FPGA DVB-S Encoder

Die Idee ist, einen DVB-S-Encoder in VHDL zu realisieren.

Software

- ETSI-Standard DVB-S
- drmpeg gr-dvbs
- TS-Erzeugung CBR

Schnittstellen

- Schnittstelle zum PC: Ethernet (UDP)
- Schnittstelle zum I/Q-Modulator: 2xDAC

Komponenten

Ethernet - FIFO - Scrambler - RS-Encoder - Interleaver - P/S-Converter - Convolutional Coder - Puncturing - Mapping - (Interpolation) - Baseband Filter

Bis nach dem Interleaver ist die Struktur byteweise aufgebaut, danach arbeitet sie bit-seriell. Die Pipeline muss vor dem RS-Encoder aller 188 Byte angehalten werden können, damit der RS-Encoder seine sechs Paritätsbytes einschieben kann.

Controller

Aufgabe:

- Datenstrom überwachen (Frame-Syncronität)
- Steuersignal für die einzelnen Komponenten erzeugen
    - Sync-Signal für Framestart an Scrambler
    - Data Valid für Scrambler
    - Reset für den Interleaver
    - Reset für den RS-Encoder
    - Reset für den Convoluational Coder

Schnittstelle:

- Eingang: Datenstrom aus FIFO (Byte)
- Ausgang: FIFO read enable (Bit)
- TBC...

Netzwerk-RX

Aufgabe:

- Empfang von UDP-Paketen (Sanity-Check)
- Weiterreichen der Nutzdaten an FIFO

Schnittstelle:

- Eingang: Netzwerk-Pins (Bits)
- Ausgang: Datenstrom (Byte)
- Ausgang: Daten valid (Bit)

Netzwerk-TX

Aufgabe:

- Auswerten der FIFO-Signale und Erzeugung von UDP-Nachichten zur Datenflusskontrolle
- Wenn FIFO fast leer: „Mach schneller“ senden
- Wenn FIFO fast voll: „Mach langsamer“ senden

Schnittstelle:

- Eingang: FIFO fast voll (Bit)
- Eingang: FIFO fast leer (Bit)

FIFO

Aufgabe:

- MPEG-Datenstrom von Ethernet entgegennehmen und an Encoder weitergeben
- Signalisierung der noch vorhandenen Daten (zu viel / zu wenig)

Schnittstelle:

- Eingang: Daten von Ethernet (Byte)
- Eingang: Schreiben aktiv (Bit)
- Eingang: Lesen aktiv (Bit)
- Ausgang: Daten an Encoder (Byte)
- Ausgang: Fast voll (Bit)
- Ausgang: Fast leer (Bit)

Scrambler

Aufgabe:

- Entsprechend der Position im Frame Scrambling anwenden
- MUX Adaptation (7 von 8 Sync Words invertieren)

Schnittstelle:

- Eingang: Reset-Sync-Logik (Byte)
- Eingang: valide Daten vorhanden (Clock Enable) (Byte)
- Eingang: Datenstrom von FIFO (Byte)
- Ausgang: Angepasster Datenstrom (Byte)

RS-Encoder

Aufgabe:

- Verkürzten RS-Code auf jeweils einen MPEG-Frame anwenden
- Paritätsbytes einfügen

Schnittstelle:

- Eingang: Daten ohne RS (Byte)
- Ausgang: Daten mit RS (Byte)

Interleaver

Aufgabe:

- Vertauschen der Byte-Reihenfolge

Schnittstelle:

- Eingang: Daten (Byte)
- Ausgang: Daten (Byte)

P/S-Converter

Aufgabe:

- Byteweisen Datenstrom in Bitweisen Datenstrom wandeln

Schnittstelle:

- Eingang: Daten (Byte)
- Ausgang: Daten (Bit, MSB first)

Convolutional Coder

Aufgabe:

- Faltungskode auf serielle Daten anwenden

Schnittstelle:

- Eingang: Daten (Bit)
- Ausgang: X (Bit)
- Ausgang: Y (Bit)

Puncturing/Mapping

Aufgabe:

- Weglassen definierter Bits aus dem convolutional Coder
- Mapping auf I/Q-Symbole

Schnittstelle:

- Eingang: X (Bit)
- Eingang: Y (Bit)
- Ausgang: I (Bit)
- Ausgang: Q (Bit)

(Interpolator)

Aufgabe:

- Einfügen von Nullen in den Datenstrom

Schnittstelle:

- Eingang: I (Bit)
- Eingang: Q (Bit)
- Ausgang: I (Bit)
- Ausgang: Q (Bit)

Baseband Filter

Aufgabe:

- Spectral Shaping mit RRC-Filter

Schnittstelle:

- Eingang: I (Bit)
- Eingang: Q (Bit)
- Ausgang: I (Bit-Vektor)
- Ausgang: Q (Bit-Vektor)

From:
https://www.loetlabor-jena.de/ - **Lötlabor Jena**

Permanent link:
https://www.loetlabor-jena.de/doku.php?id=projekte:das:dvbs&rev=1422619418

Last update: **2015/01/30 12:03**