UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

JEFFREI SOARES MOREIRA

PROJETO DE DIPLOMAÇÃO

DESENVOLVIMENTO DE MÓDULO DEMULTIPLEXADOR PARA PADRÃO BRASILEIRO DE TELEVISÃO DIGITAL

Porto Alegre
2011

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

DESENVOLVIMENTO DE MÓDULO DEMULTIPLEXADOR PARA PADRÃO BRASILEIRO DE TELEVISÃO DIGITAL

Projeto de Diplomação apresentado ao Departamento de Engenharia Elétrica da Universidade Federal do Rio Grande do Sul, como parte dos requisitos para Graduação em Engenharia Elétrica.

ORIENTADOR: Prof. Dr. ALTAMIRO AMADEU SUSIN
CO-ORIENTADOR: Eng. M.Sc. ALEXSANDRO CRISTOVÃO BONATTO

Porto Alegre
2011

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

JEFFREI SOARES MOREIRA

DESENVOLVIMENTO DE MÓDULO DEMULTIPLEXADOR PARA PADRÃO BRASILEIRO DE TELEVISÃO DIGITAL

Este projeto foi julgado adequado para fazer jus aos créditos da Disciplina de "Projeto de Diplomação", do Departamento de Engenharia Elétrica e aprovado em sua forma final pelo Orientador e pela Banca Examinadora.

Orientador: _____

Prof. Dr. Altamiro Amadeu Susin, UFRGS

Doutor pelo Institut National Polytechnique de Grenoble, França

Banca Examinadora:

Prof. Dr. Altamiro Amadeu Susin, UFRGS

Doutor pelo Institut National Polytechnique de Grenoble, França

Prof. Dr. Marcelo Götz, UFRGS

Doutor pela Universität Paderborn, Alemanha

Prof. Dr. Tiago Roberto Balen, Unilasalle

Doutor pela Universidade Federal do Rio Grande do Sul – Porto Alegre, Brasil

Porto Alegre, julho de 2011.

Dedicatória

Dedico este trabalho aos meus pais pela dignidade e pelos valores com que constituíram nossa família, pelo esforço de uma vida para mudar de vida e por sempre me apoiarem e me ajudarem a acreditar que este momento era possível, que este dia chegaria.

Agradecimentos

Agradeço a Deus pelas bênçãos em minha vida, pelo lar estruturado e pelas oportunidades que me concedeu.

Agradeço especialmente à minha família por todo o esforço e pelo apoio incondicional durante esta caminhada.

Agradeço ao meu orientador, professor Altamiro Susin, pela oportunidade de iniciação científica no LaPSI e pelas muitas aulas de sabedoria, das coisas simples às mais complexas.

Agradeço aos colegas do LaPSI pela parceria e pela permanente dispoção para ajudar, corrigir, incentivar ou simplesmente conversar, e assim criar um ambiente diferenciado de trabalho e de aprendizado.

Faço um agradecimento especial ao meu co-orientador, Alexsandro Bonatto, por nortear os trabalhos e pela paciência e dedicação na correção dos erros de percurso.

Agradeço, por fim, à Universidade, ao corpo docente do curso de Engenharia Elétrica e aos funcionários, pela oportunidade rara de uma formação pública e de qualidade no Brasil.

RESUMO

Este documento traz os relatos sobre a elaboração de um módulo demultiplexador funcional em linguagem de descrição de hardware (VHDL) para processamento de dados de entrada no sistema brasileiro de televisão digital (SBTVD), sendo objeto de projeto de diplomação para a disciplina ENG04029 do curso de Engenharia Elétrica da Universidade Federal do Rio Grande do Sul. São abordados o contexto do desenvolvimento da tecnologia no Brasil e as especificações da norma relativas ao *Transport Stream* (TS), ou Fluxo de Transporte, pacote de informações previsto em norma que deve ser analisado e decomposto pelo demultiplexador proposto. A solução desenvolvida ao longo do trabalho é apresentada, com a descrição completa do demultiplexador de TS para SBTVD e sua validação.

Palavras-chaves: SBTVD, Demultiplexador, Transport Stream

ABSTRACT

This document reports the elaboration of a demultiplexer in hardware description language (VHDL) which processes the typical input data of the Brazilian digital television system (ISDB-T). This work has been developed in the framework of a graduation project for the course ENG04029 of the Electrical Engineering degree of Federal University of Rio Grande do Sul (UFRGS). The context of the development of this technology in Brazil is depicted, as well as the main specifications and concerns about a Transport Stream (TS), the standardized packet of information which must be analyzed and decomposed by the demultiplexer. In this work a complete demultiplexer was developed, and it is fully described, as well as its validation methods.

Keywords: ISDB-T, Demultiplexer, Transport Stream

SUMÁRIO

1. INTRODUÇÃO 11

2. CONTEXTO DO PROJETO 13

2.1 O TERMINAL DE ACESSO (TA) 15
2.2 DEMULTIPLEXADOR DE TS 16

3. ANÁLISE DE ALTERNATIVAS 17

4. MÉTODOS, PROCESSOS E DISPOSITIVOS 18

4.1 DEFINIÇÕES ELEMENTARES 18
4.1.1 Payload 19
4.1.1.1 PES 19
4.1.1.1.1 Áudio 20
4.1.1.1.2 Vídeo 21
4.1.1.1.3 ES Privados (Legendas) 21
4.1.1.2 PSI 22
4.1.1.2.1 PAT 23
4.1.1.2.2 PMT 24
4.1.1.2.3 NIT 25
4.1.2 Adaptation Field 25
4.2 TS 26
4.2.1 Cabeçalho 27
4.2.2 Corpo 28
4.3 O DEMULTIPLEXADOR 29
4.3.1 Versão Anterior 31
4.3.2 Arquitetura do Demultiplexador Proposta Neste Trabalho 32
4.4 RESUMO DE OBJETIVOS 34
4.5 METODOLOGIA DE DESENVOLVIMENTO 34
4.6 TRABALHO REALIZADO 35
4.6.1 Estrutura do software 36
4.6.2 Algoritmo do software 37
4.6.2.1 PAT 37
4.6.2.2 PMT 39
4.6.2 Descrição de Hardware do demultiplexador em VHDL 41
4.6.2.1 Análise de PAT 42
4.6.2.2 Análise de PMT 43
4.6.2.3 Análise de AF 44
4.6.2.4 Análise de Legendas 45
4.6.2.5 Análise de Áudio/Vídeo 46
4.6.2.6 Módulo de Controle 46
4.6.2.7 Integração: Módulo de Gestão de Fluxo de Dados para Decodificadores 47
4.6.2.8 Módulo topo do demultiplexador 48

5. RESULTADOS ALCANÇADOS 50

5.1 SOFTWARE DE REFERÊNCIA 50
5.2 HARDWARE 51
5.3 TESTES E VALIDAÇÃO 51
5.3.1 Simulação 52
5.3.2 Síntese em placa 53

6. CONCLUSÃO 55

7. REFERÊNCIAS BIBLIOGRÁFICAS 56

Lista de Ilustrações

FIGURA 1 - MULTIPLEXAÇÃO E DEMULTIPLEXAÇÃO DE FLUXOS DE DADOS 11
FIGURA 2 - DIAGRAMA DE BLOCOS SIMPLIFICADO DO HARDWARE DO SBTVD 14
FIGURA 3 - ARQUITETURA DO TERMINAL DE ACESSO 15
FIGURA 4 - VISÃO GERAL DO DEMULTIPLEXADOR 16
FIGURA 5 - ESTRUTURA DE UM PES E INFORMAÇÕES QUE DELE PODEM SER EXTRAÍDAS 19
FIGURA 6 - ESTRUTURA E INFORMAÇÕES DE UMA PAT 23
FIGURA 7 - ESTRUTURA E INFORMAÇÕES DE UMA PMT 24
FIGURA 8 - ESTRUTURA E INFORMAÇÕES DE UM AF 25
FIGURA 9 - ESTRUTURA BÁSICA DE UM PACOTE TS 26
FIGURA 10 – INFORMAÇÕES PRINCIPAIS DO CABEÇALHO DE UM TS 27
FIGURA 11 - ÁRVORE DE INFORMAÇÕES DO CORPO DE UM TS 28
FIGURA 12 - ENTRADAS DO MÓDULO DEMULTIPLEXADOR 30
FIGURA 13 - INTERFACE DA VERSÃO ANTERIOR DE DEMUX 31
FIGURA 14 - FLUXOGRAMA DA VERSÃO ANTERIOR DO DEMULTIPLEXADOR 32
FIGURA 15 - FLUXO DE DADOS E INFORMAÇÕES DA OPERAÇÃO DO DEMULTIPLEXADOR 33
FIGURA 16 - ESTRUTURA DE DADOS DA PAT PREVISTA EM NORMA 37
FIGURA 17 - FLUXOGRAMA DE PROCESSAMENTO DE UMA PAT 38
FIGURA 18 - ESTRUTURA DE DADOS DA PMT PREVISTA EM NORMA 39
FIGURA 19 - ALGORITMO DE PROCESSAMENTO DE UMA PMT 40
FIGURA 20 - INTERFACE DO DEMULTIPLEXADOR DESENVOLVIDO NESTE TRABALHO 42
FIGURA 21 - MÓDULO DE ANÁLISE DE PAT 43
FIGURA 22 - MÓDULO DE ANÁLISE DE PMT 44
FIGURA 23 - MÓDULO DE INTERPRETAÇÃO DE CAMPOS DE ADAPTAÇÃO 45
FIGURA 24 - MÓDULO DE ANÁLISE DE PES DE LEGENDAS 45
FIGURA 25 - MÓDULO DE ANÁLISE DE PES DE ÁUDIO/VÍDEO 46
FIGURA 26 - MÓDULO DE CONTROLE DO DEMULTIPLEXADOR 47
FIGURA 27 - MÓDULO DE CONTROLE DE FLUXO 48
FIGURA 28 - DEMULTIPLEXADOR COMPLETO 49
FIGURA 29 - IDENTIFICAÇÃO DE UMA TABELA PAT EM SIMULAÇÃO 52
FIGURA 30 - IDENTIFICAÇÃO DE UMA TABELA PMT EM SIMULAÇÃO 53
FIGURA 31 – FERRAMENTA DE TRABALHO: PLACA DE PROTOTOTIPAÇÃO VIRTEX 5 54

LISTA DE ABREVIATURAS

ABNT: Associação Brasileira de Normas Técnicas

AF: *Adaptation Field*, ou Campo de Adaptação

CAT: *Conditional Association Table*

DELET: Departamento de Engenharia Elétrica

ES: *Elementary Stream*, ou Fluxo de Elementar

ISO: *International Organization for Standardization*

LaPSI: Laboratório de Processamento de Sinais e Imagens

MPEG: *Moving Picture Experts Group*

NIT: *Network Information Table*

PAT: *Program Association Table*

PCR: *Program Clock Reference*

PES: *Packetized Elementary Stream*

PID: *Packet Identifier*

PMT: *Program Map Table*

PSI: *Program Specific Information*

SoC: *System on Chip*

STB: *Set-top Box*, ou Terminal de Acesso

TA: Terminal de Acesso

TS: *Transport Stream*, ou Fluxo de Transporte

UFRGS: Universidade Federal do Rio Grande do Sul

VCO: *Voltage-controlled Oscillator*

VHDL: VHSIC *Hardware Description Language*

1. Introdução

Este projeto de criação de um módulo demultiplexador está inserido em um projeto maior, de escopo nacional. Trata-se de uma cooperação acadêmica envolvendo sistemas digitais com vistas ao desenvolvimento de tecnologia própria na área de codificação e decodificação de fluxos de dados utilizados para transmissão de programas de televisão em formato digital, seguindo o padrão brasileiro.

Os referidos fluxos estão amplamente padronizados pela norma ISO 13818-1 [1]. Neles são enviadas, por uma única via de transporte, as informações de áudio, vídeo e legendas (*closed caption*) através de pacotes com estrutura bem determinada, fato que permite a sua interpretação e posterior processamento. Além disso, são enviados na mesma frequência programas distintos em alta e baixa resolução, para televisores e para dispositivos móveis, respectivamente, os quais têm fluxos de vídeo e áudio diferentes.

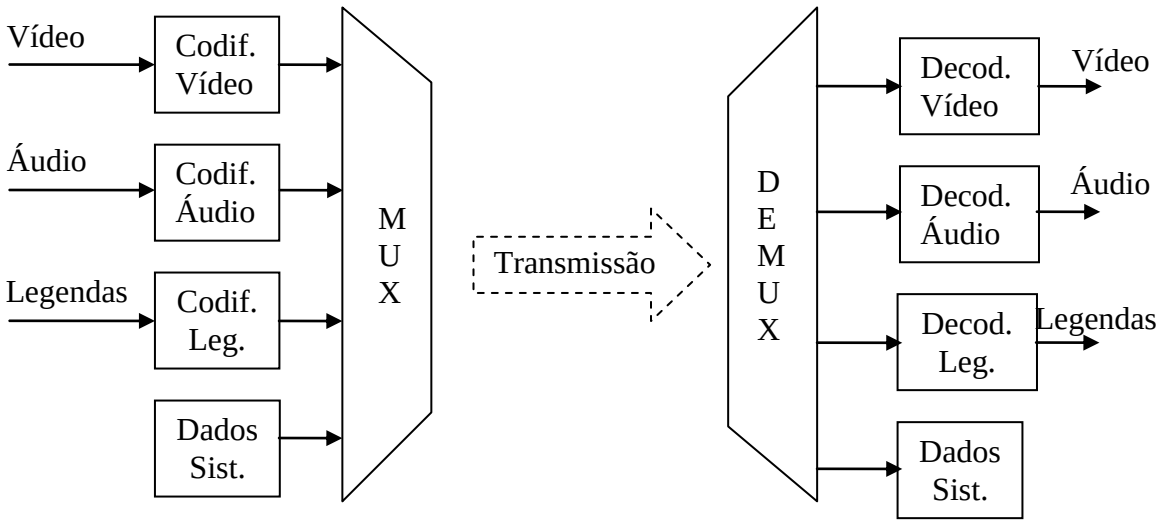

Figura 1 - Multiplexação e demultiplexação de fluxos de dados

O demultiplexador, objeto central deste trabalho, é o módulo situado na entrada do conjunto receptor denominado Terminal de Acesso (TA), sendo responsável pela análise

inicial desses pacotes, extração dos fluxos específicos de áudio, vídeo e dados, de acordo com o programa e a faixa de áudio selecionados pelo usuário, e envio dos mesmos para seus respectivos decodificadores. É este também o componente que deve recuperar as informações do sistema e disponibilizar tais dados para eventual processamento em nível superior dentro do TA. A Figura 1 exibe esquematicamente as tarefas de multiplexação e demultiplexação, explicitando o fato de que todos os fluxos envolvidos são enviados por um único canal de transmissão, sendo, no caso da SBTVD, ondas eletromagnéticas transmitidas pelo ar em frequências determinadas para cada emissora.

O desafio na elaboração deste módulo, essencial para o desenvolvimento e funcionamento dos Terminais de Acesso, reside principalmente na complexidade das estruturas normatizadas e no tempo previsto para criação e validação de um protótipo.

Este documento trará inicialmente uma contextualização que explicitará a posição exata deste trabalho dentro do referido projeto nacional, seguida por uma descrição técnica dos pacotes existentes, seus formatos e a função do demultiplexador ao examiná-los, traçando objetivos de desenvolvimento. Na sequência é apresentada a metodologia de trabalho, com as decisões de projeto que se julgou pertinentes. Por fim são apresentados os resultados obtidos ao longo deste trabalho.

2. Contexto do projeto

Conforme mencionado anteriormente, o desenvolvimento deste módulo demultiplexador faz parte de um projeto nacional, denominado SoC – SBTVD. Este recebe apoio e fomento estatal por parte da Fundação de Empreendimentos Científicos e Tecnológicos (FINATEC), tendo previsão de continuidade durante todo o ano de 2011. Participam deste programa, lideradas pela Universidade de Brasília, universidades do Rio Grande do Sul e do Rio Grande do Norte, entre outros, reforçando e legitimando o propósito de geração de conhecimento e formação de recursos humanos em todo o país.

Como a sigla que dá título ao projeto indica, este esforço visa à elaboração de um sistema em chip (SoC: *System on Chip*) contendo todo o hardware necessário para realização autônoma das tarefas relacionadas à recepção de dados digitais pela rede do sistema brasileiro de televisão, incluindo os blocos primordiais de decodificação. A integração desses módulos em um único chip permitirá a queda dos custos, promovendo a desejada Inclusão Digital. Dentro desse conjunto de atividades, foram atribuídas ao Laboratório de Processamento de Sinais e Imagens (LaPSI) da UFRGS, onde o presente trabalho é desenvolvido, tarefas focadas exclusivamente no Terminal de Acesso (TA), ou seja, na decodificação de fluxos do sistema e exibição de seus resultados para o consumidor final.

No LaPSI são atualmente estudados, desenvolvidos em hardware e/ou completados em paralelo o decodificador de vídeo, a saída de vídeo, o processador de legendas, o decodificador de áudio, a gestão dos periféricos e a integração dos mesmos. A essas tarefas se junta este trabalho na parte de demultiplexação, o qual permitirá o ensaio e a validação do sistema completo do TA. O diagrama apresentado na Figura 2 permite visualizar essa posição do demultiplexador proposto dentro do projeto nacional. Nele estão esquematizados os principais elementos envolvidos no processo completo de um sistema de televisão digital,

incluindo: captura de dados (por exemplo: realização de filmagens com áudio e vídeo); codificação, para eliminar redundâncias e otimizar o tamanho dos dados a serem enviados; multiplexação dos diferentes fluxos; modulação do sinal para adaptá-lo ao meio de transmissão; recepção e demodulação para recuperar os dados originais; demultiplexação, a fim de separar e encaminhar os fluxos; decodificação, para recuperar a quase totalidade dos dados inicialmente capturados; e, por fim, exibição da combinação dos dados para o usuário final.

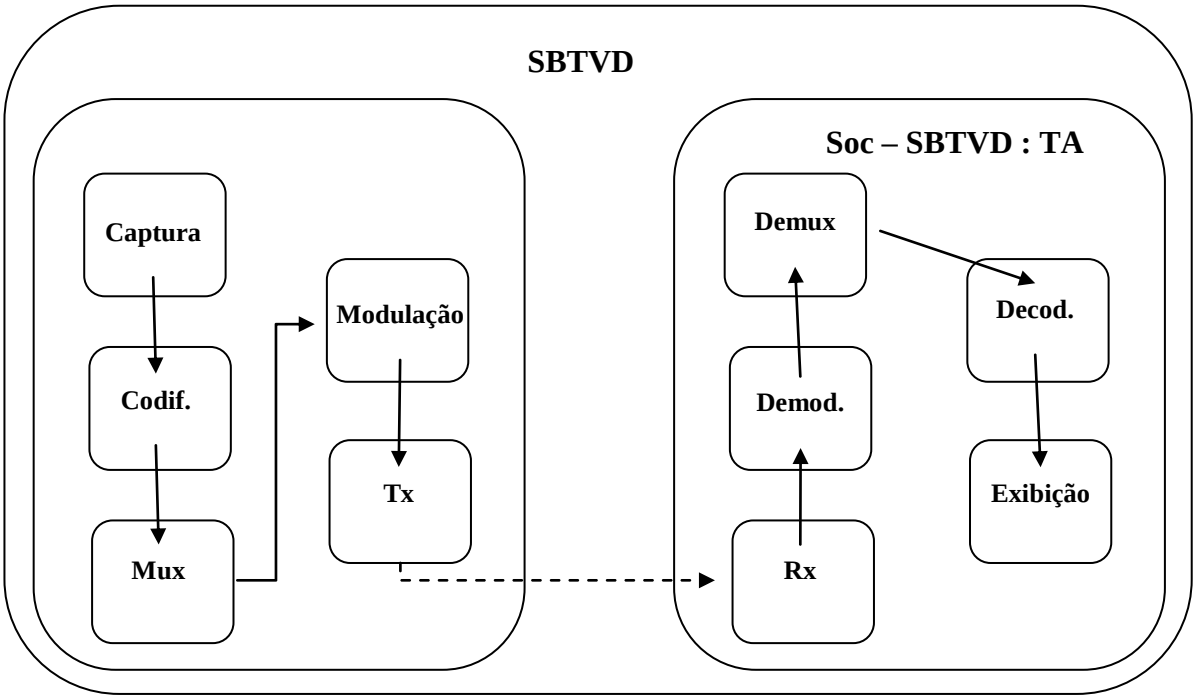

Figura 2 - Diagrama de blocos simplificado do hardware do SBTVD

A metodologia geral do projeto desenvolvido no laboratório consiste em leitura e interpretação dos documentos que constituem a norma a ser obedecida e eventual elaboração ou estudo de software de referência; por fim, o que se espera como objeto final é a realização de um módulo funcional em linguagem de descrição de hardware, em geral VHDL, que possa ser validado e enfim sintetizado em chip para futura produção comercial em larga escala.

2.1 O Terminal de Acesso (TA)

A Figura 3 apresenta a arquitetura proposta pela equipe do projeto, ao longo de anos de estudo, para o Terminal de Acesso. Nela se observa claramente a posição do demux dentro do fluxo de dados, à frente dos decodificadores.

O sistema é globalmente gerido por uma Cpu responsável pela gestão de seus periféricos básicos, que incluem portas de comunicação e entrada de dados, e dos módulos desenvolvidos no projeto, entre os quais se destaca o demultiplexador. É também responsável por todo tipo de interação com o usuário, gerenciando a interface gráfica, captando todas as suas escolhas e encaminhando-as aos módulos concernidos. Uma segunda *cpu* se encarrega exclusivamente das tarefas relacionadas à entrada de dados. Ela busca o *bitstream* recebido, já demodulado, e encaminha o *Transport Stream* (TS) diretamente, sem processamento, para o demultiplexador através do barramento AMBA – AHB indicado na figura, cujo padrão é aberto e definido em [2].

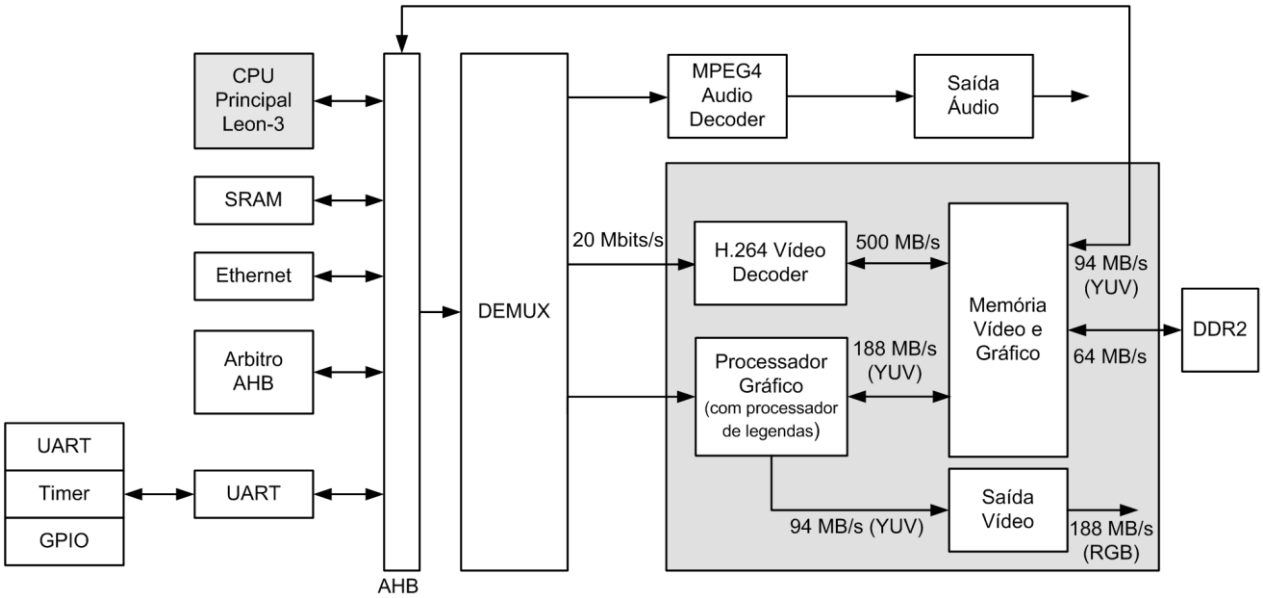

Figura 3 - Arquitetura do Terminal de Acesso

A partir do demultiplexador todo o processamento de áudio, vídeo e dados é realizado em hardware. No módulo demultiplexador o TS é processado, sendo dele extraídos os fluxos de áudio, vídeo e legendas, os quais são repassados a seus respectivos decodificadores.

2.2 Demultiplexador de TS

A Figura 4 traz uma primeira visão geral do módulo demultiplexador, objeto central deste trabalho.

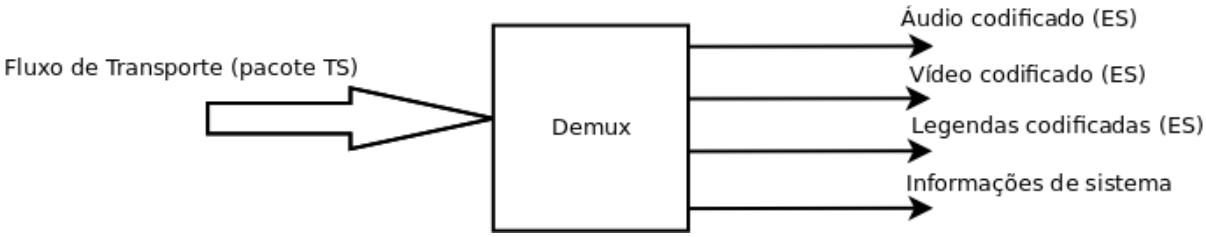

Figura 4 - Visão geral do demultiplexador

Entre as informações de sistema indicadas na figura, as quais podem ser obtidas durante a operação do demultiplexador, estão dados relativos à rede e seus parâmetros físicos (freqüência, por exemplo), além de informações de mais alto nível, como a grade de programação diária.

Em paralelo à decodificação de áudio são realizadas as decodificações de vídeo e legendas, os quais são posteriormente combinados pela saída de vídeo para compor os quadros finais, com legenda quando este item for solicitado pelo usuário.

3. Análise de alternativas

Anteriormente ao início deste trabalho foram realizados breves testes de desempenho com uma implementação extremamente básica de demultiplexador em software. De acordo com estudo anterior realizado em [3] pelos colegas de graduação Jônatas Rech, Felipe Ilha e Leonardo Faganello, ex-membros da equipe do laboratório, foi escolhido como Cpu principal o softcore Leon3 [4], responsável pela gestão do TA e pela interação com o usuário.

Inicialmente, sem conhecer o demux e sabendo apenas de sua natureza seqüencial, pensou-se em realizar a tarefa de demultiplexação completa em software nesse processador, por simplicidade de criação em linguagem de programação de alto nível. Um primeiro protótipo para validação foi então elaborado em linguagem C pelo colega Fabrício Caetano e testado em placa pelo colega Jônatas Rech. No entanto, nesses ensaios o demultiplexador em software apresentou desempenho de processamento bastante abaixo do necessário, que de acordo com as especificações de projeto seria uma taxa de processamento próxima de 2,5MB/s.

Por esse motivo foi descartado o aprofundamento em torno da versão em software, sendo que esta passou a servir apenas como parâmetro para eventuais comparações e validações. Assim, decidiu-se por esta implementação em hardware como a única alternativa capaz de responder satisfatoriamente às necessidades do projeto em termos de desempenho.

4. Métodos, Processos e Dispositivos

O demultiplexador da televisão digital, no padrão brasileiro, recebe como entrada os pacotes de informação na forma de MPEG2 – TS, cuja estrutura é descrita na norma [1]. O TS, ou fluxo de transporte, é uma camada de abstração de transporte que permite o encapsulamento de diversos tipos de mídia dentro de um mesmo pacote. Ele é, na verdade, uma sucessão de pacotes de 188 bytes identificados, no cabeçalho, por um PID que indica a natureza de seu conteúdo.

A compreensão das tarefas pertinentes para um módulo demultiplexador exige primeiramente uma noção mais precisa sobre as informações contidas em um TS. Por essa razão, esta seção está estruturada como segue: primeiramente são definidos os conteúdos que podem ser enviados através de um TS e termos a ele relacionados, para em seguida, na seção 4.2, ser possível apresentar e descrever a estrutura esperada de um fluxo de transporte. Dessa forma, na seção seguinte pode enfim ser apresentado o demultiplexador, suas entradas e saídas esperadas. Após um relato sobre discussões e opções de projeto, são então listados os objetivos definidos para este trabalho na seção 4.4. Por fim, a seção 4.5 descreve a metodologia escolhida para elaboração do módulo proposto.

4.1 Definições Elementares

Basicamente, são dois os tipos de dados que podem ser transmitidos dentro de um fluxo de transporte: *payload* e *adaptation field*.

4.1.1 Payload

Estão inclusos na definição de *payload* todo tipo de dados carregando informação útil para o correto funcionamento de um TA, o que inclui, além dos codificados de áudio e vídeo, as informações de rede e da emissora, necessárias para interação com o usuário e para correta interpretação do TS. Há dois tipos de *payload*: PES e PSI, os quais serão descritos a seguir.

4.1.1.1 PES

O PES que vai dentro do TS é, por si só, um pacote de tamanho variável que permite uma abstração do seu conteúdo, o qual pode ser fluxos elementares de áudio, vídeo ou legendas. No entanto, no caso do Fluxo Elementar Empacotado (do inglês: PES – *Packetized Elementary Stream*) trata-se de uma abstração de sistema que permite simplesmente que áudio, vídeo e legendas sejam vistos como uma mesma estrutura.

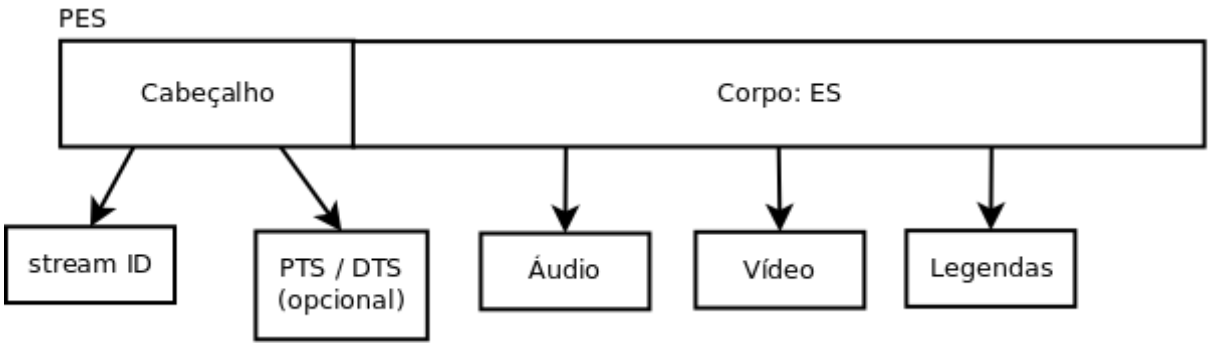

Figura 5 - Estrutura de um PES e informações que dele podem ser extraídas

Por pacote compreende-se uma estrutura com cabeçalho, contendo informações discriminando seu conteúdo, e uma seção com os dados úteis propriamente ditos. A Figura 5

ilustra essa composição. Como se pode observar, no cabeçalho, as principais informações obtidas são um *stream ID*, que identifica a natureza do fluxo elementar contido, e *timestamps* de áudio e vídeo, importantes para futura tarefa de sincronização entre as saídas.

A parte da informação útil (ex.: áudio, vídeo), a qual é enviada no corpo do PES, é chamada de Fluxo Elementar (do inglês: ES – *Elementary Stream*). Em última análise, os ES são os dados de interesse a serem extraídos do TS e repassados para os decodificadores correspondentes.

O padrão brasileiro é inspirado no sistema japonês, aqui foi adotada uma codificação mais moderna e otimizada de fluxos de mídia, seguindo o padrão MPEG-4. A seguir são descritos os principais ES, havendo distinção entre os formatos de mídia para recepção e exibição em televisores e em dispositivos portáteis.

4.1.1.1.1 Áudio

No padrão brasileiro é usado o áudio codificado em MPEG-4 HE-AAC (*Advanced Áudio Coding*), padronizado em [5] e [6]. A transmissão pode ser em modo estéreo ou multicanal (5.1) para televisores, e apenas em estéreo para dispositivos móveis.

Entre suas informações características estão um *stream_ID* obrigatoriamente da forma $11XXXXXX_b$ (onde ‘X’ denota indiferença) e seu eventual *timestamp* de apresentação (PTS). Este último é um contador amostrado a intervalos regulares durante a codificação e, por comparação com um contador semelhante no TA, permite ajustar o instante de exibição de um trecho de áudio.

4.1.1.1.2 Vídeo

As informações de vídeo, no padrão brasileiro, são transmitidos em formato MPEG-4 AVC (H.264), o qual é padronizado em [7] e [8]. A diferença de perfis de acordo com o dispositivo em que o vídeo será exibido implica a transmissão de dois formatos, denominados *one-seg* (dispositivos móveis) e *full-seg* (televisores), sendo consideravelmente grande a diferença de resolução e qualidade entre eles.

Entre suas informações características estão um *stream_ID* obrigatoriamente da forma $111XXXXX_b$ (onde 'X' denota indiferença) e seus eventuais *timestamps*. Além do de apresentação (PTS), que tem a mesma função de seu homólogo do áudio, há o de decodificação (DTS), que é transmitido em virtude da característica da codificação de vídeo utilizada, a qual foge ao escopo deste trabalho, mas que tipicamente pode conter quadros que dependem de quadros passados e futuros. Assim, um quadro futuro pode ser enviado antes de seu predecessor, que dele depende, e nesse caso o DTS auxilia no restabelecimento da ordem de quadros.

4.1.1.1.3 ES Privados (Legendas)

Nesta categoria estão os fluxos elementares privados, os quais são opcionais de acordo com a norma e podem ser específicos a cada emissora. Nela destacam-se as legendas, ou closed caption, as quais são transmitidas como PES privado tipo A, síncrono, e têm sua estrutura padronizada no SBTVD de acordo com [9].

4.1.1.2 PSI

O segundo tipo de *payload*, *Program Specific Information*, traz informações relevantes para a seleção dos fluxos elementares que devem ser efetivamente repassados aos decodificadores. Isso porque o SBTVD prevê transmissão concomitante de múltiplos programas pelo mesmo canal. Nesse caso, mais de um vídeo e diversas trilhas de áudio podem ser transmitidos através do TS em uma única faixa de freqüências. Ainda que seja um único programa da grade da emissora, ele sempre é transmitido em diferentes resoluções e isso, para a norma, é visto como programas distintos. Ou seja, há sempre, no mínimo, dois vídeos sendo transmitidos (*one-seg* e *full-seg*) e múltiplas trilhas de áudio para cada um dos programas, tudo sendo recebido pelo mesmo canal. No entanto, apenas uma faixa de vídeo deve ser decodificada (de acordo com o programa/resolução selecionado), e apenas uma trilha de áudio entre as várias associadas ao programa (de acordo com a seleção de áudio do usuário).

Portanto, é preciso primeiramente associar cada fluxo elementar ao seu respectivo programa, e isso é feito através de um PID (*Packet ID*). Essa informação consta no cabeçalho de cada pacote TS, identificando a informação nele contida. Entretanto a associação de um único PID a um determinado ES é dinâmica, dependente da emissora e da localização geográfica no país, e precisa ser informada ao TA. É basicamente essa a informação mais importante que pode ser extraída de um PSI, pois permite corresponder os fluxos decodificados à seleção feita pelo usuário.

Essas informações chegam ao TA na forma de tabelas dentro de um PSI, as quais são padronizadas em [1]. Há quatro tabelas básicas englobadas pelo tipo de *payload* denominado PSI, as quais são definidas no padrão ISO 13818-1; no entanto, na especificação brasileira presente em [10] são definidas mais algumas tabelas com diferentes informações, como por

exemplo a grade de programação diária do canal. Cada tabela é transmitida individualmente em um pacote TS. A seguir são descritas as três tabelas principais do padrão previstas no MPEG-2 TS.

4.1.1.2.1 PAT

A tabela PAT (*Program Association Table*) é a base do sistema de associação de PID a fluxos relativos a cada programa. Por isso, ela é uma das poucas entidades que apresenta um PID fixo previsto em norma (no caso “0x000”). Assim, sempre é possível reconhecê-la sem conhecimentos prévios sobre o sistema.

Como o nome indica, ela associa a cada programa sendo transmitido um PID específico. Esse PID, quando encontrado no cabeçalho de um TS, indica a presença da tabela PMT (*Program Map Table*, ver 4.1.1.2.2) associada àquele programa. A Figura 6 exemplifica a tabela que pode ser obtida e guardada em memória quando é processado um TS contendo uma PAT.

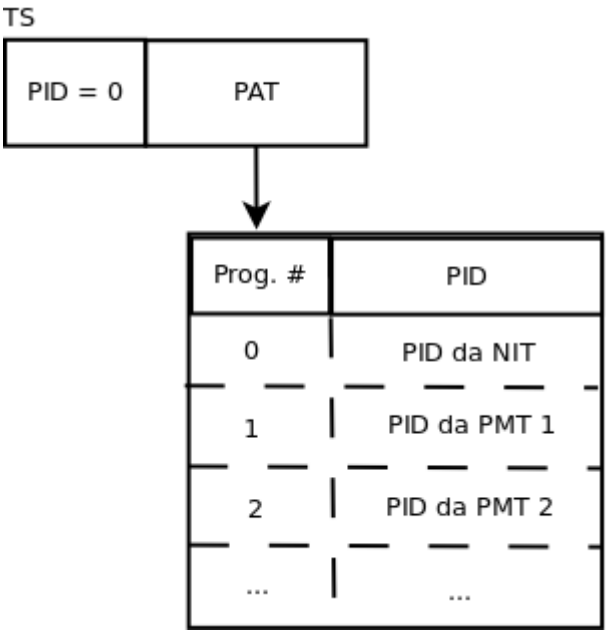

Figura 6 - Estrutura e informações de uma PAT

A norma também determina que o programa ‘0’ da tabela PAT deve sempre ser associado ao PID que identifica o TS que traz um PSI contendo a tabela NIT (*Network Information Table*, ver 4.1.1.2.3).

4.1.1.2.2 PMT

A tabela PMT (*Program Map Table*) é associada a um único programa e está contida no TS que traz o PID correspondente, indicado na tabela PAT. Esta tabela associa ao fluxo elementar de vídeo do programa um PID específico, e apresenta também os PID únicos de cada trilha de áudio (múltiplas línguas) disponível. Na Figura 7 está ilustrada essa estrutura típica de uma PMT.

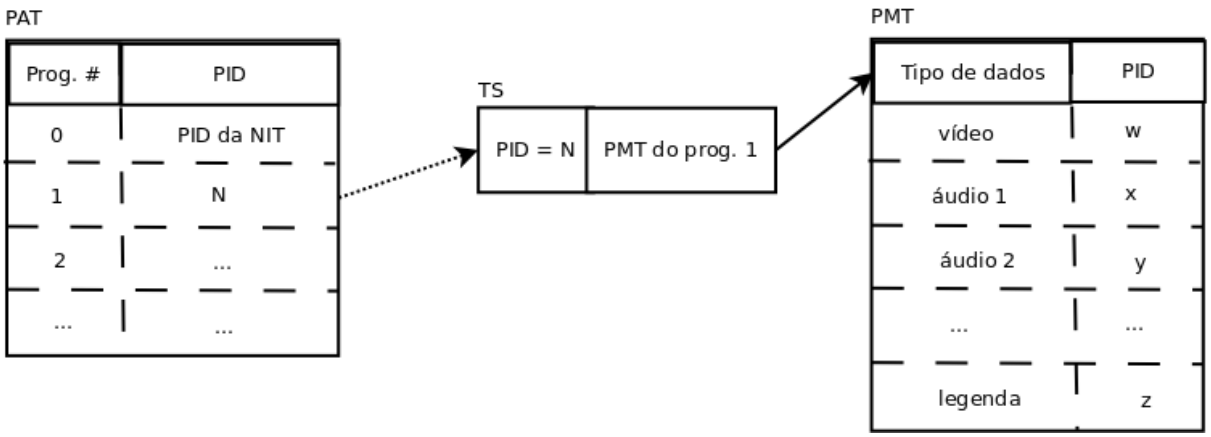

Figura 7 - Estrutura e informações de uma PMT

Portanto, a análise da PMT, associada à interpretação da seleção do usuário, permite que se identifique entre as múltiplas informações recebidas ao longo do tempo aquelas que realmente devem ser decodificadas, repassando-as para o decodificador mais adequado.

4.1.1.2.3 NIT

A tabela NIT (*Network Information Table*) traz parâmetros físicos relativos à rede e à emissora, incluindo a frequência de transmissão. Compila informações sobre o satélite utilizado na transmissão e dados sobre a modulação. Tais informações permitem um ajuste fino da sintonia do TA a fim de evitar ao máximo a perda de pacotes.

4.1.2 *Adaptation Field*

Além dos diferentes tipos de *payload* supracitados, um pacote TS pode trazer pacotes denominados *Adaptation Field* (AF). Um AF, em princípio, não carrega informações úteis para o sistema, servindo basicamente como preenchimento para manter o TS com seu tamanho fixo e para manter a taxa de transferência constante de 2,5MB/s.

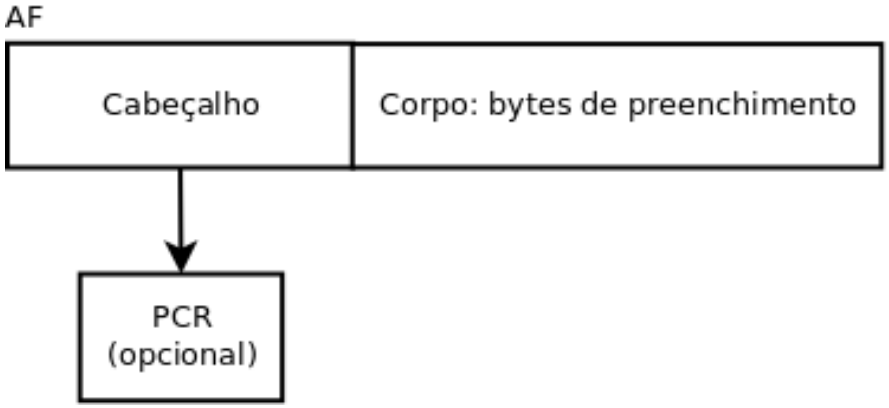

Figura 8 - Estrutura e informações de um AF

No entanto, dentro de certos intervalos de tempo, os quais não podem ultrapassar 100ms, o cabeçalho de um AF deve conter um item chamado PCR (*Program Clock Reference*). Esse campo de 42 bits nada mais é do que um *timestamp*, comparável àqueles que se encontram no cabeçalho de um PES, trazendo amostras de um contador que é

incrementado a cada borda ascendente do relógio do sistema codificador. Essa informação permite, através do VCO do TA, recuperar o relógio original do sistema de codificação e, em conjunto com os *timestamps* presentes no PES, promover a correta sincronização das saídas de áudio e vídeo. O AF especial, contendo PCR, é indicado na tabela PMT como um dos itens associados aos fluxos de um determinado programa.

4.2 TS

O *Transport Stream* (TS), ou Fluxo de Transporte, é um pacote previsto no padrão MPEG-2, o que não impede que seja usado no padrão brasileiro como invólucro para fluxos de áudio e vídeo codificados em MPEG-4. Ele tem toda a sua estrutura e composição formalmente estabelecidas pela norma ISO 13818 – 1 [1]. Trata-se de uma camada robusta, com diversas redundâncias e verificações e, portanto, adaptada para um meio bastante propício a perdas, como é o caso. Em oposição, o próprio padrão MPEG-2 prevê uma outra estrutura, chamada *Program Stream* (PS), que é mais simples e mais indicada para ambientes imunes a perdas, como o DVD, mas cuja estrutura foge ao escopo deste trabalho. Na Figura 9 se observa a composição básica de um TS.

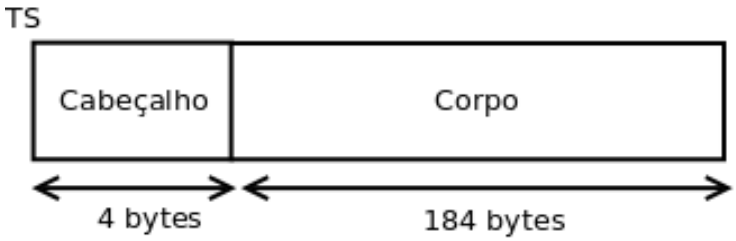

Figura 9 - Estrutura básica de um pacote TS

Sua estrutura apresenta, assim como o PES, um cabeçalho de informações relativas a seu conteúdo e um corpo com os dados úteis, a informação transportada. Ao contrário do PES, o qual tem tamanho variável, o TS é um pacote com tamanho fixo em 188 bytes,

independente de seu conteúdo. Isso explica a eventual necessidade de AF para preenchimento. A seguir são descritas em detalhes as informações que podem ser obtidas de cada seção.

4.2.1 Cabeçalho

O cabeçalho do TS é formado invariavelmente por 4 bytes. O primeiro deles é sempre um byte de sincronismo valendo 0x47 que permite identificar o início de um novo pacote. Um bit do cabeçalho, definido pelo demodulador que precede o demultiplexador dentro do TA permite identificar erros na demodulação do pacote, o que implica o descarte do mesmo.

Um par de bits indica a presença de *payload* no corpo do pacote, ou de AF, ou de ambos. A ausência de ambos é um estado proibido, mas os dois podem perfeitamente coexistir dentro de um único TS. Essa informação permite direcionar o processamento do pacote de acordo com as informações que se deseja extrair, quer seja do AF ou de um *payload*.

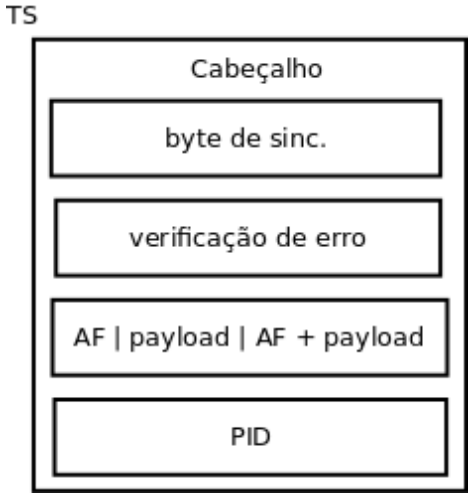

Figura 10 – Informações principais do cabeçalho de um TS

Por fim, a informação mais importante contida no cabeçalho é um campo de 13 bits correspondendo ao PID descrito anteriormente. Como já mencionado, ele permite identificar

diretamente a natureza do conteúdo do TS, quer seja um AF de interesse contendo um PCR, uma tabela PSI de referência ou um certo ES associado a um determinado programa, de acordo com as informações das tabelas PSI. A Figura 10 ilustra essas características.

4.2.2 Corpo

De acordo com as indicações do cabeçalho, os 184 bytes restantes do TS podem conter um AF, um *payload* (PES ou PSI), ou ambos em seqüência, de acordo com a necessidade.

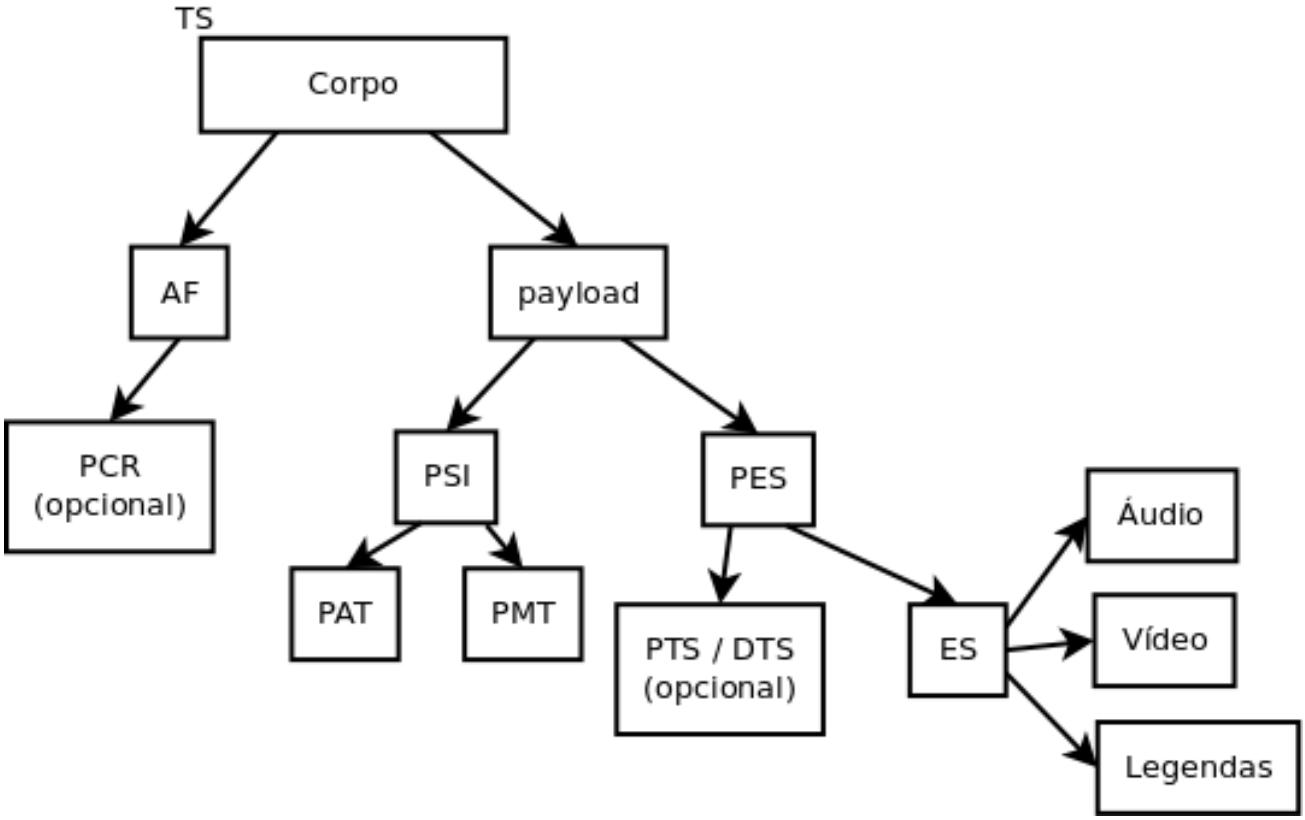

Figura 11 - Árvore de informações do corpo de um TS

Um AF, em princípio, é introduzido apenas para completar espaço deixado por um PES, já que este tem tamanho variável e não vai obrigatoriamente ocupar todo o corpo do TS.

Assim, via de regra um AF pode ser ignorado sem problemas. A única exceção acontece quando o PID do cabeçalho é identificado como o de PCR do programa selecionado pelo usuário, de acordo com a tabela PMT. Nesse caso, o conteúdo do AF precisa ser processado a fim de se extrair de seu cabeçalho a referência de relógio necessária para posterior sincronização das saídas.

No caso de um PSI, é preciso identificar o tipo de tabela e armazená-la de acordo com a necessidade. As essenciais, PAT e PMT, precisam ser registradas para consulta às listas de associação de PID a cada novo TS analisado. A extração de fluxos elementares depende integralmente da correta interpretação, antes de tudo, das tabelas, pois é a única maneira de identificar os conteúdos.

Quando o conteúdo do *payload* é um PES, este deve ser processado a fim de remover o cabeçalho e, a partir de suas informações, encaminhar o fluxo elementar para o decodificador correto. Todas essas possibilidades de informações contidas no corpo de um TS podem ser visualizadas na Figura 11.

4.3 O Demultiplexador

O módulo demultiplexador recebe como entradas o TS, byte a byte através de uma entrada serial, e um sinal de validação da entrada, além do relógio e do reset do sistema. Essa interface de entrada é conhecida como DVB-SPI, e está ilustrada na Figura 12.

Do TS, a partir das descrições vistas anteriormente, é possível extrair, entre outras, as seguintes informações principais:

- Fluxos elementares de áudio, vídeo e dados;
- Tabelas PAT e PMT, para análise da associação de PID;
- Outras tabelas da rede, contendo informações físicas, grade de programação, etc.;

- Timestamps relativos ao tempo de decodificação e apresentação de cada ES;
- PCR, contador para recuperação do relógio do sistema.

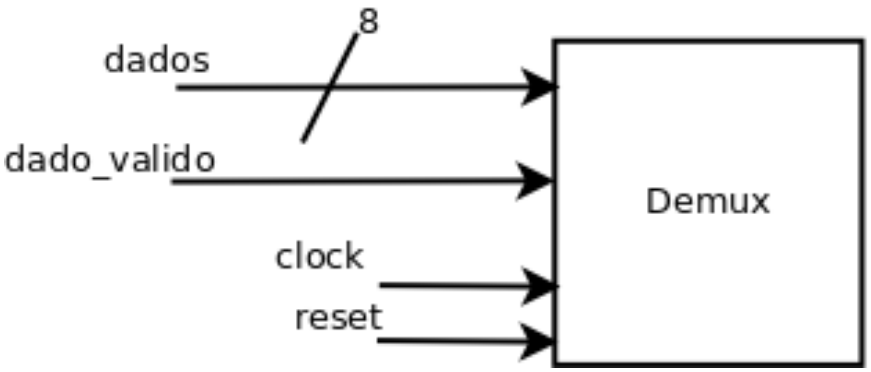

Figura 12 - Entradas do módulo demultiplexador

O demultiplexador precisa, portanto, de uma informação inicial do usuário como entrada, para encontrar os fluxos por ele selecionados. Essa informação seria o número do programa a que o usuário deseja assistir. A partir da tabela PAT descobre-se o PID do TS contendo a tabela PMT daquele programa. A tabela PMT permite, então, identificar os PIDs associados ao fluxo de vídeo do referido programa. Também deve ser informada ao demultiplexador qual, entre as diversas trilhas de áudio disponíveis para o programa, foi selecionada pelo usuário.

Assim, de posse dessas informações o demultiplexador deve abrir o pacote TS separando seu cabeçalho do corpo. Pelo PID do TS ele decide se o conteúdo deve ser repassado para algum dos decodificadores. Deve também armazenar as tabelas pertinentes. Deve, por fim, identificar a porção eventualmente ocupada pelo AF para descarte, a menos que se tenha interesse no sinal PCR que ele pode conter.

4.3.1 Versão Anterior

No semestre 2010/2 o colega Fabrício Caetano, em seu projeto de diplomação [11] precisou de um módulo demultiplexador para validação de seu trabalho, centrado no desenvolvimento do processador de legendas. Foi então elaborada uma versão simplificada, apenas para extrair legendas, cuja interface é mostrada na Figura 11.

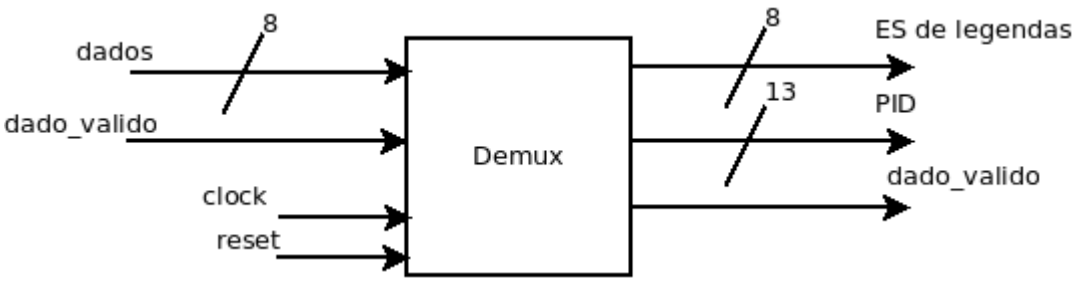

Figura 13 - Interface da versão anterior de demux

Para preparar as entradas desse módulo, foi escrito um código em VHDL com um máquina de estados que extrai vídeo e legendas de um TS. Para tanto, foi realizada uma captura de TS de uma certa emissora, o arquivo foi analisado e dele se identificou o PID que correspondia aos ES de interesse.

Assim, todas as informações das tabelas PSI se tornavam dispensáveis, bastando a análise do cabeçalho do TS para saber, através do PID, se aquele fluxo deveria ser repassado ou não para o processador de legendas. Ou seja, a interpretação de PID era estática, limitada a valores conhecidos de um TS previamente capturado, algo que não pode ser utilizado em um TA com recepção e processamento em tempo real, para múltiplas emissoras. Também os AF são descartados diretamente. O fluxograma implementado é ilustrado na Figura 2.

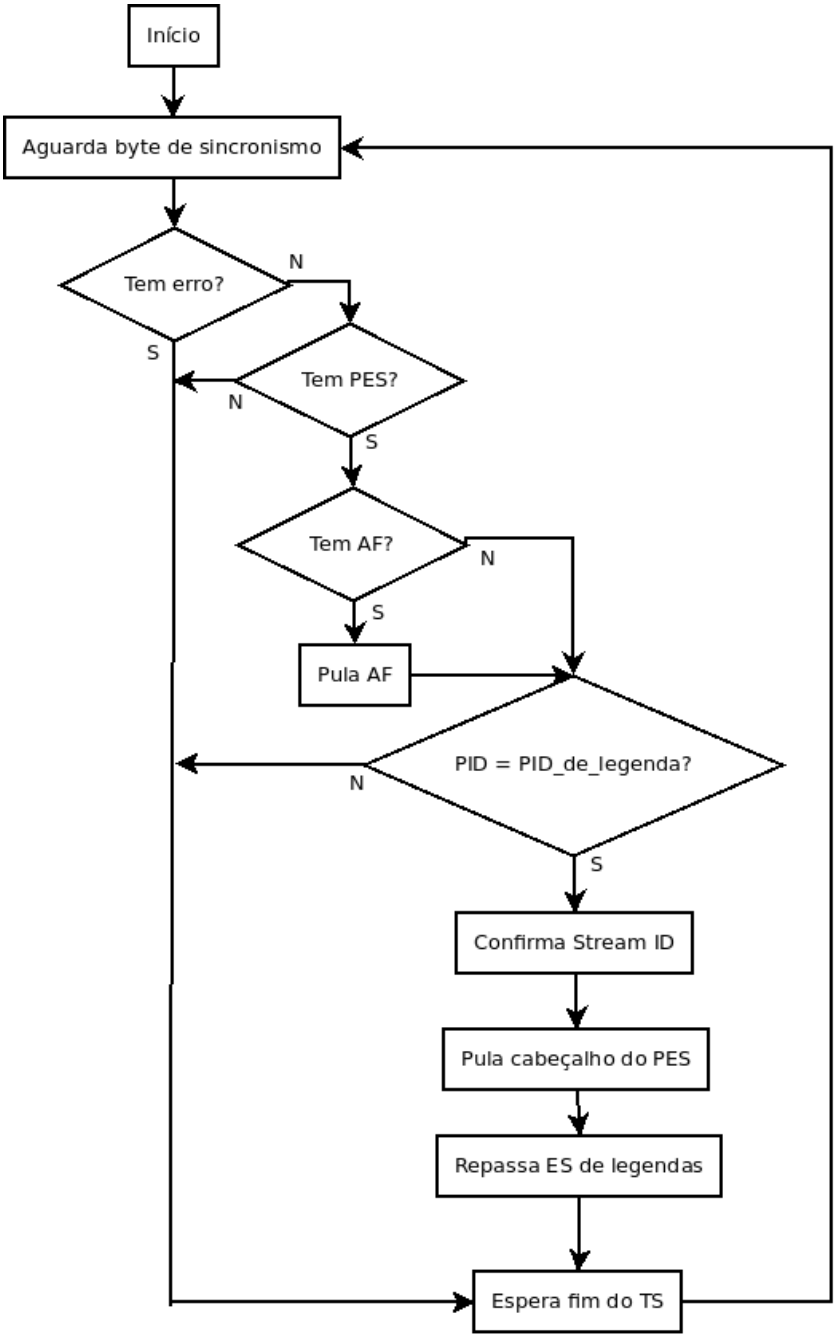

Figura 14 - Fluxograma da versão anterior do demultiplexador

4.3.2 Arquitetura do Demultiplexador Proposta Neste Trabalho

Este trabalho propõe uma versão melhorada de demultiplexador, compreendendo toda a gestão e análise das tabelas PSI principais. Tal operação permite a demultiplexação de

qualquer TS capturado, fazendo uma gestão dinâmica de PID para seleção dos fluxos elementares de acordo com a escolha do usuário, permitindo o funcionamento do TA em tempo real, em condições reais, ou seja, para qualquer emissora em qualquer região do país.

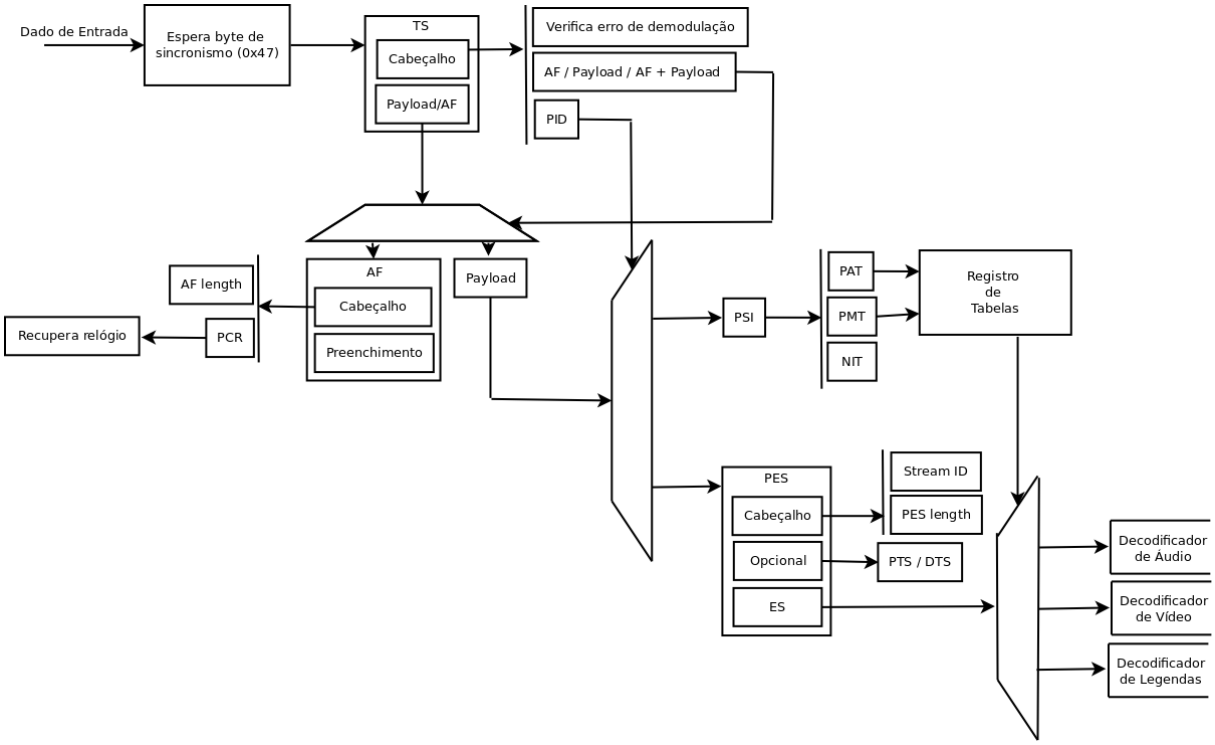

Figura 15 - Fluxo de dados e informações da operação do demultiplexador

As demais tabelas, que não interessam diretamente ao demultiplexador, mas que futuramente serão importantes para interatividade na interface gráfica com o usuário, serão disponibilizadas para o processador central, podendo ser consultadas ou armazenadas conforme são recebidas. Por fim, na questão da sincronização serão lançadas as bases que permitirão um estudo futuro, a partir da extração de *timestamps* e do sinal PCR, os quais estarão disponíveis e poderão ser processados para acertar os tempos de exibição de áudio e vídeo.

O diagrama de blocos da Figura 15 ilustra o caminho completo e todas as possibilidades do que se deseja obter com o módulo demultiplexador completo.

4.4 Resumo de Objetivos

Este trabalho deve apresentar, ao final, um módulo funcional capaz de ser utilizado em uma demonstração da primeira versão do sistema completo do TA. Esse módulo deve estar escrito em linguagem de descrição de hardware VHDL. Os objetivos são os seguintes:

- extração de todas as mídias (áudio, vídeo, legendas) do TS de uma emissora específica, com análise estática de PID;
- integração aos decodificadores.

Em um segundo momento, a prioridade será dada aos seguintes itens:

- gestão das tabelas principais (PAT e PMT), com análise dinâmica do PID;
- disponibilizar para o sistema de processamento central o conteúdo das demais tabelas, para futura utilização.

Por fim, seriam igualmente desejáveis os seguintes itens:

- disponibilizar *timestamps* dos PES para futura sincronização;
- extrair o sinal PCR para recuperação do relógio original do sistema;
- teste e validação.

4.5 Metodologia de Desenvolvimento

Para o desenvolvimento do módulo em VHDL, a tática escolhida envolve a elaboração de códigos e testes em linguagem de programação C, em um primeiro momento. Tal escolha

se deve à relativa facilidade tanto de programação quanto de depuração que essa consagrada plataforma oferece.

A ideia consiste em se valer da flexibilidade da linguagem C para comprovar a presença de todos os elementos previstos, provar o conceito e validar o algoritmo de extração de dados de um TS. Em seguida, após validação através de confirmação da natureza dos dados de áudio, vídeo e legendas recuperados, a mesma máquina de estados deve ser implementada em VHDL para integração ao projeto completo, dentro da cadeia que inclui o processador central, que fornece ao demultiplexador os pacotes TS de entrada, e os decodificadores à jusante, os quais recebem e processam os fluxos elementares oriundos do demultiplexador. O software de referência cumpre, portanto, as importantes tarefas de auxiliar na compreensão do sistema a ser projetado, gerar os casos de teste e contribuir para validação do hardware descrito posteriormente.

4.6 Trabalho Realizado

O trabalho realizado começou pelo estudo da norma e pelo aperfeiçoamento da versão anterior do demultiplexador, para que esta compreendesse e extraísse todos os fluxos elementares, ainda que de maneira estática (PID fixo, previamente identificado).

O segundo momento foi de elaboração do software de referência, o qual é descrito no item 4.6.1. Na sequência o software, validado, teve seu algoritmo adaptado para uma descrição VHDL, guardando a mesma modularidade de operações do software. Essa arquitetura em hardware é o tema do item 4.6.2.

4.6.1 Estrutura do software

O software foi desenvolvido com uma rotina central que cumpre as tarefas de sincronização e extração do cabeçalho do TS. Este é interpretado a fim de identificar o conteúdo do pacote.

Inicialmente é procurado o PID referente à tabela PAT, por ser o ponto de entrada de todo o processamento. Uma vez encontrada e armazenada a PAT, são aguardadas as tabelas PMTs referidas na PAT. Somente quando estas últimas são recebidas e registradas são liberadas as rotinas de processamento de fluxos elementares. Desse momento em diante, todo novo PID é comparado com as referências das tabelas e com a seleção do usuário (por exemplo, programa 1, primeira entrada da tabela PAT, com áudio 2, segunda trilha de áudio relatada na PMT escolhida).

Se uma nova PAT ou nova PMT é recebida, esta tem seu cabeçalho processado a fim de verificar se se trata de uma nova versão da tabela. Em caso afirmativo, a tabela é sobrescrita com os dados atualizados, caso contrário a sequência do pacote é descartada por se tratar da mesma tabela anteriormente armazenada.

A estrutura, além da rotina principal, contempla rotinas especializadas para o processamento de cada tipo de conteúdo, ou seja, um interpretador de AF (que pode tanto extrair um PCR quanto simplesmente pular bytes de preenchimento), outro de PAT, de PMT, de legendas ou de áudio e vídeo. Apenas estes dois últimos compartilham a mesma rotina, pois seus pacotes têm a mesma estrutura. Para os demais, cada casa é tratado separadamente com uma rotina dedicada.

4.6.2 Algoritmo do software

A seguir são analisados os casos de extração de dados de PAT e PMT, as tabelas principais do sistema, como referência para os procedimentos gerais típicos de análise do TS.

4.6.2.1 PAT

Pela norma, as tabelas PAT seguem o formato indicado na Figura 16, retirada de [10]. No processamento de uma PAT é utilizado um índice que serve como referência de posição durante o processamento (bytes processados em relação ao inicial) e permite, assim, identificar o fim do pacote. Dentro de um TS, essa estrutura da Figura 16 é precedida por um campo chamado *pointer*, o qual apenas indica se há dados irrelevantes entre o início do trecho de *payload* dentro do TS e o início da estrutura referida para a tabela PAT. A análise prática de arquivos de TS tem mostrado que isso nunca é utilizado pelas emissoras, *pointer* nunca indica bytes suplementares na sequência, antes da PAT. Porém não se pode ignorá-lo, e por isso a possibilidade de bytes extras é prevista no algoritmo.

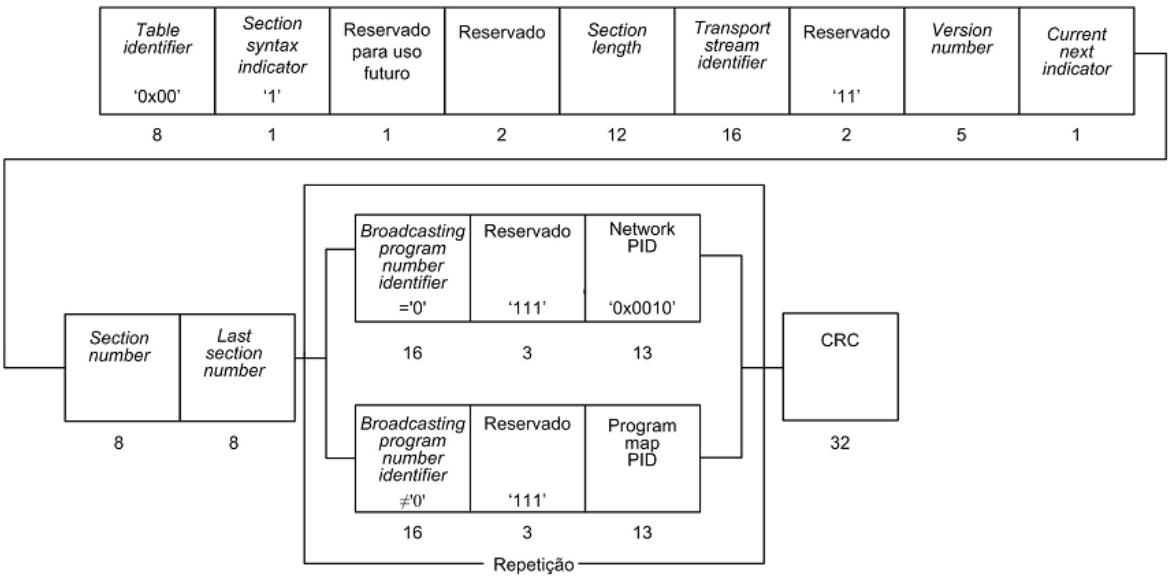

Figura 16 - Estrutura de dados da PAT prevista em norma

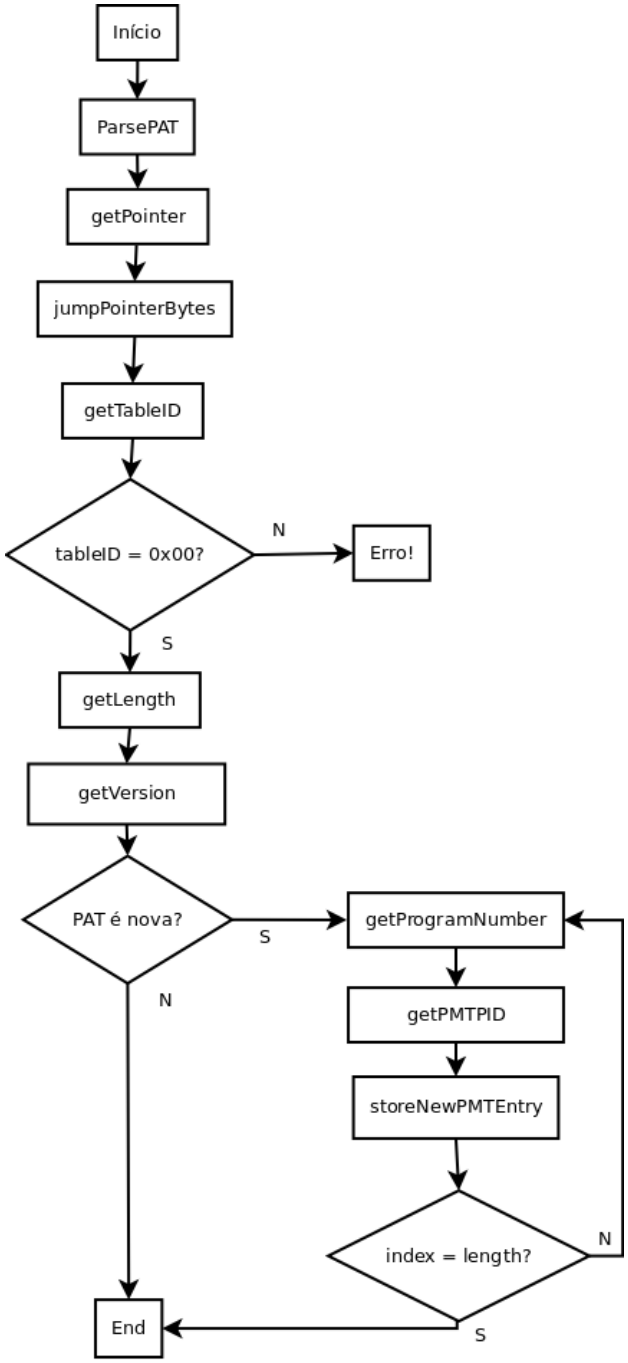

Figura 17 - Fluxograma de processamento de uma PAT

O fluxograma da Figura 17 traz os passos para extração dos dados da PAT. Como se pode observar, após saltar eventuais bytes apontados por pointer se verifica o identificador de tabelas PSI, que no caso de uma PAT, de acordo com a norma, deve ser invariavelmente igual a 0. Em caso de erro, supõe-se que se deve a um erro de dessincronização, provavelmente por perda de dados. Por esse motivo o fato é sinalizado é e o programa retorna ao nível superior

em estado fora-de-sincronismo, para que se busque uma correção do problema. Se no entanto tudo está correto, parte-se para análise dos bytes restantes. Quando se obtém a versão da PAT, esta é comparada à versão da PAT atual para definir se se trata de uma nova PAT que deve se sobrepor à anterior. Se sim, ou se nenhuma PAT havia sido encontrada ainda, parte-se para o processamento dos seus dados, o qual consiste em extrair e armazenar os pares de número de programa e o respectivo PID da PMT correspondente.

4.6.2.2 PMT

O caso das PMTs é semelhante ao das PATs. Seu formato é indicado na Figura 18, também retirada de [10].

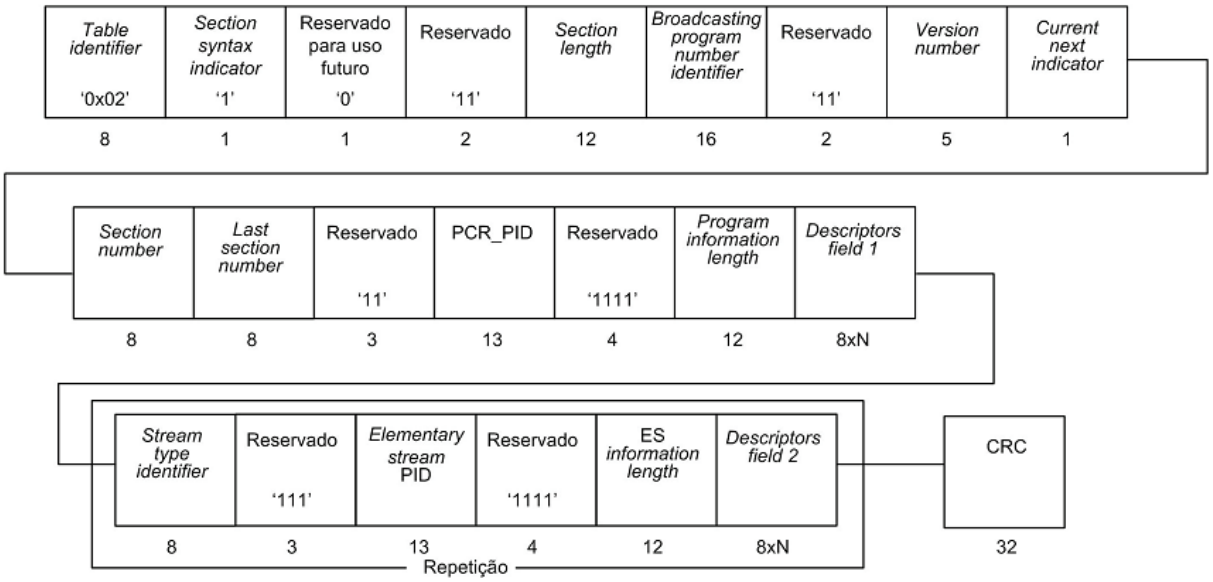

Figura 18 - Estrutura de dados da PMT prevista em norma

O fluxograma da Figura 19 explica os passos do algoritmo de processamento. Por uma questão de simplicidade do fluxograma, são omitidos estágios de processamento de informações de menor relevância.

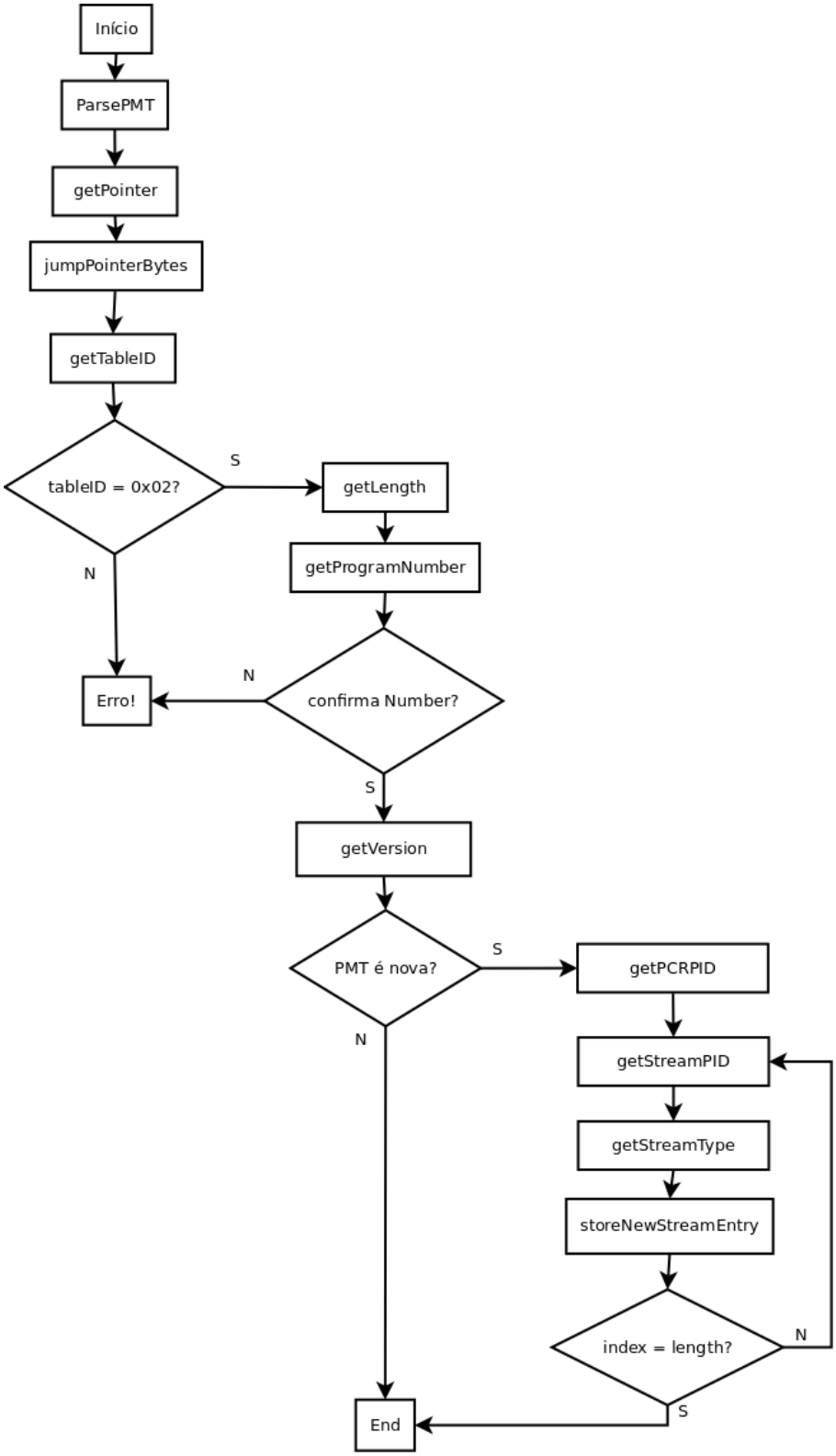

Figura 19 - Algoritmo de processamento de uma PMT

Conforme é possível perceber na Figura 19, há também um campo inicial *pointer* com o número de bytes a serem pulados antes do início da PMT, um identificador de tabela (que

no caso deve obrigatoriamente valer 0x02) e o tamanho da PMT. A seguir é encontrado o número do programa, o qual pode também indicar um estado de erro em caso de incoerência com o previsto na tabela PAT e um número de versão. No caso de se tratar de uma nova PMT, são então registrados o PID do AF contendo o PCR do programa e os pares de associação entre tipos de mídia e seus respectivos PIDs.

Além desses, estilos semelhantes de processamento de pacotes são realizados para PES de áudio e vídeo, de legendas e para campos de adaptação, cada um de acordo com sua estrutura específica prevista em norma.

4.6.2 Descrição de Hardware do demultiplexador em VHDL

Baseando-se no algoritmo já validado em C, foram descritos módulos VHDL para cumprir a mesma sequência de passos para extração de dados do TS com análise dinâmica de tabelas.

Para tanto se adotou uma estrutura semelhante àquela explicada anteriormente para o software, com entidades independentes para processamento de cada tipo de conteúdo presente em um TS, e um módulo central, de controle, para abrir o TS, sincronizar, identificar o cabeçalho e habilitar os demais módulos de acordo com o conteúdo indicado pela interpretação do PID. A Figura 20 apresenta a interface global do módulo demultiplexador desenvolvido, que terá seus pormenores detalhados na sequência. Observa-se que, além do TS, ele recebe sinais de controle por parte do usuário e por parte dos decodificadores de áudio e vídeo, os quais indicam quando suas filas de entrada estão cheias e o fluxo deve ser controlado. Entre suas saídas estão, além dos fluxos elementares dos três tipos, os timestamps extraídos e o PCR do programa selecionado.

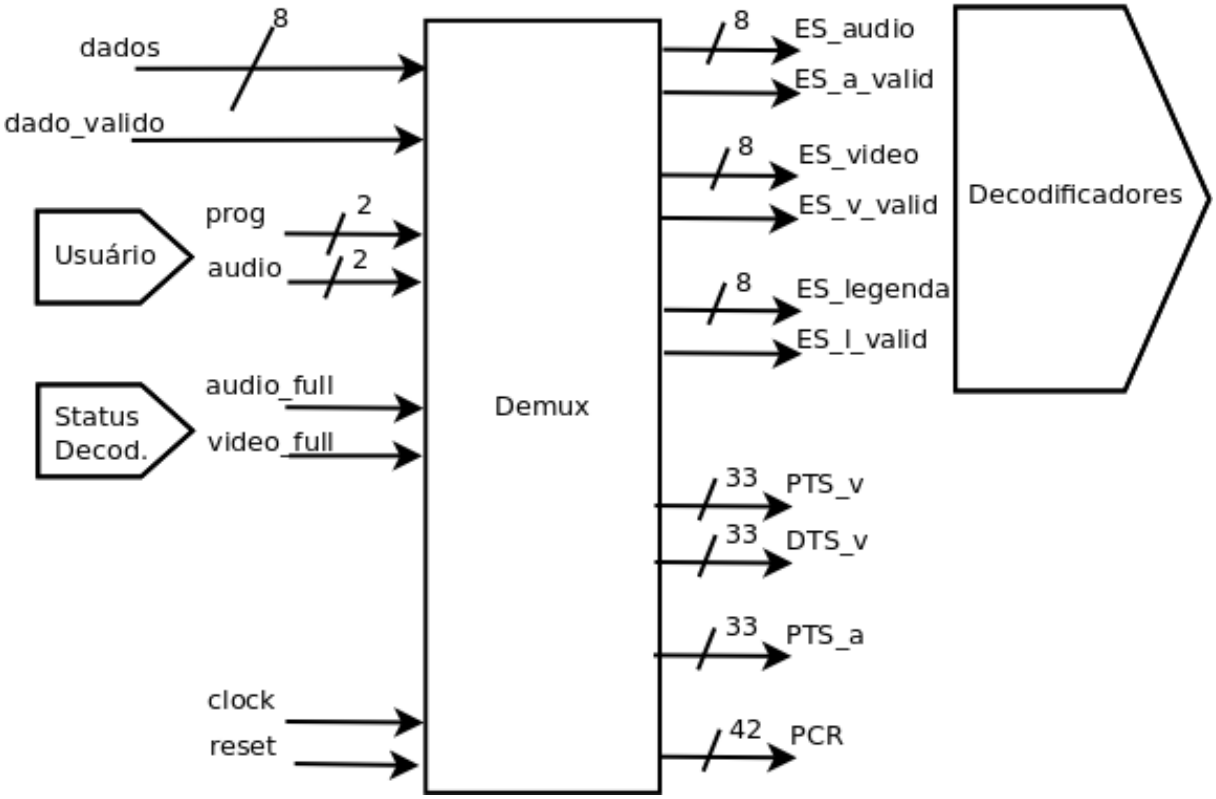

Figura 20 - Interface do demultiplexador desenvolvido neste trabalho

Nas figuras 21 a 25 são mostradas as interfaces de cada um desses módulos citados. Todos esses módulos têm em comum as entradas de dados, relativas ao TS, uma entrada de habilitação definida pelo bloco de controle, saídas de status indicando operação e estado de erro identificado, e entrada clear usada para confirmar que o controle foi informado do erro e poder, então, sair desse estado.

4.6.2.1 Análise de PAT

O módulo de análise de PAT, que pode ser visto na Figura 21 cumpre o mesmo papel da rotina do software vista anteriormente. Além dos sinais de base, ele tem saídas que indicam a versão da PAT analisada, os números dos programas (de cada PMT) e os respectivos PIDs extraídos da tabela. Por simplicidade não foram indicados, mas todos os sinais têm também um sinal de validação que os acompanha, para indicar ao controle a

presença de uma nova informação. Cabe ao controle verificar se a PAT em questão é nova e gerenciar o registro da mesma, ou desabilitar o módulo de análise de PAT se não for necessário. No caso do software em C, a tabela era registrada como *structure*, e, para manter a mesma linha, na solução VHDL as tabelas são guardadas em registros locais (*record* VHDL). Essa solução é viável porque as tabelas recebidas nunca ultrapassam o tamanho de um TS, ou seja, 184 bytes.

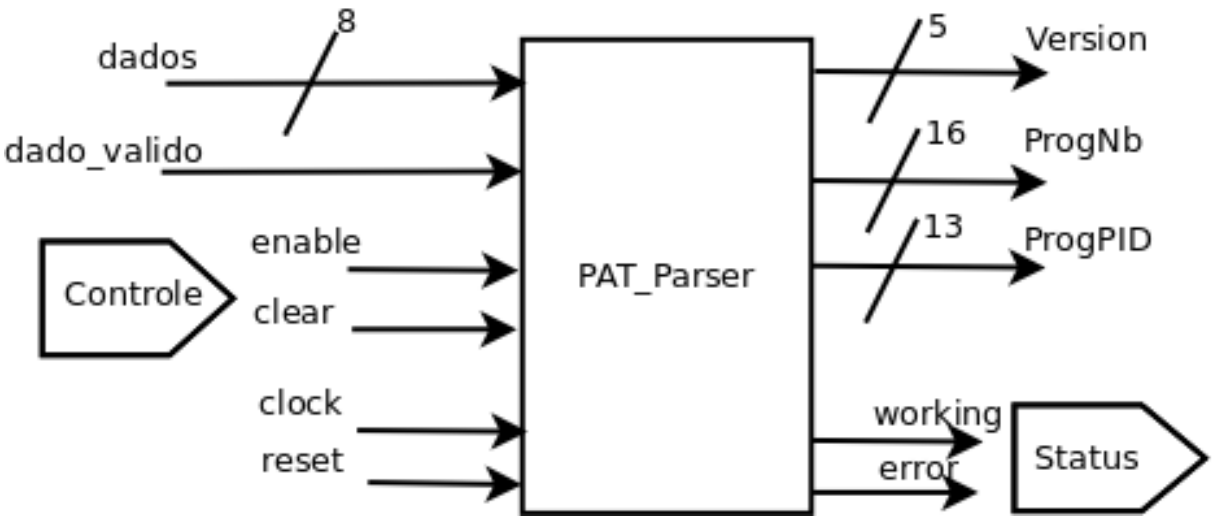

Figura 21 - Módulo de análise de PAT

4.6.2.2 Análise de PMT

O módulo de análise de PMT tem suas entradas e saídas apresentadas na Figura 22. Este também fornece ao controle o número e a versão da PMT sob análise, extrai o PID do PCR do programa e depois fornece sucessivamente o tipo de fluxo e o respectivo PID de cada um de seus elementos, para que o controle realize o registro dessas informações. Aqui também, as validações de cada sinal foram omitidas por simplicidade.

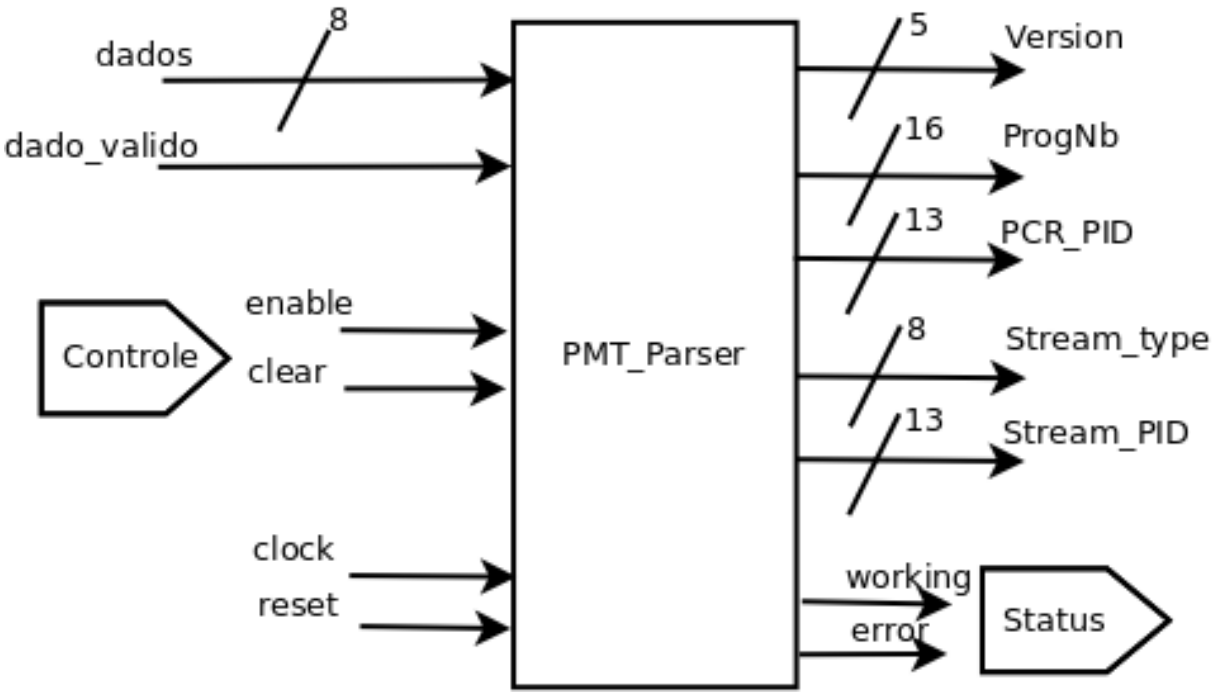

Figura 22 - Módulo de análise de PMT

4.6.2.3 Análise de AF

O módulo de interpretação de AF tem por função básica encontrar o byte incluso no cabeçalho do campo de adaptação que indica o tamanho do mesmo. A partir dessa informação, o módulo é capaz de aguardar o término do pacote de preenchimento. Se, no entanto, lhe for indicado que se trata de um AF contendo o PCR do programa selecionado pelo usuário, então o mesmo módulo se encarrega de extrair o PCR e repassá-lo ao nível superior. A presença de PCR só pode ser identificada pelo módulo de controle, que tem acesso às informações das tabelas PAT e PMT, e por isso o módulo recebe um sinal de entrada solicitando a extração de PCR, como pode ser visto na Figura 23. Além dos sinais básicos, como pode ser observado na Figura 23 o módulo oferece ao controle um sinal de status indicando que o byte atual é o último previsto para o AF. Isso é importante nos casos em que o AF divide o espaço de um TS com um payload que vem na sequência, para que o módulo de controle já faça os devidos testes e prepare a habilitação para o módulo seguinte.

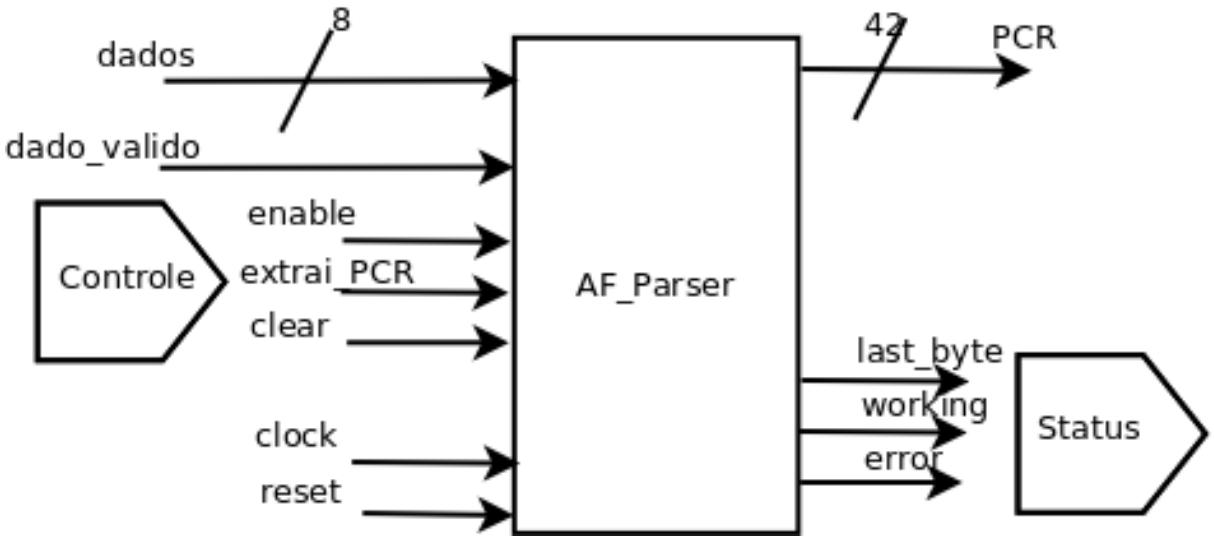

Figura 23 - Módulo de interpretação de Campos de Adaptação

4.6.2.4 Análise de Legendas

A entidade criada para análise de legendas pode ser visualizada na Figura 24. Ele recebe do módulo de controle um sinal que indica se tratar do início de um novo pacote, e fornece na saída o ES puro para o decodificador, acompanhado do seu sinal de validação.

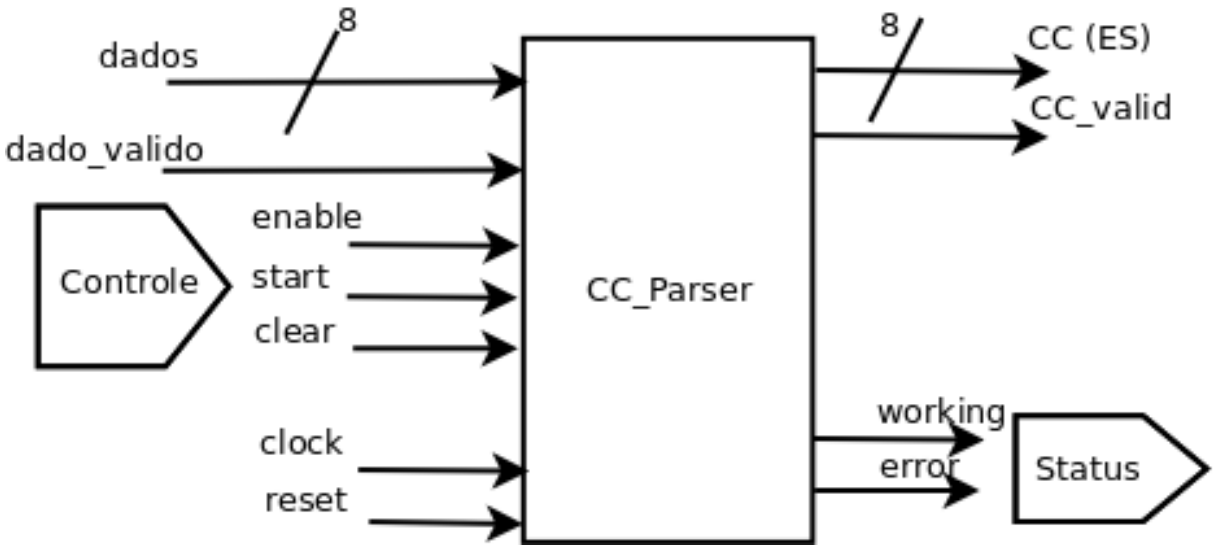

Figura 24 - Módulo de análise de PES de legendas

4.6.2.5 Análise de Áudio/Vídeo

Por terem a mesma estrutura de formação, tanto PES de áudio quanto de vídeo são processados por uma mesma entidade genérica, instanciada duas vezes no projeto. Esse módulo é apresentado na Figura 25. Ele recebe também um sinal de início de pacote, assim como o módulo de análise de legendas, e fornece na saída, além do ES para o respectivo decodificador, os sinais de PTS e DTS (este último presente apenas nos PES de vídeo).

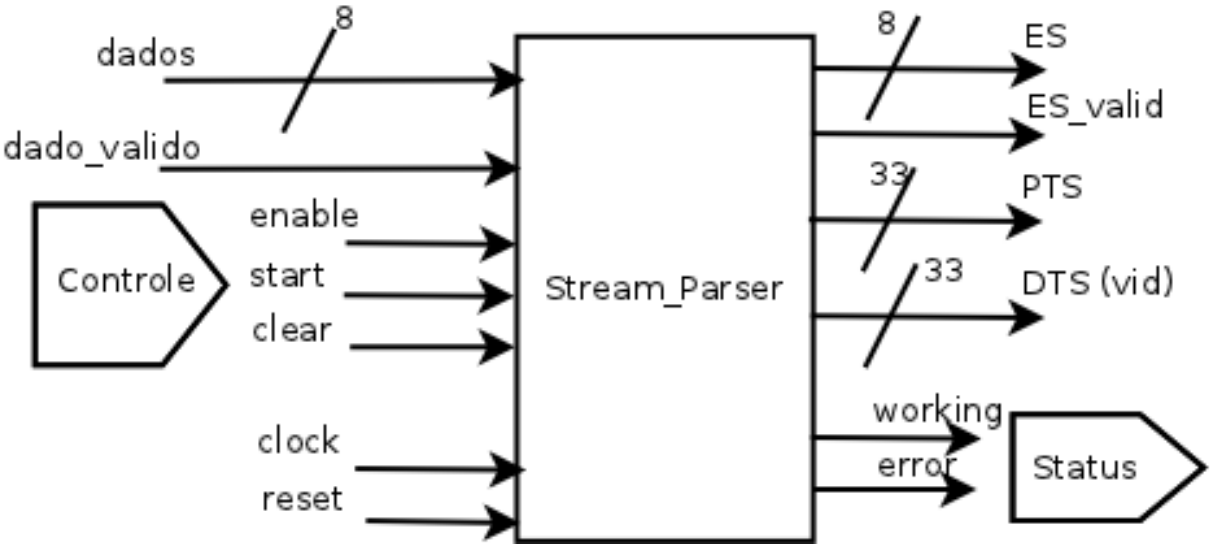

Figura 25 - Módulo de análise de PES de áudio/vídeo

4.6.2.6 Módulo de Controle

O módulo de controle é o cérebro do demultiplexador e faz a gestão de todos os outros módulos operacionais. Como pode ser visto na Figura 26, ele recebe como entrada os sinais de status e as saídas de cada módulo, e a partir disso ele gera os sinais de controle que habilitam e direcionam o processamento de cada estrutura. É este módulo de controle que armazena as tabelas extraídas, interpreta o cabeçalho de cada novo TS, compara com as tabelas e identifica a natureza de seu conteúdo e encaminha o resto do TS para o módulo mais

adequado. Ele também recebe como entradas as escolhas do usuário para escolher qual dos programas indicado na PAT deve ser repassado para os decodificadores, e também qual, entre as trilhas de áudio disponíveis, deve encaminhada ao módulo de análise de PES de áudio.

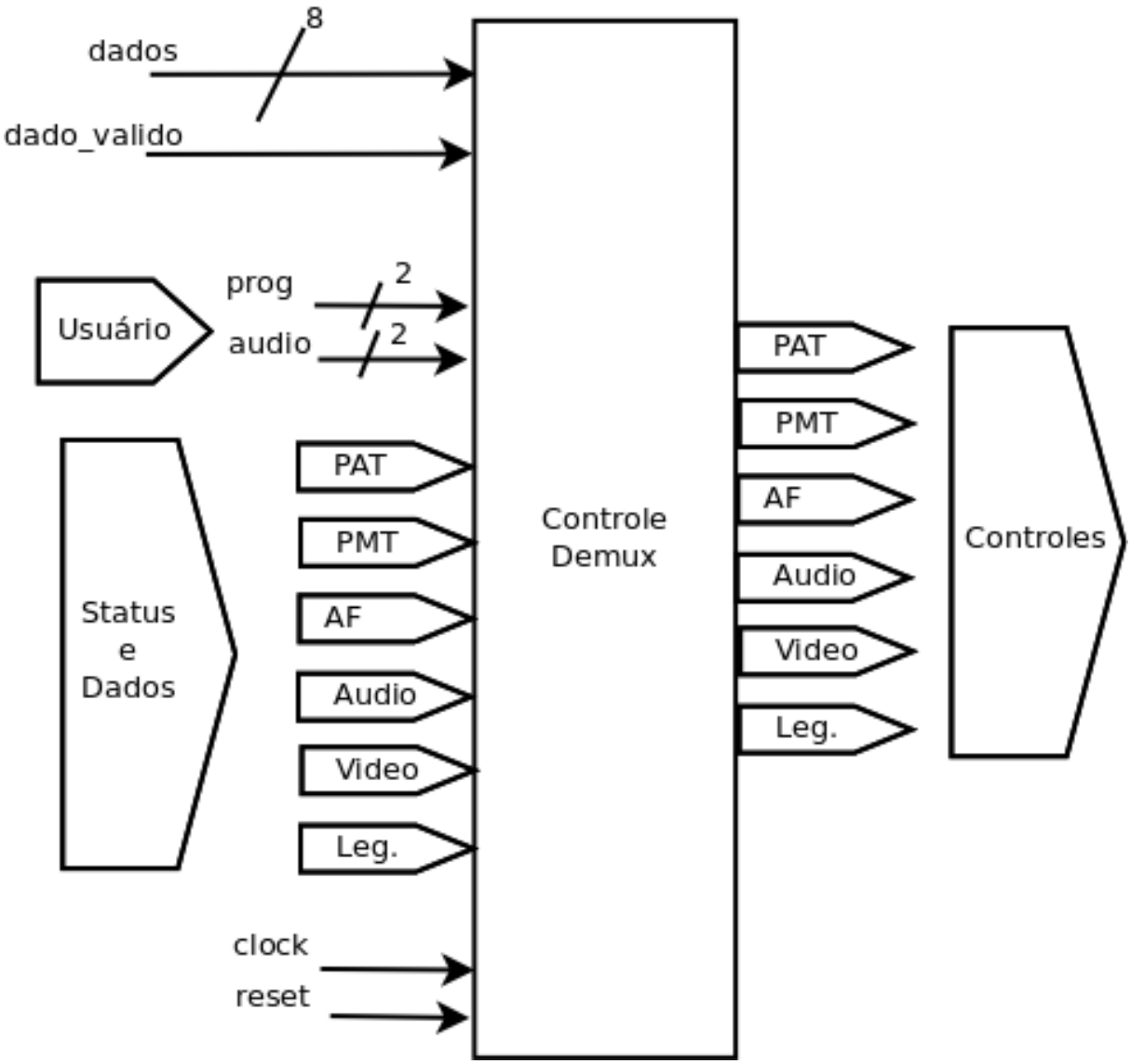

Figura 26 - Módulo de controle do demultiplexador

4.6.2.7 Integração: Módulo de Gestão de Fluxo de Dados para Decodificadores

Para a correta integração aos decodificadores de áudio e vídeo, os quais podem em alguns momentos não absorver todos os dados de ES na taxa em que eles são entregues pelo

demultiplexador, foi previsto um módulo de controle de fluxo de dados. Este módulo tem sua interface apresentada na Figura 27. Trata-se basicamente de uma máquina que controla um *buffer* (fila) e possui dois estados básicos: ou ele repassa diretamente os dados, ou ele os armazena em sua fila interna enquanto o decodificador à jusante indica estar cheio. Dessa forma ele pode aproveitar os momentos em que não há novos dados de entrada para ir repassando aquilo que foi armazenado enquanto o decodificador estava indisponível. Porém, ele também tem um limite de armazenamento, e quando este é ultrapassado ele indica o fato com um *flag* de *overflow*.

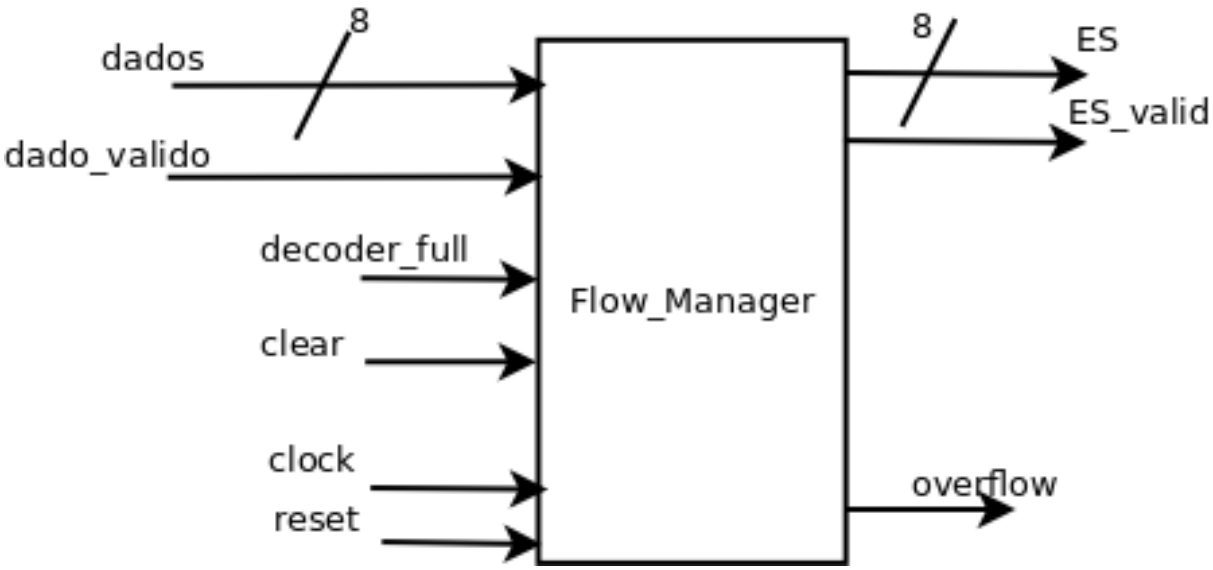

Figura 27 - Módulo de controle de Fluxo

4.6.2.8 Módulo topo do demultiplexador

A interação entre o controle e esses cinco módulos individuais de processamento específico é gerida por um topo que promove as ligações e encaminha os dados de entrada e saída. É esse topo que inclui os gestores de fluxo nas saídas de áudio e vídeo, e realiza a interface indicada anteriormente, com uma saída de dados e uma validação para cada tipo de dado. Além desses sinais, são obtidos os outros sinais previstos, a saber, *timestamps* de

decodificação e de apresentação tanto de áudio quanto de vídeo, e PCR, todos com respectiva validação.

Este sistema completo pode ser visualizado na Figura 28.

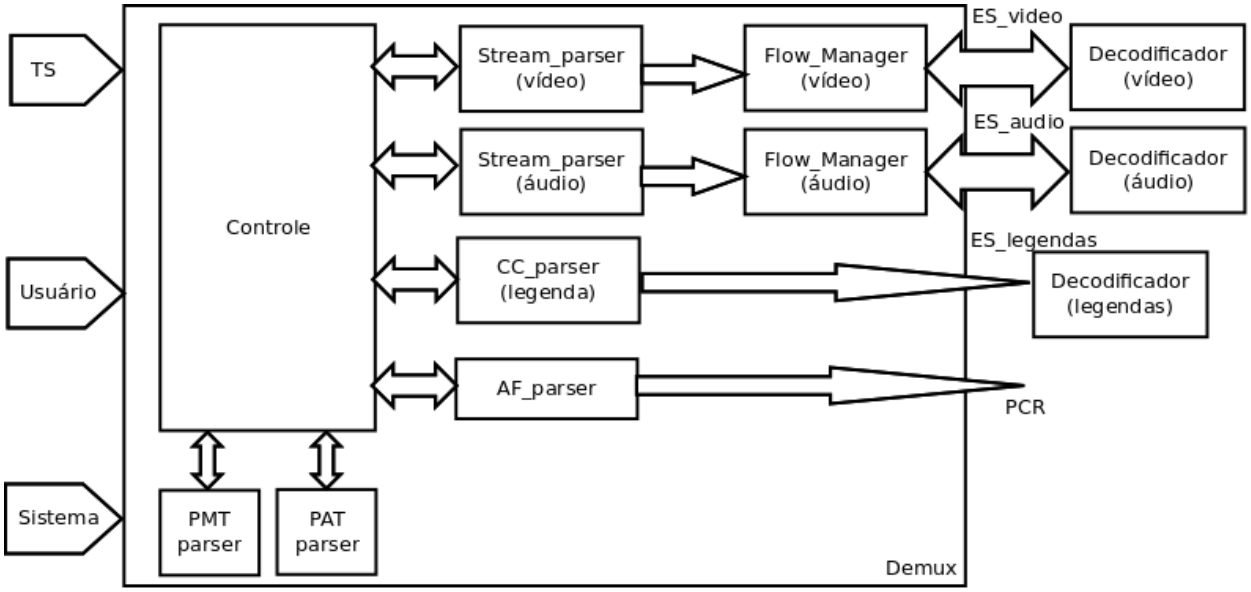

Figura 28 - Demultiplexador completo

5. Resultados alcançados

E um primeiro momento, o módulo anterior, realizado pelo colega Fabrício Caetano, foi estudado, compreendido e expandido, e agora extrai os três fluxos básicos de áudio, vídeo e legendas. A escolha continua sendo estática, de acordo com análise prévia do TS de uma emissora específica. A saída não é mais em termos de PID, mas sim de um seletor que interpreta o PID e indica a cada decodificador quando se trata do seu respectivo fluxo elementar.

Essa versão do demultiplexador foi acoplada aos decodificadores de legendas e de vídeo e teve seu funcionamento comprovado na prática.

Foram feitas capturas de novos *bitstreams* para análise de seu conteúdo. Esse estudo, realizado com o auxílio do software gratuito *TSReader [12]*, evidenciou a presença e a frequência das tabelas PAT, PMT e NIT. Também ficou clara a esperada predominância de fluxos elementares de vídeo *full-seg*, ocupando cerca de 95% dos TS capturados.

5.1 Software de Referência

Foi integralmente escrito em linguagem C o software previsto inicialmente, perfazendo um total de mais 1300 linhas de código. Ele lê arquivos de TS previamente capturados com o auxílio de um *penTV* USB e do software PRH.264 desenvolvido no LaPSI para esse fim.

A partir do arquivo de TS, sem informações prévias de PID, ele é capaz de identificar as tabelas do sistema, extrair e guardar suas informações para depois, de acordo com uma

seleção de usuário simulada em software, salvar arquivos de áudio, vídeo e legendas em arquivos separados.

A depuração do código, abrindo o arquivo hexadecimal do TS e acompanhando byte a byte a evolução da máquina de estados, além de ajudar na visualização e compreensão exata de todos os detalhes previstos em norma, permitiu ter a certeza de que o software interpreta corretamente as informações das tabelas recebidas no corpo do TS, é capaz de extrair os PCRs que chegam periodicamente e, principalmente, registra em arquivo apenas os fluxos elementares selecionados pelo usuário que, na prática, serão repassados para os decodificadores. A execução dos arquivos de áudio assim obtidos, no software VLC [13] evidencia o bom funcionamento do software e valida a sequência de passos do algoritmo proposto.

5.2 Hardware

Foi escrito em linguagem VHDL o demultiplexador proposto, com todos os seus módulos operacionais para processamento individualizado de cada tipo de informação que pode ser encontrada em um TS do SBTVD. Aos módulos operacionais se somam o módulo de controle e os gestores de fluxo, os quais, em conjunto, chegam a um total de mais de 3000 linhas escritas. Esse conjunto compõe o demultiplexador completo, conforme visto na Figura 28, o qual é capaz de demultiplexar TS de maneira dinâmica, levando em conta os dados que são enviados pela emissora na forma de tabelas e permite a identificação, em tempo real, da natureza de todos os dados recebidos.

5.3 Testes e Validação

5.3.1 Simulação

O módulo VHDL completo foi isoladamente testado em simulação com um *testbench* especialmente escrito para gerar todos os estímulos previstos. Nele foram simulados dados intermitentes de entrada (TS) e indisposição dos decodificadores por até 20 ciclos consecutivos, e em todos os casos o demultiplexador manteve o funcionamento esperado, sem perda de dados. Os dados de entrada foram capturados de uma transmissão real da emissora Record, canal 4 de Porto Alegre. Para tais dados, foi possível observar que o demultiplexador cumpria todos os passos necessários, a saber: sincronização (encontrar o início de um novo pacote), aguardar até a chegada da primeira PAT, aguardar as PMTs indicadas na PAT, identificar a PMT do programa selecionado pelo usuário e, nela, identificar o áudio selecionado, e repassar para as saídas os fluxos elementares de cada tipo. Em paralelo observou-se que eram identificados e extraídos tanto o PCR quanto os timestamps de áudio e vídeo, disponibilizados na saída. Tudo foi conferido byte a byte de acordo com a leitura do TS em sua forma hexadecimal e por comparação com os resultados do software de referência para a mesma entrada.

A Figura 29 mostra um *screenshot* do momento em que uma PAT é identificada (PID = 0x000). Isso pode ser acompanhado pela máquina de controle, que passa para o estado "PAT", e pelo sinal de habilitação do módulo de análise da PAT, que é ativado logo abaixo.

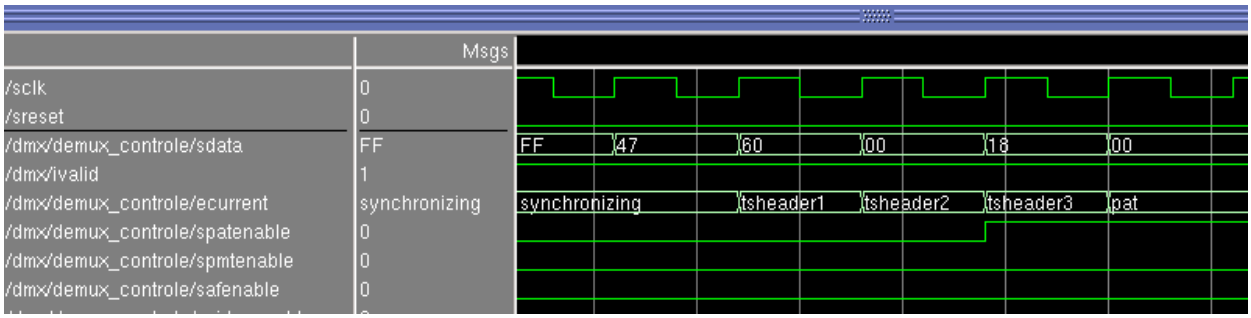

Figura 29 - Identificação de uma tabela PAT em simulação

Na Figura 30 é mostrada a correta interpretação da tabela PAT, pois o módulo de controle identifica a chegada de uma tabela PMT, cujo PID (no caso, 0x0101) havia sido anunciado na PAT que a precedeu. Observa-se que, apos a análise do cabeçalho do TS, a máquina de controle passa para o estado “pmt1” e o módulo de análise de PMT tem seu sinal de habilitação ativado. Este resultado evidencia que o módulo anterior, de análise de PAT, funcionou corretamente e teve suas inforções coletadas, registradas e consultadas como previsto e, portanto, valida a metodologia aplicada neste trabalho para extração e processamento de informações da maneira como elas são tipicamente recebidas em um TS.

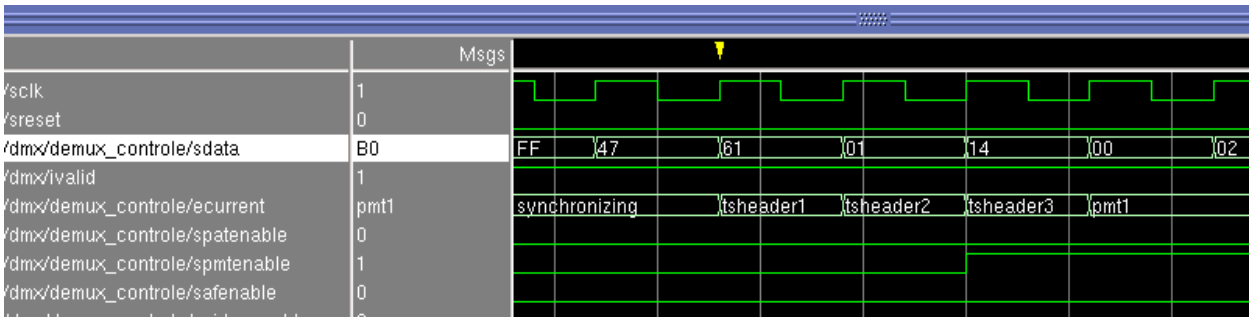

Figura 30 - Identificação de uma tabela PMT em simulação

5.3.2 Síntese em placa

Este trabalho culminou com a síntese em placa da solução em hardware desenvolvida. O módulo foi colocado em uma placa Xilinx XUPV5m mostrada na Figura 31, e teve sua integração com o decodificador de legendas testada. A entrada de dados foi realizada a partir dos mesmos dados recolhidos anteriormente e testados em simulação, sendo que para o teste em placa eles foram colocados em um gerador de padrões (memória ROM) criado especialmente para esta finalidade. A validação se deu, na prática, pela observação das legendas esperadas na saída de vídeo. Essa observação comprova o funcionamento da solução

proposta neste trabalho e mostra que a gestão e interpretação dinâmica das tabelas está sendo feita corretamente.

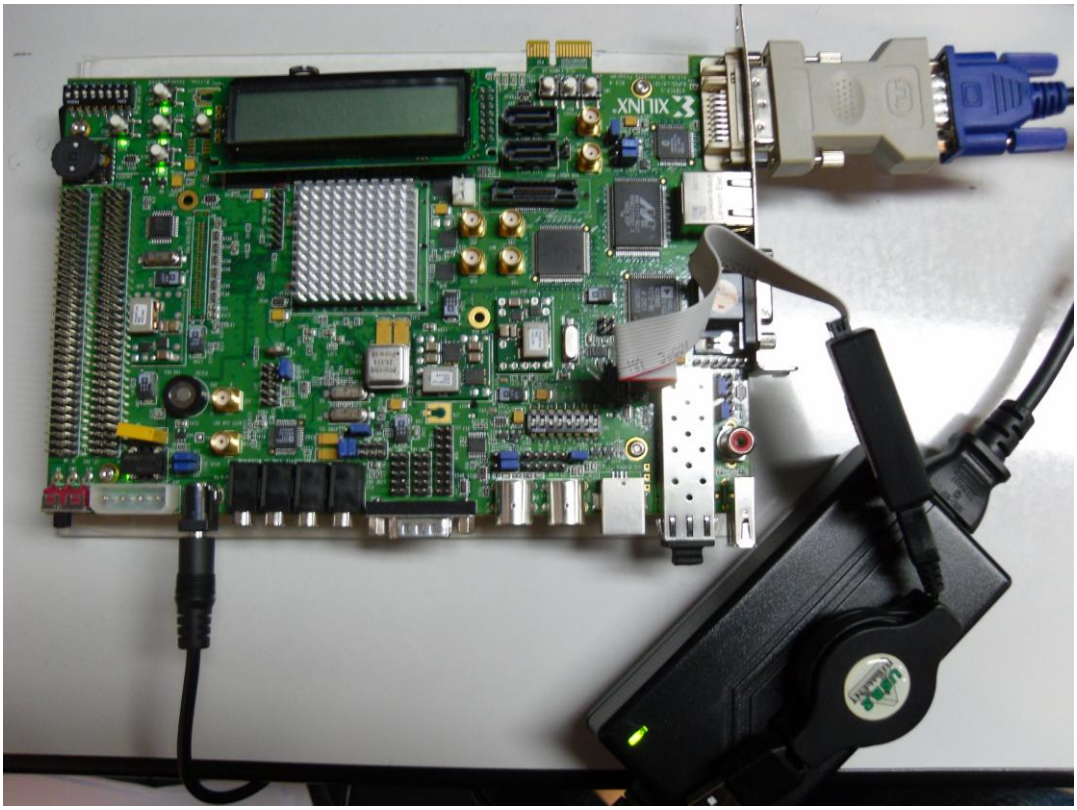

Figura 31 – Ferramenta de trabalho: Placa de prototipação Virtex 5

Ficou comprovada também a viabilidade da solução propostam pois pelo relatório de síntese se obteve os seguintes parâmetros: frequência máxima de operação de 165,865MHz, bastante acima tanto da frequência de chegada de um TS (2,5MHz), quanto da frequência de operação do barramento do processador (atualmente 60MHz, mas está previsto aumento para 80MHz). Em termos de utilização, o sistema de teste completom incluindo demultiplexadorm decodificador de legendas e saída de vídeo utilizou apenas 2% (1576) dos registradores, 4% (2891) das LUTs e 1% (2) de BlockRam/Fifo da placa.

6. Conclusão

Este trabalho apresentou as etapas percorridas durante a elaboração de um módulo demultiplexador de TS para televisão digital, segundo o padrão brasileiro. Após uma revisão técnica dos conceitos envolvidos, em particular daquilo que constitui a estrutura de um TS, foi explicada a função e a estrutura de um demultiplexador típico.

Uma proposta de implementação de um demultiplexador completo foi apresentada. Ela traz avanços na questão da gestão de tabelas PSI, permitindo assim a interpretação dinâmica de parâmetros PID, os quais identificam o conteúdo dos pacotes. Dessa forma, o módulo proposto, além de ser funcional, aproxima-se consistentemente de uma versão final para um demultiplexador completo, faltando para isso a gestão dos parâmetros de sincronização das saídas do sistema que também são recebidas através dos pacotes TS e PES.

Foi desenvolvido algoritmo do software de referência escrito em linguagem C, e este foi utilizado para a elaboração de uma descrição de hardware completa em VHDL. Deste trabalho se obteve um demultiplexador completo e funcional, pronto para ser utilizado dentro de um Terminal de Acesso em qualquer região do país e para qualquer emissora que siga o padrão estabelecido em norma para transmissão de sinal de televisão digital. O demultiplexador projetado não compromete a colocação do resto do TA em placa, pois se verifica que sua frequência mínima de operação é adequada, de 165,865MHz.

Por fim, o alvissareiro investimento nisto que se constitui em um grande esforço para crescimento tecnológico nacional é um elemento que merece destaque neste projeto. A busca pela inovação e pela formação de mão-de-obra qualificada no Brasil é uma iniciativa que precisa ser louvada para que não se torne um fato isolado, mas que ao contrário receba incentivo e continuidade.

7. Referências Bibliográficas

[1] ITU-T H.222.0 / ISO/IEC 13818-1: 2000 – "Information technology – Generic Coding of moving pictures and associated audio information: Systems."

[2] AMBA Specification – Rev 2.0 (ARM IHI 0011A), diponível em http://infocenter.arm.com/help/topic/com.arm.doc.ihi0011a/index.htm (03/07/2011)

[3] Laboratório de Processamento de Sinais e Imagens. Relatório Entregável 2 – Projeto SoC – SBTVD

[4] Aeroflex Gaisler/ GRLIB: "http://www.gaisler.com/cms/" (03/07/2011)

[5] ISO/IEC 14496-3:2001 – "Audio Lossless Coding"

[6] ABNT NBR 15602-2:2007 – "Digital terrestrial television - Video coding, audio coding and multiplexing - Part 2: Audio coding"

[7] ITU-T H.264 / ISO/IEC 14496-10 – "MPEG-4 Part 10: Advanced Video Coding."

[8] ABNT NBR 15602-1:2007 – "Digital terrestrial television - Video coding, audio coding and multiplexing - Part 1: Video coding."

[9] ABNT NBR 15606-1:2007 – "Digital terrestrial television - Data coding and transmission specification - Part 1: Data coding"

[10] ABNT NBR 15603-1:2007 – "Digital terrestrial television - Multiplexing and service information (SI) - Part 1 SI for digital broadcasting systems"

[11] F. Caetano, "Processador Gráfico", dez. 2010; http://hdl.handle.net/10183/27976

[12] "TSReader Transport Stream Analyzer": www.tsreader.com (03/07/2011)

[13] "VLC Media Player": http://www.videolan.org/vlc/ (03/07/2011)