



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ  
ÚSTAV AUTOMATIZACE A INFORMATIKY

FACULTY OF MECHANICAL ENGINEERING  
INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

## APLIKAČNÍ SERVER PRO HBBTV

APPLICATION SERVER FOR HYBRID BROADBAND TV

DIPLOMOVÁ PRÁCE  
DIPLOMA THESIS

**AUTOR PRÁCE**  
AUTHOR

**BC. VLADIMÍR LOVEČEK**

**VEDOUCÍ PRÁCE**  
SUPERVISOR

**ING. RADEK POLIŠŤUK, PH.D.**

BRNO 2014



Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav automatizace a informatiky

Akademický rok: 2013/14

## ZADÁNÍ DIPLOMOVÉ PRÁCE

student(ka): Bc. Vladimír Loveček

který/která studuje v **magisterském studijním programu**

obor: **Aplikovaná informatika a řízení (3902T001)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

### **Aplikační server pro HbbTV**

v anglickém jazyce:

### **Application server for Hybrid Broadcast Broadband TV**

Stručná charakteristika problematiky úkolu:

Rešerše a ukázka aktuálních technologií propojujících digitální televizní vysílání s širokopásmovými WWW aplikacemi - HbbTV. Rozbor možností, které tyto doplňkové informační kanály běžnému "konzumentovi" televizního vysílání nabízely a nabízejí.

Cíle diplomové práce:

Rešerše evoluce systémů, určených pro přenos doplňkových informací k běžnému televiznímu vysílání (teletext, EPG, HbbTV a další).

Ukázka řešení aplikačního serveru pro HbbTV, vyvíjeného řešitelem.

Seznam odborné literatury:

<http://www.hbbtv.com/>

Vedoucí diplomové práce: Ing. Radek Poliščuk, Ph.D.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2013/14.

V Brně, dne 25. 11. 2013



Ing. Jan Koupec, Ph.D.  
Ředitel ústavu

prof. RNDr. Miroslav Doupovec, CSc., dr. h. c.  
Děkan

## **ABSTRAKT**

Tato diplomová práce se zabývá dodatkovými službami přenášenými společně s televizním vysíláním. První část práce tvoří rešerše evoluce těchto technologií. V druhé části je pak krok za krokem řešena vytvořená ukázková aplikace založená na nejnovější platformě HbbTV. Popisovaná aplikace demonstruje možnosti streamování audiovizuálního obsahu na obrazovku televizního přijímače.

## **ABSTRACT**

This diploma thesis focuses on the additional services transmitted in conjunction with television broadcast. The first part presents the background research in the evolution of these technologies. In the second part the step-by-step creation of a sample application based on the latest HbbTV platform is dealt with. Finally, the described application demonstrates the streaming possibilities of audiovisual content on the TV screen.

## **KLÍČOVÁ SLOVA**

televizní vysílání, analogové vysílání, digitální vysílání, DVB, teletext, VPS, MHP, HbbTV

## **KEYWORDS**

television broadcast, analogue broadcast, digital broadcast, DVB, teletext, VPS, MHP, HbbTV



## PROHLÁŠENÍ O ORIGINALITĚ

Prohlašuji, že jsem diplomovou práci na téma Aplikace serveru pro HbbTV vypracoval samostatně dle pokynů a rad vedoucího práce Ing. Radka Polišůvky, Ph.D. s použitím uvedené odborné literatury.

Bc. Vladimír Loveček

## BIBLIOGRAFICKÁ CITACE

LOVEČEK, V. *Aplikace serveru pro HbbTV*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2014. 69 s. Vedoucí diplomové práce Ing. Radek Polišůvka, Ph.D..



## **POD KOVÁNÍ**

Touto cestou bych chtěl podkovat vedoucímu diplomové práce Ing. Radku Polišukovi, Ph.D. za připomínky, cenné a odborné rady při psaní této diplomové práce.



**OBSAH:**

<b>1 Úvod.....</b>	<b>13</b>
<b>2 Televizní vysílání a dodatkové služby.....</b>	<b>15</b>
2.1 DVB .....	16
2.2 Teletext.....	19
2.3 VPS a PDC.....	26
2.4 MHP .....	29
2.4.1 EPG .....	32
2.5 HbbTV.....	33
<b>3 HbbTV aplikace.....</b>	<b>41</b>
3.1 Úvodní stránka .....	41
3.2 Stránka televizního programu .....	47
3.3 Stránka ankety .....	51
<b>4 Testování HbbTV aplikace .....</b>	<b>59</b>
4.1 Testování během vývoje.....	59
4.2 Testování na hybridním přijímači .....	61
4.3 Použití přímého streamování z internetu .....	64
<b>5 Závěr.....</b>	<b>67</b>



# 1 ÚVOD

Šířené televizní vysílání neobsahuje pouze video a audio signál, ale k tomuto signálu jsou v televizních studiích přidávány doplňující informace užitečné jak pro diváka, tak pro elektroniku připojenou k televiznímu přijímači. V počátcích se jednalo o jednoduché informační služby, protože možnosti analogového vysílání byly omezené. V digitálním vysílání už jsou možnosti širší díky vylepšení části datového toku přímo pro tyto dodatečné služby.

Nejen televizní vysílání, ale také samotné televizní přijímače prošly od svých začátků značnými změnami. Tyto změny probíhaly postupně a tak ne každý televizor umí zobrazovat všechny dodatečné služby přidávané společně s televizním vysíláním. První z vyráběných modelů televizorů měly černobílou obrazovku s malou úhlopříčkou a byly kombinovány s radiovým přijímačem. Obrazovka se postupně zvětšovala a televizor se stal samostatným zařízením. Postupem času se začínají objevovat nové typy s barevnou obrazovkou, která byla odpovídající na barevné televizní vysílání. První televizory s černobílou obrazovkou dokázaly zobrazit i šířený barevný televizní signál, ale postupně byly nahrazeny televizory barevnými. Pevratnou novinkou zvyšující pohodlí diváka se stal ve své době televizor ovládaný pomocí dálkového ovladače. Po vyvinutí doplňkové služby teletext se do nových modelů přidávají také dekodéry této služby, čímž umožnily divákovi sledovat nejen samotné televizní vysílání, ale také zobrazovat skryté titulky a později i krátké informační zprávy. S přechodem z analogového na digitální vysílání došlo oproti úpravám funkcí televizorů, protože staré televizní přijímače určené pro analogové vysílání nebyly s novou technologií kompatibilní, a tak si divák musel buď koupit nový televizor, nebo alespoň ke starému připojit dekodér digitálního vysílání, tzv. set-top box. Staré televizory s CRT obrazovkami jsou poslední dekádu vytlačovány kompaktnějšími zařízeními s novými typy plochých obrazovek, dosahujícími na větší úhlopříčce výrazně vyšší obrazové kvality.

Pod pojmem televize si každý nejspíše představí zařízení pro příjem televizního vysílání i obrazovku připojenou k jinému zařízení sloužící jako jeho výstup. V posledních letech však došlo u televizních přijímačů k rozšíření jejich funkcí a televizní přijímač tedy nemusí sloužit pouze k sledování televizních programů, ale s jeho pomocí se může uživatel připojit k portálu výrobce, kde si může stáhnout nejnovější aplikace, prohlížet si přes integrovaný webový prohlížeč internetové stránky, komunikovat se svými přáteli a podobně. Tyto aktivity je možné vykonávat díky možnosti připojit přijímač k počítačové síti s přístupem na internet.

Tato diplomová práce se tedy zabývá jak dodatečnými službami analogového vysílání, tak i novějšími službami spojenými s vysíláním digitálním a nejnovější službou HbbTV, která kombinuje digitální vysílání s již zmíněnou možností využití připojení k internetu. Vytvořená HbbTV aplikace popisovaná v této práci vznikla pro firmu LIVEBOX, a. s., která se zabývá streamováním videí na internetu. Aplikace slouží jako ukázka z části projektového portfolia této firmy.



## 2 TELEVIZNÍ VYSÍLÁNÍ A DODATKOVÉ SLUŽBY

Historie televizního vysílání sahá do 20. let 20. století, kdy v USA provedl první pokusy Vladimír Zvorykin, pro kopírování záznamu obrazu pomocí elektronek. V následujícím desetiletí se začínají objevovat první televizní stanice s pravidelným televizním vysíláním. Na českém území, přesněji v tehdejší Československu, je začátek televizního vysílání datován k 1. květnu 1953 spuštěním zkušebního provozu pravidelného televizního vysílání ze studia v Praze. [1]

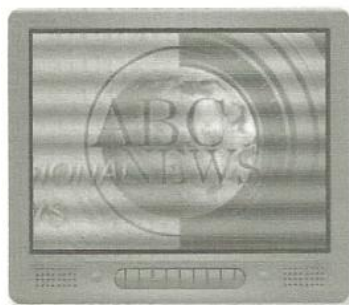
Jak již to u většiny technologií bývá, tak se i technologie televizního přenosu vyvíjí. Od počátku přenosu televizního vysílání a po dlouhou dobu potom je signál šířen analogově. To znamená, že celý průběh přenosu televizního signálu od televizního studia až po obrazovku u pozorovatele probíhá analogově. V televizním studiu tedy snímá kamera danou scénu, signál z kamery je zpracován a následně je obraz fázovou modulací namodulován na nosnou vlnu. Zvuk je přenášen nezávisle na obrazu tak, že je na nosnou vlnu modulován frekvenční modulací. Formát analogového přenosu byl v České republice, od 90. let až do závěru analogového vysílání, dán standardem PAL D/K. Označení vysílací normy se dle mezinárodní telekomunikační unie ITU skládá ze dvou parametrů, první (PAL/SECAM/NTSC) určuje kódování barevné informace a druhý vysílací systém, platný i pro černobílé televizní vysílání. Standard PAL D/K udává přenos 25 snímků za sekundu a daný snímek je tvořen 625 řádky. Aby se zamezilo blikání obrazu, tak se pro vykreslení snímku využívá prokládané řádkování, kdy každý snímek je rozdelen na polsnímky. Podle standardu PAL D/K jsou nejprve vykreslovány liché řádky obsažené v prvním polsnímku a až pak i druhém průběhu paprsku po stínítku jsou vykreslovány řádky sudé. I když je celý snímek tvořen 625 řádky, tak pouze 576 řádků nese obrazovou informaci. Zbývající řádky nenesou obrazovou informaci a jsou označovány jako řádky zatemňovacího intervalu. Tyto řádky jsou využívány pro synchronizaci a přenos vykreslovacího paprsku nebo mohou být použity pro přenos dodatkových informací, jako je například teletext, informace pro přesný záznam po dobu VPS a další. Mezi hlavní nevýhody analogového vysílání patří fakt, že na jednom kanálu s šířkou pásma 8 MHz je přenášen pouze jeden televizní kanál a na mnoha místech, kde není například viditelnost na vysílání, může docházet k degradaci přenášeného signálu jak je vidět na obr. 1. [1]



Obrázek 2.1 Sněžení v obraze způsobené nízkou úrovní signálu



Obrázek 2.2 Duchy v obraze způsobující odrazy signálu



Obrázek 2.3 Intermodulace



Obrázek 2.4 Rušení z impulzních zdrojů

Obr. 1 Možné problémy s analogovým signálem způsobené degradací signálu [1]

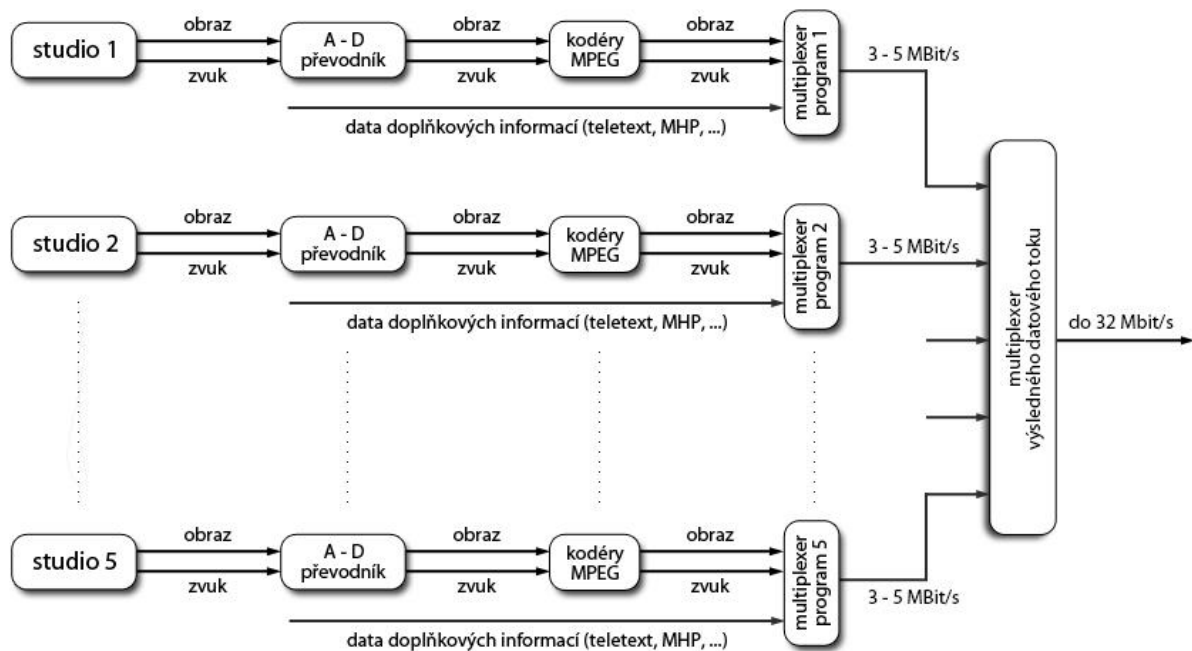
Kvůli vyerpání kmito tového pásma, jenž je zp sobeno v tšími nároky na po et kanál , a také kv li zvyšujícím se nárok m na kvalitu obrazu a zvuku, byl v Evrop vyvinut systém digitálního televizního vysílání ozna ovaný jako DVB (Digital Video Broadcasting). DVB je mezinárodní konsorcium, které bylo založeno v roce 1993 a je tvo eno 260 leny z 35 stát celého sv ta. Specifikace DVB jsou vydávány Evropským telekomunika ním institutem ETSI ve spolupráci s Evropskou unií pro televizní a rozhlasové vysílání EBU. V eské republice bylo spušt no testovací digitální vysílání v roce 2000 v Praze a Brn . [1] Za ízení pro p íjem analogového televizního televizní vysílání však v té dob vesm s nebyly schopné dekódování digitálního pozemního vysílání. Proto bylo nutné použít pro p íjem digitálního vysílání externí DVB set-top box. Teprve pozd ji se na trhu objevily televizory s integrovaným DVB dekodérem a CI sloty pro p íjem kódovaných program . [1]

## 2.1 DVB

Zkratka DVB tedy ozna uje systém pro p enos digitálního televizního vysílání, který ve své definici zahrnuje r zné zp soby p enosu televizního signálu, a to pozemní DVB-T, satelitní DVB-S, kabelový DVB-C a p enos pro p íjem televizního vysílání pomocí kapesních p íjíma DVB-H, který je nejnov jší. [2]

Obrazová kvalita m že být u DVB od velmi nízké LDTV p es standardní vysílanou kvalitu televize SDTV až po televizi s vysokou rozlišovací schopností HDTV. Rovn ž zvukový signál m že mít n kolik kvalit od monofonní, p es stereofonní až po prostorový zvuk 5.1 (Dolby Digital). [2]

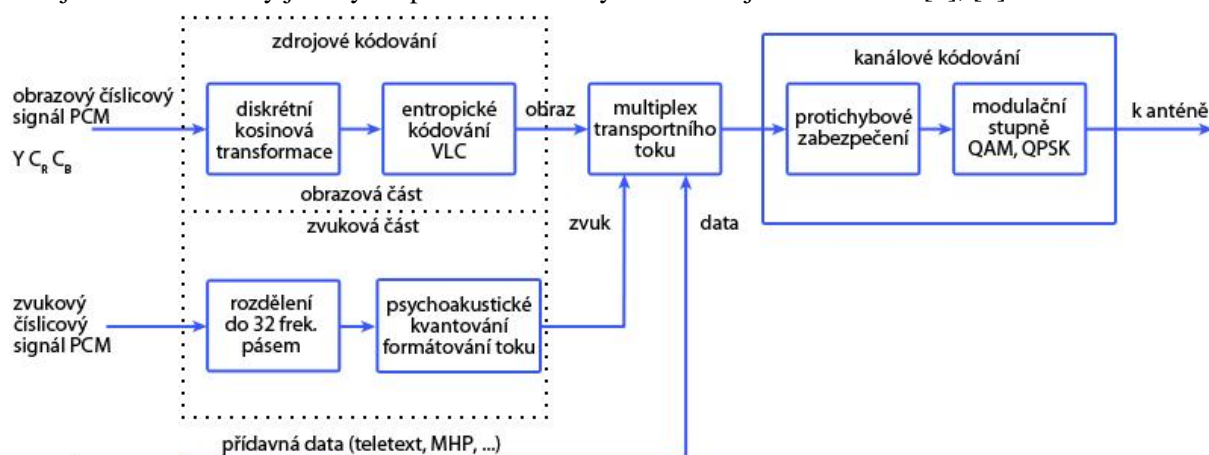
U p enosu televizního signálu analogov bylo charakteristické, že jeden televizní kanál je p enášen v jednom p enosovém kanále s ší kou pásma 8 MHz. U digitálního p enosu již toto omezení neplatí a v jednom p enosovém kanále je možné díky ú inné kompresi p enášet až 10 televizních kanál (v závislosti na kvalit ). Tímto zefektivn ěním se dosáhne snížení nárok na p enosové prost edí, m že být vysíláno více televizních kanál a také se sníží náklady na p enos jednoho televizního kanálu. Na obr. 2 je vid t p íklad vytvo ení výsledného datového toku multiplexu s p tí televizními kanály. [1], [2]



Obr. 2 Vytvo ení výsledného datového toku multiplexu [2]

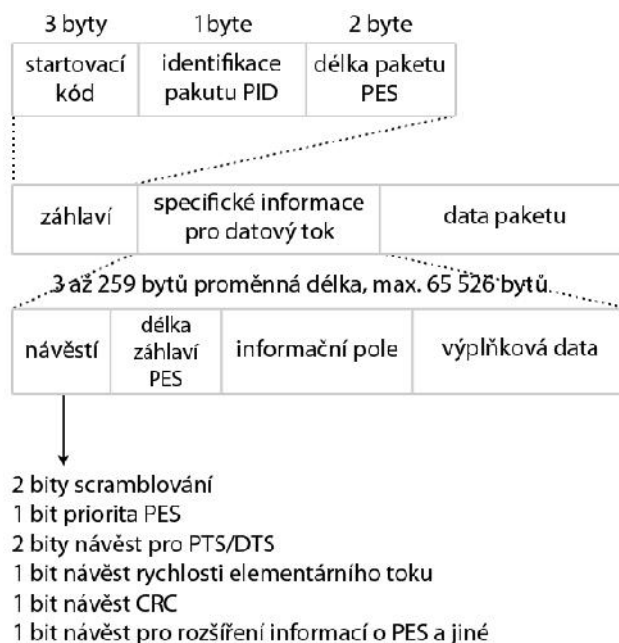
Jelikož nekomprimovaný bitový tok digitalizovaného zvukového a p edevším obrazového signálu je asi 270 Mbit/s a nebylo by jej tedy možné p enášet v p enosovém pásmu s ší kou 8 MHz, je nutné jej p ed p enosem komprimovat. Pro kompresi obou signál je u televizního p enosu používáno standardu MPEG-2. Tento standard používá ztrátovou kompresi a umož ůuje komprimovat signály v pom ru 20 – 40:1. Pon vadž se jedná o kompresi ztrátovou, tak již nemusí být možné obnovit data

do shodné podoby, v jaké byla digitalizována. Zmínou kompresi zajišuje zdrojové kódování. Obrazový signál je zpracováván diskretní kosinovou transformací a následně je entropicky kódován s proměnnou délkou slova. Zvukový signál je rozdělen do časových rámců, převeden z časové do frekvenční oblasti a rozdělen do frekvenčních pásem a nakonec je formátován pomocí psychoakustického modelu. Takto zpracované signály jsou společně s daty doplňkových informací multiplexovány do transportního toku jednoho TV programu. Následně je vytvořený transportní tok zabezpečen proti chybám při přenosu. Tímto zabezpečením se sice nepatrně zvýší nároky na přenosové pásmo, ale na příjemci je možné signál opravit. Zabezpečení je prováděno pomocí kódů FEC1 a FEC2. Po zabezpečení signálu proti chybám je ještě signál digitálně modulován. Metoda digitální modulace je použita podle druhu přenosového prostředí. Například pro pozemní přenos se používá modulace OFDM, která ovlivňuje vliv odražených signálů. Operace zabezpečení signálu proti chybám a digitální modulace jsou označovány jako kanálové kódování a zpracování obrazu před multiplexováním do transportního toku jednoho televizního kanálu je označováno jako kódování zdrojové. Schématicky jsou tyto operace zobrazeny na následujícím obrázku. [1], [2]



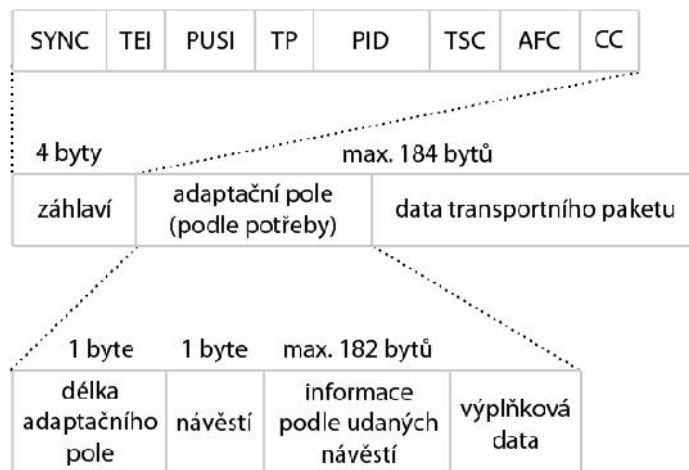
Obr. 3 Zpracování digitálního signálu jednoho televizního kanálu [1]

Přenos dat v digitálním vysílání neprobíhá v souvislém sledu informací, jako tomu bylo u analogového vysílání, ale data jsou rozdělena na menší bloky opatřené informačním záhlavím, tzv. pakety. Toto rozdělení na menší přenášené části umožňuje synchronizaci zvuku a obrazu a dalších přenášených informací v dekodéru. Další výhodou je flexibilita zpracování, protože pakety se snadno uchovávají v paměti a je možné je skládat z různých přenosových cest. Datový tok vystupující z vyrovnávacích pamětí obrazového a zvukového signálu a také přidavná data jsou označována jako paketový elementární datový tok (PES – Packetized Elementary Stream). Datový tok PES je základem pro vytvoření programového nebo transportního toku. Struktura každého paketu elementárního datového toku je složena z 6 bytů tvořících záhlaví. Záhlaví se dále dělí na 3 byty startovacího kódu, 1 byte identifikující jaký datový tok je v daném paketu přenášen a zbývající 2 byty udávají délku paketu. Za záhlavím následuje blok specifických informací s proměnnou délkou 3 až 259 bytů. V tomto bloku jsou v první části (2 byty) obsažena návští, která informují o vlastnostech paketu, následuje informace o délce záhlaví PES, informační pole a vyplňující data, která doplňují délku informačních dat na stejnou délku. Poslední částí PES paketu jsou data paketu, kde se přenáší vlastní užitečné informace. Maximální velikost přenášených dat je 65 526 bytů. Schématicky je složení paketu PES zobrazeno na obr. 4. V dekodéru na příjemci straně se pakety vybírají a ve vhodné časové souvislosti se reprodukuje podle dekodovacích (DTS) a prezentačních (PTS) značek uvedených v záhlavích paketů. [1]



Obr. 4 Složení paket pro elementární datový tok PES [1]

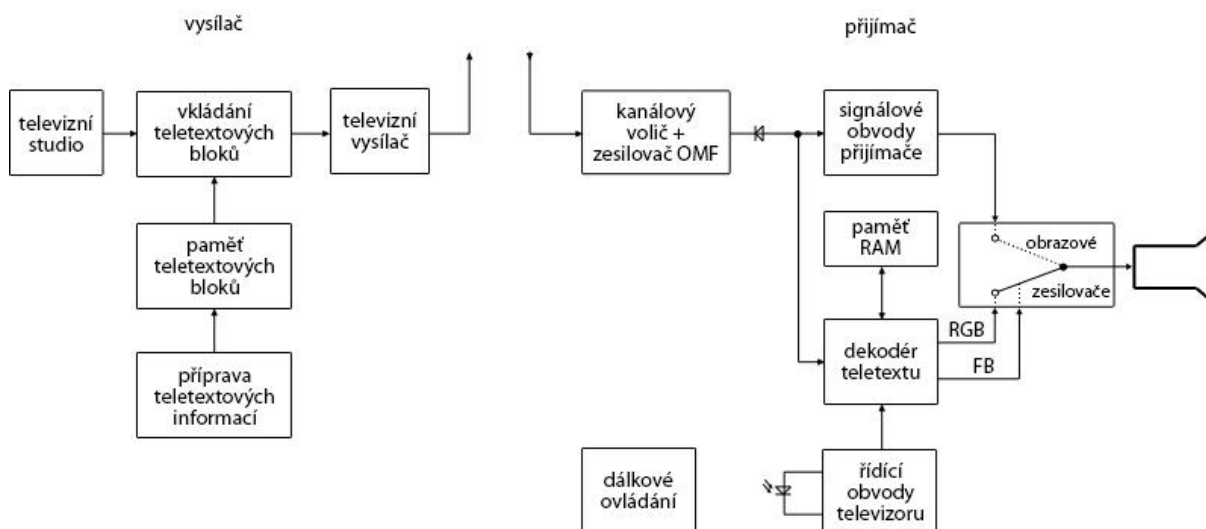
Jelikož může být transportní datový tok na své primární cestě vystaven rušení, tak se dlouhé pakety elementárních datových toků PES dělí na kratší pakety se stejnou délkou. Tento krok se nazývá transportní multiplexování a je rozhraním mezi zdrojovým a kanálovým kódováním. Pakety transportního toku jsou tvořeny 4 bytovým záhlavím, které je dále rozděleno do 8 skupin. Toto dělení je vidět na následujícím obrázku společně s celkovým složením transportního datového paketu. Synchronizační byte SYNC BYTE (Packet Sync Byte 47 hex) označující začátek paketu tvoří první skupinu. Druhou skupinu tvoří indikátor chybového přenosu TEI (Transport Error Indicator). Tento indikátor informuje o přenosu neopraveném v obvodech pro zabezpečení dat před dekodérem. Za ním následuje ukazatel začátku skupiny užitečných dat PUSI (Payload Unit Start Indicator) nesoucí informaci, zda je v transportním paketu přeneseno záhlaví paketu PES nebo jsou přeneseny informační tabulky o programech a jejich parametrech. Při přetížení primární cesty mohou být některé pakety přeneseny jednoduše a tuto informaci nese bit TP (Transport Priority) udávající přednostní přenos. Pakety skupiny identifikace paketů PID (Packet Identification Number and Priority) obsahují stejné údaje jako pakety v elementárním datovém toku. TSC (Transport Scrambling Control) je tvořena dvěma bity scamblování, které však neovlivňuje záhlaví ani adaptační pole. Následující dva bity AFC (Adaptation Field Control) slouží k řízení adaptačního pole. Stavem 01 je označena situace, kdy je 184 bytů paketu plně využito k přenosu dat bez adaptačního pole a stavem 10 je pak označena situace, kdy se v paketu nalézají užitečná data spolu s adaptačním polem. Poslední část záhlaví tvoří čísla souvislosti CC (Continuity counter) řídící pakety se stejným identifikačním znakem, čímž lze odhalit ztrátu paketu nebo nesprávné pořadí paketů. Podle potřeby se za 4 bytovým záhlavím vyskytuje již zmíněné adaptační pole, které je vytvářeno pouze v situaci, kdy paket elementárního datového toku PES nelze rozdělit beze zbytku na 184 bytové transportní pakety. Vytvářené pakety, které by měly mít rozdílnou velikost, jsou tedy doplněny adaptačním polem pro dorovnání velikosti paketu. Toto pole je složeno z informace o své délce následované návěstími a informacemi vyplývajícími z návěstí a výplňkovými bity. Mimo výplňkových dat jsou v adaptačním poli přeneseny některé řídicí informace pro řízení rekonstrukce obrazu a zvuku v dekodéru. Jednou z nejdůležitějších informací je návěst o programových referenčních hodinových impulzech PCR (Programme Clock Reference) přenesená v části s informacemi podle udaných návěstí. Toto návěstí se stará o to, aby se v dekodéru synchronizoval zdroj hodinových impulzů alespoň jednou za 0,1 s. Proto se musí adaptační pole přenášet alespoň jednou za toto časové období (nemusí se přenášet za každým paketem). Užitečná data transportního paketu jsou pak umístěna buď za adaptačním polem anebo přímo za záhlavím. Jelikož se po přenesení sledu 8 bytů provádí scamblování signálu, tak je doporučeno přenášet maximální počet užitečných dat jako násobek osmi. [1]



Obr. 5 Složení transportního datového toku [1]

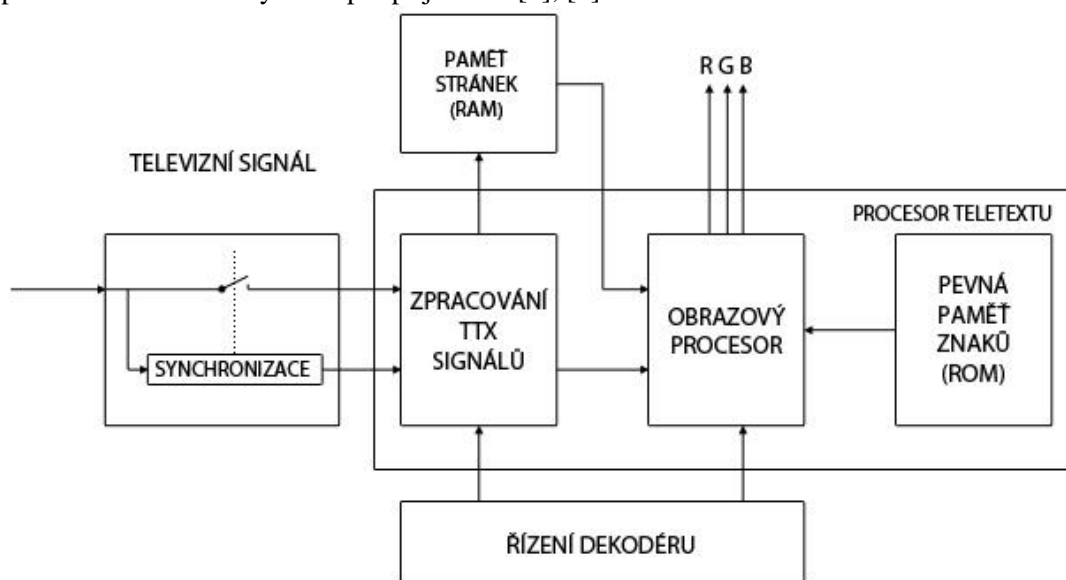
## 2.2 Teletext

Teletext je jednou z prvních a zároveň asi nejrozšířenějších dodatkových služeb předávanou již v analogovém vysílání. Myšlenka předání textových informací spolu s televizním signálem se objevila v 70. letech minulého století ve Velké Británii a tak byly položeny základy pro vznik teletextu. Tato služba má informační charakter a dovede zobrazovat textové informace a v omezeném rozsahu také grafické obrazce. Teletextové data jsou v analogovém signálu předána v záměrných řádcích. V České republice jsou pro předání teletextových dat využity řádky 7 - 15 a 19 - 20 u liché a 21 - 28 u sudých řádků. V sudém řádku jsou využity řádky 320 - 328 a 332 - 333, řádek 334 je tvořen zbytkem dat k teletextu. Datové bloky teletextu neobsahují konkrétní zobrazované informace, ale pouze instrukce pro dekodér v přijímači, který podle zadané instrukce vykresluje na obrazovku příjemce požadovaný znak nebo grafický element. Na obr. 6 je zobrazeno blokové schéma televizní předávací soustavy. Toto blokové schéma obsahuje také bloky, které jsou nutné pro implementaci teletextového předání do televizního signálu. Na vysílací straně se jedná o část přípravu teletextových informací, kde se dané informace zpracovávají do teletextových stránek a uspořádávají se do magazínů. Takto připravené stránky se dále kódují a vytváří se z nich teletextové bloky ukládané do paměti teletextových bloků, odkud jsou následně vkládány pomocí vkladatele teletextových bloků do televizního signálu. [4]



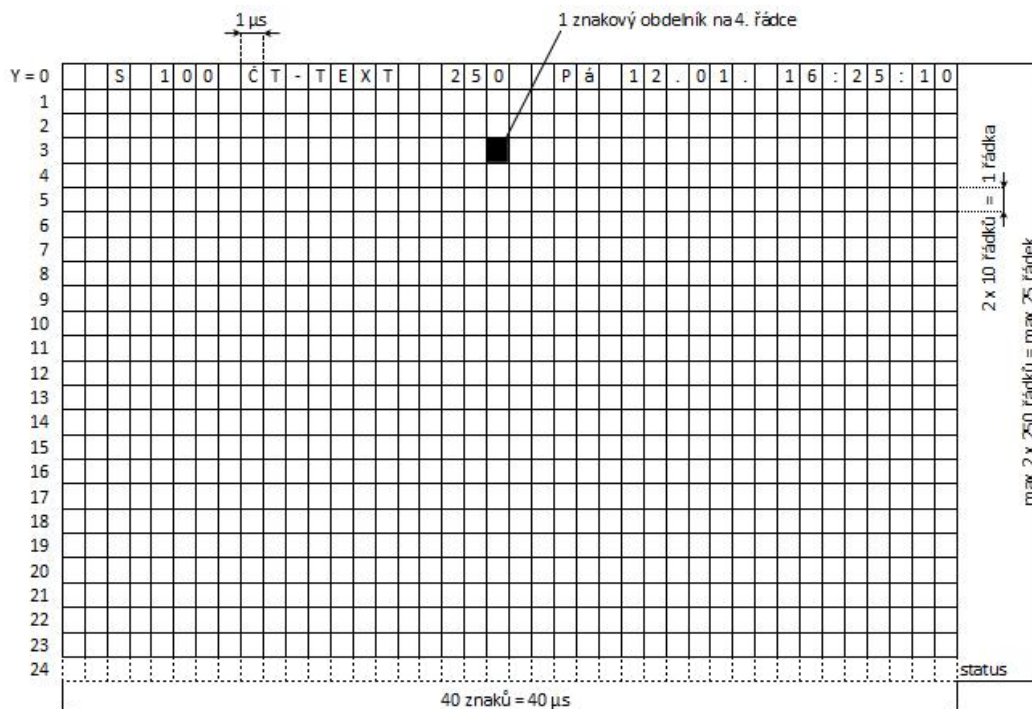
Obr. 6 Schéma televizní předávací soustavy s vkládáním bloků teletextu. [4]

Na straně přijímače je nejdůležitější částí dekodér teletextu zajišťující příjem teletextového signálu. Tento dekodér je složen z několika částí, které zpracovávají teletextová data. Schématicky je dekodér zobrazen na obr. 7. Vstupní část dekodéru je tvořena obvodem pro synchronizaci. Další obvod realizuje výběr teletextových bloků z televizního signálu, které buď uloží do paměti RAM, nebo je může rovnou předat obrazovému procesoru. Tento procesor úzce spolupracuje s pamětí ROM, v níž jsou uloženy kódy znaků. Obrazový procesor postupně zpracovává teletextové bloky z paměti RAM a za pomoci paměti znaků ROM rekonstruuje teletextovou stránku na obrazovku přijímače. Mimo dekodér je dalším důležitým prvkem na přijímací straně stykový obvod, který zajišťuje spojení výstupu dekodéru s koncovými stupni přijímače. [3], [4]



Obr. 7 Struktura teletextového dekodéru [3]

Jak již bylo zmíněno, tak struktura teletextu je tvořena z jednotlivých stránek a magazínů. Každá stránka je pak tvořena 24 řádky obsahujícími maximálně 40 znaků a jedním statusovým řádkem, jak je vidět na obr. 8. První řádek označovaný jako Y = 0 nebo X/0 je hlavičkou stránky, následujících 23 řádků může obsahovat grafické nebo textové informace a poslední 25. řádek je tzv. statusový řádek, který zobrazuje informace o stavu vysílání a zobrazuje se pouze u dekodérů druhé generace. Mezi jednotlivými stránkami se dá po vyvolání teletextu na obrazovku přecházet zadáním číselného čísla v rozmezí od 100 do 899. Celý teletext může být tedy tvořen až 800 stránkami rozdělenými na 8 magazínů. První číslice udává číslo magazínu, který sdružuje stránky s tématicky podobným obsahem jako je například sport, zpravodajství apod. Zbývající dvě číslice udávají číslo konkrétní stránky. Po zadání čísla stránky trvá několik sekund až minut, než se daná stránka zobrazí. Rychlost přenosu teletextových stránek je u analogového signálu dána počtem použitých řádků ze zatměvacího intervalu. Při využití všech 25 teletextových řádků je možné přenést za sekundu do paměti až 8 celých teletextových stránek. Při prvním vyvolání teletextu je vždy zobrazena úvodní stránka s číslem 100. Tato stránka většinou obsahuje informace o rozdělení informací na dalších stránkách. Na obr. 9 je zobrazena úvodní stránka teletextu české televize. [3], [4]



Obr. 8 Formát zobrazení teletextu na obrazovce televizního přijímače [4]

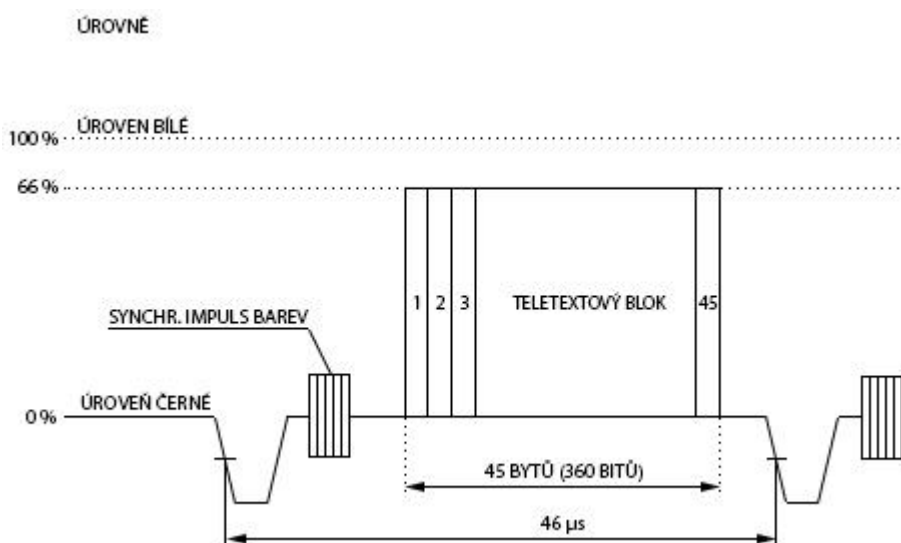


Obr. 9 Ukázka teletextu T

Vzhledem k tomu, že na které informace svým rozsahem přesahují možnosti zobrazení na jedné teletextové stránce, tak může mít každá stránka své časové dělení. Toto časové dělení vytváří různé varianty dané stránky, které se v teletextu jeví jako podstránky. Varianty stránky s časovým

dlením se po uplynutí určitého intervalu stídají. Interval stídání stránek vychází z času, který je nutný pro přetížení stránky pro mnohým divákem. Konzument prohlížející si tuto stránku si však může danou variantu stránky podržet stisknutím tlačítka HOLD na dálkovém ovladači, což má za následek zablokování naítání dalších variant až do zrušení optovným stiskem tlačítka. Po optovném stisknutí tlačítka znovu varianty optotovat. U těchto stránek bývá také nejastěji v pravém horním rohu zobrazován konkrétní počet variant spolu s íslem aktuální varianty. Varianty jednotlivé stránky jsou oznaované ty íslím, které vychází z ísového oznaení stránky a pro každou stránku je takto možné vytvo it až 3 200 variant. [3], [4]

Penos teletextových dat probíhá ve formě blok , které tvo í ucelené informa ní jednotky a nelze je již v rámci penosu d lit. Jeden teletextový blok odpovídající jedné teletextové ádce je tvo en 40 byty pro znaky stránky a 5 byty ídící funkce dekodéru. Takovýto blok je p enesen během periody zatem ovacího ádku s délkou 64  $\mu$ s. Bitová rychlost, s kterou je p enesen teletextový blok odpovídá 6,9375 Mbit/s. Na obr. 10 je zobrazen teletextový blok umíst ěný do ádku zatem ovacího intervalu televizního vysílání. [3], [4]



Obr. 10 Teletextový blok umíst ěný v ádku zatem ovacího intervalu [3]

Teletextové bloky lze rozdlit na úvodní blok a ádkové teletextové bloky. Úvodní blok je ozna ován jako Y0 a odpovídá prvnímu ádku (hlavi ce) teletextové stránky. V tomto bloku jsou p enášeny ídící informace platné pro celou stránku a za nimi jsou míst ěny kódové informace textu hlavi kového ádku teletextové stránky. Protože je teletextový signál synchronní, musí se synchronizovat. Pro tuto synchronizaci jsou určeny první dva byty, ve kterých jsou p enášeny stídav ě hodnoty logické 1 a 0. Tyto byty jsou ozna ovány jako CR (clock run-in). T etí byt ozna ovaný jako FC (framing code) slouží pro identifikaci, že za tímto bytem jsou p enášena teletextová data. Pokud tedy dekodér v tomto bytu nalezne posloupnost bit ů 11101100, pak umožní další zpracování dat. Pokud tuto posloupnost neobjeví, pak ozna í blok jako vadný a nepokra uje dále v jeho zpracování. Za skupinou synchroniza ních a identifika ního bytu (slabik) následuje dvojice slabik (6 a 8 bit ů), které nesou informaci s íslem magazínu (MG) a íslem teletextového bloku (YN). Podle těchto dvou byt ů dekodér zjistí, zda daný blok pat í do magazínu, který si uživatel vybral (bloky daného magazínu a stránky nemusí následovat souvisle za sebou). Jelikož jsou informace v těchto dvou slabikách d ležitě, tak je pro jejich ochranu používáno Hammingova kódu, který umož ůuje opravení kteréhokoliv jednoho vadného bitu. Reaguje na p enos s 2, 4 nebo 6 chybnými bity, ale nedokáže rozeznat chybu p í 3, 5, 7 a 8 bitech. V takto chrán ěných slabikách se stídají ochranné bity s bity datovými, a proto se ve slabice tvo ěné 8 bity p enáší ve skute nosti pouze 4 bity obsahující data a zbývající 4 bity jsou bity ochrannými. Ochranné bity jsou vkládány do slabik podle tabulky 1. Po penosu jsou v dekodéru testovány tyto slabiky tak, že se provedou ty í sou ty XOR A, B, C, D vybraných bit ů dle tabulky 2 a následn ě jsou tyto sou ty kontrolovány na lichou paritu. Výsledky sou t XOR jsou vyhodnocovány podle tabulky 3. Stejným typem zabezpe ení jsou v další ásti bloku

ochrání slabiky 6 a 7 nesoucí informaci o čísle stránky v BCD kódu (binárně kódované desítkové soustavě). V slabikách 8 – 11 je vysílána informace o variantě stránky jak bylo zmíněno výše. Na které bity (4. bit 9. slabiky a 3. a 4. bit 11. slabiky) nejsou využity pro přenos informace o variantě stránky a jsou tedy využívány společně s dalšími 4 a 4 bity (kvůli Hammingovu kódu) slabik 12 a 13 jako řídicí bity pro řízení dekodéru. Dekodér vykoná danou funkci, pokud logická hodnota řídicího bitu je rovna 1. Za řídicími bity se nachází skupina 32 znakových slabik (slabiky 14 až 37) s lichou paritou s jedním zabezpečovacím bitem, která obsahuje znakové kódy hlavičky dané stránky a také aktuální datum. V posledních 8 slabikách hlavičkového bloku se pokračují příznaky. Struktura řádku Y0 je zobrazena na obr. 11. [3], [4]

datové bity							
ochranné bity							
b8	b7	b6	b5	b4	b3	b2	b1
0	0	0	1	0	1	0	1
0	0	0	0	0	0	1	0
0	1	0	0	1	0	0	1
0	1	0	1	1	1	1	0
0	1	1	0	0	1	0	0
0	1	1	1	0	0	1	1
0	0	1	1	1	0	0	0
0	0	1	0	1	1	1	1
1	1	0	1	0	0	0	0
1	1	0	0	0	1	1	1
1	0	0	0	1	1	0	0
1	0	0	1	1	0	1	1
1	0	1	0	0	0	0	1
1	0	1	1	0	1	1	0
1	1	1	1	1	1	0	1
1	1	1	0	1	0	1	0

Tabulka 1 Hamming v kód pro jednu slabiku [4]

	b8	b7	b6	b5	b4	b3	b2	b1
A	o	x	o	x	x	x	o	o
B	o	x	x	x	o	o	o	x
C	x	x	o	o	o	x	o	x
D	o	o	o	o	o	o	o	o

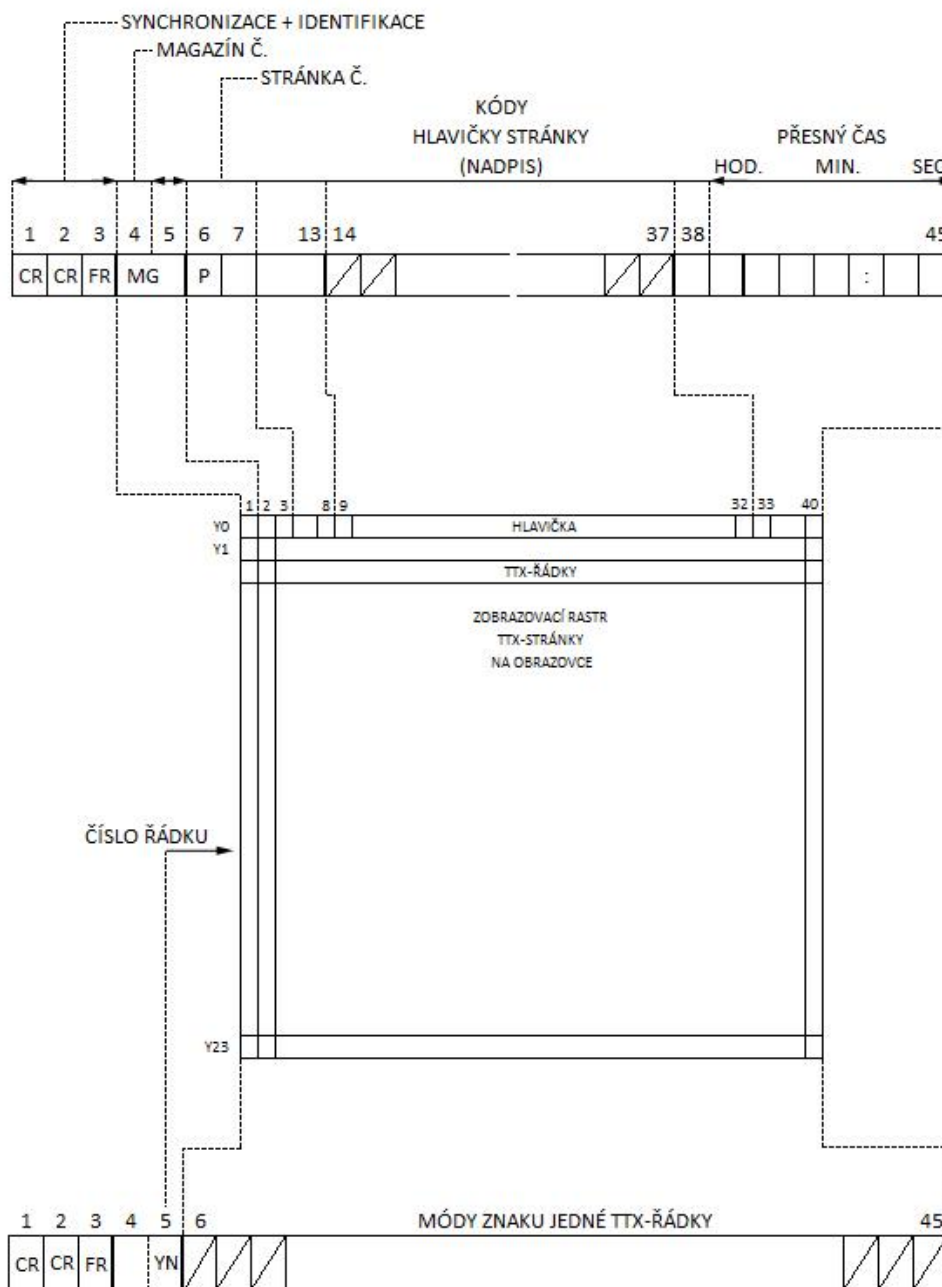
Poznámka: Písmena A, B, C, D značí jsou ty XOR testovaných bitů (označených kroužkem) na lichou paritu

Tabulka 2 Testy na lichou paritu Hammingova kódu [4]

Výsledky test parity A, B, C	parity D	Vyhodnocení	Rozhodnutí
Všechny správné	Správné	Žádná chyba	Použít datové bity HOK
Všechny správné	Nesprávné	Chyba v bitu 7	Použít datové bity HOK
Ne všechny správné	Správné	N kolikanásobná chyba	Odmítnout datové bity HOK
Ne všechny správné	Nesprávné	Jednoduchá chyba	Srovnání s tabulkou 1 identifikovat chybu. Jestliže je v datovém bitu, pak opravit. Použít datové bity. HOK

Poznámka: HOK = test Hammingova kódu v pořádku, ~~HOK~~ = kód není v pořádku

Tabulka 3 Vyhodnocení chyb při testech Hammingova kódu [4]



Obr. 11 Struktura teletextových bloku podle n s umístěním znaků na obrazovce [3]

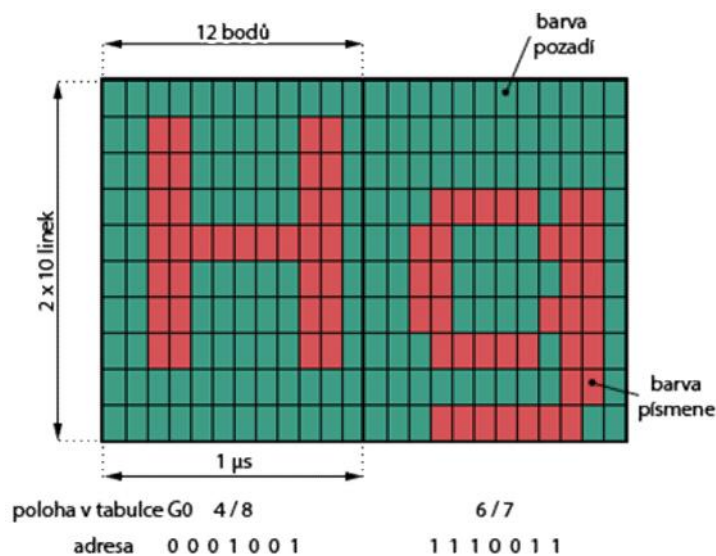
ádkové teletextové bloky (pakety Y1 až Y23) jsou tvořeny úvodními 5 byty (CR, CR, FC, MG, YN) stejně jako úvodní blok následované 40 znakovými slabikami nesoucí pouze instrukce pro dekoder, který podle této informace zobrazí znak nebo grafický element ve znakovém poli. Teletextová řádka odpovídá 10 linkám na obrazovce. Na obr. 11 je také zobrazena struktura ádkového bloku. Z tohoto obrázku je patrné, že umístění kódu v teletextovém bloku je pevně svázáno s umístěním znaku na teletextové stránce. [3], [4]

Znakové slabiky tvoří 7 datových bitů a jeden bit ochranný. Datovými bity je tedy možné adresovat  $2^7 = 128$  znaků, z kterých je 32 znaků určeno jako znaky řídicí. Tyto řídicí znaky umožňují zobrazovat různé znaky: barvu pozadí, písma, blikání písmen, výšku písma, apod. Aby bylo možné adresovat řídicí znaky i pro abecedy s více znaky než je abeceda anglická, jsou datové bity rozděleny na 4 bity ádkové ( $b_1, b_2, b_3, b_4$ ) a tři bity sloupcové ( $b_5, b_6, b_7$ ), čímž se získá tabulka s 16 řádky a 8 sloupci. Takováto tabulka pro první úroveň teletextu je zobrazena v tabulce 4. V této tabulce se ve sloupci 0 nacházejí písmena určená alfanumerickým znakům a ve sloupci 1 písmena mozaikové

grafiky. P íznakem se rozumí vlastnost p id lenou zobrazovacím znak m znakem ídícím. Dekodér rozhoduje, zda má použít alfanumerický znak nebo grafický element podle p edchozího znakového povelu. Nap . pro zobrazení celého ádku textu s ervenou barvou se první zobrazovací znak nezobrazí a znakový obdelník má barvu pozadí (první slabika p ísluší svou bitovou skladbou ádku 1 a sloupci 0 tabulky 4). Pokud by se m la v pr b hu ádku zm nit vlastnost textu, tak by muselo p edcházet této zm n prázdné znakové pole, které by obsahovalo informaci jaký p íznak se má pro následující znakový obdelník. Každý znakový obdelník je tvo en rastrem 12 x 10 bod (obr. 12). [3], [4]

bity				(b <sub>3</sub> )	0	0	0	0	0	0	0	0	0	0	0	
bity				b <sub>7</sub>	0	0	0	1	1	1	1					
bity				b <sub>6</sub>	0	0	1	1	0	0	1	1				
bity				b <sub>5</sub>	0	1	0	1	0	1	0	1				
b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	R/S	0	1	2	2a	3	3a	4	5	6	6a	7	7a
0	0	0	0	0	znaky:	grafika			0		č	P	é		p	
0	0	0	1	1	černé	černá	!		1		A	Q	a		q	
0	0	1	0	2	červené	červená	"		2		B	R	b		r	
0	0	1	1	3	zelené	zelená	#		3		C	S	c		s	
0	1	0	0	4	žluté	žlutá	ú		4		D	T	d		t	
0	1	0	1	5	modré	modrá	%		5		E	U	e		u	
0	1	1	0	6	purpurové	purpurová	&		6		F	V	f		v	
0	1	1	1	7	modrozelené	modrozelená	'		7		G	W	g		w	
1	0	0	0	8	bílé *	bilá	(		8		H	X	h		x	
1	0	0	1	9	blikání	skrytí **	)		9		I	Y	i		y	
1	0	0	1	9	ustálení (konec blikání) *, **	spojitá grafika *, **	*		:		J	Z	j		z	
1	0	1	0	10	konec vkládání *, **	oddělená grafika **	+		;		K	ř	k		á	
1	0	1	1	11	Začátek vkládání	ESC (na úrovni 1 se nepoužívá)	,		<		L	ž	l		ě	
1	1	0	0	12	normální výška *, **	černé pozadí	-		=		M	ý	m		ů	
1	1	0	1	13	dvojitá výška	nové pozadí	.		>		N	í	n		š	
1	1	1	0	14	na úrovni 1 se nepoužívá	Sevřená grafika	/		?		O	ř	o			
1	1	1	1	15	na úrovni 1 se nepoužívá	Nesevřená grafika *										

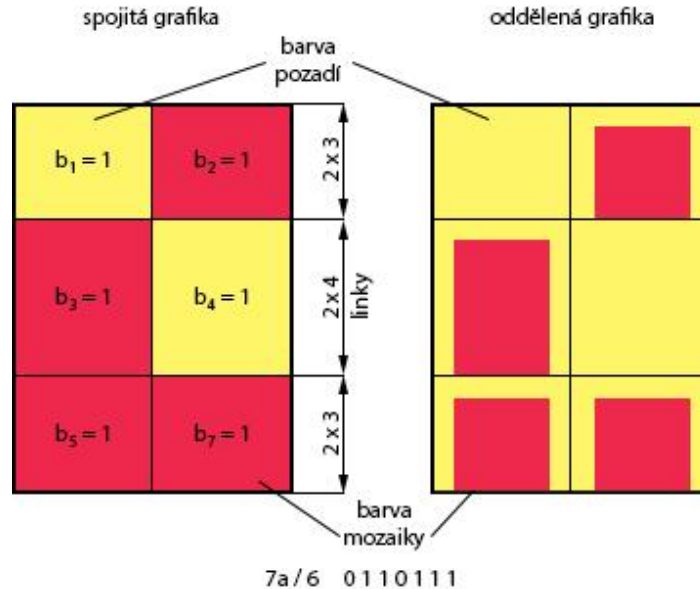
Tabulka 4 Tabulka znak , povel a grafických element teletextu první úrovn [4]



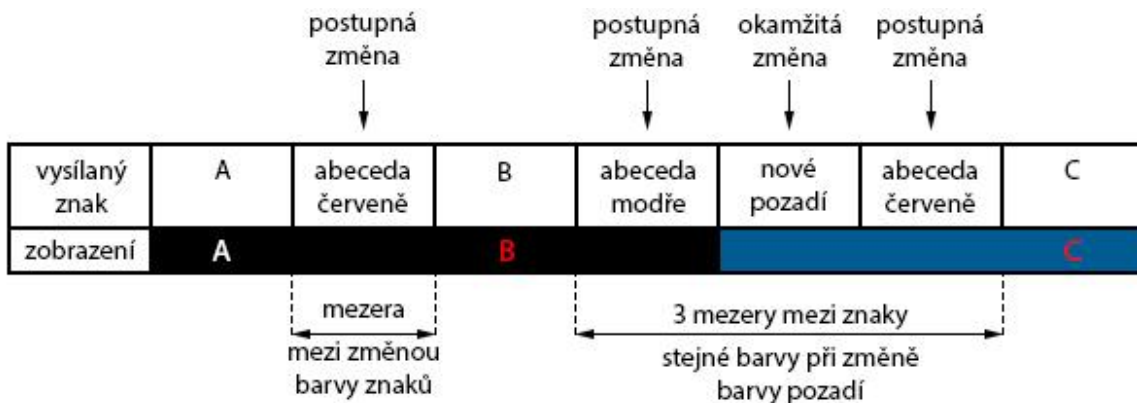
Obr. 12 Znakový obdelník 12 x 10 bod [4]

Znakové pole tvořené grafickým elementem se skládá z mozaiky s rastrem 2 x 3 plošek. Pomocí píznaku lze zobrazit grafický element jako spojitý (píznak na řádku 9 a sloupci 1), kde se zobrazované části dotýkají, nebo jako oddělenou (píznak na řádku 10 sloupci 1), u níž jsou jednotlivé elementy odděleny proužkem pozadí. Ukázka obou grafických elementů je na obr. 13. U omezeného počtu píchodů barvy se používá tzv. sevřená grafika, kdy se místo píchodového řídícího znaku zobrazí znak píchodového znaku v pívodní barvě. [3], [4]

Změna barevnosti pozadí se uskutečňuje pomocí řídícího znaku řádku 13 a sloupce 1. Barva pozadí se změní na barvu píchodového textu. Tato procedura je patrná z obr. 14. [4]



Obr. 13 Struktura grafických elementů [4]



Obr. 14 Písobení sériových píznaků v teletextu. [4]

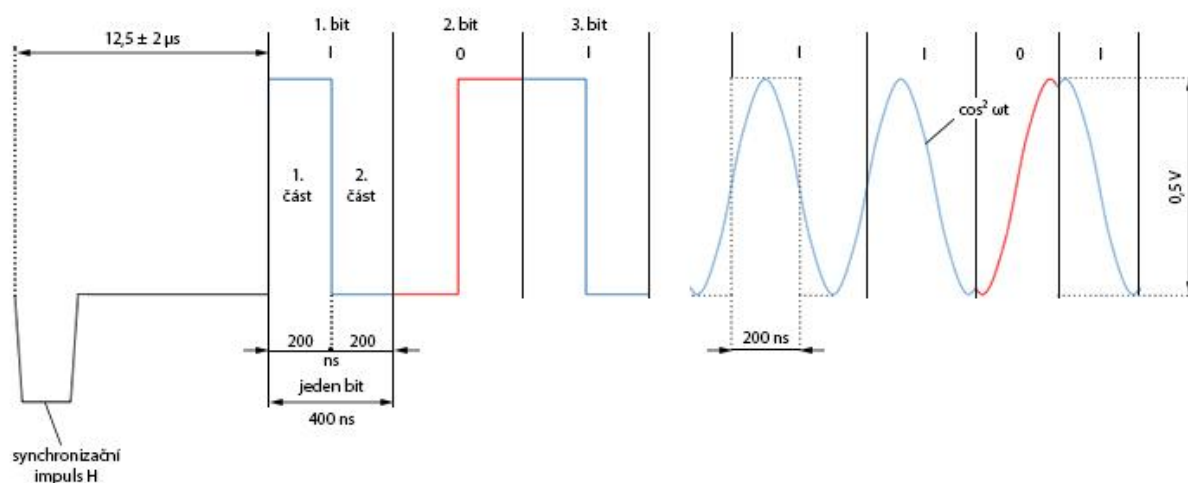
Změnou způsobu přenosu televizního vysílání z analogového na digitální není ukončen přenos teletextových informací. Přenos teletextových informací se pouze přesunul z úseku záměrného intervalu do části datového toku určeného pro dodatkové informace televizního vysílání. Grafický vzhled teletextových stránek zůstal nezměněn, ale doba na čtení jednotlivých stránek je mnohonásobně nižší, protože dekodéry digitálního vysílání obsahují v paměti a mohou mít v sobě uložené informace více teletextových stránek.

### 2.3 VPS a PDC

VPS (Video Programming system) je dodatkovou službou vysílanou podobně jako teletext v úseku záměrného intervalu analogového televizního vysílání. Tato informace vysílaná na 16. řádku obrazového signálu umožňuje naprogramování přesného zahájení a ukončení záznamu

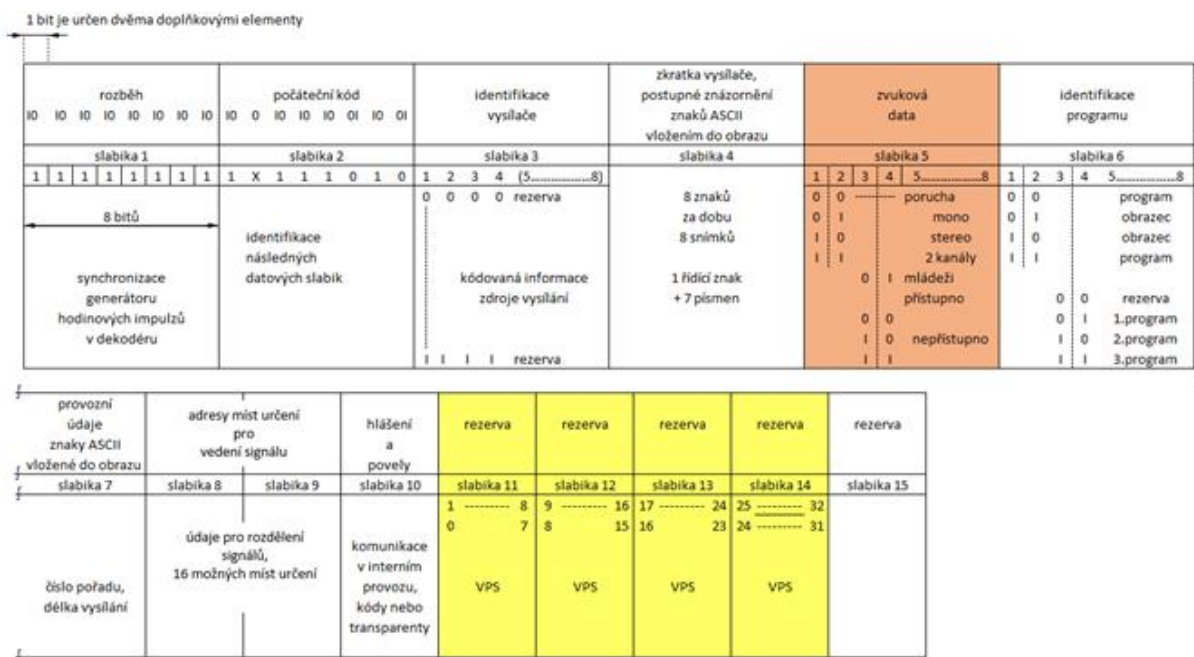
po adu na p i pojený videorekordér. Videorekordéry, které umož ůjí využití systému VPS tak mohou zaznamenat daný po ad od jeho p esného za átku, i když tento za átek m že být posunut oproti p vodnímu plánu. Naprogramování videorekordéru s použitím systému VPS se provádí zadáním íselného kódu daného po adu, který lze najít nap . v tisku s televizním programem. Tento kód se zadá do videorekordéru, který obsahuje dekodér dat VPS. Videorekordér dekóduje informace z 16. ádku obrazového signálu a porovná je s kódem zadaným uživatelem, pokud se tyto data shodují, tak spustí záznam ve správný okamžik. Videorekordér zaznamenává po ad po celou dobu, kdy je v obrazovém signále obsažena totožná informace jako informace zadaná uživatelem p i programování záznamu. Videorekordér ukon í zaznamenávání ve chvíli, kdy se v 16. ádku objeví kód jiného po adu nebo tzv. prázdný i p erušovaný kód. [4]

Data p enášená v 16. ádku obrazového televizního signálu jsou tvo ena patnácti osmi bitovými slabikami. Pro kódování jednotlivých bit dat 16. ádku se používá tzv. dvoufázová metoda, kdy je každý bit tvo en vzájemn dopl kovými logickými úrovn mi 1 a 0. Na následujícím obrázku je vid t p íklad kódování pomocí této metody. Z obrázku je patrné, že pro logickou hodnotu bitu je p enášená v první polovin intervalu vysoká úrove a p esn v polovin se zm ní na úrove nízkou, která trvá zbylou ást intervalu. P enos bitu s logickou úrovní 0 je opa ný. Díky skute nosti, že se p echody mezi nízkou a vysokou hodnotou u bitu s logickou 0 a p echody mezi vysokou a nízkou hodnotou u bitu s logickou 1 nacházejí vždy uprost ed intervalu, tak je možné využívat tohoto okamžiku k synchronizaci. [4]



Obr. 15 P íklad dvoufázové modulace v 16. ádku obrazového televizního signálu [4]

Ne všechna data v 16. ádku složená z 15 osmibitových slabik tvo í kód VPS. První slabika tohoto ádku zajiš uje synchronizaci hodinových impulz dekodéru p íjmací strany a je ozna ována jako slabika rozb hová. Startovací slabika, která je druhá v po adí, je tvo ena um le vytvá eným nesprávným sledem v druhém bitu této slabiky, ímž je identifikován následující íselný p enos. T etí slabika ve svých prvních ty ech bitech nese kód zdroje vysílání. Zbylé ty i bity t etí slabiky jsou rezervní. tvrtá slabika umož uje zobrazení zkratky zdroje vysílání ze t etí slabiky. Pátá slabika obsahuje informace jaký je spolu s obrazem p enášen zvukový doprovod a informace o p ístupnosti po adu, jenž m že blokovat nahrávání pomocí VPS. Slabiky 6 – 10 nesou informace pro ur ení obsahu vysílání a jejich možném vložení do televizního obrazu, dále také íslo p íslušného programu a jeho délce. Další ty i slabiky nesou informace VPS a poslední 15. slabika je použita jako rezerva. Schématicky je struktura 16. ádku zobrazena na obr. 16. [4]



Obr. 16 Struktura dat 16. ádku v p l snímkovém zatem ovacím intervalu [4]

Informace VPS je složena z 32 bit , které poskytují 4 slabiky 16. ádku. Následující tabulka udává po ty a po adí bit pro jednotlivé položky kódu VPS spolu s možnostmi rozlišení. [4]

	Po adí bit	Po et bit	Možnosti rozlišení
Adresa oblastí	0 a 1	2	4*
Den	2 až 6	5	31 + 1 = 32
M síc	7 až 10	4	12 + 4 = 16
Hodina	11 až 15	5	24 + 8 = 32
Minuta	16 až 21	6	60 + 4 = 64
Národnost	22 až 25	4	16
Zdroj programu	26 až 31	6	64

\* Zdroje program jsou rozd leny do ty oblastí. Zatím se používají první dv adresy II a I0. Každé oblasti m že být p i azeno až 64 zdroj program (televizních vysílání).

Tabulka 5 Skladba informace VPS [4]

Schématicky je formát dat VPS zobrazen na obr. 17, kde jsou zobrazeny také zobrazeny t i speciální kódy: stavový, prázdný a p erušovací kód. Pi správné zna ce (data VPS) je po ad vyslán v regulérním kódu a záznam je ízen podle shody s tímto kódem. Pokud však daný po ad není ozna en signálem VPS, tak zna ka obsahuje stavový kód s nelogickými údaji: den 0, m síc 15, hodina 31, minuty 63. Stavovému kódu je podobný prázdný kód, který má rozdílnou hodnotu pouze u položky hodin (30) a jsou jím ozna eny po ady, jenž se nemají zaznamenávat. Tento kód se také objevuje dv minuty p ed a po po adu, který je ízen signálem VPS. Posledním uvedeným je kód p erušení s hodnotami: den 0, m síc 15, hodina 29, minuta 63. Tento kód p edává videorekordéru p íkaz pro pozastavení záznamu, než se op t objeví regulérní kód se správnou zna kou. Použití tohoto kódu je tam, kde program s jednou regulérní zna kou p erušuje jiný program vysílaný také s regulérním kódem. [4]

		Informace VPS																																									
slabika		11						12						13						14																							
bit		1	-	-	-	-	-	8	1	-	-	-	-	-	8	1	-	-	-	-	-	8	1	-	-	-	-	-	8														
bit VPS		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31										
binární váha																																											
regulérní kód	Adresy	den binárně						měsíc binárně						hodina binárně						minuta binárně						země binárně		zdroj programu binárně															
zvláštní kódy:																																											
stavový kód	A A	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	N	-	-	N	P	-	-	-	-	P	
prázdný kód	A A	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	N	-	-	N	P	-	-	-	-	P
přerušovací kód	A A	0	0	0	0	0	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	N	-	-	N	P	-	-	-	-	P	

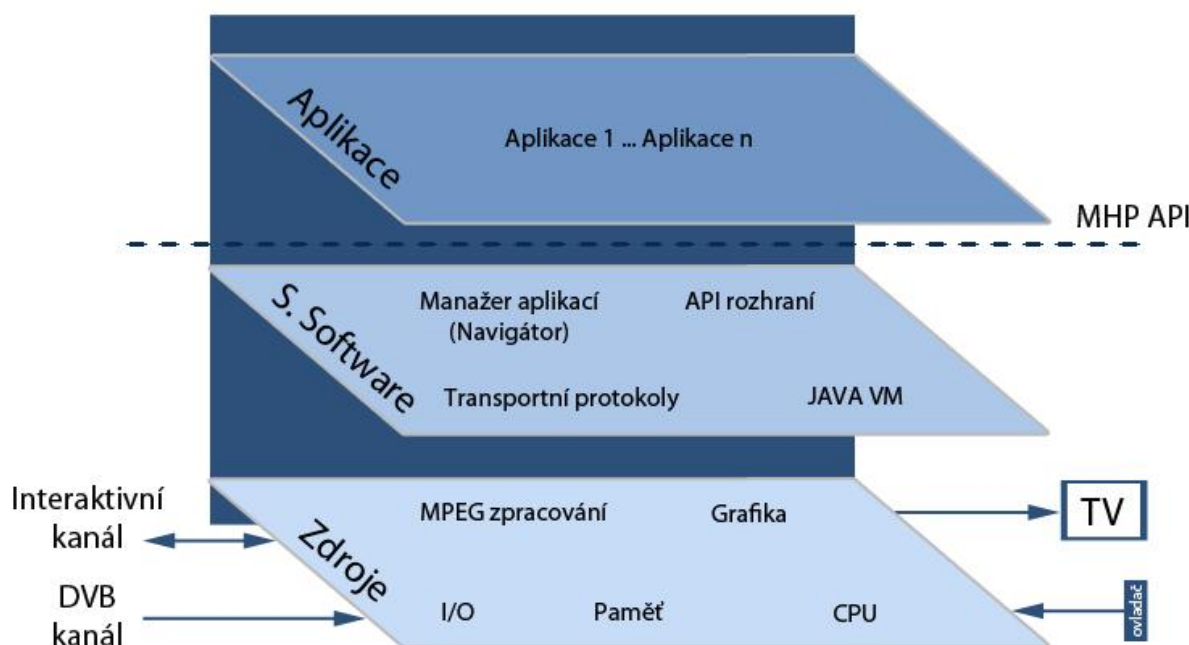
M = MSB L = LSB A = současná adresa oblasti N = současný národní kód P = současný kód programu

Obr. 17 Formát dat VPS [4]

Systém VPS byl ještě v analogovém vysílání nahrazen systémem PDC (Programme Delivery Control), který má stejnou funkci, ale informace již není přenášena přímo v rámci každého zátvarového intervalu, ale v rámci servisního teletextového bloku.

## 2.4 MHP

Evropský standard digitálního televizního vysílání DVB umožňuje fungování interaktivních služeb, jež jsou založeny na platformě MHP (Multimedia Home Platform). Tato platforma vychází z platformy DVB-J, která definuje použití Java Virtual Machine firmy Oracle Corporation pro digitální televizní vysílání. Architektura platformy je tvořena třemi vrstvami: zdrojovou, vrstvou systémového (programového) vybavení a vrstvou aplikací. Hardware není pro platformu MHP definován. Na obr. 18 jsou zobrazeny zmínované základní vrstvy platformy MHP. [1]



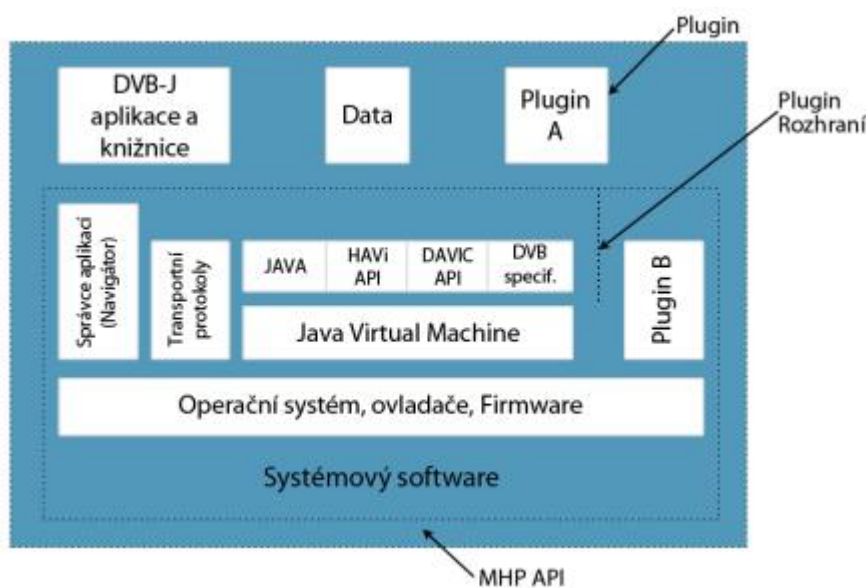
Obr. 18 Vrstvy MHP [1]

První vrstvu tvoří zdroje hardwarových a softwarových prostředků, které jsou nutné pro zpracování obrazových a zvukových dat. K typickým zdrojům patří zpracování formátu MPEG, vstupní a výstupní zařízení, procesor, paměť a grafický systém, které umožňují přístup k dostupným aplikacím.

rozhraním. Střední vrstva systémového vybavení využívá zdroj první vrstvy k propojení jednotlivých platform k daným aplikacím. Tuto vrstvu lze chápat jako operační systém. Součástí této vrstvy je také manažer aplikací (navigátor) řídící spuštění, přechy a ukončení jednotlivých aplikací. Střední vrstva mimo jiné vytváří podporu základních transportních protokolů. Součástí této vrstvy je také již zmíněný Java Virtual Machine, oddávající hardware a software od aplikačního rozhraní API. Samotné aplikace tvoří poslední vrstvu architektury MHP přistupují k platformě pouze přes aplikační rozhraní, a proto musí splňovat podmínky dané rozhraním API. Toto API je definováno systémem DVB-J. [1]

Označení platformy DVB-J vychází z toho, že Java Virtual Machine je založený na programovacím jazyku Java. Na následujícím obrázku je zobrazena struktura platformy DVB-J. Z tohoto obrázku je patrné, že systémový software je tvořen operačním systémem, ovladači a firmwarem. Transportní protokoly jsou povinné protokoly, pomocí kterých je zajištěno přenášení dat směrem od poskytovatele k divákovi (broadcast channel), a protokoly zajišťující interaktivní zpětný kanál (interactive channel). Aplikační rozhraní lze rozdělit na:

- Aplikační rozhraní JAVA obsahující:
  - základní JAVA (jazyk, utility, ...)
  - prezentační rozhraní (Java Media Framework)
  - rozhraní pro výběr služeb (Java TV)
- Rozhraní definované standardem DVB obsahující:
  - rozšíření/omezení pro rozhraní JAVA
  - rozhraní pro přístup k datům
  - rozhraní servisních informací
  - rozhraní pro přístup k vstupní-výstupním zařízením
  - rozhraní pro zabezpečení
  - další rozhraní (uživatelská nastavení, ...) [1]



Obr. 19 Platforma DVB-J [1]

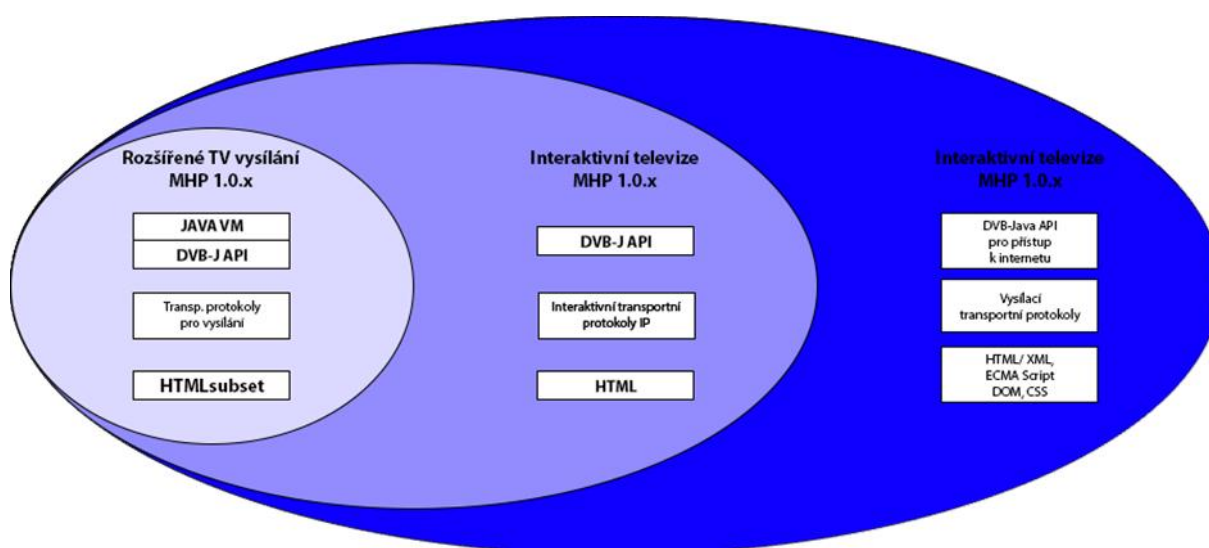
MHP je možné podle oblasti použití rozdělit do tří základních profilů (obr. 20):

- Rozšířené televizní vysílání (Enhanced Broadcast) – pro set-top boxy bez zpětného kanálu s pasivní interaktivitou (veškerá data jsou obsažena ve vysílacím kanálu); profil je definován ve specifikaci MHP 1.0, obsahuje aplikace Java VM, DVB-J API a transportní protokoly pro vysílání.
- interaktivní televize (Interactive TV) – set-top boxy obsahující zpětný kanál a umožňující tak vyšší stupeň interaktivity, profil také obsažen ve specifikaci MHP 1.0; obsahuje rozšířené aplikace DVB-J API pro interaktivitu a interaktivní transportní protokoly.

- Přístup k internetu (Internet access) – pro set-top boxy s vysokým výpočetním výkonem, velkou pamětí a nejvyšším stupněm interaktivity, profil je definován v MHP 1.1, obsahuje Java API pro přístup k internetu, transportní protokoly pro vysílání IP a další. [1]

Aplikace vytvářené pro platformu MHP mohou být napsány v jazyce Java nebo HTML. MHP aplikace nemusí být plně kompatibilní s normami obou uvedených jazyků. Obsahí všechny informace o aplikacích jsou přenášeny v rámci transportního toku spolu s MPEG-2 datovým tokem daného vysílaného programu a je složena ze dvou částí:

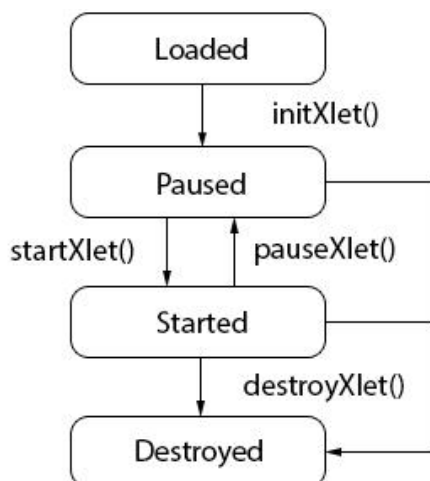
- Soubory tvořící obsah aplikace, nastavení a data jsou přenášena DSM-CC – karuselem objektů. Jedná se o systém souborů potřebovaných pro běh aplikace.
- Jelikož nestačí jen tak vysílat soubory, ale je nutné, aby příjemce byl informován o tom, že tyto přenášené soubory tvoří danou aplikaci, a také musí znát způsob, jakým má aplikaci spustit. Proto MHP definuje novou tabulku servisních informací Application Information Table (AIT), z níž se příjemce dozví, jaké aplikace jsou dostupné a dále jakým způsobem se mají spouštět (signální informace). [1]



Obr. 20 Oblasti použití platformy MHP [1]

Aplikace vytvářené v jazyce Java jsou označovány jako applety. Protože u platformy MHP se vyskytují odlišnosti od klasického jazyka Java, tak i aplikace vytvářené pod touto platformou jsou označovány odlišně jako Xlet. Tyto Xlety byly poprvé prezentovány ve specifikacích Java TV. Xlet může nabývat 4 hlavních stavů, mezi nimiž přechází během svého životního cyklu. Tyto stavy jsou: Loaded, Paused, Started a Destroyed. Na zápisích zpracovávajících digitální televizní signál může být sice spuštěno více aplikací najednou, ale kvůli hardwarovému omezení je možné v reálném čase sledovat pouze jednu aplikaci. Ostatní aplikace jsou po tuto dobu pozastaveny, aby mohly být veškeré zdroje použity pro aplikaci, která je aktuálně zobrazena. Z toho vyplývá, že Xlet není standardní Java aplikací. Každý Xlet má svůj životní cyklus:

1. Správce aplikací načte hlavní soubor Xletu (stav „Loaded“)
2. Uživatel vybere ve správci spuštění Xletu
3. Xlet se inicializuje – `initXlet()` a zůstane ve stavu „Paused“ (čas pro načtení objemnějších grafických prvků apod.)
4. Správce zavolá po inicializaci `startXlet()`, čímž přejde Xlet ze stavu Paused do stavu Started a výsledek se zobrazí na obrazovce.
5. Během spuštění Xletu může aplikace správce zavolat `pauseXlet()`, čímž přejde spuštěná aplikace do stavu „Paused“. Zavoláním `startXlet()` přejde aplikace znovu do stavu „Started“. Tyto přechody se mohou opakovat několikrát během života Xletu.
6. Při ukončení aplikace zavolá aplikace správce `destroyXlet()` a ukončí tak danou aplikaci, uvolní prostředky a daný Xlet již není možné znovu spustit. [1]



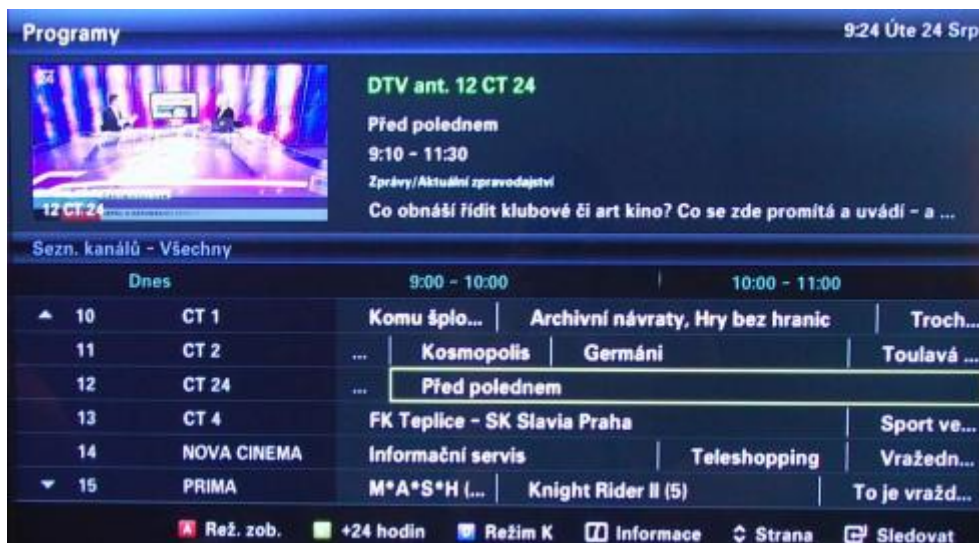
Obr. 21 Stavový diagram životního cyklu Xletu [1]

Typickým příkladem MHP aplikace je EPG (Electronic Program Guide), který využívá lokální interaktivitu. Mezi další příklady aplikací s lokální interaktivitou patří informační služby (zprávy, sport, informace z burzy) označované jako „super teletext“, dále také různé lokální interaktivní hry, které jsou synchronizovány dle televizního programu. Dále různé aplikace pro vzdělávání, nákup zboží využívají interaktivní vysílání se stejným kanálem. Příkladem aplikací s internetovým přístupem mohou být aplikace pro teniš e-mail, prohlížení internetových stránek nebo také chatování.

#### 2.4.1 EPG

EPG je obdobou tištěného televizního průvodce a je součástí každého digitálního přenášeného televizního programu. Tato MHP aplikace se vyvolává na příjímáči stisknutím příslušného tlačítka (EPG, INFO, GUIDE, apod.) a umožňuje zobrazovat televizního průvodce s informacemi o jednotlivých programech až na 7 dní dopředu. Protože norma pro EPG se zabývá pouze způsobem a formátem přenášených dat a nikterak nespecifikuje grafické zobrazení přenášených dat, tak je vizualizace aplikace EPG dána výrobcem příjímáče. Proto může EPG vypadat různě podle výrobce nebo podle dané adresy příjímáče, i dokonce i podle verze firmwaru příjímáče. Možností, jak mohou být data přenášena v digitálním televizním signále prezentována, je několik. První možností je zobrazení programu ve formě tabulky s výpisem zaátku a názvu programu vždy pro jeden konkrétní televizní kanál (single EPG). Nebo druhou možností je zobrazení EPG pro více televizních kanálů (multiple EPG) ve formě bloku, kde je lépe vidět vzájemné posloupnosti programů na různých kanálech. Obě tyto možnosti je možné vidět na následujících dvou obrázcích. [5]

Společně se záznamovými zařízeními může být EPG použito pro jednoduché programování nahrávání programu.



Obr. 22 Multiple EPG



Obr. 23 Single EPG [6]

## 2.5 HbbTV

S nástupem tzv. Smart televizí obsahujících způsoby interaktivní kanál pro připojení k internetu se objevují různé aplikace (widgety) pro sledování například archiv televizních společností, prohlížení internetu apod. Tyto widgety lze stáhnout na televizní přijímač z portálu výrobce konkrétního zařízení a aplikace jsou tedy vytvářeny pro televizory konkrétních výrobců, čímž může docházet k nekompatibilitám s různými zařízeními. Kvůli takovéto nekompatibilitě mezi výrobky různých výrobců nebo dokonce mezi různými modely jednoho výrobce se v Evropě objevuje nová platforma. Jedná se o platformu využívající technologii digitálního televizního vysílání v kombinaci s širokopásmovým internetem. Poprvé je tato platforma představena během francouzských veletrhů IBC a IFA v roce 2009, kde ji společnost France Télévision prezentovala na příležitostech prestižního tenisového turnaje Roland Garros s využitím zemského televizního vysílání (DVB-T) a IP technologie. [7]

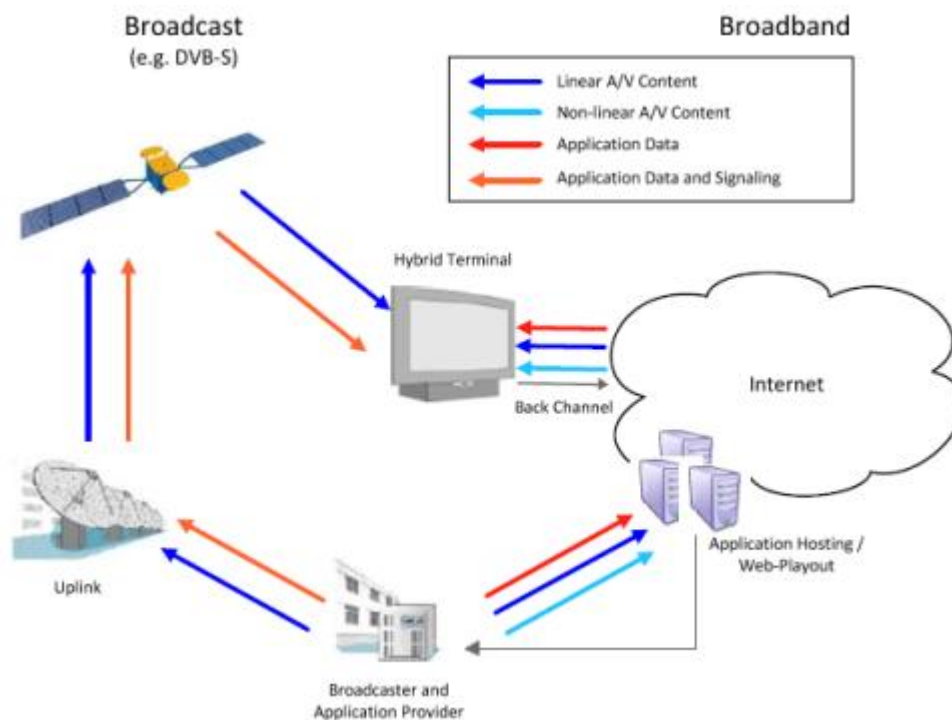
Během následujícího roku byla schválena specifikace pro platformu HbbTV s označením ETSI TS 102 796 V. 1.1.1. Na schválení normy se podílela řada organizací jako například Evropský ústav pro telekomunikační normy (ETSI), Evropská vysílací unie (EBU) nebo Evropský výbor

pro elektrotechnickou normalizaci (CENELEC). V roce 2010, kdy byla tedy schválena specifikace pro HbbTV, vzniká také stejnojmenné konsorcium sdružující n které renomované výrobce televizních přijímačů a televizních společností (Astra, Canal+, France Télévisions, Open TV, TF1, Philips, Humax, Opera Software a další). Cílem tohoto konsorcia je univerzálnost použití aplikace HbbTV na televizních přijímačích různých výrobců a typů. Tuto univerzálnost má zajistit logo HbbTV (obr. 24). Pokud tedy uživatel spatří logo HbbTV, tak si může být jist, že mu budou všechny aplikace na tomto zařízení pracovat. [7]

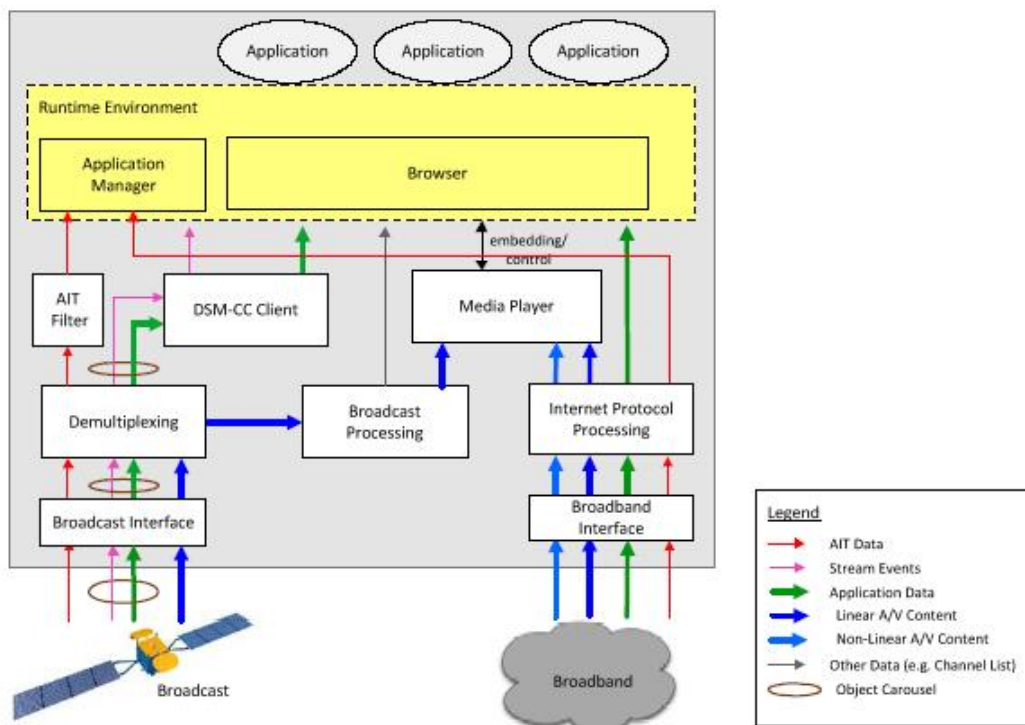


Obr. 24 Logo HbbTV

Hybridní terminál v architektuře HbbTV (obr. 25) je paralelně připojen ke dvěma sítím. Na jedné straně je připojen k jedné nebo několika platformám digitálního vysílání DVB (DVB-T, DVB-S nebo DVB-C). Tímto připojením může přijímat standardní audiovizuální obsah (lineární audiovizuální obsah) přenášený v reálném čase spolu s aplikacími daty a signálními daty aplikací HbbTV. V případě, že terminál není připojen k širokopásmovému připojení, umožňuje přijímat žádosti o související aplikace. Na druhé straně může být terminál připojen obousměrným kanálem k širokopásmové síti internet, které umožňuje obousměrnou komunikaci s poskytovatelem aplikace HbbTV. Přes toto širokopásmové rozhraní jsou přijímána data aplikací a také nelineární audiovizuální obsah. Nelineární audiovizuální obsah i aplikace jsou vyvolávány na vyžádání uživatelem a není nijak svázáno s přenosem lineárního audiovizuálního obsahu v televizním vysílání. Díky takovému přenosu nejsou zvyšovány nároky na šířku pásma televizního vysílání, protože zde je v nejjednodušším případě přenášena pouze URL aplikace. [7]



Obr. 25 Architektura HbbTV [8]



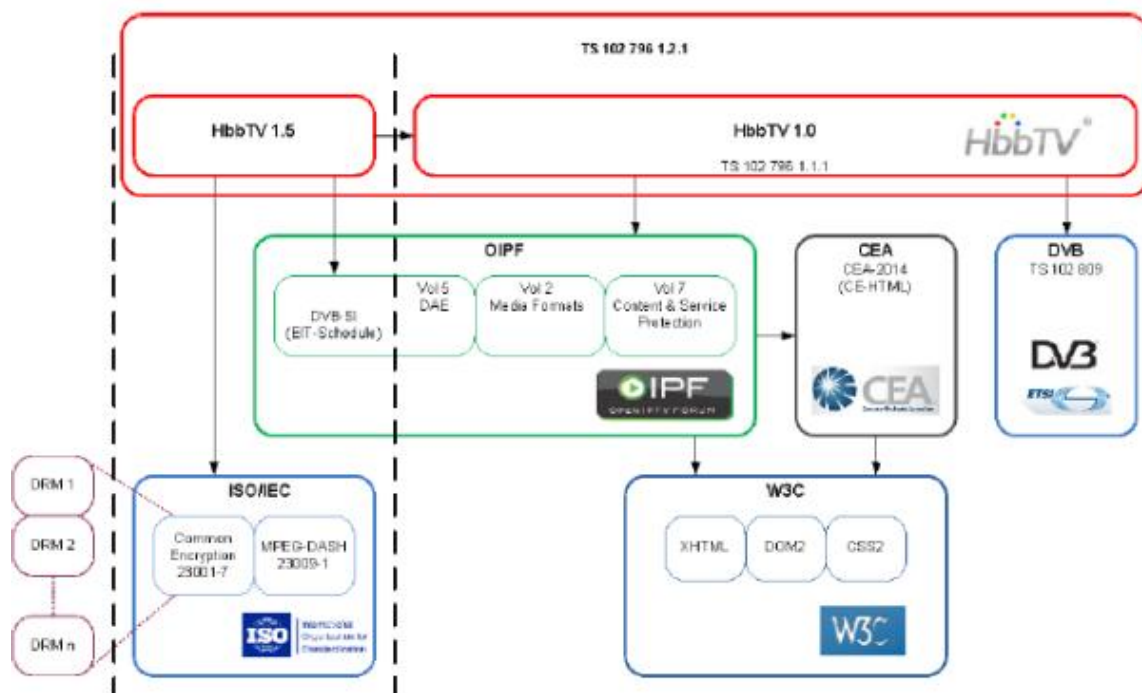
Obr. 26 Blokové schéma terminálu HbbTV [8]

Obr. 26 znázorňuje funkční komponenty terminálu HbbTV. Dle tohoto obrázku přijímá hybridní terminál prostřednictvím rozhraní televizního vysílání data tabulky AIT (Applikation Information Table), lineární audiovizuální obsah, data aplikací a také data událostí vysílaných v reálném čase. Poslední dva zmíněné datové toky jsou přenášeny v objektu karuselu DSM-CC (Digital Storage Media – Command and Control). V tomto objektu jsou přenášena nepřetržitě opakující se data pro více přijímačů najednou. DSM-CC objekt dále postupuje do bloku DSM-CC klienta, kde jsou data z objektu obnovena a předána do rozhraní Runtime Environment. Rozhraní Runtime Environment je prostředím pro spuštění, běh a zobrazování audiovizuálního obsahu aplikací HbbTV na terminálu. Z obrázku je vidět, že toto rozhraní je tvořeno prohlížečem (Browser) a správcem aplikací (Application Manager). Správce aplikací vyhodnocuje filtrovaná data z tabulky AIT sloužící pro řízení životního cyklu interaktivních aplikací. Prohlížeč pak zajišťuje provádění a zobrazení těchto aplikací. Zpracování lineárního audiovizuálního obsahu se provádí po demultiplexování transportního toku TS v bloku pro zpracování bloku televizního vysílání (Broadcast Processing) a zpracování je stejné jako u standardního DVB přijímače. Některé informace a funkce z bloku zpracování televizního vysílání mohou být přístupné v rozhraní Runtime Environment a v obrázku jsou označeny jako jiné data. Jedná se například o seznam televizních kanálů, informace o běžících nebo následujících událostech, funkce ladění atd. Navíc lze do interaktivní aplikace vložit lineární audiovizuální obsah v různé velikosti. Tyto funkce zajišťuje blok označený jako přehrávač médií (Media Player), který má na starosti veškeré zpracování audiovizuálního obsahu. [7], [8], [9]

Prostřednictvím širokopásmového rozhraní zajišťujícího připojení k internetu, které je druhou možností jak lze požádat o aplikační data ze serveru poskytovatele interaktivní aplikace. Také toto rozhraní se používá pro příjem audiovizuálního obsahu (například obsah vyžadovaný aplikací). Všechna data přicházející z internetu zpracovává blok pro zpracování internetového protokolu (Internet Protocol Processing). Prostřednictvím tohoto bloku jsou předána rozhraní Runtime Environment. Audiovizuální obsah je z bloku zpracování internetového protokolu předán dále do bloku přehrávače médií, který je také řízen rozhraním Runtime Environment a audiovizuální obsah tak může být předán aplikaci. [7]

Specifikace HbbTV byla vyvinuta tak, aby efektivně řídila rychle rostoucí množství dostupného obsahu zaměřeného na dnešního konečného spotřebitele. Celá platforma je založena na prvcích stávajících standardů a webových technologiích v rámci OIPF (Open IPTV Forum), CEA-

2014 (CE-HTML), W3C(HTML, atd.) a specifikacích DVB (ETSI 102 809). Níže uvedený graf zobrazuje vztah mezi HbbTV a dalšími stávajícími normami. [9]



Obr. 27 Pohled specifikace HbbTV [9]

CEA-2014 je standardem známým též jako HTML-CE. Jedná se základní webový protokol a rámec pro vzdálené uživatelské rozhraní v UPnP (Universal Plug and Play) sítích a internetu. Mezi důležité komponenty ze standardu CEA-2014 využívané v HbbTV standardu jsou: [9]

- definice jazyka aplikace (XHTML, CSS a Javascriptu v etn AJAXu)
- definice vkládání nelineárního audiovizuálního obsahu do aplikace
- definice DOM (Document Object Model) pro zpracování událostí (nap . klí ové události)
- specifikace statických obrazových formát [9]

CEA-2014 je již také profilováno prostřednictvím specifikace OIPF DAE (Open IPTV Forum Declarative Application Environment). Specifikace HbbTV ve své definici ještě dále profiluje HTML-CE. [9]






Dalšími důležitými komponentami definovanými ve specifikaci OIPF DAE využitými ve standardu HbbTV jsou:

- JavaScript API pro aplikace běžící v televizním prostředí (nap . zm na kanálu)
- definice vkládání lineárního audiovizuálního obsahu do aplikace

Ze standardu televizního vysílání DVB se v HbbTV využívá:

- signalizace aplikace
- využití transportu aplikace přes DVB (DSM-CC)
- událostí toku [9]

Standard HbbTV umožňuje různé způsoby zobrazení aplikace. Následující tabulka ilustruje různé možnosti zobrazení aplikace v běžném vysílání [9]

1. Standardní televizní vysílání	
2. Televizní vysílání se zobrazením dostupných dodatkových informací	
3. Televizní vysílání p ekryté dodatkovými informacemi (umož uje zobrazovat obrázky, neumož uje p ehrávání nelineárního obsahu)	
4. Samostatné zobrazení dodatkových informací bez audiovizuálního obsahu	
5. Plné zobrazení dodatkových informací s audiovizuálním obsahem	

Tabulka 6 Zp soby zobrazení aplikace z pohledu koncového uživatele [9]

Pro standard lze použít pouze dané formáty obrázk , audio a video soubor . Typy formát podporovaných v HbbTV aplikacích jsou definovány ve specifikaci OIPF. [9]

- Obrázkové formáty, jenž lze v aplikaci použít jsou: JPEG, GIF a PNG.
- Video soubory mohou být dvou typ MP4 a TS. Podrobn jší specifikace podporovaných video soubor jsou uvedeny v následující tabulce. [9]

Formát	Video formát	Audio formát	Formát titulk	MIME type
TS	AVC_SD_25 AVC_HD_25	HEAAC E-AC3 *	**	video/mpeg
MP4	AVC_SD_25 AVC_HD_25	HEAAC E-AC3 *	Není definováno	video/mp4
* Terminály budou podporovat E-AC3 v obsahu obdrženého přes širokopásmové připojení, pokud je formát podporován v televizním vysílání.				
** Terminál bude podporovat stejné formáty titulk jako v televizním vysílání				

Tabulka 7 Podporované formáty video soubor [9]

- Podporované formáty samostatných audio soubor jsou:

Audio formát	MIME type	Poznámka
MPEG1_L3	audio/mpeg	
HEAAC	audio/mp4	Jedná se o přepravu HE-AAC audio formátu uvnitř kontejneru formátu systému MP4. Tento formát musí být v souladu s požadavky stanovenými ve specifikacích multimediálních formátů DLNA.

Tabulka 8 Podporované formáty audio soubor [9]

- Ze standardu OIPF jsou u HbbTV využity také specifikace pro protokoly proudění audiovizuálního obsahu, protokoly o stahování, protokoly pro samotný přenos aplikace, HTTP hlavička a přesměrování. [9]
- Zobrazovací model standardu HbbTV definuje rozlišení aplikace na rozměr 1280 x 720 pixel s poměrem stran 16:9. Hybridní terminál musí mít minimálně toto rozlišení, pokud je rozlišení terminálu fyzicky větší, tak aplikace zobrazí s rozlišením 1280 x 720 pixel. [9]
- U hybridních aplikací jsou podporovány pouze některé druhy fontů. Ze standardních fontů používaných v počítačích se jedná o proporcionální font Sans Serif a neproporcionální typ písma Monospace. [9]

Již bylo zmíněno, že každá aplikace je ovládána pomocí dálkového ovladače hybridního aplikátora, přičemž mohou být pro ovládání aplikace použity pouze některé klávesy dálkového ovladače. Následujícím výpisem klávesy dálkového ovladače a jejich klíčové události:

- barevná tlačítka ( červené, modré, zelené, žluté) (VK\_RED, VK\_BLUE, VK\_GREEN, VK\_YELLOW)
- směrové šipky (VK\_UP, VK\_DOWN, VK\_LEFT, VK\_RIGHT)
- tlačítko Enter nebo OK (VK\_ENTER)
- číselné klávesy (VK\_0 až VK\_9)
- klávesy přehrát, zastavit, pozastavit (VK\_STOP, VK\_PLAY a VK\_PAUSE nebo VK\_PLAY\_PAUSE)
- rychlé převíjení vpřed a vzad (VK\_FAST\_FWD, VK\_REWIND)
- klávesa nahrávání (VK\_RECORD) [9]

Ostatní klávesy dálkového ovladače nemají zaručenou podporu a proto je nelze použít pro ovládání aplikace. [9]

Přechod mezi prvky, kterým lze přidat atribut aktivního prvku (focus) se realizuje pomocí směrových šipek, které mohou využívat některé mechanismy. Zde je jejich výčet podle priorit s jakou se provádí: [9]

- zachycení události stisknutí směrových kláves a provedení akce přizvané v javascriptu [9], např.:

```

case VK_RIGHT:
    document.getElementById("btn1").focus();
    break;

```
- manipulace pomocí CSS3, kdy se pro navigaci využívá vlastností nav-up, nav-down, nav-left a nav-right [9]. Např.:

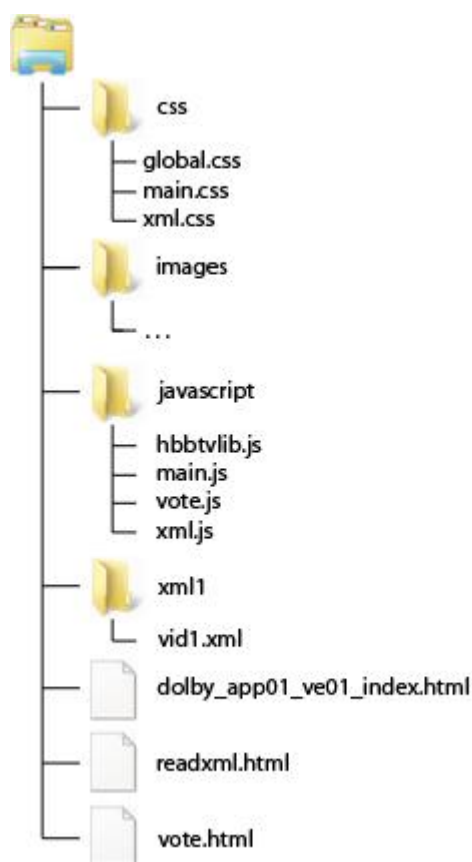
```
#back {  
  nav-left: #next;  
  nav-right: #playPause;  
  nav-up: #playPause;  
  nav-down: #back;  
}
```

- použití výchozích navigačních mechanismů, které umožní se pohybovat mezi prvky, jenž lze označit jako aktivní [9]



### 3 HBBTV APLIKACE

Vytvářená aplikace se podle zadání firmy LIVEBOX a. s. skládá ze tří obrazovek. První a zároveň úvodní stránka je ukázkou možností streamování nelineárního obsahu v televizním vysílání. Druhá stránka ukazuje možnost zobrazení televizního programu nebo třeba také načítání RSS dat, například ze zpravodajského portálu. Poslední stránka ukazuje možnost hlasování, kterého by šlo využít třeba u jaké televizní soutěže. Souborová struktura aplikace se skládá z HTML souborů tvořících kostru jednotlivých stránek. Pro vizualizaci aplikace je použito souborů s definicí kaskádových stylů (CSS). Informace obsažené v těchto souborech nejenže udávají vzhled prvků aplikace, ale v daných případech definují přechody mezi jednotlivými HTML prvky. V neposlední řadě tvoří aplikaci také soubory s definicemi javascriptových funkcí, které zajišťují chod aplikace. V adresářové struktuře (obr. 28) se vyskytují navíc ještě nezmíněné adresáře images a xml. V adresáři xml jsou umístěny XML soubory, z nichž se vytváří seznam videí, které lze v aplikaci přehrát. Posledním zmíněným adresářem je adresář images, kde je již z názvu patrné, jaký bude mít obsah.



Obr. 28 Adresářová struktura aplikace

#### 3.1 Úvodní stránka

U běžných webových stránek se standardně úvodní soubory webových stránek nazývají index.html, index.htm, default.htm a podobně. Avšak v případě vytvořené aplikace je úvodní soubor zobrazovaný při zavolání aplikace pojmenován dolby\_app\_ve01\_index.html, protože URL aplikace definovaná v testovacím datovém toku (transport streamu) je:

[http://tv-html.box.cz/hbbtv/tests/lifecycle/dolby\\_app01\\_ve01\\_index.html](http://tv-html.box.cz/hbbtv/tests/lifecycle/dolby_app01_ve01_index.html).

V tomto datovém toku je také nastaveno automatické spuštění aplikace po přepnutí hybridního přijímače na daný kanál. Po přepnutí terminálu na program, který obsahuje odkaz na HbbTV aplikaci a po jejím načtení se zobrazí stránka s šedým proužkem, červeným tmelem a nápisem HbbTV App (obr. 29). Ve stejnou dobu se na pozadí pomocí javascriptových funkcí načte přiložený XML soubor,

z kterého se vytvoří seznam přehrátelných videí a informací k nim. Pokud uživatel nestiskne červené tlačítko, tak se proužek s nápisem automaticky skryje a aplikace nebude vizuálně rušit diváka od sledování samotného televizního programu. Standardně se předpokládá, že červené tlačítko slouží k vyvolání nebo zrušení zobrazení aplikací.



*Obr. 29 Úvodní zobrazení aplikace*

Naopak pokud uživatel tlačítko na dálkovém ovladači stiskne, tak se provede sled událostí reagující na jeho stisknutí a zajistí se tím skrytí šedého proužku s nápisem a zobrazením obrazovky se seznamem videí zakrývající celou obrazovku televizoru. Tuto obrazovku na obr. 30 tvoří v horní části nadpis stránky s aktuálním časem, pod ním se nachází v levém sloupci krátký text získaný stejně jako nadpis z XML souboru. Odkazy na další stránky aplikace jsou na této obrazovce vytvořeny staticky. Vedle tohoto sloupce se nachází vložený a zmenšený obsah televizního vysílání nebo obrázek příslušného video souboru ze seznamu. Vpravo je pak umístěn sloupec s popisem video souboru, na němž je aktuálně příznak aktivního prvku obrazovky. Při první návštěvě se v pravé části automaticky zobrazí informace k prvnímu videu ze seznamu, protože aktivní prvek je nastaven na první položku seznamu. Pod těmito informacemi je umístěno logo HbbTV. Samotný seznam se nachází ve spodní části obrazovky a každá jeho položka je reprezentována obrázkem, názvem video obsahu, informacemi o kvalitě video a audio stopy tohoto videa.



Obr. 30 Úvodní stránka aplikace po stisknutí červeného tlačítka

Do této HTML kódu každé stránky je zapotřebí pro správnou funkci aplikace vložit dva objekty OIPF (Application manager a configuration embed object). Tyto dva objekty totiž zajišťují spolupráci s televizorem a umožňují pomocí funkcí obsažených v souboru javascriptu hbbtvlib.js inicializaci, zobrazení, skrytí, vytvoření, ukončení aplikace, přídání televizního vysílání do aplikace, odebrání televizního vysílání z aplikace a funkci vracející ID služby DVB. Umístění těchto dvou objektů je v této aplikaci hned za značkou `<body>` v elementu `div`, kterému je lokálně kaskádovým stylem přidána neviditelnost, šířka a výška rovna 0 (obr. 31).

```
<body onload="hbbtvlib_initialize(); hbbtvlib_show(); init('xml1');" >
  <div style="visibility: hidden; width: 0; height: 0;">
    <object type="application/oipfApplicationManager" id="appMan"></object>
    <object type="application/oipfConfiguration"></object>
  </div>
  ...
```

Obr. 31 Ústřední část HTML kódu definující vložení OIPF objektů

V ústřední části zdrojového kódu ilustrující vkládání objektů OIPF do stránky (obr. 31) je také ve značce `<body>` možné spatřit volání tří javascriptových funkcí vyvolaných při načítání HTML stránky. První funkce `hbbtvlib_initialize()` vytváří a inicializuje vložené objekty HbbTV (OIPF DAE). Tato funkce je součástí souboru `hbbtvlib.js` a musí být volána jako první. Druhá funkce je definována ve stejném souboru jako funkce první zajišťující zobrazení aplikace. Poslední funkce `init('xml1')` je inicializační funkcí samotné aplikace definovaná v souboru `main.js` a kód její definice je vidět na obr. 32. Parametrem je adresa umístění XML souboru nepřehledných videí. Celá definice funkce je uzavřena do ochranného bloku `try-catch`. Tento blok zachytí případnou chybu a tuto chybu zobrazí pomocí vyskakovacího okénka. V první části definice jsou pro další práci proměnnými přídány elementy HTML dokumentu podle ID těchto elementů. Po tomto přídání funkce testuje, zda za názvem souboru existuje nějaký query string, kterého se využívá při návratu z jiné stránky, aby se zabránilo zobrazení šedého proužku. Pokud je query string prázdný (et zecm (úvodní na tení aplikace), tak se na prvních sekund zobrazí šedý proužek s červeným tvarem klem a nápisem HbbTV App (obr. 29). Naopak pokud podmínka prázdného query stringu není splněna, tak je zobrazen hlavní obsah spolu s vysílaným televizním obsahem zmenšeným na velikost středového obrázku. Následně je také otestován parametr předaný funkci, zda se jedná o et zec. Pokud otestování dopadne kladně, tak se globální proměnné `xmlDir` při adě tento et zec a pak se se zavolá funkce `createLinkList()`, která se

dále stará o vytvoření seznamu přehratelných video souborů. Posledním příkazem v této funkci je metoda `document.addEventListener("keydown", keyHandler, false)`. Tato metoda je standardní metodou standardu W3C, která umožňuje připojit obslužné rutiny k určitému prvku. Prvním parametrem této metody je událost, která zavolá objekt nebo funkci v druhém parametru. Tímto parametrem pravdivostního typu se určuje, zda se má zahájit zachytávání. V tomto případě dokument při stisknutí klávesy zavolá funkci `keyHandler(e)`.

```
function init(_xmlDir) {
    try {
        redLabel = document.getElementById("redbtn");
        hlavni = document.getElementById("hlavni");
        navbar = document.getElementById("navBar");
        videoContainer = document.getElementById(containerId);
        navigationBar = document.getElementById(navigationId);
        playBar = document.getElementById(playBarId);
        playBtn = document.getElementById(playBtnId);
        infoArea = document.getElementById(infoAreaId);
        titleDiv = document.getElementById(titleId);
        bigImg = document.getElementById("bigImg");
        playPosMarker = document.getElementById("playPosMarker");
        playTime = document.getElementById("playTime");

        if (query_string == "") { redLabel.style.visibility = "visible"; casovac =
            setTimeout("redhidden()", 5000); }
        else {
            hlavni.style.visibility = "visible";
            hbbtvlib_init_broadcast("videoCont", "bigImg", "images/img.png");
        }

        if (typeof _xmlDir == "string") xmlDir = _xmlDir;

        if (started) { redvisible(); }
        started = true;

        createLinkList();
        document.addEventListener("keydown", keyHandler, false);
    } catch (e) {
        alert("Init: " + e);
    }
};
```

Obr. 32 Definice funkce `init(_xmlDir)`

Funkce `keyHandler` (obr. 33) ve své definici testuje, zda návratová hodnota funkce `keyCodeHandler` volaná s parametrem kódu stisknutého tlačítka vrátí hodnotu `true` (pravda) nebo `false` (nepravda). Pokud je návratová hodnota funkce `keyCodeHandler` `true`, tak zavolá metodu `preventDefault` objektu předaného jí parametrem `e` a tato metoda zruší akci bez zastavení dalších událostí.

```
function keyHandler(e) {
    try {
        if (keyCodeHandler(e.keyCode))
            e.preventDefault();
    } catch (e2) {
    }
}
```

Obr. 33 Definice funkce `keyHandler(e)`

Funkce volaná `keyCodeHandler` v podmínce funkce `keyHandler` je vždy zavolána s jedním parametrem, kterým je kód stisknutého tlačítka, jak již bylo zmíněno v předchozím odstavci. Tento předávaný parametr je rozhodovací podmínkou v tvůrčím příkazu `switch`, s jehož pomocí je provedeno vztavení pro určité stisknuté tlačítko a lze tak definovat provedení akcí při jejich stisknutí.

Již bylo také zmíněno, po zobrazení obrazovky se seznamem videí se pomocí aktivního prvku automaticky nastaví na první položku seznamu a směrovými šipkami na dálkovém ovladači vlevo a vpravo se lze posouvat tímto seznamem. Jestliže má tento seznam více než 5 položek, tak se na obou stranách tohoto seznamu zobrazují ilustrační šipky, které uživateli vizuálně napovídají, že tento

seznam obsahuje více prvků a v tomto seznamu se můžete pohybovat. Pokud se uživatel posune na poslední položku seznamu, tak šipka zmizí (obr. 34). Při stisknutí potvrzovacího tlačítka při aktivním prvku seznamu se vyvolá funkce *show(urlId)* s parametrem *urlId* určující pozici objektu v poli těchto objektů. Tento parametr je dále použit pro zavolání funkce *playVideo* s dvěma parametry, jimiž jsou URL přehrávaného videa a jeho *mime* typ.



Obr. 34 Ukazatel na konci seznamu

Po zavolání funkce se v prvním kroku do HTML dokumentu přidá značka `<object>` umožňující přehrávání audiovizuálního obsahu. V následujícím kroku se vytvoří zavoláním metody *getElementById* objekt, s jehož pomocí se bude ovládat přehrávání obsahu a postupovat k jeho natavení. V dalším kroku se nastaví funkce, která se provede při změně stavu přehrávání. Tohoto se docílí tak, že se vlastnosti *onPlayStateChange* vytvořeného objektu přidá jméno funkce, která se má použít při změně stavu přehrávání. Takto přidávaná funkce je pouze odkazem na funkci, a proto jí nelze přidat žádné parametry. Následujícím příkazem se vloženému objektu přidá příkaz pro započetí přehrávání. Aby se přehrávací objekt mohl zobrazit přes celou obrazovku a nenastal problém s vystoupením některé části úvodní obrazovky, tak jsou ostatní části dokumentu zneviditelnány. Globální proměnná *timer* se přidá jako *interval*, který každou sekundu zavolá funkci *updateplayerControls*, která slouží pro aktualizaci přehrávaného času a pro posun ukazatele aktuální pozice. Funkce *ShowPlayBar* s pravdivostním parametrem zajistí zobrazení panelu, pokud je parametr roven *true*. Tato funkce také nastaví *interval*, který po uplynutí nastavených deseti sekund automaticky skryje *PlayBar*. Samotný *Playbar* se skládá z tlačítek pro ovládání, posuvníku ukazující aktuální pozici přehrávaného videa a času. Mezi jednotlivými tlačítky pro ovládání přehrávání se uživatel pohybuje opět pomocí šipek vlevo a vpravo. Jednotlivé přechody jsou nastavené pomocí kaskádových stylů jak je vidět z následujícího obrázku. Pro každé ovládací tlačítko je také nastaveno, že při stisku klávesy nahoru nebo dolů zůstane ukazatel aktivního prvku na stejném tlačítku.

```

#back {
  nav-left: #next;
  nav-right: #playPause;
  nav-up: #back;
  nav-down: #back;
}

#playPause {
  nav-left: #back;
  nav-right: #stop;
  nav-up: #playPause;
  nav-down: #playPause;
}

#stop {
  nav-left: #playPause;
  nav-right: #next;
  nav-up: #stop;
  nav-down: #stop;
}

#next {
  nav-left: #stop;
  nav-right: #back;
  nav-up: #next;
  nav-down: #next;
}

```

Obr. 35 Nastavení pohybu mezi ovládacími tlačítky PlayBaru v kaskádovém stylu

Na obr. 36 je vidět zobrazení videopřehrávače s aktivním *playBar*em. Mezi jednotlivými video soubory lze přecházet nejen pomocí seznamu, ale také v *playBar*u pomocí prvního a posledního tlačítka. Použitím tohoto tlačítka uživatel přejde na předchozí nebo následující video soubor. Pokud se uživatel nachází na posledním souboru seznamu a pokusí se přejít na následující video, tak aplikace zajistí přechod na první video soubor. Obdobně pro opačný směr.



Obr. 36 Obrazovka při přehrávání videa

O vizualizaci posuvníku zobrazovaného v *playBar*u videopřehrávače se, jak již bylo uvedeno výše, stará funkce *updatePlayerControls()*. Tato funkce je volána každou sekundu od zvolení video souboru k přehrávání. V definici této funkce je definováno vícenásobné rozhodování podle stavu přehrávání,

které je však ještě předcházeno otestováním, zda je *playBar* zobrazen a zda také *vPlayer* existuje. Pokud jsou obě podmínky splněny a stav přehrávání je roven 1, což odpovídá stavu přehrávání souboru, tak se dopočítá hodnota pozice, na kterou se má od levého okraje posunout. Pozice se vypočítává z délky a aktuální časové pozice přehrávaného videa, které jsou získávány z objektu přehrávače *vPlayer*.

Při načítání stránky se získávají data z XML souboru umístěného v adresáři, jehož cesta se dotváří z parametru úvodní volané funkce *init(dir)*. Pro správné načtení seznamu video souborů je nutné, aby XML soubor měl správný tvar, který umí funkce pracující s tímto souborem načíst. Struktura tohoto XML dokumentu je zobrazena na následujícím obrázku.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<channel>
  <title></title>
  <link></link>
  <description></description>
  <copyright></copyright>
  <pubDate/>
  <lastBuildDate/>

  <item>
    <title></title>
    <pubDate/>
    <description></description>
    <video></video>
    <audio></audio>
    <videoLength></videoLength>
    <link></link>
    <img></img>
    <DRM></DRM>
    <guid/>
  </item>
</channel>
```

Obr. 37 Struktura XML souboru seznamu

Ihned za koncovou značkou *<channel>* se nachází položky pro nadpis a text v levém sloupci. Tyto položky jsou umístěny mezi značkami *<title>* a *<description>*. Za nimiž již následuje několik položek *<item>*, z kterých je vytvářen seznam videí k přehrávání. Položky *<item>* obsahují název videa (značka *<title>*), datum publikace (*<pubDate>*), krátký popis videa (*<description>*), informace o kvalitě videa a audia (*<video>* a *<audio>*), délku video souboru (*<videoLength>*), URL umístění samotného video souboru (*<link>*), URL umístění informačního obrázku zobrazovaného v seznamu nebo uprostřed obrazovky (*<img>*) a případně ID videa (*<guid>*). Značka obsahující URL s umístěním video souboru může mít atribut *type*, který udává mime typ video souboru.

### 3.2 Stránka televizního programu

Druhou vytvořenou stránkou aplikace je stránka, která umožňuje zobrazit televizní program všech kanálů České televize na aktuální den. Po načtení stránky se na obrazovce zobrazí nadpis a staticky vytvořený seznam kanálů (obr. 38).



Obr. 38 Stránka pro zobrazení televizního programu po na tení

Po zvolení konkrétního televizního kanálu se pod seznamem televizních kanál vypíše p ehled po ad daného kanálu pro dnešní den. Ve výpisu po ad je zahrnut obrázek po adu, as za átku a název po adu. Ukazatel aktivního prvku se p esune na první pozici v seznamu po ad a tím se v pravém sloupci zobrazí podrobn jší informace o daném po adu. Tuto situaci ilustruje obr. 39.



Obr. 39 Výpis po ad po výb ru televizního programu

Seznam po ad je získán z webové stránky eské televize. Tuto stránku lze získat dopln ním tí arametr do obecného p edpisu URL, který vypadá následovn : [http://www.ceskatelevize.cz/services/programme/xml/schedule.php?user=\[login\]&date=\[dd.mm.rrrr\]&channel=\[ct1|ct2|ct24|ct4|ct5|ct6\]](http://www.ceskatelevize.cz/services/programme/xml/schedule.php?user=[login]&date=[dd.mm.rrrr]&channel=[ct1|ct2|ct24|ct4|ct5|ct6]), kde *user* je uživatelské jméno získané po registraci, *date* je datum zadávané ve tvaru dd.mm.rrrr a posledním parametrem je jeden z množiny identifikátor televizních

kanál české televize. V případě této aplikace si funkce *createProgram* zjistí aktuální datum, které upraví do správného tvaru a spolu s parametrem, se kterým je volána funkce *createProgram*, jej doplní do URL XML souboru. Jako uživatel je v tomto případě použito jméno uživatele test. Po získání XML souboru jsou vyparsovány potřebné informace a je vytvořen seznam položek vykreslený na obrazovce terminálu. [10]

Odpovědí webové stránky s doplněnými parametry je dokument XML, který má specifický tvar zobrazený na obr. 40. V prvním řádku je vidět deklaraci XML souboru a za ní následuje kořenový prvek označovaný *<program>*, který má položky a atributy. Prvním atributem je *datum\_vyslani*, které je totožné s datem zadaným při volání webové stránky. Z druhého atributu je patrné, pro který kanál je program vygenerován. Poslední atribut udává datum a čas, kdy byl soubor vygenerován. Základním elementem každé položky je element *<porad>*. Prvním vnějším elementem je element *<linky>* s dále vnějším elementy *<program>* a *<ivysilani>*, které obsahují URL na stránku dané položky na webu české televize a URL směřující na stránku ivysílání, kde si lze danou položku přehrát. Za skupinou odkazů následuje element s kódem VPS a element času začátku položky. Dalším vnějším elementem je element *<nazvy>*, který tvoří množinu elementů s jednotlivými částmi názvu. Tyto vnějším elementy jsou *<nadtitul>*, *<nazev>*, *<original>* a *<nazev\_casti>*. Kromě elementu *<nazev>* mohou být ostatní tyto elementy prázdné podle daného názvu položky. Pokud je položka seriálem, tak je v elementu *<dil>* zapsáno pořadové číslo dílu a celkový počet dílů ve formátu díl/počet dílů. Žánr položky obsahuje element *<zahr>* následovaný délkou položky v minutách. Element *<noticka>* obsahuje krátký popis dané položky. Následující dvě značky *<regionalni>* a *<alternativa>* mohou nabývat pouze hodnot Y nebo N podle toho, zda se jedná o regionální položku u elementu *<regionalni>* a podobně u prvku *<alternativa>* je vepsáno Y, pokud se jedná o alternativu ve vysílání. Množinu elementů ohraničenou značkou *<ikony>* je tvořena deseti dalšími elementy, které kromě prvku *<zvuk>* a *<pomer>* mohou nabývat pouze hodnot 0 nebo 1 podle toho, zda se jedná o daný typ vysílání vyplývající z označení elementu. Pouze snad u prvku *<cb>* je zapotřebí vysvětlit, že se jedná o černobílé vysílání. Zmíněný element *<zvuk>* udává kvalitu zvukové stopy položky (M – mono, S – stereo, D – Dolby Digital, B – dual). Element *<pomer>* udává poměr stran videa a může tedy nabývat hodnot 16:9 nebo 4:3. Poslední element *<obrazky>* obsahuje další vnějším element s URL ilustrací obrázku položky. [10]

```

<program datum_vysilani="2014-05-27" kanal="ct24" generovano="2014-05-27 13:41:18">
  <porad>
    <linky>
      <program>
        http://www.ceskatelevize.cz/porady/1096902795-studio-6/214411010100527/
      </program>
      <ivysilani>
        http://www.ceskatelevize.cz/ivysilani/1096902795-studio-6/214411010100527/
      </ivysilani>
    </linky>
    <vps>A27050559</vps>
    <cas>05:59</cas>
    <nazvy>
      <nadtitul/>
      <nazev>Studio 6</nazev>
      <original/>
      <nazev_casti/>
    </nazvy>
    <dil/>
    <zanr>Zpravodajství</zanr>
    <stopaz>166:00</stopaz>
    <noticka>
      Přišel dýl, dal slepicom, či nastevřel okno - české, přesto pro mnohé z nás
      nesrozumitelné věty. Výrazům typickým pro určitou oblast totiž často rozumějí jen
      její obyvatelé. Na internetu proto začaly vznikat lidové slovníky, které tato
      specifická slova překládají. Jak se online slovník nářečí sestavuje a co na laické
      překlady dialektů říkají jazykovědci? Zeptáme se v 8:50!
    </noticka>
    <regionalni>N</regionalni>
    <alternativa>N</alternativa>
    <ikonky>
      <zvuk>S</zvuk>
      <skryte_titulky>0</skryte_titulky>
      <neslysici>0</neslysici>
      <live>0</live>
      <premiera>1</premiera>
      <cb>0</cb>
      <hvezdicka>0</hvezdicka>
      <puvodni_zneni>0</puvodni_zneni>
      <pomer>16:9</pomer>
      <hd>0</hd>
    </ikonky>
    <obrazky>
      <tv_program>
        http://img.ceskatelevize.cz/program/porady/1096902795/foto/program.jpg
      </tv_program>
    </obrazky>
  </porad>
</program>

```

Obr. 40 část XML odpovědi webové stránky České televize při vytváření programu

Tak jako v případě úvodní stránky, tak i zde bylo zapotřebí přidat některým tlačítkům dálkového ovládacího zařízení akce, které se mají provést po jejich stisknutí. Jelikož se na této stránce lze pohybovat pouze mezi prvky seznamu, tak akce jsou nastaveny pouze pro směrové šipky, tlačítko zpět a červené tlačítko. Jak je vidět na následujícím obrázku, tak červené tlačítko ukončí aplikaci, stisknutím tlačítka zpět se uživatel vrátí zpět na úvodní stránku a u směrových šipek je v javascriptu nastaveno, aby se uživatel mohl pohybovat pouze v prvcích seznamu, čímž jsou tak ošetřeny jejich koncové pozice.

```

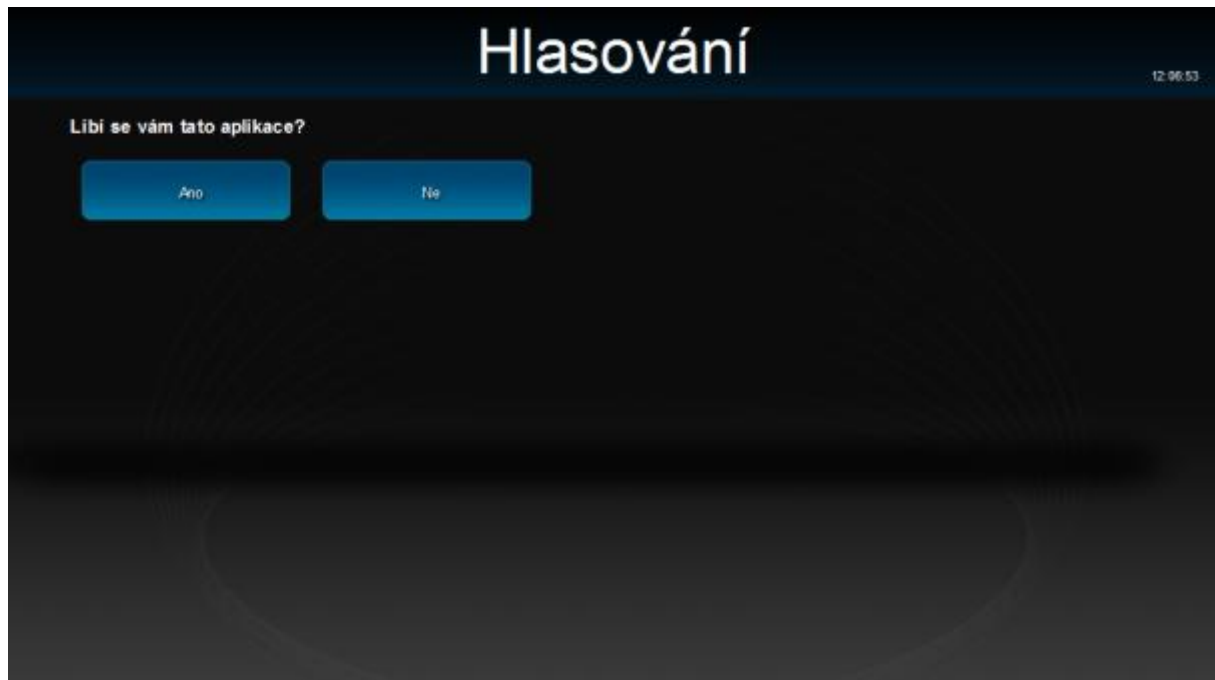
function keyCodeHandler(keyCode) {
  try {
    switch (keyCode)
      case KeyEvent.VK_UP: {
        var activeEl = document.activeElement.id;
        if(activeEl.substr(4) == 0) document.getElementById(pomCT).focus();
        else {
          if(activeEl.substr(4) > 0)
            document.getElementById("list" + (parseInt(activeEl.substr(4))-1)).focus();
        }
        break;
      case KeyEvent.VK_DOWN:
        var activeEl = document.activeElement.id;
        if(activeEl.substr(0,2) == "ct") document.getElementById("list0").focus();
        else {
          if(activeEl.substr(4) < porad.length -1)
            document.getElementById("list" + (parseInt(activeEl.substr(4))+1)).focus();
        }
        break;
      case KeyEvent.VK_RIGHT:
        var activeEl = document.activeElement.id;
        var firstPart = activeEl.substr(0,2);
        var pointPart = parseInt(activeEl.substr(2));
        if(pointPart<6)
          document.getElementById(firstPart + (pointPart+1)).focus();
        break;
      case KeyEvent.VK_Left:
        var activeEl = document.activeElement.id;
        var firstPart = activeEl.substr(0,2);
        var pointPart = parseInt(activeEl.substr(2));
        if(pointPart>1)
          document.getElementById(firstPart + (pointPart-1)).focus();
        break;
      case KeyEvent.VK_RED:
        hbbtvlib_hide();
        hbbtvlib_closeApp();
        break;
      case KeyEvent.VK_BACK:
        window.location.href = "dolby_app01_ve01_index.html?id=1";
        break;
      default:
        return false;
    }
    return true;
  }
  catch (e) {
    alert(e);
  }
}

```

Obr. 41 Funkce tla ítek stránky televizního programu

### 3.3 Stránka ankety

Poslední ástí ukázkové HbbTV aplikace je stránka s anketní otázkou. Po na tení této stránky se zobrazí obrazovka (obr. 42), na které je v hlavi ce zobrazen na st ed zarovnaný nadpis a v pravé ástí aktuální ás. Pod hlavi kou se nachází text otázky následované tla ítky s možnými odpov ími na danou otázku. Na této stránce má uživatel možnost se pohybovat pouze mezi odpov ími na otázku. Návrat na úvodní stránku umož ůje tla ítko back na dálkovém ovlada i. P i na tení stránky je op t volána funkce *init()* podobn ě jako u ostatních stránek (obr. 43). Tentokrát nemá tato funkce žádný parametr a její definice zahrnuje pouze spoušt ní funkce zobrazení a aktualizace aktuálního ás, nastavení aktivního prvku na tla ítko kladné odpov ídi na otázku a v neposlední ad také nastavení obslužné rutiny pro stisknutí tla ítko dálkového ovlada e.



Obr. 42 Obrazovka hlasovací stránky

```

<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-loose.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>HBBTv</title>
    <link rel="stylesheet" href="css/main.css" type="text/css" media="all" />
    <link rel="stylesheet" href="css/global.css" type="text/css" media="all" />
    <link rel="stylesheet" href="css/xml.css" type="text/css" media="all" />
    <script type='text/javascript' language='javascript' src='javascript/hbbtvlib.js'></script>
    <script language="javascript" type="text/javascript" src="/javascript/vote.js"></script>
  </head>
  <body onload="init();" >
    <div style="visibility: hidden; width: 0; height: 0;">
      <object type="application/oipfApplicationManager" id="appMan"></object>
      <object type="application/oipfConfiguration"></object>
    </div>
    <div id="hlavni">
      <div class="head">
        <div class="headline" id="title">Hlasování</div>
        <div class="actualTime" id="actualTime"></div>
      </div>
      <div id="safe_area">
        <div id="question">
          <h2>Líbí se vám tato aplikace?</h2>
          <br />
          <a class="vector2" href="javascript: void(0);" id="Yes"
            onclick="javascript:voteYes();">Ano</a>
          <a class="vector2" href="javascript: void(0);" id="No"
            onclick="javascript:voteNo();">Ne</a>
        </div>
        <div id="results"></div>
      </div>
    </div>
  </body>
</html>

```

Obr. 43 HTML kostra stránky vote.html

```

function voteYes()
{
    var httpRequest = new XMLHttpRequest();
    httpRequest.open('POST', 'http://hbttvtest.stage.livebox.cz/SetData.aspx?id=1', true);
    httpRequest.send(null);
    showResult();
}
function voteNo()
{
    var httpRequest = new XMLHttpRequest();
    httpRequest.open('POST', 'http://hbttvtest.stage.livebox.cz/SetData.aspx?id=0', true);
    httpRequest.send(null);
    showResult();
}

```

Obr. 44 Funkce tlačítka odpovědí

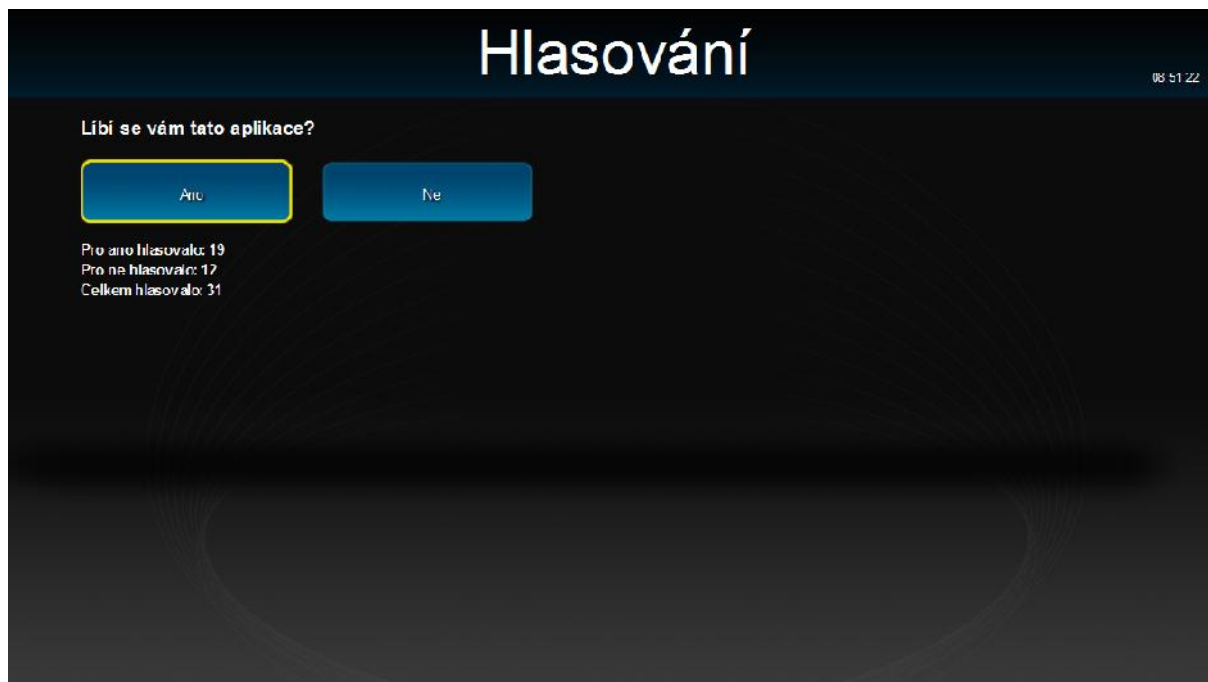
Po stisknutí některé z možných odpovědí se provede funkce určená pro provedení po stisku daného tlačítka v HTML kódu (obr. 43). Tato funkce odešle požadavek na webovou stránku, které předá v parametru uživatelskou odpověď (obr. 44). Volaná webová stránka *SetData.aspx* obsahuje script, který pokud je této stránce předán správný parametr, tak hodnotu tohoto parametru zapíše do databázové tabulky společně s datem a časem odeslání odpovědi. V definici hlasovacích funkcí (obr. 44) je vždy na konci volána funkce *showResult()*, která následně zavolá funkci pro načtení XML dat z dané URL. Tato volaná stránka po zpracování scriptu vrátí jako odpověď dynamicky vytvořený XML dokument ve tvaru na obrázku obr. 45, kde hodnota v první značce *<count>* určuje počet hlasů pro odpověď „Ne“ a druhý počet patří odpovědím Ano. Po načtení tohoto XML jsou data zpracovávána v javascriptu, dále je také dopočten celkový počet odeslaných odpovědí a informace jsou vypsány pod hlasovacími tlačítky. Na obr. 46 je možné vidět vzhled obrazovky po provedení zápisu odeslané odpovědi do databáze a výpisu stavu hlasování.

```

<votes>
  <vote>
    <count>12</count>
  </vote>
  <vote>
    <count>19</count>
  </vote>
</votes>

```

Obr. 45 Odpověď stránky *GetData.aspx*



Obr. 46 Obrazovka hlasovací stránky po odeslání hlasování

Stránky provádějící zápis a čtení dat do/z databáze jsou vytvořeny s pomocí technologie ASP.NET. Script vykonávající zápis dat do databáze, stejně jako script stránky načítající data z databáze jsou napsány v jazyce C#. V prvním případě, kdy jsou data zapisována do databázové tabulky, si stránka po jejím zavolání zkontroluje zadání neprázdného parametru *id* a pokud je tato podmínka splněna, tak se provede následující kód (obr. 47), kterým se hodnota parametru vloží do databáze. Výše bylo zmíněno, že do databázové tabulky je spolu s hodnotou odpovědi ukládáno také datum a čas odpovědi, ale ve scriptu se vkládá do databáze pouze hodnota odpovědi, protože aktuální datum a čas je automaticky generováno v SQL tabulce, jak bude vysvětleno níže.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.SqlClient;
using System.Data;

namespace HbbTVSQL
{
    public partial class SetData : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Request.QueryString["id"] != null)
            {
                SqlConnection cnn = new SqlConnection();
                cnn.ConnectionString =
                System.Configuration.ConfigurationManager.ConnectionStrings["SQLconn"].ConnectionString;
                cnn.Open();
                SqlCommand cmd = new SqlCommand();
                cmd.CommandText = "select * from vote";
                cmd.Connection = cnn;
                SqlDataAdapter da = new SqlDataAdapter();
                da.SelectCommand = cmd;
                DataSet ds = new DataSet();
                da.Fill(ds, "vote");
                SqlCommandBuilder cb = new SqlCommandBuilder(da);
                DataRow drow = ds.Tables["vote"].NewRow();
                drow["vote"] = Request.QueryString["id"].ToString();
                ds.Tables["vote"].Rows.Add(drow);
                da.Update(ds, "vote");
                string script = @"<script language="javascript"> alert('Information have been
                Saved Successfully.....!!!!!!'); </script>";
                Page.ClientScript.RegisterStartupScript(this.GetType(), "myJScript1", script);
            }
        }
    }
}

```

Obr. 47 Kód stránky zapisující hlasování do databáze

Stránka vracející po ty hlasovacích odpovědí ve tvaru XML dokumentu získá vypisovaná data po připojení k databázi odesláním dotazu, který je vidět na obr. 48. Tento SQL dotaz vrátí součet po této hlasování seskupených podle hodnot odpovědí.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data.SqlClient;
using System.Data;

namespace HbbTVSQL
{
    public partial class GetData : System.Web.UI.Page
    {
        private string connectionString = @"Data Source=DEVEL;Initial Catalog=HbbTVtest;Persist
Security Info=True;User ID=HbbTVtest;Password=HbbTVtest123";
        protected void Page_Load(object sender, EventArgs e)
        {
        }
        protected override void Render(HtmlTextWriter writer)
        {
            Response.ContentType = "text/xml";
            if (Request.HttpMethod.ToUpper() == "GET")
            {
                using (SqlConnection connection = new SqlConnection(connectionString))
                {
                    using (SqlCommand command = connection.CreateCommand())
                    {
                        command.CommandText = "SELECT COUNT(*) AS count FROM vote GROUP BY vote";
                        DataSet set = new DataSet("votes");
                        set.Tables.Add(new DataTable("vote"));
                        DataTable table = set.Tables[0];
                        try
                        {
                            connection.Open();
                            SqlDataReader reader = command.ExecuteReader();
                            table.Load(reader, LoadOption.Upsert);
                        }
                        catch (Exception ex)
                        {
                            writer.WriteLine(@"<Error> + ex.Message + "</Error>");
                        }
                        table.WriteXml(writer);
                    }
                }
            }
        }
    }
}

```

Obr. 48 Stript prováděný na pozadí stránky *GetData.aspx*

Databázová tabulka, do níž se zapisují jednotlivé hlasy, se skládá ze tří sloupců. Prvním sloupcem je unikátní id, které je zároveň primárním klíčem této tabulky a jednotlivým řádkům tabulky je přidělován automaticky. Druhý sloupec tvoří samotné hlasování a v posledním sloupci je uloženo datum a čas hlasování. Toto datum a čas je generováno automaticky funkcí *getDate()*. Všechny tři sloupce tabulky jsou definovány jako neprázdné (NOT NULL). Pro lepší představu jak vypadá tato tabulka, je její část vyobrazena na následujícím obrázku.

id	vote	voteTime
185	1	29.4.2014 16:10:40
186	0	29.4.2014 16:10:47
187	0	29.4.2014 16:10:52
188	1	29.4.2014 16:22:25
189	1	29.4.2014 16:22:26
190	0	29.4.2014 16:24:35
191	0	29.4.2014 16:24:36
192	1	30.4.2014 8:48:37
193	0	30.4.2014 8:58:26
194	0	30.4.2014 9:01:46
195	0	30.4.2014 9:02:08
196	0	30.4.2014 9:02:44
197	1	30.4.2014 9:03:15
198	1	30.4.2014 9:04:30
199	1	30.4.2014 9:05:30
200	1	30.4.2014 9:05:35
201	1	30.4.2014 9:05:37

*Obr. 49 Ukázka databázové tabulky*



## 4 TESTOVÁNÍ HBBTV APLIKACE

### 4.1 Testování během vývoje

Během vývoje aplikace byla aplikace testována pomocí prohlížeče Mozilla Firefox s nainstalovaným doplňkem FireHbbTV verze 1.3.9. Tento prohlížeč spolu s nainstalovaným doplňkem FireHbbTV umožňuje vývoj a testování HbbTV aplikací. Po nainstalování doplňku se v liště prohlížeče zobrazí ikonka monitoru (obr. 50). Pokud doplněk detekuje zobrazování HbbTV aplikace, spustí se automaticky, protože samotný prohlížeč tento typ aplikací nedokáže zobrazit. Doplněk lze také pro běžné webové stránky zapnout a vypnout ručně kliknutím na ikonu (obr. 50). [11]



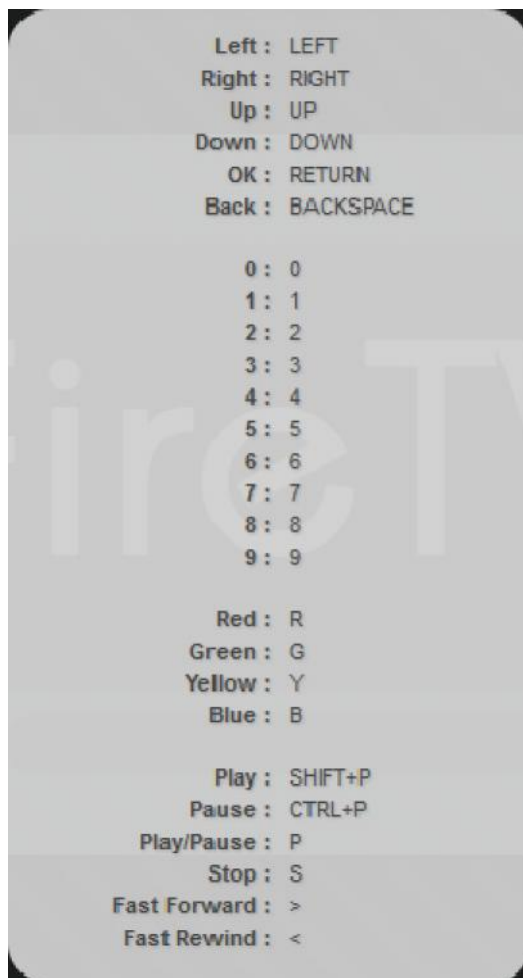
Obr. 50 Ikona FireHbbTV (a) neaktivní, b) aktivní

Po načtení aplikace se tedy automaticky aktivuje plugin FireHbbTV a v okně prohlížeče se zobrazí aplikace s panelem možností pro zobrazení (obr. 51). Na tomto obrázku je zobrazená ukázková aplikace spolu s okrajem ochranné zóny (safe zone), kterou doporučuje standard HbbTV. Zobrazení tohoto okraje lze vypnout tlačítkem Show Margins (Zobrazit okraje) v panelu pluginu. Pro možnost ovládní aplikace stejně jako s použitím dálkového ovladače televizoru jsou na klávesnici pi azeny funkce tohoto tlačítka a jejich pohled lze vyvolat stisknutím tlačítka Remote Control (Dálkové ovládní). Pohled funkcí kláves je zobrazen na obr. 52). [11]

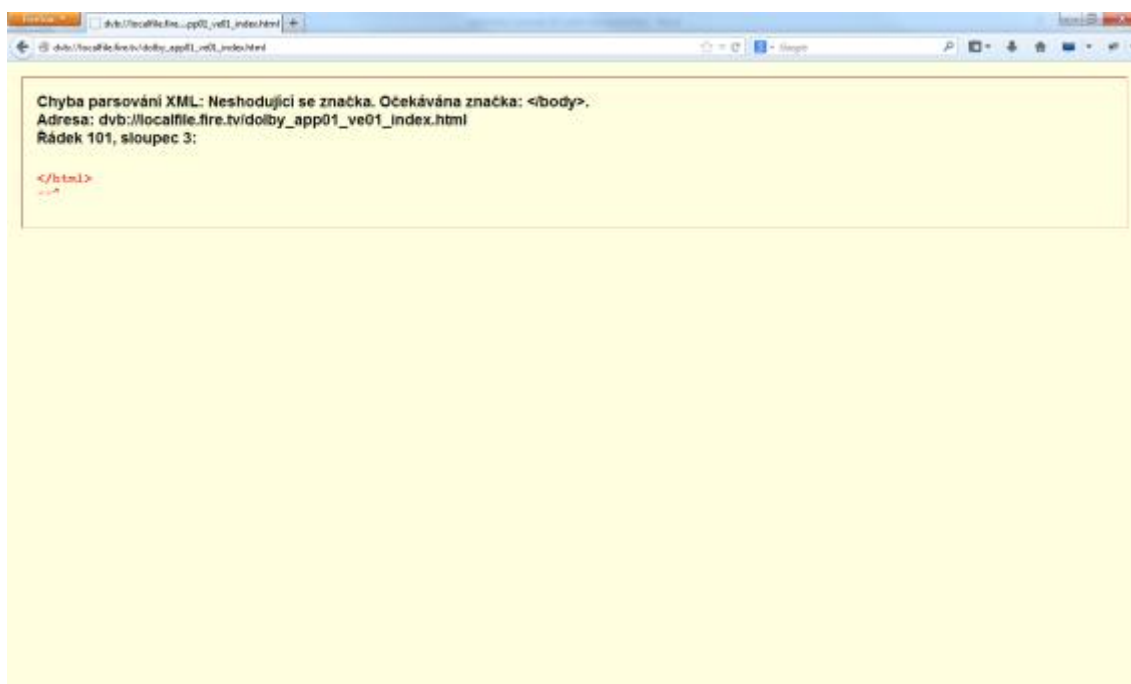


Obr. 51 Zobrazení aplikace v prohlížeči Firefox

Výhodou tohoto pluginu je automatická kontrola syntaktických chyb. Pokud se v HTML kódu aplikace objeví syntaktická chyba, tak tento doplněk nezobrazí aplikaci, ale pouze informaci o chybě v parsování XML, jak je vidět na obr. 53. V tomto případě se jedná o vynechanou koncovou značku `</body>`.



Obr. 52 P ehled funk níh kláves dopl ku FireHbbTV



Obr. 53 Chyba syntaxe HTML souboru HbbTV díky chyb jící zna ce </body>

## 4.2 Testování na hybridním přijímači

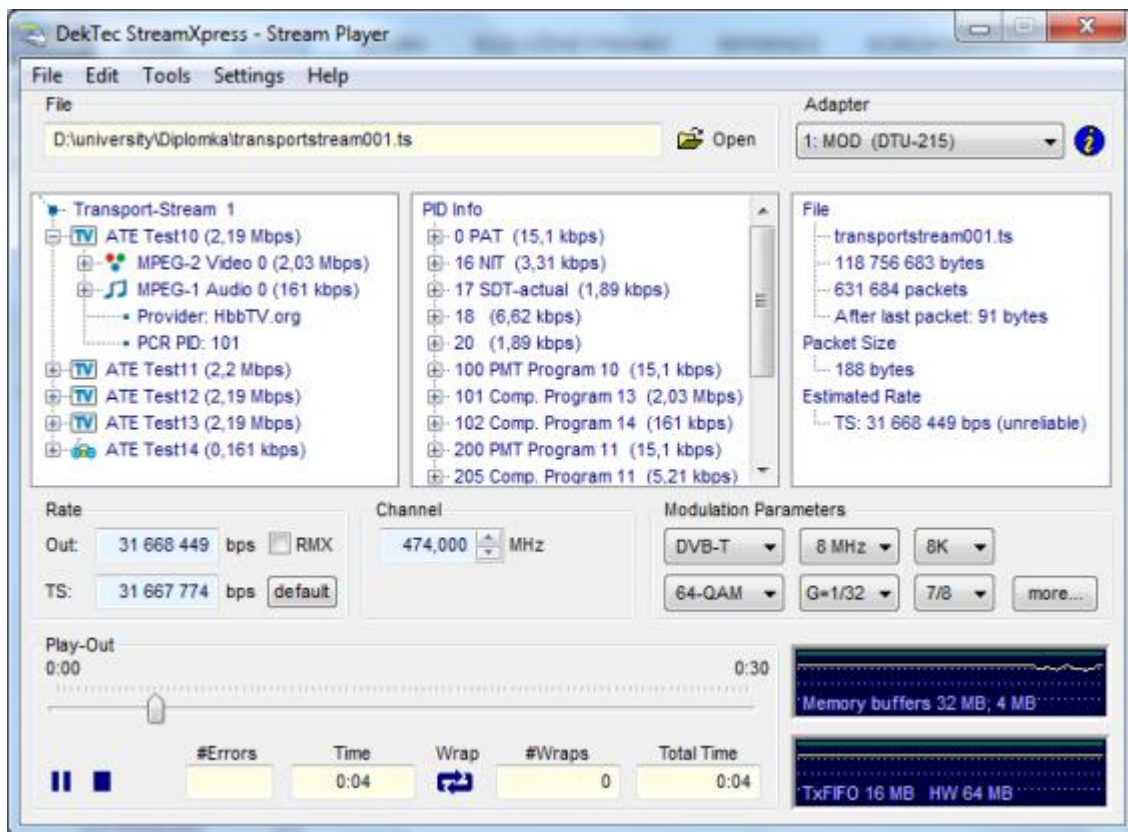
Po dokončení programování aplikace bylo zapotřebí otestovat aplikaci přímo na hybridním přijímači. Z architektury standardu HbbTV, zobrazené na obr. 25 je patrné, odkud a jak proudí data lineárního i nelineárního obsahu. Z tohoto schématu je také patrné, že pro dodržení architektury je vytvořenou aplikaci nutné umístit na webový server, aby k ní mohlo být přistupováno pomocí protokolů internetu. Pro tento účel byla tedy aplikace nahrána na webový server pod operačním systémem Windows, kde byl vytvořen pro aplikaci hosting s nastavenou doménou. Tímto se tedy vytvořila část se zprostředkovanou aplikací.

V druhé části architektury je řešeno vysílání lineárního obsahu (televizního vysílání) ke koncovému uživateli. Za normálních okolností je lineární obsah vysílán přes běžné vysílání DVB-T/S/C. Proto bylo tedy zapotřebí vytvořit situaci jak dostat testovací datový tok (transport stream) obsahující odkaz na vytvořenou aplikaci z počítače, kde byl tento datový tok uložen, na anténní vstup hybridního přijímače. K tomu účelu bylo využito USB-2 VHF/UHF modulátoru DTU-215 vyráběného firmou DekTec (obr. 54). Pomocí tohoto modulátoru lze z transportního toku uloženého v počítači simulovat šíření televizního vysílání stejně, jakoby se jednalo o signál televizního vysílání šířeného pomocí DVB-T, DVB-S nebo DVB-C. Výstupní signál vychází přes 75 Ω RF konektor. Frekvenční rozsah je mezi 47 – 1000 MHz s šířkou pásma od 2,7 do 8,0 MHz. Nastavování výstupních parametrů lze provést pomocí softwaru StreamXpress, který je dodáváný spolu s tímto modulátorem.



Obr. 54 USB-2 VHF/UHF modulátor DTU-215

Po spuštění programu StreamXpress a připojení modulátoru lze z počítače na výstupu uložený transportní tok. Bez připojení modulátoru k počítači nelze s programem provádět žádné operace. Po načtení souboru s připojením se v okně aplikace na displeji zobrazí informace z tohoto transportního toku, jak je možné vidět na následujícím obrázku.



Obr. 55 DekTec StreamXpress

Pod nabídkovou lištou se nachází cesta k právě na tenému souboru transportního toku. Napravo od této cesty je seznam připojených adaptérů. V tomto případě se jedná o již zmíněný modulátor DTU-215. Níže jsou zobrazeny tři sloupky s informacemi o transportním toku. V levém sloupci lze pozorovat název transportního toku se seznamem televizních kanálů. Po rozkliknutí jednotlivého programového kanálu se zobrazí podstrom s podrobnějšími informacemi o jednotlivých programech. Za každou položku je v závorce zobrazena velikost datového toku, kterou zabírá v daném transportním toku. V prostředním sloupci je výpis PID paketů obsažených v transportním toku spolu s velikostí datového toku. V posledním sloupci jsou pak zobrazeny informace o souboru, jeho názvu, velikosti, počet paketů, velikost transportního paketu a další. [12]

Pod informací získanou z daného souboru transportního toku následuje část pro nastavení výstupních parametrů modulátoru. V levé části lze nastavit výstupní datový tok. Uprostřed lze nastavit frekvenci, na které lze najít vysílaný multiplex. V pravé části se nachází konkrétní nastavení modulace. Nastavení modulace lze provést pro několik modulacích standardů, podle kterých se určují další možná nastavení. Aplikace StreamXpress dovoluje modulaci těchto standardů:

- ATSC standard pozemního televizního vysílání používaný v USA, Kanadě a Mexiku
- DVB-C standard kabelového digitálního vysílání
- DVB-H standard pozemního digitálního televizního vysílání pro mobilní zařízení
- DVB-T standard pozemního digitálního televizního vysílání
- QAM-B modulací standard kabelového televizního vysílání používaný nejčastěji v severní Americe
- QAM-C modulací standard kabelového televizního vysílání používaný nejčastěji v Asii [12]

Pro testování této aplikace je důležité nastavení pro modulacích standard DVB-T, který po jeho výběru dovoluje nastavení těchto parametrů:

- šířka pásma modulovaného kanálu (5, 6, 7 nebo 8 MHz)
- mód modulace (modulace DVB-T podporuje 3 módy 2K, 4K a 8K)
- režimy modulace (QPSK, 16-QAM, 64-QAM)

- délka ochranného intervalu (představuje trvání prázdného symbolu  $v$  i trvání užitečného symbolu a může nabývat hodnot  $1/32, 1/16, 1/8, 1/4$ )
- poměr proti chybovému zabezpečení FEC ( $1/2, 2/3, 3/4, 5/6$  nebo  $7/8$ )
- další nastavení [12]

Poslední část obrazovky aplikace tvoří část umožňující spuštění otevřeného transportního datového toku. Je zde zobrazen posuvník zobrazující aktuální pozici, na které se přehrávaný datový tok nachází, dále také políčko zobrazující počet chyb, za ním následuje aktuální čas v přehrávaného transportního toku. Uprostřed se nachází tlačítko, po jehož stisknutí se zapne/vypne smyčka transportního toku. Textové políčko #Wraps udává počet opakování a vedle nahoře se zobrazuje celkový čas přehrávání transportního toku. [12]

Posledním článkem z architektury standardu HbbTV je samotný hybridní terminál. Pro tento konkrétní případ byl vybrán televizor firmy Samsung označením ES5500 (obr. 56). Dle specifikací výrobce se jedná o LCD televizor s LED technologií. Úhlopětka tohoto přijímače je 32 palců (80 cm) a Full HD rozlišením (1920 x 1080 px). Jedná se o tzv. Smart televizor, který umožňuje stahovat z portálu výrobce různé widgety pro použití přijímače nejen jako klasického televizoru. Díky možnosti připojení k síti pomocí konektoru RJ-45 se může tento televizor připojit také k internetu a umožňuje tak uživateli prohlížení webových stránek bez použití jiného připojeného zařízení. K prohlížení internetových stránek je v přijímači integrován webový prohlížeč. [13]



Obr. 56 Samsung ES5500 [13]

Po propojení zmíněných komponent vznikla sestava zaznamenaná na obr. 57. Na připojeném notebooku je v programu spuštěn program StreamXpress, který přes modulátor DTU-215 vysílá z uloženého datového toku data anténní vstup hybridního přijímače. Na tomto přijímači bylo před samotným testováním vyhledat kanály vysílané přes modulátor. Po vypnutí na nalezený kanál s číslem 11 se zobrazí šedý proužek s červeným tvarem kempu a nápisem HbbTV App (), který indikuje, že HbbTV aplikace se přijímači načítá. Po následném stisknutí červeného tlačítka se zobrazí obrazovka se seznamem videí, kterým lze listovat a vybrané video přehrát.



*Obr. 57 Sestava pro testování HbbTV aplikace*

### **4.3 Použití přímého streamování z internetu**

Doplňkové služby přenášené společně s televizním vysíláním jsou v dnešní době zcela běžné. V počátcích televizního vysílání tomu tak však nebylo, protože v původní analogovém vysílání se nedaly přenášet žádné další služby, ale i přesto se v průběhu vývoje podařilo využít část přenosového signálu, do které přidaly další informace. Ke zlepšení možností přenášet další informace do televizního vysílání došlo až s nástupem digitalizace přenosu televizního vysílání, kde je již část datového toku vyhrazena pro tyto další informace.

Většina technologií popsaných v této práci je již mezi uživateli zažita. Nejnovější technologie HbbTV je zatím hojně rozšířena v západní Evropě (Německo, Francie), kde dochází k velkému rozmachu této doplňkové služby a většina televizních kanálů má svou vlastní HbbTV aplikaci. V České republice je prozatím jediným poskytovatelem veřejné jednatelství česká televize, ale v blízké době se nejspíše objeví i aplikace dalších televizních společností. Česká televize má svou vlastní HbbTV aplikaci, v níž má v současné době přístupná všechna svá archivní videa společně se zpravodajským portálem a upravenou verzí teletextu. Podobně jako česká televize by mohly využívat technologii

spojující širokopásmový internet a televizní vysílání i další české televizní společnosti, čímž by umožnily uživateli sledovat pořady z jejich archívu přímo na televizních obrazovkách bez potřeby dalšího externího zařízení s internetovou konektivitou (běžná a nebo kompaktní PC, tablety a jiná zařízení s TV výstupem apod.). Technologie HbbTV by také mohli využívat i poskytovatelé kabelového nebo satelitního vysílání pro vytváření nových videoték nebo informačních portálů.

Zmíněný problém podpory technologie HbbTV v příjímácích je také v dnešní době již zanedbatelný, protože v příjímáky prodávány i na našem trhu již podporu technologie HbbTV standardně obsahuje. V příjímáky domácností splňuje také druhou podmínku pro její příjem, kterou je připojení k internetu.

Vytvořená aplikace založená na standardu HbbTV je, jak již bylo zmíněno, ukázkou možností streamování audiovizuálního obsahu na obrazovku televizního příjímáče. Každá taková aplikace je však pevně spojena s datovým tokem daného televizního kanálu a tak se naskytuje otázka: „Proč nestreamovat tento audiovizuální obsah na televizní obrazovky přímo bez vázanosti na digitální datový tok televizního vysílání?“

Část populace, především mladší lidé, by možnost využít televizní příjímáky jako zařízení pro příjem streamovaného obsahu z internetu uvítala. Tuto část tvoří lidé, kteří se o novinky na poli elektronických zařízení zajímají a příjemně využívají technologií je potěší. Zbývá část populace je stále ještě zastáncem názoru, že primárním úkolem televizoru je příjem televizního vysílání, například zobrazování audiovizuálního obsahu z připojených přehrávačů. Tato část populace by navrhouvanou možnost streamování audiovizuálního obsahu přímo z internetu nejspíše neuvítala nebo kdyby zde tato možnost byla, tak by ji zpočátku nejspíše nevyužívala. Postupem času by se možná i konzervativní uživatelé uvolili k jejímu používání, když by zjistili, že si nemusí k televizoru připojovat žádné další zařízení, ale tento obsah si mohou prohlédnout přímo z internetu.

V době, kdy v příjímáky domácností má možnost internetového připojení je proto asi nejlepší odpovědí na položenou otázku: „Proč, ne?“. Bude-li i samotné televizní vysílání streamováno prostřednictvím internetu, sjednotí se způsob jeho přenosu - a díky myšlence síťové neutrality uživateli odpadnou historické starosti o dostupnost kabeláže, obsah poskytovaný lokálním kabelovým/satelitním operátorem a nebo koncesovanými pozemními televizními společnostmi. Do doby než toto nastane, zůstane technologie HbbTV "překlenovacím mostem" mezi běžným televizním vysíláním a streamovaným audiovizuálním obsahem z internetu. První krok této změny, připojení televizních příjímáček k internetu, již byl proveden.



## 5 ZÁVĚR

Hlavními cíli této diplomové práce bylo zachytit evoluci vývoje dodatkových technologií televizního vysílání a také vytvoření demonstrační aplikace založené na technologii HbbTV, kombinující klasické televizní vysílání s internetovým obsahem.

Předložená diplomová práce tedy ve své rešeršní části sumarizuje vývoj technologií doplňkového obsahu předávaného konzumentovi televizního vysílání, od teletextu až po hybridní televizory. U každé z jednotlivých technologií je vždy popsán princip jejího provozu. Z této sumace je patrné, že možnosti, jak obsahového rozsahu, tak i grafické stránky služeb se zlepšily až s nástupem digitálního vysílání. V další části jsou pak krok za krokem vysvětleny hardwarové požadavky a také postup, kterým si musí vývojář HbbTV aplikací projít při jejich vytváření. V této části jsou také vysvětleny možnosti, které nabízejí jednotlivé části ukázkové aplikace pro koncového uživatele.

Tato aplikace byla úspěšně vytvořena pro zadavatelskou firmu jako demonstrace možností nejnovější technologie na poli spojujícím běžné televizní vysílání spolu s obsahem streamovaným pomocí širokopásmového internetu, ve kterém se tato firma pohybuje. Vytvořená aplikace sestává z modulu demonstrujícího streamování kontextového videa, ukázky stránky s programovou nabídkou a z modulu komunikujícího s aplikací připojenou k databázovému serveru pro divácké hlasování. Tato aplikace je základem pro rozsáhlejší aplikace nabízené ke komerčnímu využití nyní i budoucím zákazníkům zadavatelské firmy.

Jelikož je technologie HbbTV technologií mladou, tak i její rozšíření je v současné době především v západních zemích. Česká republika v implementaci mírně zaostává, nicméně díky české televizi je už i u nás k dispozici programová nabídka s vlastními HbbTV aplikacemi, ke které se v blízké době jistě připojí i další vysílací společnosti. Technologii HbbTV je v současné době možné považovat za určitý „nenásilný mezikrok“ mezi klasickou koncepcí „konzumace jednosměrně vysílaného televizního obsahu“ a trendem k interaktivnímu streamovanému „vysílání na požádání“, s bohatým doplňkovým obsahem který je divákům k dispozici na internetu.



## SEZNAM POUŽITÉ LITERATURY

- [1] LEGÍ , Martin. *Televizní technika DVB-T*. 1. vyd. Praha: BEN - technická literatura, 2006. ISBN 80-730-0204-3.
- [2] HANUS, Stanislav. *Základy televizní techniky III: p ednášky*. Vyd. 1. V Brn : Vysoké u ení technické, Fakulta elektrotechniky a komunika ních technologií, Ústav radioelektroniky, 2010. ISBN 978-80-214-4206-1.
- [3] MACK, Zden k a Miloš SEQUENS. *Teletext*. Praha: UTRIN, 1988, 22 s.
- [4] VÍT, Vladimír. *Televizní technika: P enosové barevné soustavy*. 1. vyd. Praha: BEN - technická literatura, 1997, 719 s. ISBN 80-860-5604-X.
- [5] *Electronic Programme Guide (EPG); Protocol for a TV Guide using electronic data transmission* [online]. [cit. 2014-03-23]. Dostupné z WWW: <[http://www.etsi.org/deliver/etsi\\_en/300700\\_300799/300707/01.02.01\\_40/en\\_300707v010201o.pdf](http://www.etsi.org/deliver/etsi_en/300700_300799/300707/01.02.01_40/en_300707v010201o.pdf)>.
- [6] VŠETE KA, Roman. *Digitální televize za pár korun. Otestovali jsme dev t nejlevn jších set-top box* [online]. 24. 10. 2008,[cit. 2014-05-06]. Dostupné z: <[http://www.samsung.com/hk\\_en/consumer/tv-av/televisions/smart-led-tv/UA40ES5500JXZK-spec](http://www.samsung.com/hk_en/consumer/tv-av/televisions/smart-led-tv/UA40ES5500JXZK-spec)>
- [7] Í NÝ, Václav. *Co je to hybridní televize HbbTV a jak tato platforma funguje?* [online]. 5. 12. 2011, [cit. 2014-04-10]. Dostupné z: <<http://www.digizone.cz/clanky/co-je-hybridni-televize-hbbtv-a-jak-funguje/>>.
- [8] *Specification: HbbTV* [online]. [cit. 2014-05-01]. Dostupné z WWW: <[http://hbbtv.org/pages/about\\_hbbtv/specification.php](http://hbbtv.org/pages/about_hbbtv/specification.php)>.
- [9] *Technical Specification: Hybrid Broadcast Broadband TV* [online]. [cit. 2014-03-25]. Dostupné z WWW: <[http://www.etsi.org/deliver/etsi\\_ts/102700\\_102799/102796/01.02.01\\_60/ts\\_102796v010201p.pdf](http://www.etsi.org/deliver/etsi_ts/102700_102799/102796/01.02.01_60/ts_102796v010201p.pdf)>.
- [10] TV program – XML [online]. [cit. 2014-05-07]. Dostupné z WWW: <<http://www.ceskatelevize.cz/xml/tv-program/>>
- [11] *Fire HbbTV: Features detail* [online]. [cit. 2014-05-01]. Dostupné z WWW: < <http://tum-iptv.aw.atosorigin.com/firehbbtv/#features> >
- [12] *DTC-300-SP: StreamXpress® Software* [online]. [cit. 2014-04-25]. Dostupné z WWW: <<http://www.dektec.com/products/Apps/DTC-300/Downloads/DTC-300-SP%20StreamXpress%20Manual.pdf>>.
- [13] *40" ES5500* [online]. [cit. 2014-05-06]. Dostupné z: <[http://www.samsung.com/hk\\_en/consumer/tv-av/televisions/smart-led-tv/UA40ES5500JXZK-spec](http://www.samsung.com/hk_en/consumer/tv-av/televisions/smart-led-tv/UA40ES5500JXZK-spec)>.