

**İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ**

**CANLI İÇERİK SAĞLAMA AMAÇLI, MPEG-2 SİSTEM UYUMLU BİR  
YÖNTEM TASARIMI VE UYGULAMASI**

**YÜKSEK LİSANS TEZİ**

**Müh. Engin DOĞAN**

**504031508**

**Tezin Enstitüye Verildiği Tarih : 22 Aralık 2006**

**Tezin Savunulduğu Tarih : 29 Ocak 2007**

**Tez Danışmanı : Yrd. Doç. Dr. Osman K. EROL**

**Diğer Jüri Üyeleri : Doç. Dr. Sema OKTUĞ**

**Doç. Dr. Serdar ÖZOĞUZ**

**OCAK 2007**

## **ÖNSÖZ**

Yüksek lisans eğitimim esnasında önüme çıkan en büyük engeli aşmamda bana yardımcı olan lisans danışmanım Prof. Dr. Ahmet Coşkun Sönmez'e ve danışmanlığını sadece tez çalışması ile sınırlamayan, her konuda bana yardımcı olmaya çalışan yüksek lisans danışmanım Yrd. Doç. Dr. Osman Kaan Erol'a teşekkürü bir borç bilirim.

**Aralık 2006**

**Engin Doğan**

## İÇİNDEKİLER

<b>ÖNSÖZ</b>	<b>ii</b>
<b>İÇİNDEKİLER</b>	<b>iii</b>
<b>KISALTMALAR</b>	<b>iv</b>
<b>TABLO LİSTESİ</b>	<b>v</b>
<b>ŞEKİL LİSTESİ</b>	<b>vi</b>
<b>SEMBOL LİSTESİ</b>	<b>vii</b>
<b>ÖZET</b>	<b>viii</b>
<b>SUMMARY</b>	<b>xi</b>
<b>1. GİRİŞ</b>	<b>1</b>
<b>2. TEMEL KAVRAMLAR</b>	<b>3</b>
2.1 Sayısal Yayıncılık Temelleri	3
2.1.1 Sıkıştırma	4
2.1.2 Çoklama	6
2.2 Sayısal Yayın Kaydediciler	9
2.2.1 Donanım	10
2.2.2 Yazılım	13
<b>3. ‘CANLI İÇERİK’ KAVRAMI VE TASARIMI</b>	<b>16</b>
3.1 PSI / SI Tabloları	17
3.2 Canlı İçerik Tablosu	21
3.2.1 Tablo Yapısı	22
3.2.2 Canlı İçerik Açıklayıcıları	24
<b>4. ‘CANLI İÇERİK’ UYGULAMASI</b>	<b>26</b>
4.1 SoftHE Yazılımı	26
4.2 SoftTV Yazılımı	29
4.3 Örnek Kullanım Senaryosu	32
<b>5. SONUÇLAR VE TARTIŞMA</b>	<b>36</b>
<b>KAYNAKLAR</b>	<b>38</b>
EK A. JPEG Kodlayıcı/Kodçözücü Blok Diyagramları	40
EK B. STi5100 Tümdevre Kılavuzu	41
EK C. PAT/PMT Bölüm Dizilimleri	42
EK D. SoftHE ve SoftTV Yazılımlarının Kaynak Kodları	44
<b>ÖZGEÇMİŞ</b>	<b>45</b>

## KISALTMALAR

<b>B</b>	: Bidirectionally Predicted Picture
<b>CAT</b>	: Conditional Access Table
<b>CRC</b>	: Cyclic Redundancy Check
<b>DCT</b>	: Discrete Cosine Transform
<b>DTS</b>	: Decoding Time Stamp
<b>DVB</b>	: Digital Video Broadcasting
<b>EIT</b>	: Event Information Table
<b>ES</b>	: Elementary Stream
<b>I</b>	: Intra Picture
<b>I<sup>2</sup>C</b>	: Inter-Integrated Circuits
<b>IF</b>	: Intermediate Frequency
<b>ISO</b>	: International Organisation for Standardisation
<b>JPEG</b>	: Joint Photographic Experts Group
<b>LCT</b>	: Live Content Table
<b>MPEG</b>	: Motion Pictures Experts Group
<b>MP3</b>	: MPEG-1 Level 3
<b>NIT</b>	: Network Information Table
<b>P</b>	: Predicted Picture
<b>PAT</b>	: Program Association Table
<b>PCR</b>	: Program Clock Reference
<b>PES</b>	: Packetized Elementary Stream
<b>PID</b>	: Packet Identifier
<b>PMT</b>	: Program Map Table
<b>PSI</b>	: Program Specific Information
<b>PTS</b>	: Presentation Time Stamp
<b>PUSI</b>	: Payload Unit Start Indicator
<b>PVR</b>	: Personal Video Recorder
<b>RISC</b>	: Reduced Instruction Set Computer
<b>RTOS</b>	: Real-Time Operating System
<b>RF</b>	: Radio Frequency
<b>SD</b>	: Standard Definition
<b>SDT</b>	: Service Description Table
<b>SI</b>	: Service Information
<b>SSI</b>	: Section Syntax Indicator
<b>TDT</b>	: Time and Date Table
<b>TS</b>	: Transport Stream
<b>UTC</b>	: Universal Time Co-ordinated
<b>UML</b>	: Unified Modeling Language
<b>VLSI</b>	: Very Large Scale Integration
<b>XPM</b>	: X PixMap

## TABLO LİSTESİ

	<b><u>Sayfa No</u></b>
<b>Tablo 2.1:</b> TS paketi başlık yapısı .....	8
<b>Tablo 3.1:</b> Tablo bölümlerinin başlığı.....	17
<b>Tablo 3.2:</b> PAT içeriği.....	18
<b>Tablo 3.3:</b> PMT içeriği .....	19
<b>Tablo 3.4:</b> TDT bölüm yapısı .....	20
<b>Tablo 3.5:</b> Genel açıklayıcı yapısı .....	21
<b>Tablo 3.6:</b> Canlı İçerik Tablosu bölüm yapısı.....	22
<b>Tablo 3.7:</b> Canlı İçerik nesne tiplerine atanmış açıklayıcı etiketleri.....	24
<b>Tablo C.1:</b> PAT bölüm dizilimi .....	42
<b>Tablo C.2:</b> PMT bölüm dizilimi.....	43

## ŞEKİL LİSTESİ

	<b><u>Sayfa No</u></b>
<b>Şekil 2.1:</b> Çerçevesel-arası kestirim.....	5
<b>Şekil 2.2:</b> ES çerçevelerinin paketlenmesi.....	6
<b>Şekil 2.3:</b> TS paketlerinin oluşumu .....	7
<b>Şekil 2.4:</b> Genel PVR yapısı .....	10
<b>Şekil 2.5:</b> Genel yazılım bileşeni katmanları .....	13
<b>Şekil 2.6:</b> Uygulama örnekleri.....	15
<b>Şekil 3.1:</b> Grafik nesnesi açılabilir yerleşimi.....	25
<b>Şekil 4.1:</b> SoftHE yazılımının girdi/çıktıları.....	27
<b>Şekil 4.2:</b> SoftHE sınıflarının UML diyagramları .....	27
<b>Şekil 4.3:</b> SoftTV yazılımının genel yapısı .....	29
<b>Şekil 4.4:</b> SoftTV sınıflarının UML diyagramı.....	30
<b>Şekil 4.5:</b> Canlı İçerik Tablosu'nun SoftHE ile hazırlanması .....	32
<b>Şekil 4.6:</b> Canlı İçerik Tablosu'nun TS'e gömülmesi.....	33
<b>Şekil 4.7:</b> Uzaktan kumanda grafik arayüzü .....	33
<b>Şekil 4.8:</b> TRT DVB-T test yayını .....	33
<b>Şekil 4.9:</b> SoftTV menüsü üzerinden kayıt işlemi.....	34
<b>Şekil 4.10:</b> SoftTV menüsü üzerinden oynatma işlemi.....	34
<b>Şekil 4.11:</b> Canlı İçerik sunumu.....	35
<b>Şekil A.1:</b> Basitleştirilmiş JPEG kodlayıcı blok diyagramı .....	40
<b>Şekil A.2:</b> Basitleştirilmiş JPEG kodçözücü blok diyagramı .....	40
<b>Şekil B.1:</b> STi5100 tümdevre kılavuzu.....	41

## SEMBOL LİSTESİ

<b>A</b>	: Audio
<b>V</b>	: Video
<b>D</b>	: Data
<b>b</b>	: Bit
<b>B</b>	: Bayt
<b>KB</b>	: Kilo Bayt
<b>GB</b>	: Giga Bayt
<b>Mb/s</b>	: Mega Bits per Second
<b>MHz</b>	: Mega Hertz
<b>  </b>	: VEYA

## ÖZET

Kişisel Görüntü Kaydedici (PVR) cihazlar, kullanıcılarına; tek-dokunuşla kaydetme, listeden kayıt, yetenekli modlarla (dondurma, ileri/geri alma) yeniden oynatma ve zaman-kaydırma gibi birçok avantajlar sunarlar. PVR kullanıcıları, genellikle; beğendikleri programları cihazın kayıt listesine eklerler ve boş vakitlerinde seyrederek. Ancak kullanıcılar, sabit diskte saklı yayınlarını izlerken, ayarlayıcı (tuner)'dan gelmekte olan canlı yayın içeriklerinden (son dakika haberleri, özel reklamlar, vb.) de haberdar olmak isteyebilirler. Bu tez çalışmasında, kullanıcıların; MPEG-2 Program Özel Bilgisi (PSI) tabloları yapısında hazırlanmış canlı yayın içeriklerine erişebilmelerini sağlayacak bir yöntem önerilmektedir.

Önerilen yöntemin gerçekleştirilmesi için, MPEG-2 TS uyumlu bir tablo tasarımı yapılmıştır. Bu tablo içinde taşınacak olan canlı içerik (LC) ayıklanıp işlenerek, kullanıcıya yeniden oynatma esnasında sunulacaktır. Ayrıca, önerilen yöntem; SoftHE ve SoftTV adlı iki program geliştirilerek, bir Linux sistemi üzerinde gerçekleştirilmiştir.

### ▪ Yeni Bir Tablo Tasarımı

PSI tabloları ile taşınan veriler, sadece, alıcıların yayın içindeki servislerin temel bileşenlerini ayrıştırıp çözebilmeleri için bilgi sağlamaktadır. Herhangi bir servis için LC verisi taşıyabilmek için DVB Servis Bilgisi (SI) tabloları gibi tablolara ihtiyaç vardır. Her iki PSI ve SI tabloları, TS paketleri içine taşınabilmeleri için sözdizimsel bölümlere ayrılmışlardır.

Canlı İçerik Tablosu (LCT) diye adlandırılan, canlı içerik verilerini taşıyabilmek amacı ile yeni bir tablo tasarlanmıştır. LC verileri, LC bölümlerinin gövdesindeki etiket-uzunluk-değer yapısında dizilmiş açıklayıcılar üzerinden taşınacaktır. LC verilerinin açıklayıcılar şeklinde tasarımı, ileride, kolayca yeni canlı içerik türleri tanımlamayı sağlar. Yeni bir canlı içerik tipine ihtiyaç duyulduğu vakit, ilişkili bir açıklayıcının bölüm gövdesine eklenmesi yeterli olacaktır.

Bu tez çalışması kapsamında, ilkin; etiketlerine göre ayırt edilebilen Başlık, Alt-başlık ve Metin Gövdesi gibi çeşitli metinsel LC açıklayıcıları tanımlanmıştır. Böylece yayıncılar, bu açıklayıcıları kullanarak metin tabanlı canlı içerikler



oluşturabilirler. Daha sonra, grafik ve görüntü gibi metinsel olmayan içerikler için, ek olarak yeni LC açıklayıcıları da tanımlanmıştır.

- Gerçekleme

Bu tezde öne sürülen fikir, bedava bir Linux sürümü çalıştıran kişisel bilgisayar ortamında gerçekleşmiştir. İki adet uygulama programı yazılmıştır; bunlardan biri canlı içerik tabloları oluşturabilmeye yarayan SoftHE ve diğeri ise temel bir televizyon olmakla beraber, kaydetme/oyunatma özelliklerine de sahip SoftTV yazılımıdır.

SoftHE Java'da yazılmıştır. Kullanıcı-dostu grafik arayüzü aracılığı ile, canlı içerik tabloları oluşturulabilir. Tablo oluşturmanın yanı sıra, bu program; oluşturulan tabloları TS paketlerine böler ve de var olan bir TS ile beraber bu paketleri birleştirebilir. Kullanıcıların sadece, oluşturacakları yeni tablonun başlık ve açıklayıcı bilgilerini programa girmesi yeterlidir.

SoftTV, MPEG-2 ve MPEG-1 görüntülerini çözebilen ve Genel Kamu Lisansı (GPL) ile kullanılabilen bedava bir kütüphane olan libmpeg2'yi kullanır. Sabit diskte kayıtlı bir TS dosyasını okuyarak temel bir televizyon gibi çalışır. Kullanıcılar, adı komut satırı parametresi olarak girilen TS dosyasında bulunan servisler arasında dolaşabilirler. Ayrıca SoftTV yazılımı, kaydetme/oyunatma gibi fonksiyonları da yerine getirir.

- Uygulama

Bu tezin son kısmında, bütünlüklü bir senaryoya yer verilmiştir.. İlk adım olarak, kimlik numarası 2 olan bir servis (TRT test yayınından TRT-2) için canlı içerik tablosu oluşturulmuştur ve açıklayıcı alanlarına bazı basit metinler girilmiştir. Metinlere ek olarak; X Pencere Sistemi'nin kullandığı, ASCII (American Standard Code for Information Interchange) tabanlı XPM formatında, bir adet grafik nesnesi (İTÜ logosu) canlı içerik içine gömülmüştür. SoftHE uygulaması, tabloyu oluşturduktan sonra, önceden TS formatında kaydedilmiş olan TRT test yayını ile harmanlamak üzere, tabloyu TS paketlerine bölmüştür.

İkinci adım olarak; SoftTV uygulaması, harmanlanmış TS dosyası ismini komut satırından okuyacak şekilde çalıştırılmıştır. Bu çalıştırma esnasında, TRT-2'den kayıt yapılmış ve yapılan kayıt daha sonra tekrardan oynatılmıştır. SoftTV uygulaması, canlı içerik tablosunu yakalayıp, açıklayıcıları ayrıştırdıktan sonra, içerikleri yeniden oynatmayı durdurarak ekrana çıkarmıştır.

- Sonu

Bu tez alıřmasında; kullanıcıya, sabit diskte kayıtlı bulunan yayınlarını yeniden oynatma esnasında, canlı ierikler sunmaya yarayan bir yntem nerilmiř ve bahsedilen yntem, bir Linux platformu zerinde gereklenmiřtir.

## **DESIGN AND IMPLEMENTATION OF AN MPEG-2 SYSTEM COMPATIBLE METHOD FOR PROVIDING LIVE CONTENT**

### **SUMMARY**

A personal video recorder (PVR) set provides many advantages to its user such as one-touch recording, scheduled recording, playback with trick modes (pause, forward, and rewind) and time-shifting. Users mostly add their favorite programs to the scheduled recording list of a PVR and replay them in their leisure times. However, users may want to be notified about live broadcasting content (latest news, special advertisements, etc.) coming from the tuner while they are watching those previously recorded streams from hard disk. In this thesis, a method is proposed for those users to be able to access live broadcasting content which is formatted as MPEG-2 PSI table structure.

In order to achieve the proposed method, a new MPEG-2 TS compatible table has been defined. The live content (LC) carried within that table can be filtered and processed in order to present to the user during playback. The proposed idea has been also implemented on a Linux platform by having developed two utility programs; SoftHE and SoftTV.

- **Defining a New Table**

The data carried within PSI tables only provides information for receivers to demultiplex and decode the elementary streams of services within the multiplex. In order to carry LC data specific for each service, it is needed a new table such as Digital Video Broadcasting (DVB) Service Information (SI) tables. Both PSI and SI tables shall be segmented into sections which are syntactic structures used for mapping tables into TS packets.

A new table called LCT is defined in order to carry LC (Live Content Table) data. The LC data should be carried within the payload of an LC section via the descriptors organized as tag-length-value (TLV) structure. Organizing LC data as descriptors provides defining new LC content types very easily, in the future. When a

new type of LC content is needed, a new corresponding descriptor should be added into the payload.

In the context of the thesis study, several text-based LC descriptors were defined such as Title, SubTitle, and Text which are distinguished according to their tags. Thus, broadcasters can generate text-based LC content having a title, subtitle and a text area by using those descriptors. Then, additional LC descriptor types were defined; such as Graphic and Video which provide non-text contents like a graphic object and a video elementary stream.

- Implementation

The idea proposed in this thesis is implemented on a personal computer (PC) running a free Linux distribution. Two applications were developed; one is Software Head-End (SoftHE) which is a utility program to generate LCTs, and the other one is Software Television (SoftTV) which is a basic zapper television which has record/playback features.

SoftHE has been written in Java. Users can form LCTs via its user-friendly graphic interface. This program can also split the LCTs into TS packets and merges these packets with an existing TS file. Users only enter the header and descriptor information through the graphical user interface to create a new LCT.

SoftTV uses libmpeg2 which is a free library for decoding MPEG-2 and MPEG-1 video streams under the terms of the General Public License (GPL). It reads a TS file from HDD and serves as a basic zapper television. Users can navigate the services available in the TS file whose name is passed as a command-line argument. It also has record/playback capabilities.

- Demonstration

In the last section of the thesis, a complete scenario is demonstrated. Firstly, an LCT was created for a service whose identification number is 2 (TRT-2 of TRT Test Multiplex) and some simple test strings were also entered to the fields of descriptors. Additionally, a graphics object (ITU logo) in the X PixMap (XPM) format, which is an ASCII image format used by the X Window System, was also embedded to the live content. After the SoftHE utility generated the LCT, it was mapped into TS packets which were, then, merged with an existing TS file that was recorded from TRT Test Multiplex.

Secondly, SoftTV was executed with the name of the merged TS file as command-line parameter. During this execution, TRT-2 service was recorded and then replayed. SoftTV utility caught the LCT, parsed it and displayed the content of the descriptors on the screen during playback.

- Conclusion

In this thesis; a method is proposed aiming to provide live content to the user when he/she replays the recorded streams from HDD of his/her PVR. The outputs of an implementation on a Linux platform is also presented.

## 1. GİRİŞ

Geçtiğimiz son birkaç on sene içerisindeki Çok Geniş Ölçekli Tümeleşirme (VLSI) teknolojilerindeki gelişmeler, sayısal sistemlerin giderek daha az yer kaplamasına ve ucuzlamasına neden oldu ve hala olmaya devam etmektedir. Bunun en doğal sonucu olarak analog sistemlerin yerlerini sayısal sistemler almaktadır. Bu yer değiştirmenin en somut örneklerine, günlük yaşantıda sıkça kullanılan tüketici elektroniği ürünlerinde rastlanmaktadır. Örnek olarak; taşınabilir kaset-çalarlar yerlerini MP3-çalarlara; içerisinde sayılı poz bulundurabilen fotoğraf makinaları ise yerlerini, çektiğimiz fotoğrafları, sayısız kere kopyalayıp silebileceğimiz sayısal modellerine bıraktılar.

Analog sistemlerin yerini sayısal sistemlere bırakmaya başladığı diğer bir alan ise televizyon yayıncılığıdır. Yayının kendisi ile beraber; onu alan, işleyen, kaydeden ve de oynatan bütün sistemler analogdan sayısala evrilmeye başladı. Kayıt ortamı olarak video-kasetler yerine CD (Compact Disk), DVD (Digital Versatile Disk), vb. ortamlar; kayıt aygıtları olarak da, video yerine sayısal kaydediciler kullanılmaya başlandı. Evlerin baş köşelerini de artık tüplü CRT (Cathode-Ray Tube) televizyonlar yerine LCD (Liquid-Crystal Display) ve plazma gibi yassı televizyonlar süslüyor. Televizyon dünyasında gözlemlenen bu evrime yol açan sayısal devrimin maddi koşullarını; VLSI teknolojisi ile beraber, özellikle doksanlardan sonra ses ve görüntünün sıkıştırılması konusunda yapılan başarılı çalışmalar oluşturmuştur. Böylelikle; bilgisayar teknolojisi, sadece donanım olarak kalmayıp, yazılım olarak da bu alanda belirleyici bir rol üstlenmeye başlamıştır.

Televizyonculuğun sayısallaşmaya başlaması ile beraber bu alanda boy gösteren oyuncular, oyunun kimi kurallarını değiştirmek ya da yeni kurallar koymak durumunda kaldılar. Özellikle coğrafi ve ekonomik parametrelere göre yeni uluslararası standartların temelleri atıldı ve bu standartlar geliştirilmeye devam edilmekteler. Ülkeler, politikalarını yeni standartları oturtmak üzere planlarken, tüketici elektroniği şirketleri ise yeni ürünler piyasaya sürme yarışındalar. Piyasaya sürülen bu ürünler, emekleme aşamasında olan sayısal televizyonculuk dünyasının geliştirilmeye en açık yönlerinden birini oluşturmaktadır.

Bilgisayar Mühendisliđi alanında sunulacak bu tez; bilgisayarların, gerek yazılım gerekse de donanım olarak teknolojik açıdan sayısal yayıncılık alanın temel bir öđesi olduđu fikrinden hareket ederek, bu alanda sunulan ürünleri, standartlara uygun olarak geliřtirmeye odaklanmıřtır. Konu olarak, sayısal yayıncılıđın son ürünlerinden olan sayısal kaydediciler için özgün bir fikir ortaya atılarak; kaydedicinin depoladıđı cansız yayınların yeniden oynatımı esnasında, kullanıcıya, canlı yayın takibi yapmaya olanak sađlayan, standartlarla uyumlu bir yöntem üzerinde çalıřılmıřtır. Tez çalıřması kapsamında ayrıca, bilgisayar ortamında, teorinin uygulaması da gerçekenmiřtir.

Tezde çalıřılan konuya ışık tutması açısından, ikinci bölümde, sayısal yayıncılık ile ilgili temellerin kısaca üzerinden geçilmiř ve sayısal yayın kaydedicilerin yapısı ve çalıřma prensibi anlatılmıřtır. Bölüm üç, tezde ortaya atılan ana fikrin standartlar kapsamında nasıl tasarlandıđı üzerinde durmaktadır. Bu bölüm, sayısal yayının yapısı hakkında da kısmi bilgiler içermektedir. Tez kapsamında gerçeken uygulmaya ise dördüncü bölümde yer ayrılmıřtır. Geliřtirilen yazılımlar tanıtılmakta ve de örnek bir senaryo üzerinden uygulama ayrıntılı olarak anlatılmaktadır. Son bölüm yani beřinci bölüm ise bu tez çalıřmasının sonuçlarına dayanarak çeřitli yorumlar içermekte ve de devam niteliğinde yapılabilecek çalıřmalar hakkında çeřitli öngörülerde bulunmaktadır.

Tezde kullanılan dile de ayrıca özen gösterilmeye çalıřılmıřtır. Özellikle son teknoloji kavramlarının mümkün oldukça dildeki karřılıkları kullanılmıř ve bu konuda ‘Bilgisayar Terimleri Karřılıklar Kılavuzu’<sup>1</sup> ana kaynak olarak izlenmeye çalıřılmıřtır.

---

<sup>1</sup> <http://www.tdk.gov.tr>

## 2. TEMEL KAVRAMLAR

Sayısal yayın; hiç bozulmadan iletilebilmesi ve kopyalanabilmesi, gürültü ve karışımdan az etkilenmesi, band genişliğinin az olması nedeni ile analog yayına göre daha üstün özelliklere sahiptir [1]. Sayısal yayıncılık alanında yapılan çalışmalar, ancak 90'lı yıllardan sonra boy göstermeye başlamıştır. Bunun nedeni ise daha verimli sıkıştırma algoritmalarının hızlı bir şekilde bu tarihlerden itibaren geliştirilmesi ve geliştirilen bu algoritmaların, tümleşik devre teknolojileri sayesinde, ucuz bir şekilde donanım yardımı ile gerçekleştirilebilmesidir [2].

### 2.1 Sayısal Yayıncılık Temelleri

Sayısal yayıncılığın temelinde, analog görüntü (**V**) ve de ses (**A**) işaretlerinin sayısal hale dönüştürülmesi vardır. Fakat, sıkıştırma yapılmadan bu işaretlerin iletimi için çok yüksek ikil-hızı (bit-rate)'na ihtiyaç vardır. Örnek olarak; televizyon ekranında herhangi bir anda gördüğümüz 525 ya da 625 satırlık<sup>1</sup> bir resmin sayısal olarak iletilebilmesi için 108 - 270 **Mb/s** arası bir ikil-hızına ihtiyaç duyulacaktır [2]. Bu ise, iletimi, mevcut yayın sistemi üzerinde olanaksız kılmaktadır. Sıkıştırma teknikleri kullanılarak, bu yüksek ikil-hızı, SD yayınlar<sup>2</sup> için, 4 - 9 **Mb/s** arasına kadar düşürülebilmektedir. İkil-hızındaki bu düşüş, sıkıştırmanın, sayısal yayıncılığın gelişiminde, neden bu kadar önemli olduğu gerçeğini ortaya koymaktadır.

Analog yayında bir kanal için en fazla 30 – 40 **Mb/s** ikil-hızına çıkıldığı [2] göz önüne alınırsa, sayısal yayına geçildiği zaman tek bir analog kanal yerine 4 – 10 arası SD servis<sup>3</sup> iletiminin mümkün olduğu görülür. Bu ise çoklama yöntemi ile yapılır. Birden fazla servise ait **A**, **V** işaretleri sıkıştırıldıktan sonra paketlere ayrıştırılıp, çeşitli veri (**D**) paketleri de aralarına katılarak harmanlanır.

---

<sup>1</sup> Bir televizyon resmi Amerika standartlarına göre 525, Avrupa standartlarına göre ise 625 satırdan oluşur.

<sup>2</sup> MPEG-2 standardı resim çözünürlüğüne göre 4 seviye tanımlamıştır. Bunlardan ilk ikisi yani 360x288 ile 720x576 çözünürlüğe sahip seviyeler SD; diğer ikisi de (1440x1152 ile 1920x1152 çözünürlüklü) HD (High Definition) olarak nitelendirilir.

<sup>3</sup> Servis sözcüğü, sayısal yayıncılıkta, analog yayıncılıktaki kanal sözcüğü yerine kullanılır.



Sıkıştırma ve de çoklama işlemleri sayısal yayıncılığa özgü en önemli iki basamağı oluşturur.

### 2.1.1 Sıkıştırma

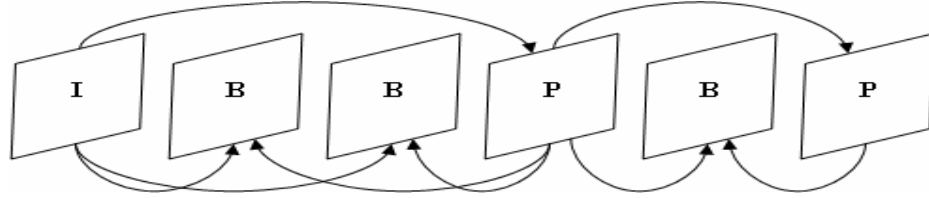
Görüntü işareti sayısala dönüştürüldükten sonra, sıkıştırma işlemine başlanır. Görüntü sıkıştırma, başlıca üç noktaya dayanır. Bunlardan ilki, hareketsiz resimler için uzamsal benzerliklerdir; diğer bir deyişle, resim içinde birbirine konum olarak yakın piksellerin, değerlerinin de birbirlerine yakınlığıdır. İkinci nokta ise, hareketli resimler için zamansal benzerliklerdir; zaman içerisinde birbirini takip eden resimlerin birbirlerine benzerliklerinin fazla olmasıdır. Bir diğer nokta ise insan gözünün görme yeteneğinin sınırlarıdır. İnsanların görsel sistemi, hareketsiz bir resimdeki çok ince detayları, hareketli görüntülerde de geçişleri ancak belli bir seviyeye kadar yorumlayabilmektedir [3].

1990 yılında, ISO hareketsiz görüntülerin sıkıştırılması amaçlı JPEG adında bir grup oluşturdu. Aynı yıl endüstrinin çoğu alanından biraraya gelen şirketlerin de katılımı ile, sayısal A, V işaretlerinin sıkıştırılması ile ilgili MPEG adı ile bilinen bir başka çalışma grubu daha oluşturdu. İlk hedef olarak CD ortamına sıkıştırılmış görüntü kaydetmeyi planlayan grup, sırası ile, JPEG algoritmasını da kullanarak, MPEG-1, MPEG-2, MPEG-4 ve de MPEG-7 standartlarını oluşturdular [2, 3].

JPEG algoritması altı ana aşamadan oluşur; sayısal bir görüntü ilk olarak, 8x8 piksellik öbeklere ayrılır. Bu ayırma işi renk ve ışık bileşenlerinin her biri için ayrı ayrı yapılır. İkinci aşamada, her bir öbek DCT'den geçirilerek yeni 8x8'lik katsayı matrisleri oluşturulur. Oluşan yeni matrislerin sol üst köşesindeki değer, ortalama renk ya da ışık büyüklüğünü verir. Sol üst köşeden, sağ alt köşeye gidildikçe, katsayılar hızlı bir şekilde düşerler ve sıfıra yakın (bazen sıfır) değerler alırlar. Bu katsayıların düşük olması yani ışık seviyesinin belli bir değer altında olması, insan gözünün bu ayrıntıları fark edemeyeceği anlamına gelir. O nedenle, üçüncü aşamada, bu katsayı değerleri elenir. Belli bir seviyenin altında olan değerler sıfıra eşitlenir, diğer katsayılar ise belli bir aralığa denk gelecek şekilde yuvarlatılır. JPEG algoritmasında bilgi kaybının olduğu yer bu aşamadır. Dördüncü aşamada, içinde bolca sıfır değeri bulunduran matrisler söz konusudur. Bu matris zig-zag yöntemi ile tek boyutlu hale getirilir. Beşinci aşamada, oluşmuş olan tek boyutlu dizi üzerinde tekrarlanan sıfır değerlerinin sayısı kodlanır. Son aşamada ise, Huffman kodlaması ile, yani en çok tekrarlanan değerlere daha kısa kodun denk düşürülmesi ile sıkıştırma tamamlanmış olur [2]. Basitleştirilmiş JPEG kodlayıcı ve çözücünün blok diyagramları [Ek A](#)'da gösterilmiştir [4].

Sayısal yayıncılık uygulamaları için MPEG komitesi; A, V işaretlerini sıkıştırmak üzere MPEG-2 standardını geliştirmiştir. MPEG-2’de iki tip sıkıştırma söz konusudur; uzamsal ve de zamansal. Uzamsal ya da konumsal sıkıştırma JPEG algoritmasında anlatıldığı gibi yapılıdır. Bu sıkıştırma, hareketsiz resimlerde, komşu piksel değerlerinin birbirleri ile olan yakınlığını kullanmakta idi. Zamansal sıkıştırma ise ardarda gelen resimlerin benzerliklerinden faydalanır ve ardışıl resimlerin arasındaki farkı kodlar.

MPEG sıkıştırma yapılırken, iki çeşit zamansal kodlama yöntemi uygulanır. Bunlardan ilki, *çerçeveler-arası kestirim*dir. Çerçeve kavramı, MPEG standardının ortaya çıkardığı üç tip resimden gelir. Bunlar; I, P ve de B resimleridir ve çerçeve olarak adlandırılırlar. I çerçevesi, resimlerin doğrudan JPEG algoritması ile kodlanması ile oluşturulur. MPEG kodlanmış bir görüntü çözüldüğünde, ilk olarak bu çerçeveler açılır. I çerçeveleri, P ve B çerçevelerinin çözülmesi için de gereklidir. Çünkü; P çerçevesi kendinden önce gelen I (ya da yine P) çerçevelerinden elde edilmiştir. B çerçevesi ise hem kendinden önceki hem de sonraki I ya da P çerçevelerinin arasına yerleştirilecek şekilde hesaplanır. Şekil 2.1 bu çerçeve tiplerinin dizilimini göstermektedir. Resimlerin bu yöntemle kodlanması, P çerçeveleri için I çerçevesinin yarısı kadar, B çerçeveleri için de I çerçevesinin dörtte biri kadar veri kullanılarak sıkıştırma yapılmasını sağlar [3].



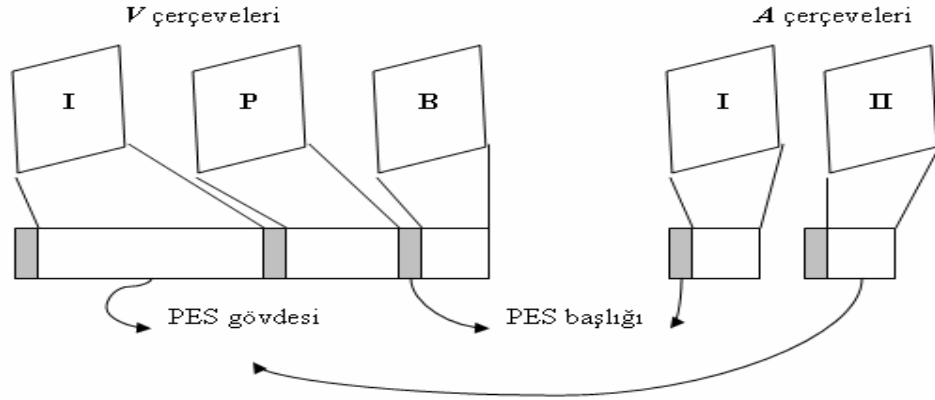
**Şekil 2.1:** Çerçeveler-arası kestirim

MPEG’in kullandığı ikinci kodlama yöntemi ise I çerçevesinden P ve B çerçevelerini hesaplamaya dayanan *hareket kestirim*dir. Bu yöntem ise hareketli görüntülerde, genelde nesnelerin yer değiştirdiği ve görünüşlerinin pek değişmediği fikrine dayanmaktadır. İşleyiş olarak; bir çerçevedeki 16x16’lık makro öbekler, diğer çerçevedeki makro öbekler ile karşılaştırılarak, en yakın olan makro öbek belirlenir. Bu aşamaya *öbek eşleştirme* adı verilir. Eşleştirme sonucu bulunan öbek ile ilk öbek arasındaki fark hesaplanarak *hareket vektörü* oluşturulur ve sadece, bu vektörler sıkıştırmada kullanılır. Hareket halindeki bir nesnenin hızının sabit olduğunu düşünürsek, hesaplanan vektör değişmeyecektir ve de sadece fark olarak sıfır vektörü kodlanacaktır. Bu da büyük bir sıkıştırma miktarı demektir [3].

## 2.1.2 Çoklama

MPEG sıkıştırması sonucu oluşan ikil dizisi Temel Akım (ES) olarak adlandırılır. Bir sayısal servis için en azından bir tane A ES'i, bir tane de V ES'i olmakla beraber, sayısal yayıncılıkta bunlar birden fazla da olabilir. Örnek olarak; bir servis iki farklı dilde yayın yapabilir. Yayınlanacak servislerin A, V ES'leri, çoklama aşamasında birbirleri ile harmanlanırlar ve tek bir akım oluşturulur. İçinde birden fazla servise ait ES'ler içeren bu tek akım Taşıma Akımı (TS) diye adlandırılır.

Harmanlama işleminin gerçekleştirilebilmesi için ES'lerin paketlenmesi gerekir. Böylelikle, tek bir akım içerisinde, istenilen bir ES'e bağımsız olarak erişilebilir. Paketlenmiş ES'lere ise PES adı verilir. PES paketleri değişken uzunlukludur, nedeni ise ES içerisindeki çerçevelerin tek bir pakete konulmasıdır. Şekil 2.2'de A ve V çerçevelerinin paketlenmesi görsel hale getirilmiştir. Şekilde A çerçeveleri, sıkıştırma katmanına göre I ve II olarak adlandırılmıştır. Bu çerçeveler sayısal yayıncılıkta kullanılmaktadır<sup>1</sup>.



Şekil 2.2: ES çerçevelerinin paketlenmesi

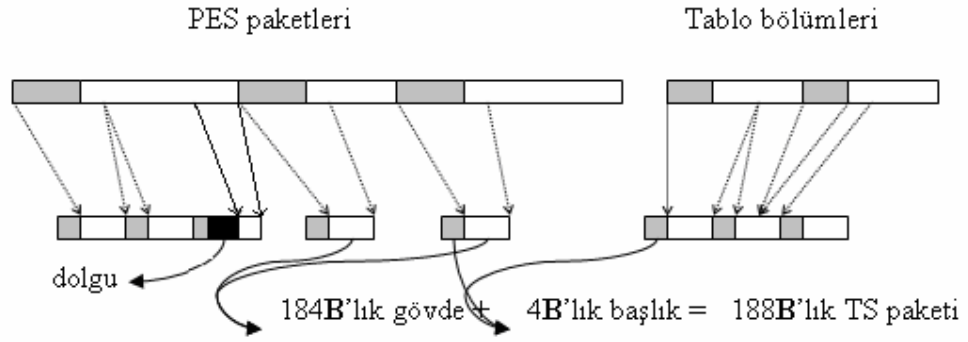
PES gövdelerinin çerçeve içeriklerini taşıdığı belirtilmişti, başlıkları ise ilgili çerçevelerin, alıcı tarafta, ne zaman çözülmesi (DTS) ve de ne zaman gösterilmesi (PTS) gerektiği ile ilgili zamanlama bilgileri taşır.

A ve V ES'lerine ek olarak, yayıncı, *altyazı* ve de *teletext* gibi, bir servise ait olabilecek başka bileşenleri de PES yapısında yayına yerleştirebilir. PES'lerle beraber; harmanlanacak servislerle ilgili ya da bütün bir harmanı taşıyacak TS ile ilgili bir takım bilgilerin yer aldığı tablolar da TS ile taşınır. Bu tablolar bölüm

<sup>1</sup> III. katman MPEG ses sıkıştırması MP3 olarak tanınmaktadır ve daha çok ağ uygulamalarında kullanılmaktadır.

(section) adı verilen yapılar şeklinde düzenlenmişlerdir. Uzunluğu azami 16 **KB** olabilen tablo bölümleri de sabit uzunluklu başlık (8 **B**) ve de gövdeden oluşurlar<sup>1</sup>.

Bütün PES yapıları ve de tablo bölümleri, MPEG-2 Sistem [5] standardına uygun paketlenerek TS'i oluşturur. Şekil 2.3'te PES ve tablo bölümlerinin TS paketlerine bölünmesi gösterilmiştir. PES paketlerinin aksine, TS paketlerinin uzunlukları 188 **B** olarak sabitlenmiştir. Paket uzunluğunun 188 **B** olarak seçilme nedeni, bu paketleri, 47 **B**'lık<sup>2</sup> paket gövde uzunluğuna sahip ATM (Asynchronous Transfer Mode) ağları üzerine de kolayca taşıyabilmektir [3].



**Şekil 2.3:** TS paketlerinin oluşumu

Herhangi bir TS paketi, sadece bir adet PES'in verilerini taşıyabilir. TS paketlerine gömülürken, Şekil 2.3'te görüldüğü gibi, PES'in son paketlenen kısmındaki bayt sayısı 184'ten küçük ise, dolgulama (*stuffing*)<sup>3</sup> yapılır. Fakat aynı durum tablo bölümleri paketlenirken geçerli değildir. Bu, bir TS paketi içinde herhangi bir bölümün bittiği yerde, aynı tabloya ait olmak koşulu ile, başka bir bölümün başlayabileceği anlamına gelir. Herhangi bir pakette, yeni bir tablo bölümünün başlayıp başlamadığı PUSI ikilisinin değerine göre kontrol edilir.

TS paketlerinin; hangi **A**, **V**, **D** ya da tabloya ait olduğu, PID numaralarına göre ayırılır. 13 ikiliden oluşan bu değer, her servisin değişik bileşenleri için, o servise ve bileşene özel, ve de tek olacak biçimde ayarlanmıştır. Böylelikle, yayın alıcısı, ilgili bir servise ait paketleri ayırt ederek, yeniden PES ve tablo bölümlerini oluşturabilmektedir. PID, PUSI, vb. TS paket başlığını oluşturan ikili ya da ikililerin yerleşimi, Tablo 2.1'de verilmiştir.

<sup>1</sup> Bu kurala yalnız, özel tanımlanmış tablolar uymaz, bu tablolar bölüm başlıklarında yeri sabit olan SSI ikilisinin değeri ile ayırt edilirler.

<sup>2</sup>  $188 = 47 \times 4$

<sup>3</sup> Dolgulama işlemi, paketlerin belli kısımlarını 0xFF değeri ile doldurmaktır.

**Tablo 2.1:** TS paketi başlık yapısı

Alan	Açıklama	İkil Sayısı
<b>sync_byte</b> <i>Eşzamanlama Baytı</i>	Değeri 0x47'dir ve alıcının, paketlerin başladığı yeri algılamasını sağlar.	8
<b>transport_error_indicator</b> <i>Taşıma Hatası Belirteci</i>	Bu ikil '1' ise, bu paketten önceki paketlerde hata olduğunu gösterir.	1
<b>payload_unit_start_indicator</b> <i>Gövde Birimi Başlama Belirteci</i>	Bu ikil '1' ise, paket içinde yeni bir PES yada tablo bölümü vardır.	1
<b>transport_priority</b> <i>Taşıma Önceliği</i>	Taşıyıcı sistem öncelik değeri '1' olan paketlere öncelik verir.	1
<b>PID</b> <i>Paket Kimliği</i>	Paketlerin hangi servisin hangi bileşene ait olduğunu gösterir.	13
<b>transport_scrambling_control</b> <i>Taşıma Karıştırma Denetimi</i>	Paketin baytlarının karıştırılma tipini belirtir. Şifreli yayınlarda kullanılır.	2
<b>adaptation_field_control</b> <i>Uyarılama Alanı Denetimi</i>	Paketin gövdesinde dolgu ya da başka amaçlı verinin olması ile ilgili durumu gösterir.	2
<b>continuity_counter</b> <i>Süreklilik Sayacı</i>	Paketlerin sıralı olmasını sağlar. En büyük değerden sonra başa döner.	4

TS başlığındaki ikili ve ikililerin, Tablo 2.1'deki açıklamaları ışığında, yayın alıcı tarafında kullanılmasına şöyle bir örnek verilebilir. Sayısal yayın alıcı; belli bir servise ait görüntü PES'ini TS'den ayıklayacak olursa; ilk olarak yayın ile eşzamanlılığı sağlamalıdır. Bunun için; yayından aldığı her baytın 0x47 olup olmadığını kontrol etmelidir, bu değeri bulursa, bunun gerçekten eşzamanlama baytı mı yoksa herhangi bir 0x47 değeri olup olmadığını sınımalıdır. Paket başlığından emin olunduktan sonra, bu paketin istenilen PES'e ait olup olmadığı, PID değerine bakılarak saptanır. PID denetiminden geçildikten sonra da, bu paketin içinde PES'in başlayıp başlamadığı, PUSI değeri ile kontrol edilir. Bu son kontrol de başarılı ise bu paket, arandığı PES'in ilk paketidir. Uyarılama alanları geçilerek, PES'in verilerine ulaşılabilir. PES'in verilerinin bulunduğu diğer paketleri de bulabilmek için, ilk paketin süreklilik sayaç değeri not edilir. İlk paketten PES başlığı ayrıştırılarak uzunluk hesaplanır ve bu uzunluğa göre yeni paketler alınır. Bir sonraki paket, PID denetimini geçtikten sonra, PUSI denetimine bakılmaksızın, süreklilik sayacı kontrol edilir. Bu değer, not ettiğimiz değerden, mod 15'e göre bir fazla olması gerekir<sup>1</sup>. Son paketin de verisi alındıktan sonra PES oluşturulmuş demektir.

TS ile ilgili bir diğer önemli nokta ise zamanlamadır. Bir servisin **A** ve **V** bileşenlerinin birbirleri ile eşzamanlı olması gerekir. Bunun sağlanması için de, TS

<sup>1</sup> 15 (=0xF) değeri dört ikiliden oluşan süreklilik sayacının alabileceği azami değerdir.

oluşturulurken, paketler arasına sistem zamanını da içeren özel PCR paketleri eklenir. Sistem saatinin frekansı sabittir ve 27 MHz'dir. Bu paketlere ek olarak, her PES paketinin de başlığında, ne zaman çözülüp sunulacağına dair bilgiler yer aldığına değinilmişti. PCR değerlerini kullanarak, sistem saati ile kendisini eşzamanlayan sayısal yayın alıcı, ayıkladığı PES başlıklarından elde ettiği DTS ve de PTS değerlerini kullanarak, PES'leri doğru zamanda çözüp sunabilir.

## 2.2 Sayısal Yayın Kaydediciler

Sayısal yayın kaydedici aygıtlar, en basit tanımıyla, içlerinde yayını kaydedebilecekleri depolama birimi bulunduran yayın alıcılardır. Sayısal Görüntü (Video) Kaydedici olarak da adlandırılmalarına rağmen, ürün ismi olarak Kişisel Görüntü Kaydedici (PVR) adlandırması daha ön plana çıkmıştır. Kişisellik vurgusu, kullanıcının, gelen yayını, sabit diski üzerinden seyretmesinden ötürü, kolaylıkla yönlendirebilmesinden kaynaklanmaktadır. Şöyleki; gelen canlı yayın, depolama biriminde tamponlandığından, kullanıcının isteğine göre durdurulabilir; tampon yayın üzerinden geri gidilebilir; bir kere geriye gidilmiş ve geriden izlenilmeye başlanmışsa, tekrar canlı yayına, tampon yayın ileri yönde hızlı oynatılarak yetişilebilir [6].

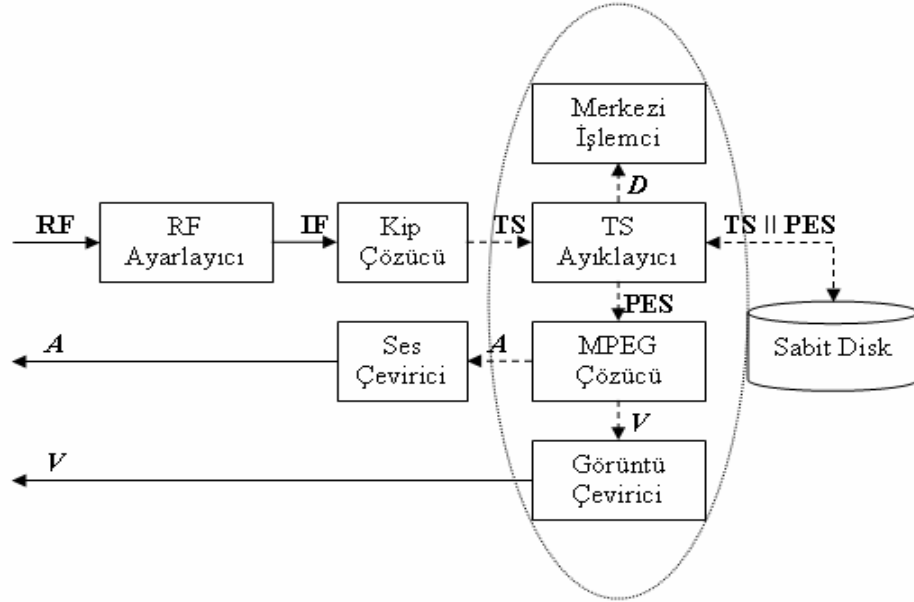
PVR'larda depolama birimi olarak, genellikle sabit diskler kullanılır. Normal yayın kaydetmenin dışında, PVR'ların en önemli özelliği, gelen yayını doğrudan bu sabit diske yazıp, buradan oynatmasıdır. Bu özelliğe, zaman-kaydırma (time-shifting) adı verilir. Bu sayede, kullanıcı istediği an canlı yayını dondurabilir ve belli bir süre sonra, kaldığı yerden izlemeye devam edebilir. Bu işlemde, kullanıcı canlı yayını geriden izlemeye başlar ve normal yayın akışı ile arasında zaman kayıklığı meydana gelir. Kullanıcı istediği takdirde, canlı yayını tekrar yakalayabilir.

PVR'larda zaman-kaydırma yapılabilmesi için sabit disklerde tampon bir bölge ayrılmıştır. Bu bölgenin büyüklüğü, kullanıcının ne kadarlık bir zaman kaydırma yapabileceğini belirler. Ayrılmış olan bu bölge, çoğu PVR yazılım sistemlerinde, *dairesel dosya* olarak isimlendirilir. Kayıt esnasında dosya sonuna geldiği vakit, tekrar başa dönülür ve kaydetmeye devam edilir. Aynı durum oynatma esnasında da geçerlidir.

Tek kumanda tuşu ile anında kaydetmeye başlayabilen PVR'lar, sayısal yayının nimetlerinden de faydalanarak, kullanıcıya haftalık program listesinden kayıt listesi oluşturabilmesini [7], kayıtları arasında akıllı taramalar yaparak kolayca dolaşabilmesini [8] de sağlar.

## 2.2.1 Donanım

Şekil 2.4'te sabit disk dışında kalan birimler, günümüz sayısal yayın alıcılarında bulunan tipik birimlerdir. PVR'lar da genel anlamda bir sayısal yayın alıcı olduklarından bu birimleri donanım olarak içerirler ve bunlara ek olarak yayını kaydedebilecekleri sabit disk içerirler. Sabit disk dışında kalan birimler, tek bir kart üzerinde dizilebilecek tümdevrelerden oluşmaktadırlar. Günümüz teknolojisinde, yarı-iletken firmaları, sayısal yayıncılık ürünleri için özel tümdevreler üretmektedirler ve de çoğunlukla tek bir tümdevreye Şekil 2.4'te dikey elips içine alınmış birimleri sığdırabilmektedirler. Tümdevreyi özellikle PVR için özelleştirmek amacı ile sabit disk arabirimini de içerecek şekilde üretirler. Böyle bir tümdevre diyagramı [Ek B](#)'de gösterilmektedir [9]. Ayrıca son dönemde, RF ayarlayıcı dışındaki birimlerin de (Kip Çözücü, Ses Çevirici) ana birimlerin bulunduğu tümdevreye alınarak, tek bir tümdevre olarak üretilmeye başlanmıştır. RF ayarlayıcının ise *silikon ayarlayıcı* (silicon tuner) adı verilen, hacimce daha az yer kaplayan modelleri üretilmektedir.



Şekil 2.4: Genel PVR yapısı

PVR sistemi kapalı bir kutu olarak düşünüldüğü vakit; dış dünyadan RF yayını girdi olarak alıp, çıktı olarak ise genellikle televizyon sistemine ses ve görüntü olarak verir. Şekil 2.4'te analog işaret geçişleri düz oklar ile, sayısal işaret geçişleri de kesikli oklarla çizilmiştir. Böylelikle hangi birimlerin sayısal-analoga çevirici ve analogdan-sayısal çevirici devreler içerdiği anlaşılabilir. Bu durumda, çıkışın bağlanacağı televizyon sistemi analog düşünülmüştür.

İstenilen frekanstaki yayının seçilmesi, RF ayarlayıcı (tuner)'nın görevidir. RF ayarlayıcı istenilen frekanstaki yayını yakaladıktan sonra, belli bir ara frekansta (IF) sabitler, bu frekans değeri 480 MHz'dir. Ayarlayıcı birimi, genellikle, merkezi işlemci tarafından I<sup>2</sup>C protokolü<sup>1</sup> ile kontrol edilir.

Kip çözücü (demodulator), IF işaretlerini çözerek sayısal TS'i oluşturur. Bu birim IF işaretlerini sayısalan-analoga çevirici üzerinden sayısallaştırır. İletim ortamında kullanılan kipleme yöntemine<sup>2</sup> göre bu birim özelleştirilmiştir. TS paketi bazında hata tespit ve de düzeltilmesi de yapar. Çıktı olarak TS'i, 8 b'lik veri, saat ve kontrol işaretleri ile beraber paralel olarak verir. Kontrol işaretleri arasında, çözemediği hata durumlarını belirten hata işareti de vardır.

RF ayarlayıcı ile kip çözücünün oluşturduğu ikiliye, son alıcının ön bileşenleri anlamında Son-Ön (Front-End) adı verilir. Genelde birini üreten firma, diğerini de üretmektedir.

TS ayıklayıcı (demultiplexer), TS'den istenilen görüntü ve de ses paketlerini ayıklayarak PES'leri, ayrıca istenilen veri paketlerini de ayıklayarak tablo bölümlerini oluşturur. Bu işlem için, merkezi işlemci tarafından programlanabilen filtreleri mevcuttur. Filtrelere geçirilecek ana parametre, ayıklanması istenilen TS paketlerinin PID değerleridir. Tablo bölümleri söz konusu olunca, filtre parametreleri artmaktadır. Tablo bölümlerinin numaraları, tablo versiyonu, tablo tipi, vb. parametreler söz konusudur. Filtre parametrelerini saklayıcılarında tutan ayıklayıcı, oluşturduğu bölüm ya da PES'leri sistem belleğindeki tampon alanlara kopyalar.

PVR'lara özel olarak, Şekil 2.4'te de görüldüğü üzere, TS ayıklayıcıda filtrelenen TS paketleri ya da filtrelenip oluşturulan PES paketleri sabit diske de kopyalanabilir. Ayıklayıcı ile sabit disk arasındaki bu bağlantı, kaydın yapıldığı yoldur. PVR uygulamalarında, sabit diske kaydedilecek paket cinsi (PES ya da TS) farklılık gösterebilmekle beraber, PES yerine TS paketleri kaydetmek daha fazla tercih edilmektedir. Kayıttan oynatma esnasında, eğer PES paketleri kaydedilmiş ise; **A**, **V** eşzamanlaması yapmak karmaşık bir işlemdir. TS kaydedildiği durumda, TS tekrar ayıklayıcı üzerinden dolaştırılarak, PES'lerin oluşturulması sağlanır.

MPEG çözücü, TS'den ayıklanıp oluşturulan PES'leri çözer ve sıkıştırılmış sayısal görüntü ile sesi yeniden oluşturur. Eşzamanlamayı TS'den ayıklanan PCR ile sağlar.

---

<sup>1</sup> I<sup>2</sup>C protokolü, tümdevreler-arası seri iletişim sağlamak amacı ile Philips tarafından geliştirilmiştir.

<sup>2</sup> Karasal (terrestrial) ortam için COFDM (Coded Orthogonal Frequency Division Multiplex), kablo (cable) için QAM (Quadrature Amplitude Modulation) ve uydu (satellite) için QPSK (Quadrature Phase Shift Keying) kipleme yapılır.



Merkezi işlemci, bütün donanım birimlerini kontrol etmenin yanında, kullanıcıdan kumanda ile gelen istekleri karşılama, TS'den ayıklanan bölümlerden tabloları oluşturma, kullanıcı menülerini gösterme, vb. işleri yapar. Sayısal yayıncıya gelen zengin içerikle beraber, kullanıcıya fazla uygulama sunma işlemini yerine getirebilmek için günümüz sistemlerinin çoğunda, 32 b'lik güçlü RISC işlemciler<sup>1</sup> kullanılmaktadır [2].

Görüntü çevirici, MPEG çözücünden sayısal olarak çözülmüş görüntüyü, ilgili formata çevirdikten sonra analoga dönüşümü işlemini de yapar. Görüntü çevirici, ilgili format sayısal ise, analog yerine sayısal çıkış da verebilir, bu durumda analog dönüşümü yapmaz.

Ses çevirici de MPEG çözücünden çıkan sayısal formattaki sesi analoga dönüştürerek, ses çıkışlarına verir. Eğer çıkış sayısal bir format ise, bu birim sadece format çevrimi yapar.

PVR'larda kullanılan sabit diskler, genel amaçlı bilgisayarlarda kullanılan sabit disklerden farklı değildirler. Paralel ya da seri olmak üzere iki sabit disk tipi de PVR'larda kullanılmaktadır. Yalnız, görüntü ve ses gibi çokluortam dosyalarını ve yayını depolayacağı için, yüksek kapasitede depolama özelliği olmalıdır. PVR'larda kullanılan kapasite genelde 100 GB'lar mertebesindedir.

Yukarıda tanıtılan ve Şekil 2.4'te gösterilen birimlerin dışında herhangi bir sayısal alıcıda; çevre aygıtları ile haberleşmek için çeşitli arabirimler, şifre çözücü birim, kart okuyucu, vb. başka birimler de bulunabilir. Haberleşme arabirimlerine örnek olarak UART (Universal Asynchronous Receiver-Transmitter), I<sup>2</sup>C ve de son zamanların vazgeçilmezi olan USB (Universal Serial Bus) gösterilebilir. Bu arabirimler aracılığı ile alıcı sistem çevre aygıtlara bağlanabilir. Örnek olarak, bir modeme bağlanılarak, yerel ya da genel ağ bağlantısı yapılabilir.

Şifre çözücü birim ise, yayının şifreli olduğu durumlarda, yayını çözmeye yarar. Şifreli yayınlar, belli bir algoritma ile çeşitli anahtarlar kullanılarak yayıncı şirket tarafından şifrelenir. Şirket, şifrenin çözülebilmesini sağlayacak anahtarları ise akıllı-kartlara (smart-card) kopyalayarak abonelerine verir. Alıcı kutuda, şifre çözücü birim TS ayıklayıcıya gelmeden önce devreye girer ve kart okuyucu ile haberleşerek gerekli anahtarları alıp yayını çözer.

---

<sup>1</sup> RISC işlemciler, komut kümelerinin azlığı nedeni ile sahip oldukları basit mantık devreleri ile daha az yer kaplayan ve de güç tüketiminin daha az olduğu işlemcilerdir. Bu özellikleri ile, sayısal yayıncılık sistemlerinin de dahil olduğu gömülü sistem uygulamalarında tercih nedenidirler.

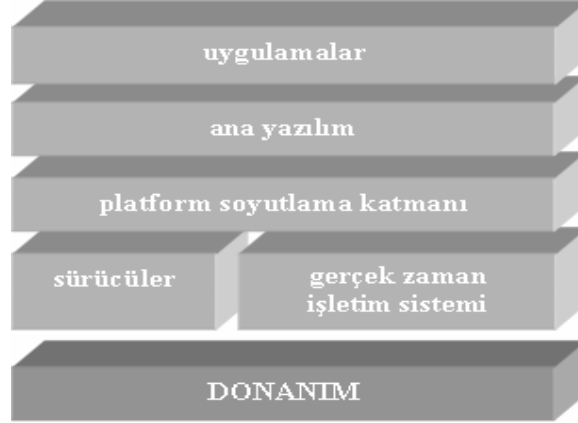
## 2.2.2 Yazılım

Genel amaçlı yazılmış yani belli bir uygulama için özelleşmemiş, üzerinde çalışacağı donanım mimarisinden bağımsız bir yazılımın yapısı Şekil 2.3'te görüldüğü gibi katmanlara ayrılmıştır. Böyle bir yazılım mimarisi, sayısal yayıncılık uygulamaları için oldukça idealdir. Bunun nedenlerini şöyle sıralayabiliriz:

Birincisi; tümdevrelerin (dolayısı ile de sürücülerinin) gün geçtikçe evrimleştiği düşünülürse, yazılımın bundan etkilenmemesi için üzerinde çalışacağı donanım ile arasında bir soyutlama katmanına gerek vardır.

İkinci olarak; gerçek zamanda çalışan ve birbirlerinden bağımsız birden fazla işlemi paralel yürütmek zorunda olan bir yazılım söz konusu olduğundan, gerçek zaman işletim sistemine (RTOS) ihtiyaç vardır.

Son olarak da; sayısal yayıncılık belli standartlar üzerine oturduğundan, değişmez bir takım kuralları vardır. Örnek olarak, TS paketlerinin uzunluğu her zaman sabittir. İstisnasız bütün uygulamalarda, TS ayrıştırılırken, ön tabloların oluşturulması gerekir. Bütün bu değişmez ve de gerekli kısımların kodlandığı ana bir yazılım bileşenine ihtiyaç vardır.



**Şekil 2.5:** Genel yazılım bileşeni katmanları

Yukarıda bahsedilen katmanları içeren bir yazılım, ana yazılım katmanında bulundurduğu modüller ölçüsünde uygulama katmanına hizmet sunarlar. Örnek olarak; bir PVR sisteminde, ana yazılım katmanının kaydetme/oynatma işlemlerini gerçekleştirmesi gerekir. Bunun için de ilgili sabit disk sürücülerinin yazılıma eklenmiş olması şarttır. En üst aşamada da, kullanıcıya bu özelliğin menüler ya da kumanda gibi girdi/çıkı araçları ile kontrol edebilmesi sunulmalıdır. Yukarıdaki katman mimarisi göz önünde bulundurularak, bileşenlerin ayrıntıları aşağıda sıralanmıştır.

Sürücüler, mevcut bir donanımın ana fonksiyonlarını yerine getirmesini sağlayan yazılımlardan oluşur. Örnek olarak; bir RF ayarlayıcının belli bir frekansa bağlanmasını sağlayan bir C fonksiyonu gösterilebilir. Yazılımcı kişi, ayarlayıcının elektronik yapısını ve de saklayıcılarının adresleri ile uğraşmaz, yalnızca ilgili fonksiyonu doğru parametreler ile çağırarak yetinir. Genellikle, donanımı üreten firma tarafından C kütüphaneleri ve başlık dosyaları olarak, donanım ile beraber verilir.

İşletim sistemi, merkezi işlemciyi ana yazılımın ya da uygulamanın yapması gereken işlere paylaştıran en temel yazılım bileşenidir. Sıradan bir sayısal yayın uygulamasının, bir taraftan kullanıcıdan kumanda ile gelen istekleri yerine getirmesi gerekirken, bir taraftan da sayısal yayını ayrıştırması gerekir. Bunlara ek olarak, ekrana bir de grafik çizmesi gerekebilir. Bütün bu görevlerin eşzamanlı yapılıyor olması şarttır. Fakat fiziksel olarak bu imkansızdır, çünkü tek bir işlemci vardır. İşletim sistemi, işlemcinin hızından faydalanarak, bahsi geçen görevleri, belirli sürelerle yerine getirir ve kullanıcıda bir eşzamanlılık hissi uyandırır. Sözde eşzamanlılık adı verilen bu yöntemi gerçeklemek işletim sisteminin görevidir. Gerçek zamanlılık sıfatı ise, işletim sisteminin yapması gereken işlere bir öncelik atamasından gelir. Örnek olarak, sayısal yayında paketler gerçek zamanda sürekli gelir ve de paketleri kaçırmamak en yüksek önceliklidir. Ekrana çizilen bir grafik biraz gecikebilir ve bu sistem için çok önemli olmayabilir ama paketlerin işlenmesindeki bir gecikme sistemde yanlışlıklara ve daha önemli sonuçlara yol açabilir. Bu nedenle, önceliğin yayının işlenmesine verilmesi gerekir.

İşletim sisteminin yerine getirdiği diğer bir önemli görev de, bellek yönetimidir. Çalışma anında, uygulamalar gerek duydukları belleği işletim sistemi üzerinden alabilir ve daha sonra işletim sistemine geri verebilir. Ayrıca farklı işleri yerine getiren yazılım parçalarının aynı bellek bölgesi üzerinde ortak çalışabilmeleri; birbirleri ile eşgüdümü de işletim sisteminin görevleri arasındadır.

Sayısal yayıncılık yazılımında başlıca kullanılan gerçek zamanlı işletim sistemlerine uCOS, Nucleus, vxWorks, vb. örnek verilebilir. Bunların bir kısmı açık lisanslı, bir kısmı ise lisanslıdır. Bunların yanında, Microsoft'un gömülü sistemler için ürettiği WinCE ile Linux'un gerçek zamanlı türevleri de mevcuttur.

Ana yazılım, çeşitli modüllerden oluşur. Bunların içerisinde başlıca, sistemin durumlarını tutan ve belli bir andaki durumdan diğerine geçmesini sağlayan ana kontrol modülü, TS'den çıkan tabloları yöneten modül, menülerin çizilmesini sağlayan grafik modülü yer alır. PVR işleri (kaydetme/kontrol) için de gerekli denetimleri ana kontrol modülü yerine getirebilir. Fakat, sabit disk üzerindeki

dosyaların yönetimini sağlayan bir dosyalama sisteminin arayüzünün de ana yazılım içerisinde bulunması gerekir.

Ana yazılım modüllerinin çoğu standart işleri yerine getirdiğinden bunların bir kere yazılmış olmaları yeterlidir. Daha sonra çeşitli nedenlerle farklı bir platform üzerine taşınmaları söz konusu olabilir. Bunun için, ilk kez geliştirildikleri platform sürücülerini ile aralarında bir soyutlamanın olması gerekir. Bu soyutlama ile beraber, farklı bir platform söz konusu olduğunda, sadece soyutlama katmanı üzerinde çalışılacak, ana katmandaki modüller değiştirilmeyecektir. Platform değişikliğine neden olabilecek gelişmeler arasında; yeni üretilmiş bir MPEG çözücünün fiyatının daha düşük olması, daha hızlı bir işlemciye gereksinim duyulması, kullanılan RF ayarlayıcının performansının beğenilmemesi, vb sayılabilir.

Uygulama katmanında, sayısal yayıncılıkla beraber, sayısız uygulamadan bahsedilebilir. Bunlardan başlıcaları; altyazı uygulaması, haftalık program rehberi (EPG), yazılım güncelleme, oyunlar, vb. uygulamalardır.

Uygulamalar arasında en yaygın olanlarından biri, EPG uygulamasıdır ve bir örneği Şekil 2.6.a'da gösterilmektedir. Uygulama, programların yaklaşık bir haftalık verilerini TS yapısına gömülen tablo bölümlerini toplayarak elde etmektedir. Bunun için yazılım ana katmanında bulunan tablolardan sorumlu modülü kullanmaktadır. Daha sonra bu verileri uygun olarak sınıflayarak, kullanıcıdan ilgili kumanda tuşuna basmasını beklemektedir. Kullanıcı rehberi görmek istediğini, tuşa basarak sisteme bildirdiğinde, uygulama bu sefer, grafik modülünü kullanarak, bilgileri grafik arayüzünde kullanıcıya sunmaktadır.



(a) EPG uygulaması



(b) Altyazı uygulaması

Şekil 2.6: Uygulama örnekleri

Altyazı uygulaması da benzer şekilde, altyazı PES'lerini TS ayrıştırıcı sürücülerini kullanarak alıp çözmekte ve de ilgili kullanıcı isteği geldiğinde de yine grafik arayüzünü kullanarak, Şekil 2.6.b'ye benzer şekilde ekrana çıkarmaktadır.

### 3. ‘CANLI İÇERİK’ KAVRAMI VE TASARIMI

Bu bölümde, bu tezin kavram olarak ortaya çıkardığı ‘canlı içerik’ kavramının ne olduğu, neden ve nasıl oluşturulduğu gibi soruların yanıtları açıklanmaya çalışılmıştır.

Bölüm II’de genel olarak çalışma prensibi tanıtılan PVR’ların, geleneksel televizyon seyretme alışkanlığını köklü bir şekilde değiştireceği bir gerçektir. Öyleki; televizyon seyircileri, artık kendi kaydettikleri ve üzerlerinde kes/kopyala/yapıştır işlemleri ile istedikleri değiştirmeleri yaptığı sabit diskteki yayınlar ile canlı olarak aldıkları yayınları karıştırarak, kendilerine özgü yayın akışları yaratabilir ve bunları seyredebilir [6].

Kullanıcının bir kaydının (dizi film, spor karşılaşması, sinema, vb.) ortalama olarak bir saat süreceği düşünülürse, kayıttan oynatma zaman dilimi içerisinde kullanıcı, canlı olarak yayınlanan bütün içeriklerden yoksun kalacaktır. Kullanıcı her ne kadar kaydını, içerik olarak canlı yayınlanan hiçbir programı izlemediği bir zaman diliminde seyretse bile, ‘son dakika gelişmeleri’ gibi önceden kimsenin hesaplayamayacağı içerikler söz konusudur ve bunlardan haberdar olmak isteyebilir.

Kayıttan oynatma yapan bir kullanıcı, ayrıca, bu kayıt üzerinde kes/kopyala/yapıştır işlemleri yapabileceği için özellikle reklam gibi ara içerikleri silebilir. Bu, özellikle kayıtlarını aylarca saklayacak kullanıcılar için çok cazip bir uygulamadır. Yeniden oynatma esnasında, aylar öncesinin reklamlarının hiçbir al-benisi kalmamıştır. Bu durumda, tekrar oynatma esnasında, bazı kullanıcılar hiç değilse güncel reklamları görme gibi bir lüksleri bulunduğunu bilseler, böyle bir uygulamayı da tercih edebilirler.

Son dakika gelişmeleri, güncel reklamlar gibi bazı duyurular da kullanıcıya PVR’larında yeniden oynatma yaptıkları esnasında sunulabilir. Bu tez kapsamında; PVR kullanıcılarına, tekrar oynatma esnasında, canlı yayın içerisinden sunulabilecek güncel bu içerikler, canlı içerik olarak adlandırılmıştır. Kullanıcılar, kendi tercihleri konusunda bu içerikleri aktifleştirebilir, engelleyebilir ya da bir kısmını filtreleyebilir.

### 3.1 PSI / SI Tabloları

Canlı içerikleri, sayısal yayın ile birlikte sunabilmek için, onları TS'in içine gömebilmek gerekir. TS'in içine bazı bilgilerin tablo olarak hazırlanıp bölüm yapısında gömüldüklerine Bölüm II'de değinilmiştir. O halde, canlı içerikler de, kullanıcıları kayıtlarını seyredirken onları bilgilendirmek amacı taşıdığından, bilgi tabloları yapısında düşünülebilir ve tablo bölümleri olarak tasarlanabilir.

Canlı içerik tablo tasarımına geçilmeden önce, öncelikle MPEG-2 Sistem sayısal yayıncılık standardının [5] tanımladığı TS bilgi tabloları (PSI) ile DVB<sup>1</sup> standardının [10] tanımladığı servisler hakkında bilgiler içeren (SI) tablolarının yapıları incelenecektir.

MPEG-2 Sistem standardı dört adet tablo tanımlamıştır: *program<sup>2</sup> ilişki tablosu* (PAT), *program eşleştirme tablosu* (PMT), *koşullu erişim tablosu* (CAT) ve de *şebeke bilgi tablosu* (NIT). Bunlardan sadece, isteğe bağlı NIT hariç, diğerlerinin yapısını da tanımlamıştır ve bu tablolar zorunludur. Eksik veya hatalı olmaları durumunda, alıcı TS'i doğru çözemez.

Bütün tabloların *bölüm* diye adlandırılan yapılardan meydana geldiğine değinilmiştir. Özel bölümler dışında, bölümler sırası ile 8 B'lık başlık, veri ve de 32 b'lık CRC'den oluşur. Başlık yapısı Tablo 3.1'de gösterilmektedir. Bütün PSI tabloları genel bölüm dizilimine sahiptir, bu nedenle *bölüm\_dizilim\_belirteci* ikil değerleri '1'dir.

**Tablo 3.1:** Tablo bölümlerinin başlığı

Dizilim	İkil Sayısı
<b>tablo_no</b> ( <i>table_id</i> )	8
<b>bölüm_dizilim_belirteci</b> ( <i>section_syntax_indicator</i> )	1
0	1
ayrılmış ( <i>reserved</i> )	2
<b>bölüm_uzunluğu</b> ( <i>section_length</i> )	12
<b>tablo_uzantısı</b> ( <i>table_extension</i> )	16
ayrılmış ( <i>reserved</i> )	2
<b>versiyon_numarası</b> ( <i>version_number</i> )	5
<b>şimdiki_sonraki_belirteci</b> ( <i>current_next_indicator</i> )	1
<b>bölüm_numarası</b> ( <i>section_number</i> )	8
<b>son_bölüm_numarası</b> ( <i>last_section_number</i> )	8

<sup>1</sup> DVB projesi, Avrupa merkezli oluşturulmuş, sayısal yayıncılığın uygulamasına dönük bir standartlar topluluğudur. Kısa sürede başarılı olmuş ve Avrupa dışında da benimsenmiştir.

<sup>2</sup> Program kavramı, sayısal yayıncılıkta servis olarak kullanılan kavrama denk düşmektedir.

Yapısı Ek C’de gösterilmiş olan PAT, TS’in ana giriş tablosudur. PAT bölümlerinin *tablo\_no*’su ve de PID değeri 0’dır ve bu değerler sabittir. Bu tablo TS’in içinde hangi servislerle ilgili içerik bulunduğunu ve bu içeriklerin de her servis için hangi PID’li PMT tablosundan bulunacağını söyler. PAT’ın *tablo\_uzantısı* bilgisi, TS no’sunu taşır. Bir şebekede bulunan TS’lerin kendilerine özgü no’ları mevcuttur. O halde, yayın alıcı PAT’ın bu değerinden hangi TS’i alıyor olduğunu öğrenir. PAT birkaç bölümden oluşabilir ve kaç bölümden oluştuğu da *son\_bölüm\_numarası*’na bir eklenerek<sup>1</sup> hesaplanabilir. PAT başlığındaki diğer bir önemli alan ise *versiyon\_numarası*’dır. Bu numara yayının herhangi bir anında değişirse, alıcının TS ile ilgili bütün bilgileri güncellemesi gerekecektir. O zaman, alıcının sürekli PAT bölümlerinin versiyonlarını kontrol etmesi gerekir.

PMT tablosunun da *tablo\_no*’su 2’dir ve de sabittir. En temel amacı ise, TS’de bulunan herhangi bir servise ait temel bileşenlerin PID’lerini vermektir. Yapısı [Ek E](#)’de gösterilen PMT’nin *tablo\_uzantısı* bilgisinde ait olduğu servisin no’su vardır. Buradaki bilgi, tekrarlanmış bir bilgidir, çünkü PAT içerisinden PMT PID’i bulunurken, servis no (program no) bilgisi de öğrenilmektedir. Standart, PMT’nin bölüm sayısını tek olarak sınırlamıştır. O halde, PMT bölümlerinde, *bölüm\_numarası* *son\_bölüm\_numarası*’na, o da 0’a eşit olmalıdır. PMT’nin versiyonu da ilişkili olduğu servis açısından önemlidir. Yeni bir versiyon PMT’nin yayınlanması durumunda, alıcının o servisle ilgili bilgilerini güncellemesi gerekir.

Sayısal yayın alıcı, bir TS’i almaya başladığı ilk an, PAT bölümlerini toplamaya başlamalıdır. PAT içeriğini ayıkladığı zaman Tablo 3.2’ye benzer bir yapı ile karşılaşır. Bu içeriğin anlamı şudur; ilgili TS’de 3 adet servis bulunmaktadır ve de *şebeke\_bilgi\_tablosu* (NIT) da, zorunlu olmadığı halde, TS içerisine eklenmiştir. NIT’in TS’e eklendiği, servisler arasında no’su 0 olan bir servisin bulunmasından anlaşılır. Her servisin içeriklerine (A, V ya da D) erişebilmek için, servislere özgü yönlendirme tablo (PMT) PID’leri ise sıralandığı gibidir.

**Tablo 3.2:** PAT içeriği

Servis No	PMT PID
0	0xXYJ
1. Servis Numarası	0xXYK
2. Servis Numarası	0xXYL
3. Servis Numarası	0xXYM

<sup>1</sup> Bölümler numaralandırılırken 0’dan başlanır. Bu nedenle;  $bölüm\_sayısı = son\_bölüm\_numarası + 1$

Alıcı ilk açıldığı zaman TS’de bulunan ilk servisi televizyonda göstermek isterse, PAT’tan okuyacağı ilk servisin PMT PID’ine göre, PMT tablosunu oluşturmaya başlayacaktır. Daha sonra bu tabloyu bileşenlerine ayrıştırarak, Tablo 3.3’tekinе benzer PID’lere ulaşır. Alıcının amacı, televizyona ilk etapta, görüntü ve ses vermek olduğu için, ilk servisin görüntü ve de ses PID’lerini okuyarak, TS ayrıştırıcıya bu değerleri yollayıp PES’leri oluşturmasını sağlar. Bu PES’ler daha sonra MPEG çözücünden geçirilerek, televizyona görüntü ve ses iletilmiş olur.

**Tablo 3.3: PMT içeriği**

İçerik Tipi	ES PID
Görüntü	0xXYZ
Ses (Almanca)	0xYZX
Ses (İngilizce)	0xZXY

MPEG çözücünün ses ve görüntü eşzamanlılığını sağlayabilmesi için PCR bilgisine ihtiyacı olduğuna değinilmişti. Bu bilginin de TS içinde paket yapısında geldiği biliniyor. Alıcının, çözücüye PCR bilgisini verebilmesi için, TS ayrıştırıcıdan bu bilgiyi alması gerekir. Bunu da, yine PMT’den çıkaracağı PCR PID değerini, TS ayrıştırıcıya yollayarak yapar. Tablo 3.3’te gösterildiği üzere; ses, görüntü ve de veri bilgileri temel akış (ES) olarak adlandırılmışlardır. PMT’si çözülen servisin, bir adet görüntü ES’i, Almanca ve İngilizce dillerinde olmak üzere de iki adet ses ES’i bulunmaktadır. Alıcı, kullanıcının yaptığı dil ayarlarına göre, öntanımlı sesi televizyona göndermek durumundadır. Kullanıcı istediği zaman, sayısal alıcının ayar menüleri aracılığı ile dil seçimini değiştirebilir.

PSI tabloları sadece TS genelinde bilgiler içerirken, SI tabloları servislere özgü detay bilgiler içerirler. DVB standardının, yapıları ile beraber tanımladığı dört adet zorunlu SI tablosu bulunmaktadır: *şebeke\_bilgi\_tablosu* (NIT), *servis\_açıklama\_tablosu* (SDT), *olay\_bilgi\_tablosu* (EIT) ve *zaman\_tarih\_tablosu* (TDT).

NIT, TS’in içinde yayınlandığı şebeke ile ilgili bilgiler içerir ve de alıcılar tarafından pek kullanılmaz. SDT ise TS’de bulunan servislerin isimleri başta olmak üzere, servis sağlayıcı benzeri bilgileri de içerir. Örneğin, sadece PSI bilgileri ile yetinilse idi, bir servisin sadece no’su bilinecek, bunun dışında ismi dahi bilinemeyecekti. Alıcı, servisleri listeleyeceği vakit, sadece kullanıcı için anlamsız numaraları sıralayacaktı. Halbuki SDT tablosundan, kullanıcı için anlamsız olan bu numaraların isim karşılıkları bulunabilir. Örneğin, İngiltere’de yayın yapan 0x1041 servis no’lu servisin isminin BBC1 olduğu SDT’den çıkmaktadır. Bunun yanında, bu servisi sağlayan kuruluşun bilgisi de BBC diye çıkar. SDT tablosunun da PID değeri (0x11) ve *tablo\_no* değeri (0x42) sabittir.



DVB'nin tanımladığı ve zorunlu olan diğer bir tablo da EIT tablosudur. Bu tablonun da PID değeri (0x12) ve *tablo\_no* değeri (0x4E) sabittir. Bu tablonun asıl amacı, ilgili servisin programları düzeyinde bilgiler vermektir. Sunulan bilgiler başlıca; programların tarih ve saat olarak ne zaman başlayıp ne zaman biteceği, ne kadar süreceği ve de konusunun ne olduğudur. EIT tablosunun hangi servise ait bilgi taşıdığı, bölüm başlıklarında bulunan *tablo\_uzantısı* bilgisinden bulunur, bu değer servis no'sunu taşır.

EIT tablosunun birkaç çeşidi söz konusudur. Bunlardan zorunlu olanı, sadece o an oynayan program ve bir sonraki program ile ilgili bilgiler içerirken, isteğe bağlı olarak yayınlanan başka bir çeşidi de, ilgili servisin bir haftalık ya da 8 günlük bütün programları hakkında bilgiler sunar. Bu tablolarda, program açıklamaları da daha uzun olabilir. Öyleki, oynayan ya da oynayacak olan bir sinema filmine dair, oyuncu listesi ve diğer yapımcı kadrosunun isimleri dahi EIT ile taşınabilir.

EIT tablosunun EPG uygulamasının belkemiğini oluşturduğuna Bölüm II'de yazılım kısmında değinilmişti. EPG üzerinden zamanlayıcı kurma ya da kaydetme listeleri oluşturma, vb. kullanıcıların en sık kullandığı uygulamalar, EIT tablosundan çıkan bilgilere dayanır.

Üzerinde durulması gereken diğer bir tabloda TDT'dir. Bu tablo da zorunlu olarak yayınlanmaktadır. Amacı, alıcının saati ve tarihinin yayınıki ile eşzamanlı çalışmasını sağlamaktır. Yayın zamanı UTC formatında kodlanmıştır. Yalnız TDT, Tablo 3.4'te görüldüğü gibi, özel yapıdadır, yani genel bölüm yapısındaki gibi başlık, veri ve CRC'den oluşmaz. Bu nedenle *bölüm\_dizilimi\_belirteç* değeri 0'a eşittir. Tek bir bölümden oluşacağı ve bölüm uzunluğunun da sabit olacağı bellidir. TDT'nin PID değeri (0x14) ile *tablo\_no*'su (0x70) sabittir.

**Tablo 3.4:** TDT bölüm yapısı

Dizilim	İkil Sayısı
zaman_tarih_bölümü( {	
<i>tablo_no</i>	8
<i>bölüm_dizilimi_belirteci</i> = 0	1
ayrılmış	1
ayrılmış	2
<i>bölüm_uzunluğu</i>	12
UTC_zamanı	40
}	

### 3.2 Canlı İçerik Tablosu

Tasarlanacak olan Canlı İçerik Tablosu için, zorunlu PSI/SI tablolarının yapısını örnek almak, tasarımı kolaylaştırmakla beraber, tasarlanacak tabloların da kullanım alan ve kolaylığını arttıracaktır. O halde, tasarım kriterlerinden ilki, canlı içeriklerle ilgili oluşturulacak tabloların bölüm diziliminin genel bölüm dizilimine uyması olmalıdır.

Canlı içeriklerin güncel reklam, son dakika gelişmesi ve de duyuru gibi içerikler olabileceğine değinilmişti ama bu içeriklerin yapısına değinilmemişti. Yapıdan kastedilen, içeriğin karakter tabanlı mı, grafik tabanlı mı yoksa daha farklı mı olacaktır. Bu durumda, birden fazla içerik yapısı söz konusudur ve bu içerik yapılarının ileride artabileceği de düşünülmelidir. Örneğin, ileriki bir süreçte çoklu ortam nesnelere de canlı içerik olarak TS'e gömülebilir. Bu nedenle, tablo yapısının esnek olması gerekir ki; bu da bir diğer tasarım kriteridir. Esnek tablo yapısı için, dizilim olarak *açıklayıcılar* kullanmak, her iki standartta (MPEG-2 Sistem ve DVB) da uygulanan bir yöntemdir.

Genel bir açıklayıcının yapısı Tablo 3.5'te görüleceği üzere oldukça yalındır. Birinci *etiket* alanı açıklayıcının tipini belirtir. Önceki paragrafta adı geçen canlı içerik yapıları farklı tipte olabilirler. Örneğin; karakter tabanlı bir içerik hazırlanacak olursa, tip olarak *başlık* veya *altbaşlık* benzeri tipler söz konusu olabilir. Bütün bu tipler, *etiket* alanı içinde kodlanmaya elverişlidir.

**Tablo 3.5:** Genel açıklayıcı yapısı

Dizilim	İkil Sayısı
canlı_içerik_açıklayıcı() {	
etiket	8
uzunluk	8
değer	
}	

İkinci alan *uzunluk* ise açıklayıcının toplam verisinin kaç bayt olduğunu göstermektedir. Buradan çıkarılacak sonuç, uzunluğun azami değerinin 0xFF (= 255) olacaktır. Grafik tabanlı içerikler dışında, bu uzunluk uygundur, yalnız sıkıştırılmamış grafik nesnelere için açıklayıcılar dışında yöntemler daha verimli olabilir.

Son *değer* alanında ise, içerik doğrudan bulunabilir. Tasarımda kullanılan içeriklerin açıklayıcılar ile tasarımına Bölüm 3.2.2'de daha ayrıntılı değinilecektir.

### 3.2.1 Tablo Yapısı

Tasarım kriterlerinden ilki, bu tablonun genel bölüm yapısına uygun olması idi. Bu durumda, genel format; sıra ile, 8 **B** başlık, asıl bilginin taşındığı gövde ve de 32 **b** CRC şeklinde olacaktır. Gövde kısmının da, ikinci tasarım kriteri gereği, *açıklayıcı*lardan oluşması gerekiyordu. Öyleyse; daha genelleştirilecek olursa, tablonun bölüm yapısı;

$$bölüm = başlık + n * açıklayıcı + CRC (n = açıklayıcı sayısı)$$

şeklinde olacaktır. Bölüm dizilimi Tablo 3.6'da gösterilmiştir.

**Tablo 3.6:** Canlı İçerik Tablosu bölüm yapısı

Dizilim	İkil Sayısı
canlı_içerik_bölümü() {	
tablo_no	8
bölüm_dizilimi_belirtecisi	1
0	1
ayrılmış	2
bölüm_uzunluğu	12
servis_numarası	16
ayrılmış	2
versiyon_numarası	5
şimdiki_sonraki_belirtecisi	1
bölüm_numarası	8
son_bölüm_numarası	8
for (i=0; i<N; i++) {	
açıklayıcı()	
}	
CRC_32	32
}	

Canlı İçerik bölümünün her bir alt-alanının görevi aşağıda listelenmiştir:

- *tablo\_no*: MPEG-2 Sistem ve DVB tablolarının PID'leri sabittir. Ama PID değerleri ile tablolar 1-1 örtüşmez. Şöyleki; bir PID değeri birden fazla tabloya verilebilir. Örnek olarak, DVB SI talolarından TDT ve TOT aynı PID değerine sahiptir. Bu iki tablo, TS'den filtrelendikten sonra, *tablo\_no*'larına göre ayırt edilirler. Ayrıca, bir tablonun farklı çeşitleri söz konusu olabilir. Buna örnek olarak EIT tablosu verilebilir. Zorunlu olarak TS'de bulunması gereken EIT ile 8-günlük EPG bilgisi taşıyan EIT tabloları da *tablo\_no* olarak birbirlerinden ayırt edilirler. Standardın gelecekte kullanım için ayırdığı atanmamış *tablo\_no*'lar söz konusudur ve bunlardan bir tanesi Canlı İçerik tablosu için kullanılabilir.

- *bölüm\_dizilimi\_belirteci*: Bu ikilin, bir tablonun genel bölüm yapısında tasarlanıp tasarlanmadığını belirttiğine önceden değinilmişti. O halde, Canlı İçerik tablo bölümleri kodlanırken, bu ikil değeri '1' seçilmelidir, çünkü tasarımıımızın temel kriterlerinden ilki Canlı İçerik tablolarını genel bölüm yapısında tasarlamak idi.
- *bölüm\_uzunluğu*: Bu değer, kendinden sonra gelen başlık baytları ile gövde ve de CRC baytlarının toplamını verir. TS paketlerinden bölüm yapıları oluşturulurken bu uzunluk kullanılır. Genelde bu uzunluk kadar bir tampon bölge bellekte oluşturulur ve paketler başlıklarından ayrıştırıldıktan sonra buraya kopyalanırlar. Tampon bellek dolana kadar bu kopyalama işlemi devam eder. Bellek dolduğunda, son alınmış 4 bayt CRC baytlarıdır. Bütün tampon, CRC denetimini doğru geçer ise; bölüm TS'den doğru bir şekilde alınmıştır demek. Aksi takdirde, tampon alanın temizlenmesi ve yeni bir bölümün oluşturulması gerekir.
- *servis\_no ya da program\_numarası*: Genel bölüm başlık yapısı Tablo 3.1'de verilirken, bu alan *tablo\_uzantısı* olarak genel isimlendirilmişti. Tablodan tabloya buraya konulan değer değişmektedir. Canlı İçerik tasarımında da, bu alan, canlı içeriğin hangi servise ait olduğunu belirtmek için *servis\_no* değerine atanmıştır.
- *versiyon\_numarası*: Canlı içerikler anlık oldukları için, belli bir anda tek bir versiyona sahiptirler. Eğer bir servis için yeni bir canlı içerik TS'e eklenecekse, versiyonu bir arttırılmalıdır. Örneğin; canlı içerik olarak güncel bir reklam yayınlayan bir servis, ani gelişen bir haberi yayınlamak istediği vakit, reklam içeriğinde kullandığı versiyon numarasını arttırmalıdır ki; alıcı taraf yeni bir içerik yollandığını anlayabilsin.
- *şimdiki\_sonraki\_belirteci*: Bu ikil değerinin pek bir uygulama alanı yoktur. Alıcı taraf genellikle 'şimdiki' değeri gösteren yani ikilin değerinin '1' olduğu bölümlere odaklanır. Bu değer '0' olması, toplanan bölümün o an için geçerli olmadığını gösterir.
- *bölüm\_numarası*: Uzunluk değeri 0xFFF'den fazla olan bir Canlı İçerik tablosu birden fazla bölümden oluşmak zorundadır. Bu durumda, ilk bölüm numarası 0 olacak şekilde numaralandırılır.
- *son\_bölüm\_numarası*: Alıcının bir tabloyu oluşturabilmesi için onun kaç bölümden oluştuğunu bilmesi gerekir. Bu nedenle, her bölüm başlığına *son\_bölüm\_numarası* mutlaka eklenir.
- *CRC\_32*: Alınan bölümün doğrulanması için CRC denetimi baytları sona eklenir.

### 3.2.2 Canlı İçerik Açıklayıcıları

Bu tez çalışmasında; canlı içerik sunum nesnesi olarak; metin, grafik ve de çokluortam nesnelere için açıklayıcılar tasarlanmıştır. Burada amaç; kullanıcıya PVR’ında kayıt seyrederken, hem de metinsel içeriğin, hem grafiksel bir içeriğin, hem de görüntü ve ses gibi çokluortamsal bir içeriğin sunulabileceğini göstermektir.

#### 3.2.2.1 Metin Tabanlı İçerikler

Metin tabanlı içeriklerde kullanıcıya sadece, canlı içerikler yazılı olarak sunulur. Bir metnin ana öğeleri başlık, alt-başlık ve de metin gövdesi olarak ayrı ayrı açıklayıcı şeklinde tablo bölümlerine kopyalanabilir.

- **Başlık:** Canlı İçerik ister son dakika gelişmesi, ister güncel reklam, isterse de duyuru olsun; başlık alanına sahip olması, sunum esnasında bu alanın, kullanıcının dikkatini çekmeye yönelik, daha vurgulu yapılmasına yarayabilir. Açıklayıcılar, sadece içerik taşımaktalar; yani sunum esnasında alacakları görüntü sayısal alıcının denetimindedir.
- **Alt-başlık:** Bu alan, daha çok isteğe bağlı kullanılabilir ve başlık alanının bir devamı olarak düşünülebilir. Metin gövdesine sığdırılmayan bazı yazılar için de bu alan kullanılabilir.
- **Metin Gövdesi:** Ana metnin kopyalanacağı açıklayıcı budur. Metnin uzunluğu kullanılacak karakter tablosundan bağımsız olarak, açıklayıcının uzunluk alanı ile sınırlıdır. Bu değer en fazla 255’tir, bu da tek baytlık karakter tablosu kullanan sistemler için, metin gövdesinin en fazla 255 karakterden olacağı anlamına gelir.

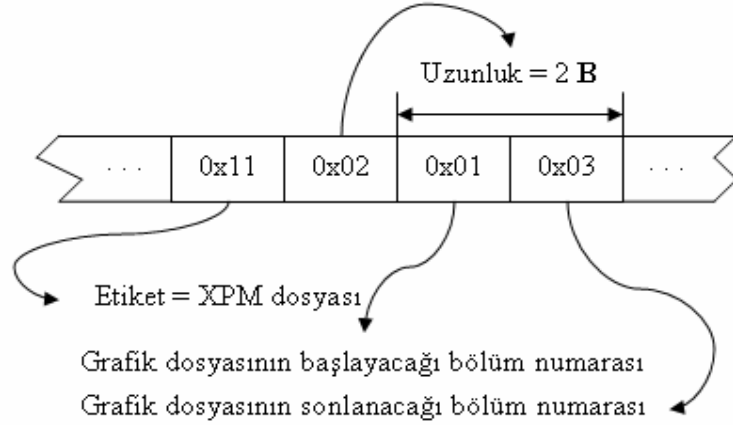
**Tablo 3.7:** Canlı İçerik nesne tiplerine atanmış açıklayıcı etiketleri

<b>İçerik Nesnesi</b>	<b>Nesne Tipi</b>	<b>Açıklayıcı Etiketi</b>
METİN	Ayrılmış	0x00
	Başlık	0x01
	Alt-başlık	0x02
	Metin Gövdesi	0x03
	Ayrılmış	0x04 – 0x0F
GRAFİK	Ayrılmış	0x10
	XPM dosyası	0x11
	Ayrılmış	0x12 – 0x1F
ÇOKLUORTAM	Ayrılmış	0x20
	Video PID	0x21
	Ayrılmış	0x22 – 0x2F
-	Ayrılmış	0x30 – 0xFF

### 3.2.2.2 Grafik Tabanlı İçerikler

Sıkıştırılmış grafik dosyaları bile açıklayıcıların değer alanına sığmayacaklarından, bu içeriklerin bölüm yapısına eklenmesi için farklı bir yöntem seçilmiştir. Bu yöntemde göre; canlı içerik sadece bir adet grafik dosyası taşıyabilecek ve dosyanın sıralanmış baytları, yeni bir bölümün gövdesinin ilk baytından başlanarak bölümlere ayrıştırılacaktır.

Bu durumda, açıklayıcının değer alanı 2 bayttan oluşacak; ilk bayt dosya verisinin başladığı bölüm numarasını, ikinci bayt ise bittiği bölüm numarasını gösterecektir. Şekil 3.1’de örnek değerlerle bu durum görselleştirilmiştir. Şekle göre, grafik nesnesi, XPM formatında bir dosyadan oluşmakta ve de birinci, ikinci ve de üçüncü bölümleri kapsamakta.



Şekil 3.1: Grafik nesnesi açılacağı yerleşimi

### 3.2.2.3 Çokluortam Tabanlı İçerikler

Çokluortamdan kasıt, kullanıcıya canlı içeriği metin ya da grafik nesnesi dışında görüntü ya da ses formatında sunabilmektir. Yöntem olarak da, kullanıcının TS’de var olan bir A ya da V ES’ine bağlanması tasarlanmıştır. Bu yöntem, şöyle de düşünülebilir; kullanıcı PVR’ında kayıtlı bir yayını seyredirken, bir son dakika gelişmesinden metin tabanlı bir Canlı İçerik tablosu ile haberdar olup, alıcının yönlendirmesi ile gelişmeyi canlı olarak yayınlayan servise bağlanıyor. Alıcı bu bağlanma işini, Canlı İçerik içinden ayıklayacağı açıklayıcılardan bulacağı PID değerleri ile yapabilecektir.

Bu durumda, açıklayıcının değer alanına A ya da V PID’leri girilecektir. PID uzunluğu 13 b olduğundan, bu tip açıklayıcıların uzunluğu 2 bayt olacaktır.

## 4. ‘CANLI İÇERİK’ UYGULAMASI

Canlı İçerik olgusunun uygulamaya dönüştürülmesi için, temel iki farklı uygulama yazılımı sözkonusudur. Bunlardan ilki, yayıncı tarafında çalışacak olan yazılımdır ve temel olarak, yayıncının ‘canlı içerik’leri oluşturmasını sağlayacaktır. Diğeri ise, sayısal yayın kaydedici yani kullanıcı tarafında çalışacak olan yazılımdır ve kaydedicinin kayıttan oynatma yaptığı esnada, ‘canlı içerik’ takibi yapmasını sağlayacaktır. Bu yazılım, kaydedicinin ana yazılımına eklenecek bir modül olarak tasarlanabilir.

Bahsi geçen uygulamalar, tez çalışması kapsamında, kişisel bilgisayar ortamında geliştirilmiştir. Bundan dolayı, yayın ortamı ve televizyon sistemi de kişisel bilgisayar ortamında canlandırılmaya çalışılmıştır. Yayıncı tarafında çalışacak yazılıma, yayın ortamının yazılımla benzetimi olduğu için ‘software’; yayın tarafında olduğu için de ‘head-end’ İngilizce sözcüklerinin birleşimi olan Software Head-End (SoftHE) adı verilmiştir. Benzer şekilde, kaydedici tarafında çalışacak yazılıma da Software Television (SoftTV) adı verilmiştir.

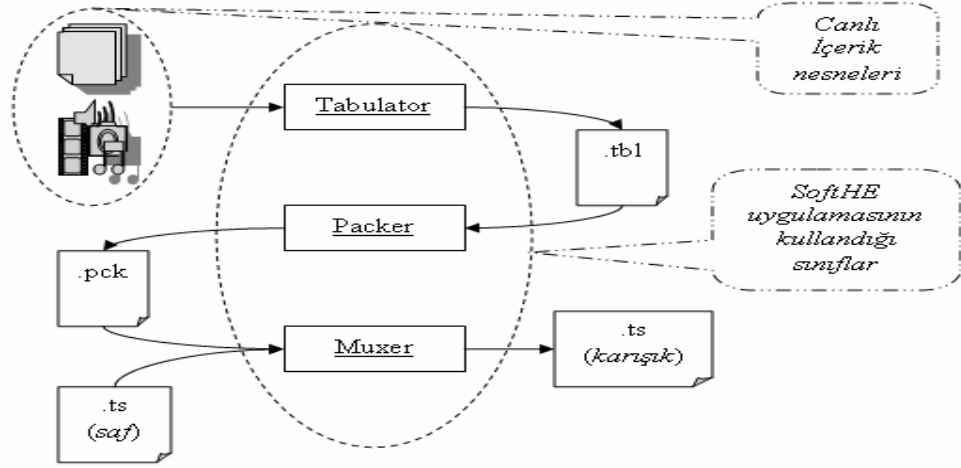
### 4.1 SoftHE Yazılımı

SoftHE yazılımı; işletim sistemi bağımsızlığı, kullanıcı arayüzünün kolay geliştirilebilirliği ve kolay erişilebilen, zengin kütüphaneleri içermesi göz önüne alınarak, Java teknolojisi ile geliştirildi. Şekil 4.1’de görüldüğü üzere, üç temel sınıfı barındırır; Tabulator (tablolu), Packer (paketleyici) ve Muxer (harmanlayıcı) sınıfları.

Ana uygulama sınıfı; kullanıcıya sunduğu menü üzerinden, kullanıcının belli bir servise canlı içerik nesnesi eklemesini sağlamaktadır. Şekil 4.1’de resmedildiği üzere; kullanıcıdan grafik tabanlı bir form aracılığıyla alınan bu nesnelere, Tabulator sınıfının ilgili metodları ile tablo yapısına çevrilir. Tablo yapısı, *.tbl* uzantılı bir dosyada, bölümlerin baytlar dizisi şeklinde dosyaya yazılması ile tutulur.

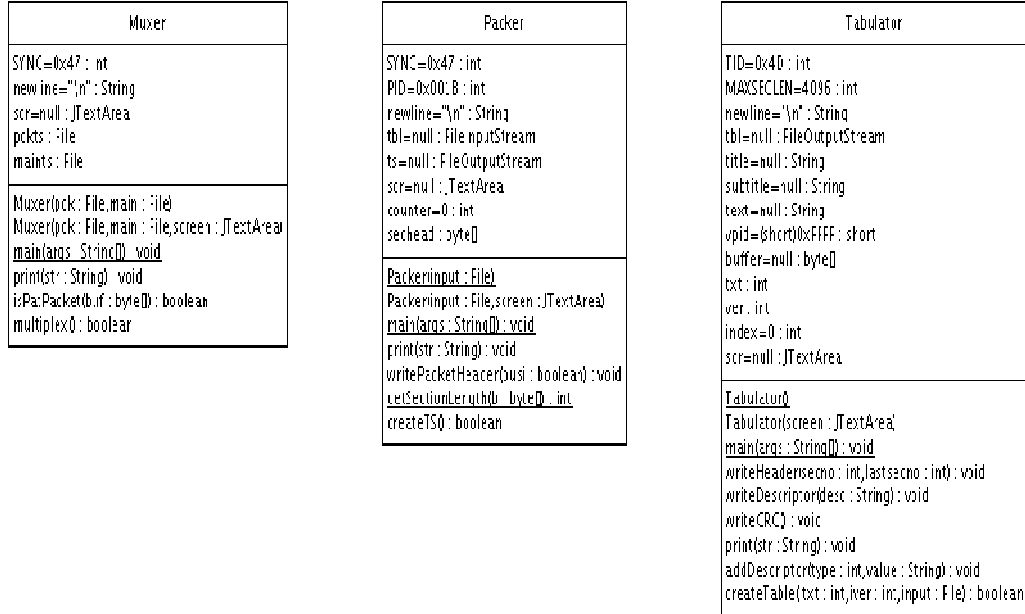
SoftHE uygulamasının diğeri bir özelliği de; tablo bölümlerini 188 B’lık TS paketlerine çevirebilmesidir. Bu işlemi yerine getirirken; ilkin, *.tbl* dosyasından

bölmeleri okuyup standarda uygun bir şekilde paketlere ayrıştırır. Sonra da, oluşan paketleri *.pck* dosyasına bayt dizisi olarak kopyalar.



Şekil 4.1: SoftHE yazılımının girdi/çıkışları

SoftHE yazılımı, Muxer sınıfı aracılığıyla, önceden bir dosyaya bayt bayt kopyalanmış bir TS'e (Şekil 4.1'de *.ts(saf)* ile gösterilen dosya), *.pck* dosyasında tutulan paketleri gömebilir. Aslında, bu işlem; TS paketleri içeren iki dosyayı harmanlamaktadır. Sonuç olarak, girdi olarak alınan *.pck* ile *.ts(saf)* dosyaları ile çıktı olarak verilen *.ts(karışık)* dosyasının yapıları aynıdır.



Şekil 4.2: SoftHE sınıflarının UML diyagramları



UML diyagramları Şekil 4.2’de verilen SoftHE yazılımının temel sınıflarının önemli sayılabilecek metot ve değişken açıklamalarına aşağıda değinilmiştir:

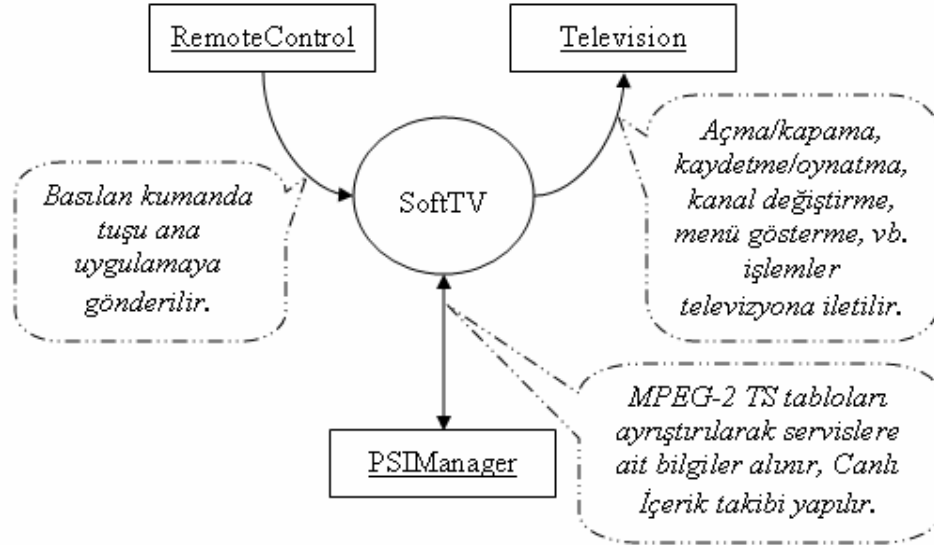
- *Tabulator* sınıfı, `addDescriptor()` metodu aracılığı ile, uygulama kapsamında tanımlanmış açıklayıcıları tutan `title`, `subtitle`, `text`, vb. değişkenlerine atama yapılmasını sağlar. Böylelikle ana uygulama form üzerinden aldığı canlı içerik nesnelarını tabloya dönüştürmesi için *Tabulator* sınıfına göndermiş olur. Bütün açıklayıcılar geçildikten sonra, oluşacak tablo bölümlerinin `tablo_uzantısı`, versiyon gibi alanlarının parametre olarak geçildiği `createTable()` metodu ile de tablo bölümlerinin oluşturularak, `.tbl` dosyasına yazılması sağlanır. Tablo yaratma işleminde, sınıf, kendine özgü (`private`) `writeHeader()`, `writeDescriptor()` ve de `writeCRC()` metotlarını kullanır. Bu sınıfın sabit olarak atanmış iki özel (`private`) değişkeni vardır. Bunlardan biri, oluşturulacak Canlı İçerik Tabloları’na tekil olacak şekilde seçilmiş `0x4D` `tablo_no` değerini tutan TID değişkenidir. Bu değişkenin değeri, hem PSI tablolarını tanımlayan TS standardına [4], hem de SI tablolarını tanımlayan DVB standardına [10] göre, gelecek kullanım için ayrılmış değerlerden seçilmiştir. Diğer bir özel değişken de, yine standartların koyduğu bir bölüm uzunluğunun alacağı azami değer olan 4096’yı tutan `MAXSECLN` değişkenidir.
- *Packer* sınıfı, `Packer()` yapıcı metodu aracılığı ile paketlere bölüneceği tablo bölümlerinin yazılı olduğu `.tbl` dosyasını öğrenir. Ana uygulama `createTS()` metodunu çağırarak, sınıfın `.pck` dosyasına bölümleri ayrıştırdığı paketleri yazmasını sağlar. Sınıf, bu metot bünyesinde, `writePacketHeader()` ve `getSectionLength()` özel metotlarını kullanarak paket başlıklarını oluşturur. Bu sınıfın da kendine özgü sabit iki değişkeni vardır. Bunlardan ilki, `SYNC` değişkenidir ve paketlerin başlangıç baytı olan `0x47` değerini tutar. Diğerisi ise, yine standartlara uygun olarak seçilmiş ve Canlı İçerik Tablo bölümlerinin paketlerine özel olarak atanmış `PID` değerini tutan `PID` değişkenidir.
- *Muxer* sınıfı, harmanlayarak birleştireceği paketleri tutan dosyaları `Muxer()` yapıcı metodu aracılığı ile alır. Bu dosyalardan biri, sadece Canlı İçerik Tablo paketlerinin tutulduğu `.pck` dosyası; diğeri ise orjinal TS kaydının tutulduğu yani bütün bir sayısal yayının paketlerinin bulunduğu `.ts` dosyasıdır. Ana uygulama bu sınıfın `multiplex()` metodunu çağırarak, yeni ve birleştirilmiş `.ts` dosyasının oluşturulmasını sağlar. Sınıf, bölüm paketlerini araya yerleştireceği yerin tespiti için `PAT` paketlerini referans olarak kullanır. Bu nedenle, `isPatPacket()` metoduna ihtiyaç duyar.

## 4.2 SoftTV Yazılımı

SoftTV yazılımı, Genel Kamu Lisansı (General Public Lisence - GPL) ile kullanılabilen ve MPEG-1, MPEG-2 görüntülerini çözen açık C kaynak kodlu bir proje olan *libmpeg2* [11] kütüphanesi kullanılarak geliştirilmiştir. Linux yazılım geliştirme araçları kullanılarak C++ programlama dilinde yazılmıştır.

SoftTV yazılımı temel üç görevi yerine getirir. İlki; TS yapısında kaydedilmiş bir sayısal yayını çözerek, kullanıcının TS içindeki servisleri izlemesini sağlamaktır. Böylece kullanıcı bu servisler arasında dolaşabilecek, istediklerini seyredebilecektir. Yazılımın ikinci görevi ise, temel PVR fonksiyonları olan kaydetme ve yeniden oynatma işlemlerini yerine getirmektir. Bu özellik sayesinde, kullanıcı, sevdiği bir servisi kaydedebilir, ileriki bir zamanda yeniden oynatabilir. Yazılımın son görevi ise, bu tezin konusu olan canlı içerik takibini yapmak ve gerektiği zamanda içeriği kullanıcıya sunmaktır.

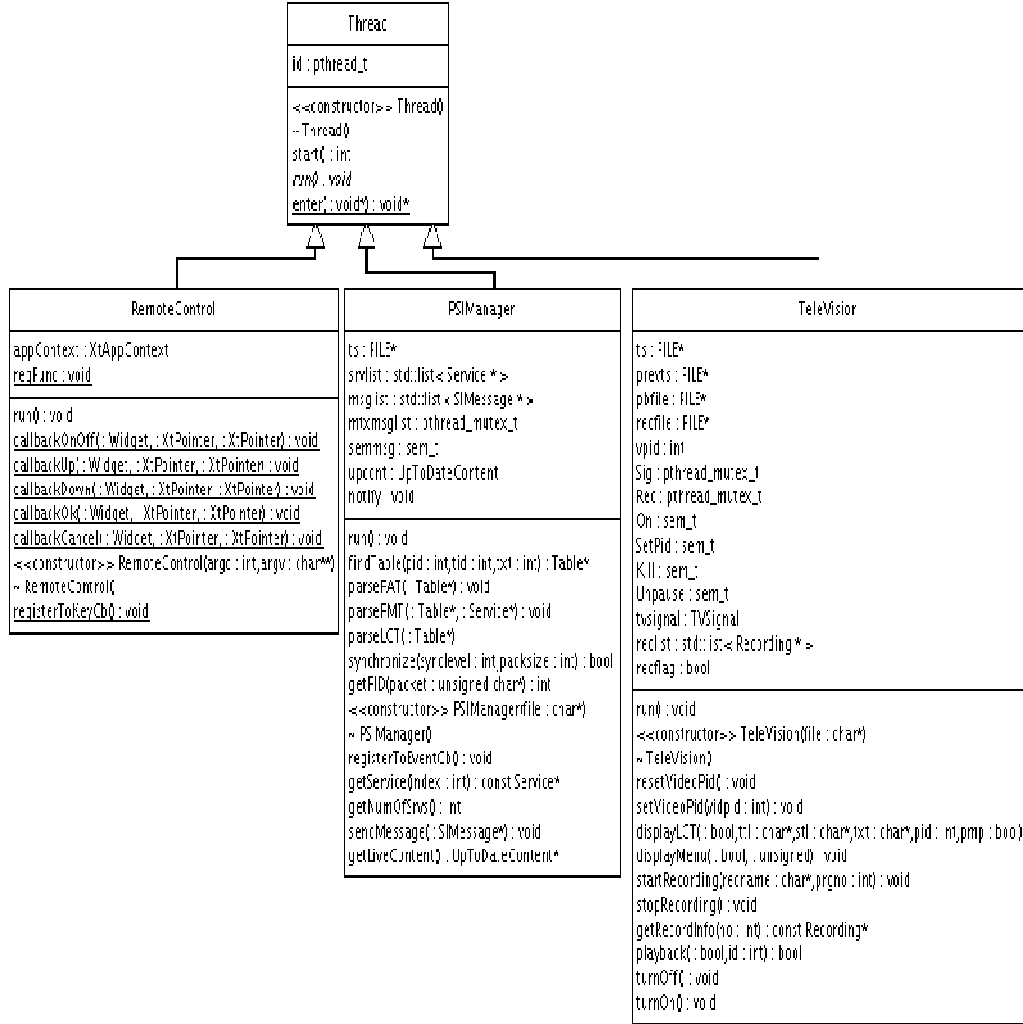
Bu görevlerin yerine getirilmesi amacıyla Şekil 4.3'te görülen üç sınıf geliştirilmiştir: RemoteControl, PSIManager ve Television sınıfları. Her bir sınıf ise kendilerine özgü çalışma metotları sayesinde sözde-paralel çalışacak şekilde tasarlanmışlardır.



Şekil 4.3: SoftTV yazılımının genel yapısı

RemoteControl, uzaktan kumanda görevini yerine getirmektedir. Ekrana çizilen tuşlar arayüzü ile kullanıcı girdilerini sisteme yollar. PSIManager, kendisine verilen parametrelere göre tablo (PAT, PMT, canlı içerik, vs.) bölümlerini yayından filtreler ve tabloları oluşturur. Ana yazılım bu tablolar aracılığı ile servis bilgilerini oluşturur.

Kullanıcıdan gelen istekler doğrultusunda ilgili servisi Television sınıfı metodlarını çağırarak ekranda oynatır. SoftTV, Television sınıfından ayrıca, PVR işlemlerini yerine getirirken de yararlanır.



Şekil 4.4: SoftTV sınıflarının UML diyagramı

UML sınıf diyagramları Şekil 4.4'te gösterilen bu ana sınıfların daha detaylı açıklamaları aşağıda listelenmiştir:

- *RemoteControl* sınıfı, bir grafik arayüz sınıfıdır. Ekrana uygulamanın ihtiyacı olan kumanda tuşlarını çizer ve uygulama boyunca kullanıcının bastığı tuşları ana uygulamaya geçirir. Bu işlemin olabilmesi için, ana uygulama bu sınıfın `registerToKeyCb()` metodunu kullanarak, tuş basıldıkça çağrılmasını istediği kendi fonksiyonunu kaydeder. Kaydedilen bu fonksiyon değeri `regFunc` özel

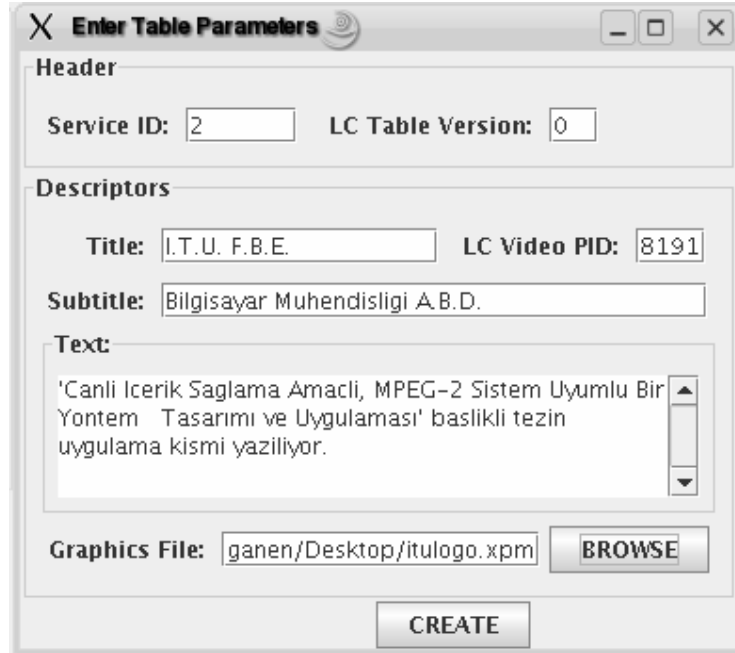
değişkeninde tutulur. Böylece, her tuşa basıldığında, basılan tuş bilgisi parametresi ile RemoteControl, kendisine kaydedilen bu fonksiyonu çağırır.

- *PSIManager* sınıfı, sayısal yayını *.ts* dosyasından paket paket okuyarak, tablo bölümlerini bulur ve bu bölümleri birleştirerek sayısal yayın tablolarını oluşturur. Kendisine *findTable()* ile geçirilen parametrelere uygun olarak istenilen bütün tabloları TS'de varoldukça oluşturabilir. Uygulama, üç tabloya ihtiyaç duyar; PAT, PMT ve Canlı İçerik Tablosu (Live Content Table - LCT). Bu tablolar oluşturulduktan sonra, yine bu sınıfın ilgili arayüzleri ile ayrıştırılır, bu arayüzler sırası ile; *parsePAT()*, *parsePMT()* ve *parseLCT()*'dir. PAT ayrıştırılarak, TS'de kaç servis olduğu, bu servislerin no'ları ve de PMT PID değerlerine ulaşılır. Bütün bu değerler bir servisi temsil ederler ve *svrlist* liste değişkeninde tutulurlar. Uygulama servis bilgilerine ihtiyaç duyduktan sonra, *getService()* arayüzü ile bunlara ulaşabilir. Uygulama PAT ayrıştırıldıktan sonra, her servisin PMT tablolarının yine *findTable()* ile bulunmasını ister. Oluşturulan her tablo, uygulamaya, sınıfın *notify* değişkenine *registerToEvent()* metodu ile kaydedilen fonksiyon çağrılarak haber verilir. Oluşturulan PMT'ler, ayrıştırıldıktan sonra, servis V PID'lerine ulaşılır ve bunlarla servis listesi güncellenir. Uygulama, LCT takibini ise, kullanıcı kaydettiği bir yayını yeniden oynatmaya başlayınca başlatır. LCT oluşturulduktan sonra, diğer tablolar gibi ayrıştırma işlemini yapar ve kaydı durdurup canlı içeriği kullanıcıya göstermek için *getLiveContent()* metodunu çağırarak gerekli içeriği alır.
- *Television* sınıfı, MPEG-2 görüntülerini ekrana çizmek için kullanılan *libmpeg2* kütüphanesine arayüz sağlar. *turnOn()* ve *turnOff()* metotları aracılığıyla, görüntünün ekrana basılmasını ve ekrandan yok edilmesini sağlar. *setVideoPid()* ile hangi servisin görüntüsünün basılacağını; *resetVideoPid()* ile de basımının durdurulacağını ilgili kütüphaneye bildirir. PVR fonksiyonlarını sağlamak üzere ise kayıt işlemleri için *startRecording()* ve *stopRecording()*; oynatma için de *playback()* arayüzlerini içerir. Başlatılan her kaydı, *reclist* özel liste değişkeninde tutar. Ana uygulama, bu listeye erişim için *getRecordInfo()* metodunu çağırır. Oynatma işleminin başlatılıp durdurulması *playback()* metoduna gönderilen parametrelerle ayarlanır. Uygulama menünün gösterilmesi için de *displayMenu()* metodunu çağırır, ancak menü üzerinde dolaşımı ayarlamak uygulamanın kontrolündedir; bu sınıf sadece, menüyü V görüntüsü ile beraber ekrana çizmekle yükümlüdür. Uygulama, yeniden oynatma esnasında, herhangi bir canlı içeriğe rastlarsa, *displayLCT()* metodunu kullanarak, *Television* sınıfının oynatmayı durdururup, canlı içeriğin sunması sağlanır.

### 4.3 Örnek Kullanım Senaryosu

Bu bölümde SoftHE ve SoftTV yazılımları kullanılarak, bir Canlı İçerik senaryosu uygulanmaya çalışılmıştır. Uygulama, TRT'nin 2006 yılı içinde başlattığı DVB-T test yayınının [12] kaydı üzerinden oluşturulmuştur. Senaryo kapsamında, kullanıcı SoftTV uygulamasını kullanarak, sayısal yayın servislerini dolaşacak, beğendiği bir servisi kaydedecek ve bir müddet sonra kaydetmiş olduğu servisi yeniden oynatacaktır. Yeniden oynatma esnasında, kaydetmiş olduğu servis ile ilgili canlı bir içeriğe yayın içerisinde rastlanırsa, kayıttan oynama durdurulacak ve canlı içerik ekrana çıkarılarak, kullanıcıya sunulacaktır.

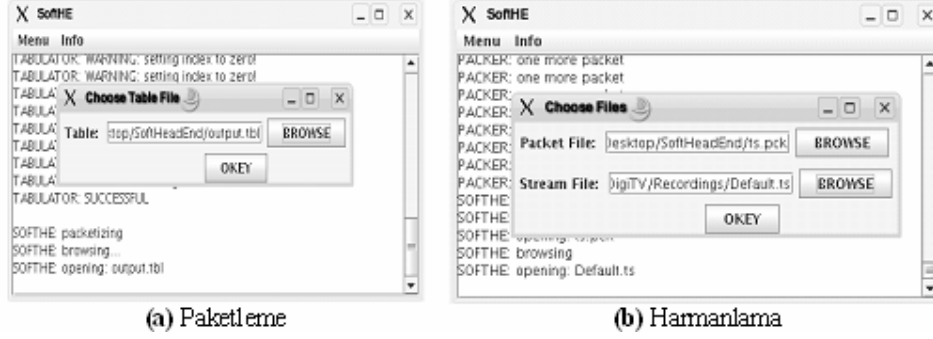
Kullanıcının TRT-2 de gösterilen bir belgeseli kaydedeceğini varsayarak, ilk adımda, SoftHE yazılımı kullanılarak, TRT'nin önceden kaydedilmiş TS dosyasına, canlı içerik nesnelere eklenmiştir. Metin nesnelere olarak; başlık, alt-başlık, metin gövdesi'nin; grafik olarak da bir XPM dosyasının eklendiği ekran çıktısı Şekil 4.5'te gösterilmiştir.



Şekil 4.5: Canlı İçerik Tablosu'nun SoftHE ile hazırlanması

Canlı İçerik Tablosu hazırlanırken, TRT-2'nin *servis\_no* bilgisinin 2 olduğu önceden bilinmektedir. Tablo versiyonu 0-15 arası sayılardan 0 seçilmiştir. Şekil 4.5'te görülen *LC Video PID* değeri ise ön tanımlı olarak bırakılmıştır. Öntanımlı bu değer (8191 = 0x1FFF) canlı içerik olarak herhangi bir görüntü nesnesinin olmadığını göstermektedir.

Grafik nesnesi olarak da, İstanbul Teknik Üniversitesi'nin logosu kullanılmıştır. Metinsel içerik ile ilgili grafiğin, kullanıcının TRT-2'den önceden yapmış olduğu bir kaydını izlediği esnada ekranda çıkması beklenmektedir. Tablo *CREATE* butonuna basılıp oluşturulduktan sonra; sırası ile, TS paketlerine ayrılmış ve TRT'nin önceden kaydedilmiş bir yayınına gömülmüştür. Şekil 4.6 bu adımları göstermektedir.



Şekil 4.6: Canlı İçerik Tablosu'nun TS'e gömülmesi

Senaryonun kullanıcı tarafında ise, ilk olarak, kullanıcı Şekil 4.7'de gösterilen arayüz ile SoftTV uygulamasının televizyon kısmını kumanda eder ve mevcut servisleri dolaşır. Uygulama kumandası olabildiğince sade tasarlanmaya çalışılmıştır. Ekranda menü yokken, *UP* ve *DOWN* butonları servisler arası dolaşmayı sağlar. *SELECT* ile menüye girilmişse, tuşlar bu sefer menü maddeleri arasında dolaşmayı sağlar.



Şekil 4.7: Uzaktan kumanda grafik arayüzü

Test yayınında, Şekil 4.8'teki gibi dört adet servis bulunmaktadır. Kullanıcı ilgisini çeken TRT-2'deki belgeseli Şekil 4.9'teki menü aracılığı ile kaydedecektir.



Şekil 4.8: TRT DVB-T test yayını

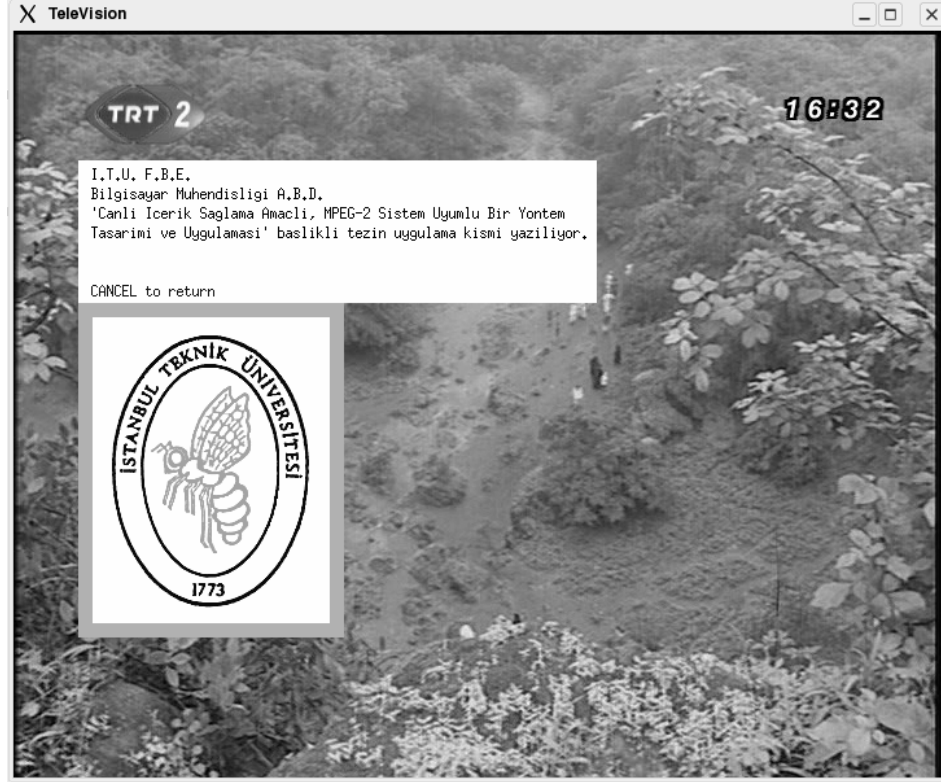


Şekil 4.9: SoftTV menüsü üzerinden kayıt işlemi



Şekil 4.10: SoftTV menüsü üzerinden oynatma işlemi

Kullanıcı kaydetme işlemini menü üzerindeki *CANCEL* butonu ile sonlandırır. Bundan sonra, kullanıcı normal televizyon izlemeye geçer ve TRT-1'de oynayan diziyi seyrederek. Bu dizi bitmek üzere iken de, kaydetmiş olduğu belgeseli izlemek ister. Şekil 4.10 dizinin bitişiyle beraber kullanıcının menüden, kaydettiği yayını seçmek üzere olduğu anı göstermektedir.



Şekil 4.11: Canlı İçerik sunumu

Kullanıcı, kayıtlı belgeselini izlerken, SoftTV bir taraftan, TRT-2 ile ilgili canlı içerik takibi yapmaya başlar. Bu servise, senaryonun yayıncı tarafındaki kısmında, TRT-2 ile ilgili içerikler yerleştirilmiştir. SoftTV, oluşturduğu Canlı İçerik Tablosu'nun *tablo\_uzantısı* kısmını, TRT-2'nin *servis\_no* değeri ile karşılaştırır ve eşit olduğu anda kayıttan yapılan yayını dondurur. Kullanıcıya, tablodan ayrıştırdığı bilgileri, Şekil 4.11'deki gibi gösterir.

Kullanıcı, *CANCEL* ile kayıttan oynatmaya, kaldığı yerden devam edebilir. Bu durumda, SoftTV; versiyonu bir fazla olan tabloları takip etmeye başlar. Senaryo başlangıcında yayına, 0 no'lu haricinde 1 no'lu versiyonlu başka içerikler de gömülmüş olsa idi, ikinci canlı içerik sunumunda bu içerikler görüntülenmiş olacaktı.



## 5. SONUÇLAR VE TARTIŞMA

Bu tez çalışmasının ana hareket noktası; bilgisayar mühendisliğinin etkin olmaya başladığı uygulama alanlarından birinin de sayısallaşmaya başlayan görüntü ve ses yayıncılığının olduğu fikri idi. Bu noktadan hareketle, mevcut sayısal yayıncılık standartları, piyasaya çıkmış ve/veya çıkacak olan ürünler, son teknolojik araştırma ve geliştirmeler incelenerek, yenilikçi ve yaratıcı bir konu üzerinde çalışmak hedeflenmiştir. Sonuç olarak; mevcut sayısal yayıncılık standartları ile uyumlu yeni bir yöntem önerilmiş ve bu yöntem sayesinde, son teknolojik ürünlerden olan sayısal yayın kaydedicilerde, kullanıcıların kayıtlı yayınlarını izlerken, canlı içeriklere de erişimlerini sağlayacak bir uygulama üzerinde çalışılmıştır<sup>1</sup>.

Tez kapsamında ortaya konan çalışmalar, yöntemin ana noktalarının tasarlandığı ve uygulamada sınındığı kısımlardan oluşmaktadır. Şüphesiz ki; ikincil olarak tasarlanacak ve uygulama kapsamında gerçekleştirilecek bir çok yön söz konusudur. Bu bakış açısıyla; tasarım açısından tezin geliştirilmeye açık noktalarına, bu bölümde yer vermeye çalışılmıştır.

Hangi içerik türlerinin, canlı içerik sınıfında tanımlanabileceğine dair, tez kapsamı içinde kavramsal olarak canlı içerik açıklaması yapılırken değinilmişti. Bunlar; son dakika gelişmeleri, güncel reklamlar ve de bazı duyurular idi. Bunlara ek, daha farklı türden de içerikler söz konusu olabilir. Tezde sunulan yöntemin önerdiği altyapı, içeriğin türünden bağımsız olduğundan, örnek olarak verilen türlerin dışında farklı türlerin de önerilmesinin yönü açıktır.

Önerilen yöntem, yapısal açıdan, canlı içerik nesnelere tanımlanması ile ilgili birtakım kısıtlamalar getirmektedir. Örneğin, metin gövdesi 255 karakterden fazla olamaz. Metin gövdesi ya da farklı alanların uzunluğunun daha fazla olduğu durumları da içerecek şekilde daha karmaşık açıklayıcılar tasarlanabilir.

İçerik nesne türleri ile ilgili üç ana grup oluşturulmuştur; metin, grafik ve de çokluortam. Ana grupların ise başlıca alt-grupları oluşturulmuştur. Örneğin, grafik nesnelere için sadece Linux sistemlerde kullanılan ve sıkıştırılmamış bir

---

<sup>1</sup> Bahsi geçen yöntemin TPE (Türk Patent Enstitüsü)'ye başvuru süreci tamamlanmıştır.

grafik dosyası formatı olan XPM tanımlanmıştır. Burada, sıkıştırılmış dosya formatları kullanmak, oluşacak tablo boyunu azaltacağından daha tercih edilebilir<sup>1</sup>.

Tasarımın geliştirilmeye açık noktalarından burada değinilmesi gereken son nokta ise tez kapsamı içinde bir çalışmanın yapılmadığı ses olgusudur. Çokluortam nesnesi olarak sınıflandırılması gereken ‘ses’li içerikler daha çok ‘görüntü’ nesnelere üzerinden işlenmeye çalışıldı. Tasarıma bu içerik nesnelere de çokluortam alt-grup nesnelere olarak eklenmesi, tez çalışmasının bir sonraki adımda tamamlaması gereken işlerin başında gelmelidir.

Uygulama tarafında da tasarımın geliştirilmeye açık noktalarının gerçekleşip, test edilmesi şarttır. Uygulamada, basitlik açısından metinsel içerikler daha ön plana çıkarılmıştır. Grafik tabanlı içeriklerin, özellikle ağ uygulamalarında çokça kullanılan formatlarının (.png, .jpg, vb.) TS içine gömülebildiği uygulamalar geliştirilebilir. Bu formatlar sıkıştırılmış olduklarından, tablo boyutlarının küçük kalmasını sağlayacaklardır.

Uygulama, kullanmış olduğu kütüphane gereği, ‘ses’ çözme ve işleme becerisine sahip değildir. Bu özelliğin ya mevcut kütüpheneye ve/veya SoftTV yazılımına eklenmesi ya da ‘ses’ desteği de veren bir kütüphane ile uygulamanın yeniden tasarımının yapılması tezin bir sonraki adımda önüne koyması gereken hedeflerden olmalıdır.

Son olarak, tasarım; kişisel bilgisayarların sunduğu kolay ve hızlı geliştirme ortamlarında SoftTV ya da benzeri farklı uygulamalarla gerçekleştirilerek sınanıp belli bir olgunluğa eriştikten sonra, asıl hedef platform üzerinde yani PVR gibi kayıt yeteneğine sahip sayısal yayın alıcı ürünlerinde geliştirilmeye başlanmalıdır.

---

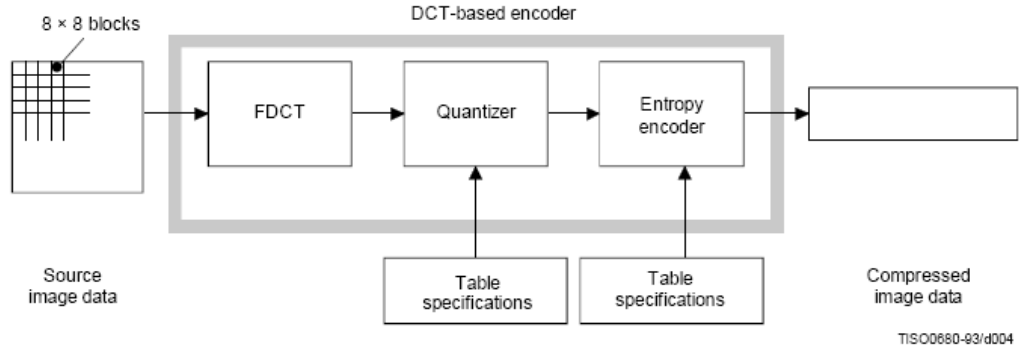
<sup>1</sup> Tezde XPM formatının kullanılmasının nedeni, sıkıştırılmamış formatın ekrana çizilmesinin uygulama açısından daha az çaba gerektirmesi idi.

## KAYNAKLAR

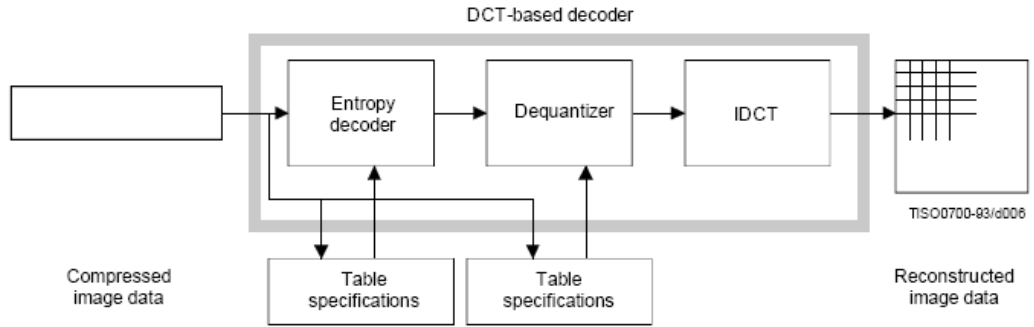
- [1] **Morgül, A.**, 2002. Dijital Televizyon: Uydu ve Karasal Yayın Sistemleri, Anten Dünyası Dergisi, İstanbul.
- [2] **Benoit, H.**, 2002. Digital Television: MPEG-1, MPEG-2 and principles of the DVB system - 2nd edition, Focal Press, Oxford.
- [3] **ACTERNA**, 2001. *MPEG-2 Digital Broadcast Pocket Guide*, Acterna, Maryland.
- [4] **ISO/IEC 10918**, 1993. Information Technology – Digital Compression and Coding of Continuous-Tone Still Images – Requirements and Guidelines, *International Telecommunication Union*.
- [5] **ISO/IEC 13818-1**, 1994. Information Technology – Generic Coding of Moving Pictures and Associated Audio Information: Systems, *International Organisation of Standardisation*, Cenevre.
- [6] **Oliphant, A. ve Scudamore B.**, 2001. TV Gets Personal, *IEE Review*, **9**, 9-13.
- [7] **Lu, W., Yu, S. ve Chu, J.**, 2004. A Unified Program Resource Management Scheme of Personal Video Recorder, *IEEE International Symposium on Consumer Electronics*, 1-3 Eylül, s. 351-352.
- [8] **Nakane, K., Otsuka, I., Esumi, K.**, Divakaran, A. ve Murakami, T., 2003. A Content-Based Browsing System for a HDD and/or Recordable-DVD Personal Video Recorder, *IEEE International Conference on Consumer Electronics*, 17-19 Haziran, s. 26-27.
- [9] **STi5100 Data Brief**, 2004. Low-cost interactive set-top box decoder, *STMicroelectronics*, Cenevre.
- [10] **ETSI EN 300 468 V1.6.1**, 2004. Digital Video Broadcasting (DVB); Specification for Service Information (SI) in DVB Systems, *European Telecommunications Standards Institute*, Cedex.

- [11] *libmpeg2 – a free MPEG-2 video stream decoder*. 3 Haziran 2006, <http://libmpeg2.sourceforge.net>
- [12] **Paçacı, K. K.**, 2006. Karasal Sayısal TV Yayıncılığı (DVB-T), Geniřbant Pazarına Etkisi ve Dzenleme Perspektifi, *Uzmanlık Tezi*, Telekomünikasyon Kurumu, Ankara.

## EK A. JPEG Kodlayıcı/Kodçözücü Blok Diyagramları




Şekil A.1: Basitleştirilmiş JPEG kodlayıcı blok diyagramı



Şekil A.2: Basitleştirilmiş JPEG kodçözücü blok diyagramı

## EK B. STi5100 Tümdevre Kılavuzu



**STi5100**

---


Low-cost interactive set-top box decoder

DATA BRIEF

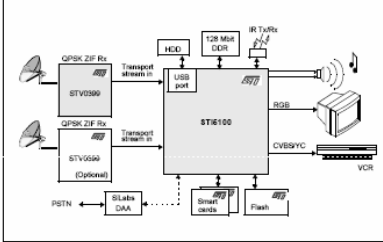
**Description**

The STi5100 provides the power and features required to meet the specifications of the most demanding broadcasters and can be used in satellite, terrestrial, cable or IP set-top boxes. The STi5100 also provides a USB2.0 host interface which allows consumers to connect a wide range of peripherals to their STB such as digital cameras, memory sticks, printers and hard disk drives for DVR applications.

The STi5100 features the ST20 CPU that is standard across the OMEGA range but boosts the clock speed to 243MHz which, when combined with the DDR memory interface, gives the new device exceptional levels of performance. The new chip also includes a unified DDR memory interface and a modern interface compatible with Silicon Laboratories' silicon DAA.



Application: Cost effective set-top box with optional hard disk



May 2004 1/2

**STi5100**

**Features**

- **Enhanced ST20 32-bit VL-RISC CPU**
  - 243 MHz, 8Kbyte I-Cache, 8Kbyte D-Cache, 4 Kbyte SRAM
- **Unified Memory Interface**
- **Dual Transport Stream Merger**
  - Supports DVB and DIRECTV\*
  - Integrated DES-ECB, DVB and ICAM decoders
  - NDS RASP compliant
  - Low cost DVB-CI and Cable Card support
- **MPEG-2 MP@ML Video Decoder**
  - Trick modes including smooth fast forward and rewind
- **Audio**
  - MPEG-1 layers III, MP3
  - Dolby® Digital decoding
  - Dolby Pro Logic® compatible output
  - PCM input, mixing and sample rate conversion
  - DRS/TruSurround™ virtual surround sound
  - Simultaneous MPEG audio decode and output of Dolby streams
- **Graphics/Display**
  - 4 display planes
  - 2, 4 and 8 bpp CLUT graphics, 256 x 30 bits (AYCBO) CLUT entries
  - 15 bpp true color graphics
  - Alpha blending, anti-aliasing, anti-flicker, anti-flicker filters
- **PAL/NTSC/SECAM encoder**
  - RGB, CVBS, Y/C and YUV outputs with 10-bit DACs
  - CGMS, Teletext, WSS, VPS and close caption
- **On-Chip Peripherals**
  - 4 ASCs (UARTS)
  - 4 parallel 8-bit I/O banks
  - 2 smartcard interfaces and clock generators
  - 3 SSCs for I2C/SPI master/slave interfaces
  - Silicon Labs line side (DAA) interface
  - High-speed USB OHCI/EHCI compliant host interface
  - DISEQC interface
- **Enhanced Security**

**Ordering information**

Device number	Description
STi5100UA	Basic STi5100
STi5100-Mboard	Development platform
STi5100-Ruf	Example STB design
STi5100-ST0VB	STAP1 drivers for DVB

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without the express written approval of STMicroelectronics.

The ST logo is a registered trademark of STMicroelectronics.  
All other names are the property of their respective owners.

© 2004 STMicroelectronics - All Rights Reserved

STMicroelectronics Group of Companies  
Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy  
Japan - Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States

www.st.com

2/2 

Şekil B.1: STi5100 tümdevre kılavuzu

## EK C. PAT/PMT Bölüm Dizilimleri

**Tablo C.1:** PAT bölüm dizilimi

Dizilim	İkil Sayısı
program_association_section() {	
table_id	8
section_syntax_indicator	1
0	1
reserved	2
section_length	12
transport_stream_id	16
reserved	2
version_number	5
current_next_indicator	1
section_number	8
last_section_number	8
for (i=0; i<N; i++) {	
program_number	16
reserved	3
if (program_number == '0') {	
network_PID	13
}	
else {	
program_map_PID	13
}	
}	
CRC_32	32
}	

**Tablo C.2: PMT bölüm dizilimi**





<b>Dizilim</b>	<b>İkil Sayısı</b>
TS_program_map_section() {	
table_id	8
section_syntax_indicator	1
0	1
reserved	2
section_length	12
program_number	16
reserved	2
version_number	5
current_next_indicator	1
section_number	8
last_section_number	8
reserved	3
PCR_PID	13
reserved	4
program_info_length	12
for (i=0; i<N; i++) {	
descriptor()	
}	
for (i=0; i<N; i++) {	
stream_type	8
reserved	3
elementary_PID	13
reserved	4
ES_info_length	12
for (i=0; i<N; i++) {	
descriptor()	
}	
}	
CRC_32	32
}	



## EK D. SoftHE ve SoftTV Yazılımlarının Kaynak Kodları


### 1. SoftHE Yazılımı

#### SoftHeadEnd





 SoftHE.java    Tabulator.java    Packer.java    Muxer.java

### 2. SoftTV Yazılımı

#### SoftTeleVision

 mpeg2dec-0.4.0

 src

 television.cpp    television.h    siparser.cpp    siparser.h

 remotecontrol.cpp    remotecontrol.h    thread.cpp    thread.h

 softtv.cpp    makefile

## **ÖZGEÇMİŞ**

24.09.1980 tarihinde Almanya'nın Bielefeld kentinde dünyaya geldi. İlk, orta ve de lise eğitimini, sırasıyla, Elazığ Gazi Osman Paşa İlkokulu, Elazığ Anadolu Lisesi ve Bursa Fen Lisesi'nde tamamladı. İ.T.Ü. Bilgisayar Mühendisliği bölümünden 2003 yılında mezun oldu. Aynı yıl Beko Elektronik A.Ş.'nin ARGE bölümünde yazılım tasarım mühendisi olarak çalışmaya başladı ve hala bu işine devam etmektedir.