada hedef IP adresi ve port belirlenir, UDP soketi oluşturulur ve hedefe bağlanacak şekilde yapılandırılır. Ardından Route 0 konfigüre edilir ve telemetri gönderimine hazır hale gelir. Telemetri Paketlerinin Alınması (TO_CustomProcessData)

2. Telemetri Paketlerinin Alınması

Uygulama tarafından üretilen telemetri paketleri TO uygulamasına iletilir. Bu paketler Transfer Frame (TF) içine eklenir. TF’de boş alan kalırsa, kalan alan Idle Packet ile doldurularak frame’in sabit boyutu korunur.

3. Transfer Frame’in Hazırlanması

Transfer frame tamamlandıktan sonra isteğe bağlı olarak OCF (Operational Control Field) eklenebilir. Ayrıca güvenlik önlemleri uygulanır; örneğin, şifreleme veya diğer veri koruma işlemleri gerçekleştirilir. Son olarak frame, CADU (Channel Access Data Unit) formatına senkronize edilerek iletme hazır hale getirilir.

4. Telemetri Paketinin Gönderimi

Hazırlanan Transfer Frame (TF), UDP soketi üzerinden hedef IP ve port’a gönderilir; gönderim sırasında bir hata oluşursa route kapatılır ve durum log’a kaydedilir, kısmi gönderim gerçekleşmişse de route kapatılır ve durum kaydedilir.

5. Telemetri Çıkışının Durdurulması

Yer istasyonu komutu alındığında veya uygulama kapanırken, UDP soketi kapatılır ve telemetri çıkışı devre dışı bırakılır (usOutputEnabled = 0).

nos3\fsw\apps\ci\fsw\examples\multi

ci_custom.c Dosyası

Bu kod, Command Ingest (CI) uygulamasının özel bir katmanını tanımlar ve CI'nin RS422 seri port ile UDP socket üzerinden veri almasını sağlar. Kodun amacı, bu kanallardan gelen CCSDS formatındaki mesajları okuyup doğrulamak ve uygun şekilde işlemek veya yönlendirmektir.

Veri Akışı

1. Başlatma (CI_CustomInit)

Bu aşamada RS422 seri portu ve UDP socketi gerekli ayarlarla başlatılır ve açılır. Her iki haberleşme kanalı select setine eklenir, böylece aynı anda dinlenebilir hâle gelir. Ardından CI için özel bir child task olan CI_CustomMain oluşturulur ve veri akışını yönetecek görev başlatılmış olur.

Child Task: cFS uygulamasında ana task tarafından oluşturulan ve belirli işleri sürekli veya paralel olarak yerine getiren alt görevidir.

2. Child Task Çalışması (CI_CustomMain)

Oluşturulan child task sürekli döngü içerisinde çalışır. Burada select mekanizması kullanılarak hem RS422 hem UDP socketi aynı anda dinlenir. Eğer herhangi bir kanaldan veri gelirse, hangi kanalın aktif olduğuna bakılır ve uygun okuma fonksiyonu çağrılır.

3. Veri Okuma

Veri okuma aşamasında iki farklı senaryo vardır. Eğer veri RS422 üzerinden geldiyse, önce CCSDS başlık bilgisi okunur ve mesajın boyutu belirlenir. Ardından tüm mesaj alınarak işleme hazır hâle getirilir. Eğer veri UDP üzerinden geldiyse, mesaj doğrudan buffer'a aktarılır. Her iki durumda da elde edilen mesaj bir sonraki işleme aşamasına gönderilir.

4. Mesaj İşleme (CI_CustomProcessUpMsg)

Okunan mesajlar önce CCSDS checksum kontrolünden geçirilir. Eğer mesaj hatalıysa düşürülür ve hata raporlanır. Mesaj geçerli ise içerdiği MID değerine bakılır. Eğer mesaj bir Gate komutu içeriyorsa, CI_CustomGateCmds fonksiyonu çağrılarak hemen işlenir. Gate komutu değilse, mesaj yazılım bus'ı üzerinden ilgili diğer uygulamalara iletilir.

5. TO (Telemetry Output) Aktifleştirme (CI_CustomEnableTO)

CI uygulaması, gerek duyulduğunda Telemetry Output (TO) uygulamasını etkinleştirebilir. Bunun için TO'ya özel bir etkinleştirme mesajı gönderilir. Böylece sistem, RS422 üzerinden alınan dosya tanımlayıcısı ile TO'nun downlink telemetry çıkışı başlatılmış olur.

6. Temizlik (CI_CustomCleanup)

Uygulama kapatıldığında ya da görev sonlandırıldığında, RS422 portu kapatılır ve UDP soketi temiz bir şekilde kapatılır. Ayrıca select seti de boşaltılır. Bu sayede kaynak sızıntıları önlenir ve uygulama düzgün şekilde sonlandırılmış olur.

nos3\fs\apps\ci\fs\examples\multi_tf

ci_custom.c Dosyası

Bu kod, yer istasyonundan veya diğer kaynaklardan gelen komutları farklı iletişim kanalları üzerinden alıp doğrulayarak yazılım bus'ına aktarmak ve gerektiğinde doğrudan işlemek suretiyle uçuş yazılımının komut kontrolünü sağlamaktır.

Veri Akışı

1. Başlatma

Uygulama çalıştırıldığında öncelikle RS422 portu ve UDP soketleri yapılandırılır. Bu aşamada dinleme görevini üstlenecek child task oluşturularak sürekli çalışacak şekilde devreye alınır.

2. Dinleme

Child task, select fonksiyonu kullanılarak hem RS422 hem de UDP kanallarını aynı anda izler. Böylece hangi kanaldan veri gelirse gelsin, aynı görev üzerinden etkin biçimde işlenebilir.

3. Veri Alma

İlgili kanaldan bir mesaj geldiğinde, uygun okuma fonksiyonu devreye girer ve mesaj belleğe alınır.

4. Doğrulama

Alınan mesaj, öncelikle checksum kontrolünden geçirilir. Bu kontrol, mesajın bozulmadan ve eksiksiz ulaşmış olup olmadığını test eder. Kontrolü geçemeyen paketler işleme alınmaz ve hata olarak raporlanır.

5. İşleme

Doğrulamadan geçen paketler değerlendirilir. Eğer paket Gate komutu içeriyorsa, bu komut doğrudan CI uygulaması tarafından işlenir. Diğer komutlar ise yazılım bus'ına yönlendirilir ve ilgili uygulamalara iletilir.

6. TO Aktifleştirme

Bazı durumlarda Telemetry Output (TO) uygulamasının aktif hale getirilmesi gerekebilir. CI, bu durumda TO'yu devreye sokar ve downlink üzerinden telemetry akışının başlatılmasını sağlar.

7. Kapatma ve Temizlik

Uygulama sonlandırıldığında, açık olan RS422 portu, UDP soketleri ve kullanılan select yapıları kapatılır. Böylece tüm sistem kaynakları temizlenmiş olur.

nos3\fs\apps\ci\fs\examples\rs422

ci_custom.c Dosyası

Bu kodun amacı, RS422 seri port üzerinden alınan CCSDS mesajlarını güvenli ve doğru bir şekilde işleyip CI (Command Ingest) uygulamasına ileterek komut ve telemetri akışını sağlamaktır.

Veri Akışı

1. Başlatma ve Port Konfigürasyonu

RS422 portu CI_CustomInit() ile başlatılır; cihaz adı, baud rate, zaman aşımı ve minimum byte ayarları yapılır, TO mesajı hazırlanır ve child task (CI_CustomMain) oluşturulur.

Baud Rate: Bir iletişim hattında saniyede iletilen simge (signal) sayısını ifade eden bir terimdir. Basitçe, dijital veya seri iletişimde verinin hızını belirler.

2. Mesaj Alma (Uplink)

Child task porttan sürekli mesaj header'ı bekler. Header alındığında mesaj boyutu ve veri kısmı belirlenir; eksik veya buffer'ı aşan mesajlar düşürülür ve hata kaydı tutulur.

3. Mesajın Tamamını Okuma ve Doğrulama

Mesajın geri kalanı okunur ve CI_CustomProcessUpMsg() ile CCSDS checksum kontrolü yapılır. Hatalı mesajlar düşürülür, doğruluğu sağlanır.

4. Komutların İşlenmesi

Gate komutları CI_CustomGateCmds() ile anında işlenir, diğer mesajlar Software Bus üzerinden CI uygulamasına iletilir. Böylece komutlar hızlı işlenir, normal veri akışı kesintisiz devam eder.

5. Telemetry Output (TO) İşlemleri

CI_CustomEnableTO() ile TO'ya mesaj gönderilir ve telemetri akışı başlatılır; alınan veriler diğer sistemlere iletilir.

6. Temizlik ve Kapanış

CI_CustomCleanup() portu güvenli şekilde kapatır ve kaynakları serbest bırakır, veri iletimi güvenli sonlandırılır.

nos3\fs\apps\ci\fs\examples\udp

ci_custom.c Dosyası

Bu ci_custom.c dosyasının amacı, UDP soketi üzerinden alınan CCSDS mesajlarını işleyerek CI (Command Ingest) uygulamasına iletmek ve Gate komutlarını doğrudan yönetmektir. Dosya, ayrıca Telemetry Output (TO) mesajlarını gönderme ve telemetry akışını sağlama işlevlerini de kapsar.

Veri Akışı

1. **Başlatma ve UDP Konfigürasyonu**
CI_CustomInit() fonksiyonu, UDP soketini başlatır ve yapılandırır. IP adresi, port numarası ve zaman aşımı ayarlanır. Ardından child task (CI_CustomMain) oluşturulur ve veri alım işlemleri arka planda başlatılır.
2. **Mesaj Alma (Uplink)**
Child task çalışmaya başladığında UDP soketinden gelen verileri sürekli dinler. Veri geldiğinde buffer'a alınır ve mesaj boyutu hesaplanır.
3. **Mesajın İşlenmesi ve Gate Komutları**
Alınan mesaj Gate komutu ise CI_CustomGateCmds() fonksiyonu ile doğrudan işlenir. Diğer mesajlar Software Bus üzerinden CI uygulamasına iletilir.
4. **Telemetry Output (TO) İşlemleri**
CI_CustomEnableTO() fonksiyonu çağrıldığında TO mesajı hazırlanır ve iletilir, böylece telemetry akışı başlatılır.
5. **Temizlik ve Kapanış**
Görev sona erdiğinde veya sistem kapatıldığında CI_CustomCleanup() fonksiyonu çağrılır ve UDP soketi güvenli şekilde kapatılır.

nos3\fs\apps\ci\fs\examples\udp_tf

ci_custom.c Dosyası

Bu kodun amacı, UDP üzerinden alınan Telecommand Transfer Frame (TCTF) formatındaki Communication Link Data Units (CLTU) verilerini işleyerek CI (Command Ingest) uygulamasına iletmek, gerekli güvenlik işlemlerini yapmak ve Gate/TO mesajlarını yönetmektir. Dosya, CLTU verilerini alır, işleyip doğrular ve Software Bus (SB) üzerinden CI uygulamasına aktarır.

Veri Akışı

1. **Başlatma ve UDP Konfigürasyonu**
CI_CustomInit() fonksiyonu UDP socketini başlatır ve yapılandırır; IP adresi, port ve zaman aşımı ayarları yapılır. Ardından select yapısı başlatılır ve child task (CI_CustomMain) oluşturularak veri alımı arka planda başlatılır.
2. **Mesaj Alma ve CLTU Okuma**
Child task, select mekanizması ile UDP socketini dinler. Veri geldiğinde CI_CustomReadCltuSocket() çağrılır ve CLTU mesajları buffer'a alınır.
3. **Güvenlik ve İşleme**
Alınan CLTU mesajları Crypto_TC_ProcessSecurity fonksiyonu ile işlenir; bu aşama, TC çerçevelerinin güvenliğini ve bütünlüğünü sağlar. İşlem başarılı olursa mesaj, Software Bus'a iletmek üzere hazırlanır.
4. **Gate ve Telemetry Output (TO) İşlemleri**
Eğer mesaj Gate komutu içeriyorsa CI_CustomGateCmds() ile işlenir. CI_CustomEnableTO() çağrıldığında TO mesajı hazırlanır ve iletilir, telemetry akışı başlatılır.
5. **Temizlik ve Kapanış**
Görev sona erdiğinde CI_CustomCleanup() fonksiyonu çağrılır. UDP socketi ve select yapısı güvenli şekilde temizlenir, kaynaklar serbest bırakılır.