

MPEG-2 TS MULTIPLEXER IN FPGA TECHNOLOGY

Denis Pavliha, Andrej Trost

University of Ljubljana, Faculty of Electrical Engineering, Ljubljana, Slovenia

Key words: Moving Picture Experts Group 2 Transport Stream (MPEG-2 TS), data multiplexer, Field Programmable Gate Array (FPGA), video, Ethernet.

Abstract: This paper discusses design and testing of a Moving Picture Experts Group 2 (MPEG-2) Transport Stream (TS) multiplexer in Field Programmable Gate Array (FPGA) technology that would transmit multiplexed packets via Ethernet. Multiplexer is designed in accordance with standard ISO/IEC 13818-1, thus it is fully compatible with standard players. The digital design is very flexible since adding additional payload sources is effortless. Use of a microprocessor has been dropped since it often provokes non-deterministic situations, hence the design is very robust yet fully programmable.

Podatkovni multiplekser MPEG-2 TS v vezju FPGA

Ključne besede: Moving Picture Experts Group 2 (MPEG-2) transportni pretok, podatkovni multiplekser, Field Programmable Gate Array (FPGA), video, Ethernet.

Izvleček: V članku je predstavljen razvoj in preizkušanje podatkovnega multiplekserja za transportni pretok Moving Picture Experts Group 2 (MPEG-2 TS) v tehnologiji Field Programmable Gate Array (FPGA), ki oddaja multipleksirane podatkovne pakete prek omrežja Ethernet. Multiplekser je izdelan v skladu s standardom ISO/IEC 13818-1, zaradi česar je popolnoma kompatibilen s standardnimi predvajalniki. Digitalno vezje je zelo fleksibilno, saj je vključevanje dodatnih podatkovnih virov enostavno. Uporaba mikroprocesorja je bila opuščena, saj ta pogosto izzove nedeterministične situacije. Vezje je torej zelo robustno, vendar še vedno popolnoma programabilno.

1. Introduction

As result of rapid evolution of consumer electronics having been faced recently we can notice an immense increase of data transfers worldwide. Predictions have been made about global Internet Protocol (IP) traffic quintupling from 2008 to 2013 /1/. Extensive bandwidth requirements are consequence of technological convergence which is an operators' concept of bundling together various structure-independent services the user can then access /2/, which can be seen from the engineer's point of view as merging of technologies and processes from multiple industries. It is only in the last decade when convergence has taken places in our everyday, joining mobile and stationery communications with broadband internet, terminal applications, digital television and other technologies.

When discussing digital television the data-transferring aspect of such an application should be considered. Nearly real-time transmission of moving picture (i.e. streaming) is very pretentious about required bandwidth, especially when transmitting high-definition (HD) content. Required transmission bitrates for Standard-Definition Television (SDTV) and for High-Definition Television (HDTV) are normally up to 7 Mbps and 30 Mbps, respectively /3/. Because of these bandwidth demands one should reflect on optimizing the data path not only by the means of improving payload compression, but mainly by using a proper information coding system providing an optimal transmission of transferred content. Today it is already very common to have network connections up to 100 Mbps over copper /4/ or 1000 Mbps over fiber; nevertheless an optimal coding system is required because transmission of video is a time-critical application.

If we take a look at a common digital television (DTV) program we can notice it consists of several elements. First, there is the video signal that describes the moving picture, together with one or more audio signals and probably equipped with some other data, such as Electronic Program Guide (EPG). Therefore we need to establish the best possible way of coding data before transmitting it, especially because we want to transmit multiple streams via a single communications channel. Such coding system is described in standard ISO/IEC 13818-1 /5/, known as Moving Picture Experts Group 2 (MPEG-2) Transport Stream (TS). It defines delivery of multiple elementary streams using services of multiplexing, timing, buffer management and control data transmission /6/.

Since recent hardware development tends toward solutions that are not only highly compact but also complex enough to perform sophisticated functions, we developed a solution that embeds an MPEG-2 TS Multiplexer carried out within a Field Programmable Gate Array (FPGA) circuit using digital logic elements only. Therefore the use of an embedded microprocessor has been dropped, which was possible mainly because a multiplexer is actually a fundamental unit of FPGA designs. Therefore a complex multiplexer like MPEG-2 TS can be carried out using basic smaller multiplexers with some additional complex control logic. Nevertheless such a system needs to be flexible enough to embed all the mechanisms that are required by standard ISO/IEC 13818-1 which fully determines MPEG-2 TS /5/.

2. Methods

An ordinary digital television (DTV) program consists of a video signal, an audio signal and auxiliary data (e.g. Electronic Program Guide - EPG). Those fundamentals can be interpreted as elementary streams of that program. Because we want them transferred over a common communications channel we need to packetize the elementary streams and conduct them through the process of multiplexing. An example data multiplexer is presented in Fig.1.

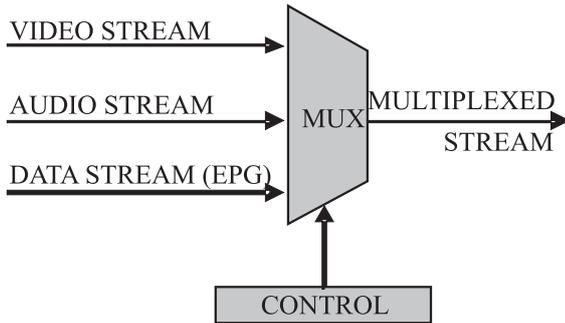


Fig. 1: Example data multiplexer.

The three elementary streams we can see in Fig.1 are coupled into a single multiplexed data stream that embeds both video and audio, together with some additional data (e.g. EPG). Such a stream represents a single DTV program. Nevertheless, if hardware is powerful enough we can perform multiplexing of more than just one DTV program to join them into a single stream that can be transferred via a communications channel. An example of multiplexing three programs into a single stream is shown in Fig.2.

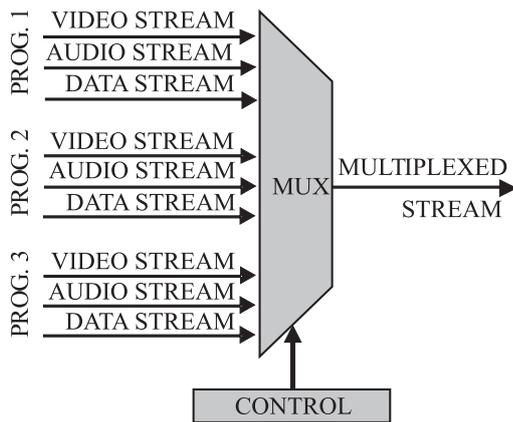


Fig. 2: Example multiprogram data multiplexer.

In order to achieve a configuration capable of multiplexing several digital television (DTV) programs into a single data stream, one should consider a proper hardware configuration that would allow construction of a compact yet powerful device. The solution we developed is based on a Field Programmable Gate Array (FPGA) integrated circuit, together with some additional communication periphery.

2.1 Digital design

Schematics of Field Programmable Gate Array (FPGA) circuit contents are shown in Fig.3. The Moving Picture Experts Group 2 Transport Stream (MPEG-2 TS) multiplexer shown in Fig.3 is basically a complex digital switch, made of several small multiplexers. Due to the FPGA technology being carried out mainly with look-up tables (LUT) and flip-flops (FF), a multiplexer is actually a "natural" unit of an FPGA digital design.

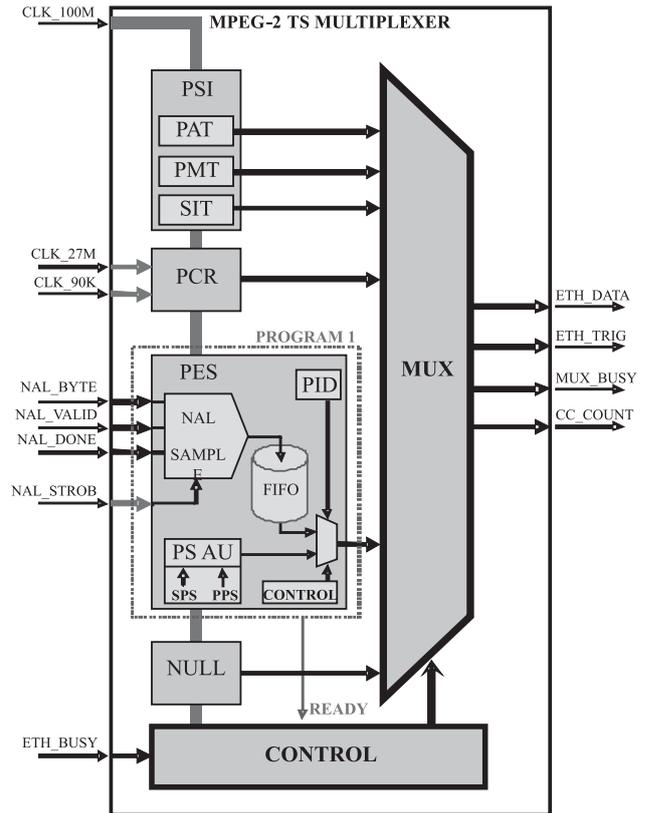


Fig. 3: Contents of FPGA circuit.

The main multiplexer in the MPEG-2 TS digital design (MUX in Fig.3) switches between various sources and it is based on 100 MHz clock that is captured directly from an external oscillator, hence jitter is negligible.

Program Specific Information (PSI) Table generation unit produces tables that are needed in MPEG-2 TS to specify what elements are contained within the Transport Stream. These tables are Program Association Table (PAT), Program Map Table (PMT) and Selection Information Table (SIT), and carry information about elementary streams that are then packetized and multiplexed together with other payload data into single Transport Stream.

Program Clock Reference (PCR) section handles two counters that are sampled separately but joined together into a single data packet. The packet contains samples of 27 MHz and 90 kHz clocks and is needed for the decoder to synchronize and properly present the decompressed content at correct time. Since the 27 MHz counter actually represents System Time Clock (STC) it has to have a clean

source, hence it is driven directly from a 27 MHz oscillator so that jitter is insignificant. Clock with frequency 90 kHz is, however, produced using a Digital Clock Manager (DCM) unit and therefore additional jitter is present.

Program Block, marked as "Program 1" on Fig. 3, contains logic to handle, temporary store and multiplex a single video signal, represented in the form of a Network Abstraction Layer (NAL) bit stream. Such a representation of video signal is typical of standard Motion Picture Experts Group 4 – Advanced Video Coding (MPEG-4/AVC), more known as H.264 /7/. A stream like this can be handled by our MPEG-2 TS multiplexer quite effortlessly. Program Block incorporates logic to sample NAL bit stream (input NAL_BYTE) regarding appropriate flags (inputs NAL_VALID, NAL_DONE and NAL_STROB) and stores it into a First-In-First-Out (FIFO) buffer, generated as data vector with a pointer that act together as a circular buffer. When first half of buffer is full the Program Block adds the proper Packet Identifier (PID) and sends the NAL bit stream to the main multiplexer as a Packetized Elementary Stream (PES), while still continuing to store the incoming NAL bit stream into the second half of FIFO. Whether the incoming NAL stream does not include Presentation Set Access Units (PS AU), the Program Block can add both Stream Parameter Set (SPS) and Picture Parameter Set (PPS).

Since main benefit of multiplexing video or audio signals is the possibility of packing more such signals into a single transport stream, the Program Block (PES in Fig. 3) can be multiplied into several blocks, each handling its own NAL bit stream. Adding parallel blocks is not an issue in FPGA circuits, however the number of available logic blocks and speed of the outgoing Ethernet connection represent the limitation.

Additionally, so called NULL packets can be added to the transport stream. NULL packets are packets that are identified with a Packet Identifier (PID) of 0x1FFF and contain only 0xFF values as their payload. Their purpose is to maintain a constant bit rate of the output transport stream by inserting them when no other packet is scheduled to be multiplexed. However, an upper limitation of inserting these packets has to be implemented in the multiplexer control logic. If not, the Ethernet connection can be flooded and the receiver could not handle proper packet reception anymore.

Output of the multiplexer is a 188-bytes-long data vector (ETH_DATA in Fig. 3) sent directly to the LocalLink interface /8/ of the Xilinx Tri-Mode Ethernet Media Access Controller (TEMAC) core, together with trigger (ETH_TRIG) and busy (MUX_BUSY) signals. The output represents a single MPEG-2 TS packet; therefore an Ethernet frame can contain only one MPEG-2 TS packet with a single higher-level packet (PSI, PES or NULL). This is actually a quite strict limitation, which was imposed in order to minimize the time of designing the multiplexer core. Nevertheless, performance is not affected significantly because use of Gigabit Ethernet is applied.

Because MPEG-2 TS packets are normally sent through User Datagram Protocol (UDP) over Internet Protocol (IP) and there is no feedback about success of transmission,

/5/ defines a special section that is contained in the header of TS named Continuity Counter which is a 4-bit value that increases every new transmission of a packet with the same PID number. Since there is the restriction of a single packet within one Ethernet frame, we can implement the Continuity Counter outside the MPEG-2 TS multiplexer. In our case the Continuity Counter resides in the LocalLink interface module. Consequently we need to send out of the core a flag vector to signalize the PID of the packet being currently transmitted (CC_COUNT in Fig. 3) so that the Continuity Counter can increase properly regarding the PID of the packet being sent.

Digital design described in this chapter was realized with FPGA circuit XC5VFX70T-1FFG1136 /9/ that resides on evaluation platform ML507 /10/.

2.2 Hardware configuration

ML507 Evaluation Platform /10/ is a general-purpose development board, based on XC5VFX70T-1FFG1136 that is an FPGA circuit from Virtex-5 FXT Family /9/. The FPGA itself incorporates 12 Digital Clock Managers (DCMs) that can be used to generate various clock signals. 5,328 kilobits of Block Random Access Memory (BRAM) is available and a maximum of 640 single-ended (or 320 differential-pair) input-output (I/O) pins can be utilized. The designer can also bring into use four Gigabit Ethernet Media Access Controller (GEMAC) blocks and 16 High-Speed *Rocket I/O GTX* transceivers.

ML507 board was mainly selected because of the need for rapid prototyping having been faced. The board embeds both clocking components (27 MHz and 100 MHz oscillators) and connectivity periphery; integrated circuit Marvel 88E1111 is connected to GEMAC using Serial Gigabit Media-Independent Interface (SGMII) and acts as the Physical layer of Ethernet IEEE 802.3 connection /11/ together with Halo RJ-45 connector with magnetics. Hardware configuration that has been used is shown in Fig. 4.

As seen in Fig. 4 all the Network Abstraction Layer (NAL) signals (NAL_BYTE, NAL_VALID, NAL_DONE and NAL_STROB) are brought into circuit from outside (as input pins) while clock signals (CLK_100M and CLK_27M) are captured from on-board oscillators.

When implemented the MPEG-2 TS design takes 23% of Flip-Flops (FF) and 40% of all available slices in XC5VFX70T. Amongst those, 24% are used as Look-Up Tables (LUT) only, 44% as Flip-Flops (FF) only and the rest (32%) utilizing both LUT and FF. Beside logic elements, 3 units of 18k Block Random Access Memory (RAM) are used. Implementation results are shown in Fig. 5.

2.3 Operational process

The whole digital design is synchronized to run at 100 MHz main clock (CLK_100M in Fig. 3 and Fig. 4). Frequency of multiplexing can be modified through an internal signal, however it is intended to leave the setting at its default value in order to achieve timings that are prescribed by /5/.

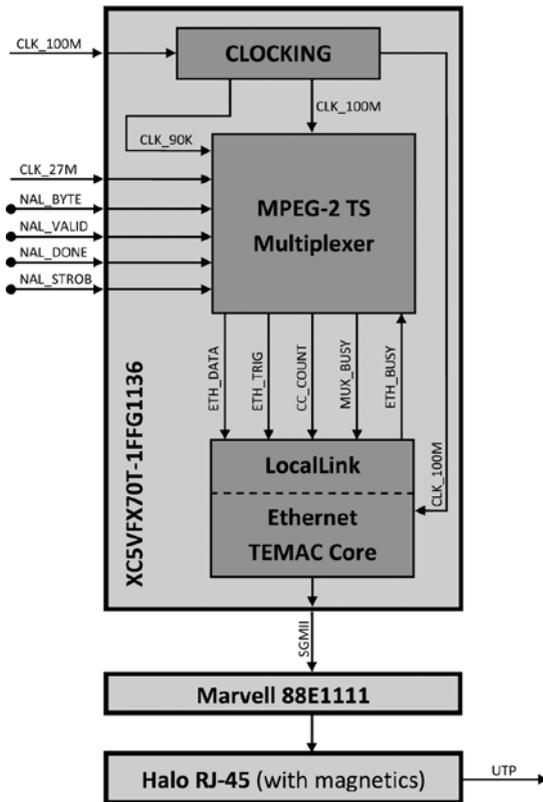


Fig. 4: Hardware configuration.

The Network Abstraction Layer (NAL in Fig.3) handling logic located in Program Block (*Program 1* in Fig.3) is designed to produce a READY internal flag that signalizes a whole Packetized Elementary Stream (PES) packet is waiting in First-In-First-Out (FIFO) buffer to be sent.

Main multiplexer (*MUX* in Fig.3) is designed as a priority switch. If the READY internal flag is detected, PES packet is sent to Ethernet LocalLink interface; else an appropriate table (PAT, PMT, SIT) or counter (PCR) is generated accordingly to timing specifications of /5/ and transmitted to LocalLink interface in the form of a 188-byte vector. If none of the payload-transmitting conditions are met, a NULL packet may be sent instead. Nevertheless, since transmission of NULL packets should be limited in order not to flood the Ethernet connection, not sending anything at all is frequently more appropriate.

The output data vector is then sent to the LocalLink interface of the Xilinx Tri-Mode Ethernet Media Access Controller (TEMAC) core together with flags, where the Continuity Counter (CC) is added and data is encapsulated into the User Datagram Protocol over Internet Protocol (UDP/IP) and sent as an Ethernet frame.

3. Results

Circuit design has been developed using Xilinx ISE Foundation v10.1 K31 (Xilinx, San Jose, USA, 2008) in Very High Speed Integrated Circuit Hardware Description Language (VHSIC HDL - VHDL). Reception of generated multiplexed

Device Utilization Summary			
Slice Logic Utilization	Used	Available	Utilization
Number of Slice Registers	10,444	44,800	23%
Number used as Flip Flops	10,444		
Number of Slice LUTs	7,720	44,800	17%
Number used as logic	7,708	44,800	17%
Number using O6 output only	7,407		
Number using O5 output only	213		
Number using O5 and O6	88		
Number used as exclusive route-thru	12		
Number of route-thrus	230	89,600	1%
Number using O6 output only	225		
Number using O5 output only	5		
Slice Logic Distribution			
Number of occupied Slices	4,524	11,200	40%
Number of LUT Flip Flop pairs used	13,816		
Number with an unused Flip Flop	3,372	13,816	24%
Number with an unused LUT	6,096	13,816	44%
Number of fully used LUT-FF pairs	4,348	13,816	31%
Number of unique control sets	1,107		
IO Utilization			
Number of bonded IOBs	62	640	9%
Number of bonded IPADs	6		
Specific Feature Utilization			
Number of BlockRAM/FIFO	2	148	1%
Number using BlockRAM only	2		
Total primitives used			
Number of 18k BlockRAM used	3		
Total Memory used (KB)	54	5,328	1%
Number of BUFG/BUFGCTRLs	7	32	21%
Number used as BUFGBs	7		
Number of BUFDSs	1	8	12%
Number of BUFRs	1	32	3%
Number of DCM_ADVs	1	12	8%
Number of GTX_DUALs	1	8	12%
Number of TEMACs	1	2	50%

Fig. 5: Implementation results.

test stream was performed on a workstation based on the AMD Athlon64 3000+ Processor (2.0 GHz) with 1.5 GB of DDR SDRAM, equipped with Gigabit Ethernet Network Interface Controller (NIC) Realtek RTL8111B.

To verify the correct reception of User Datagram Protocol (UDP) over Internet Protocol (IP) *Wireshark* software /12/ was used as it performs network protocol analysis. Results from *Wireshark* are shown in Fig.6 where it is noticeable that UDP/IP packets are sent via Ethernet to Media Access Controller (MAC) of the workstation and are properly received. Each received packet consists of 231 bytes of data which is the sum of 188 bytes representing MPEG-2 TS payload data and 43 bytes of MAC/IP/UDP headers. Packets are marked as part of the multicast group with IP address 239.192.1.100 on port 5500.

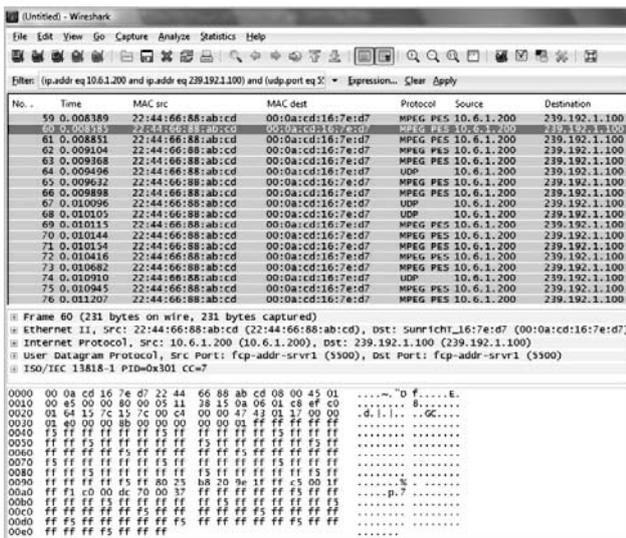


Fig. 6: Network Protocol Analysis results from Wireshark.

Received MPEG-2 TS stream is then isolated using *Wireshark* to include only payload without MAC/IP/UDP headers and after that analyzed using *MPEG-2 Transport Stream Analyzer Enhanced* software by Manzanita Systems, Ltd /13/. Structure of the analyzed stream that has been produced using our MPEG-2 Multiplexer is shown in Fig. 7.

PAT Version 0 PID 0x0000	NIT_PID 0x001F		
	Program 1 PID 0x0100	PMT Version 0	Descriptor 0x88 registration_descriptor
			0x1011 MPEG-4 / AVC / H.264 Video avc_video_descriptor
			0x1001 PCR

Fig. 7: Structure of analyzed MPEG-2 Transport Stream that includes sample H.264 video payload.

Ratio of individual elements is also analyzed using *MPEG-2 Transport Stream Analyzer Enhanced* software and is shown in Fig.8 where it can be seen that main part of the stream is taken by video payload. All the other elements occupy less than 2% each. Since table timing limitations prescribed in /5/ are met, such ratio is expected and correct.

To confirm that produced stream is compliant with standard /5/ we performed another analysis. The second tool used is *Stream Analyzer* by Elecard /14/. Results of analysis are illustrated in Fig.9.

Stream structure in Fig.9 is the same as in Fig. 7; it consists of a single program containing one H.264 video stream. Packet analysis shown in the lower part of Fig.9 also confirms that Presentation Set Access Units (PS AU) of Sequence Parameter Set (SPS) and Picture Parameter Set (PPS) are properly generated, since their decoding is successful.

As mentioned in (2.1.) Program Block (PES in Fig.3) can be multiplied into several blocks in order to multiplex various NAL bit streams into a single MPEG-2 TS. A com-

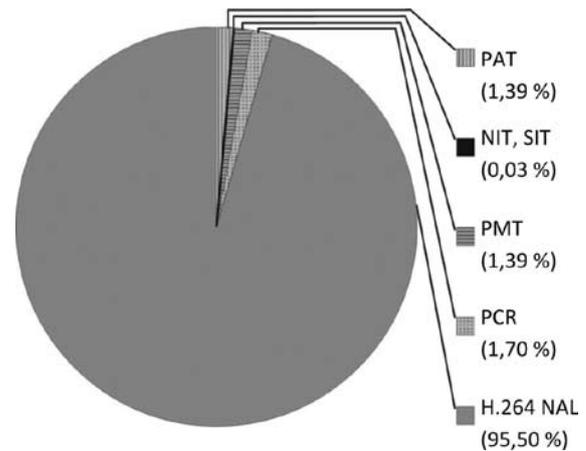
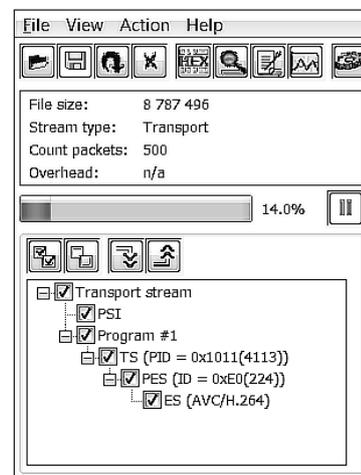


Fig. 8: Ratio of individual elements in the analyzed MPEG-2 Transport Stream.



0x00009A38	Transport Packet { PID = 0x1011,2 Payload = Yes
0x00009AF4	Transport Packet { PID = 0x1011,2 Payload = Yes
0x00009BB0	Transport Packet { PID = 0x1011,2 Payload = Yes
0x00009C6C	Transport Packet { PID = 0x1001,2 Payload = No
0x00009D28	Transport Packet { PID = 0x1011,2 Payload = Yes
0x00009DE4	Transport Packet { PID = 0x1011,2 Payload = Yes
0x00009DE8	PES Packet { stream_id = 0xE0 (video stream)
0x00009DF6	H264 Sequence Parameter Set 227
0x00009E02	H264 Picture Parameter Set 228
0x00009E0A	H264 I slice #1 229
0x00009EA0	Transport Packet { PID = 0x1011,2 Payload = Yes
0x00009F5C	Transport Packet { PID = 0x1011,2 Payload = Yes
0x0000A018	Transport Packet { PID = 0x1011,2 Payload = Yes
0x0000A0D4	Transport Packet { PID = 0x1011,2 Payload = Yes
0x0000A190	Transport Packet { PID = 0x1011,2 Payload = Yes
0x0000A24C	Transport Packet { PID = 0x1011,2 Payload = Yes

Fig. 9: Analysis of MPEG-2 TS using Stream Analyzer.

parison is shown in Fig.10 where utilization of FPGA circuit XC5VFX70T-1FFG1136 can be seen when only one program block is used (a) or another one is added (b).

As seen in Fig.10, when adding a second program stream, the number of used slices with only Look-Up Table (LUT) basically does not increase; however, increase of Flip-Flop (FF) and combined (LUT with FF) slices is obvious.

Since MPEG-2 TS multiplexer is a time-critical application, timing performance is significant. Timing report that has been generated by the tools /15/ claims all timing con-

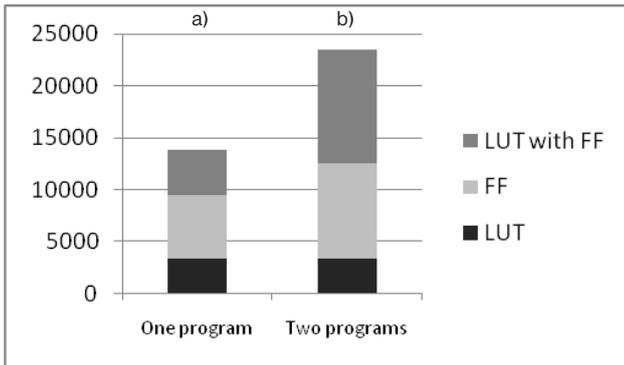


Fig. 10: LUT and FF utilization comparison.

straints have been met. There have been 4179 paths and 1003 endpoints analyzed without any errors. Minimum period is 7.609 ns, which represents a maximum frequency of 131.42 MHz.

4. Discussion

This paper discusses design of a Moving Picture Experts Group 2 (MPEG-2) Transport Stream (TS) multiplexer in Field Programmable Gate Array (FPGA) integrated circuit. FPGA allows creating digital designs that are fully programmable while lacking the use of a microprocessor.

The latter is frequently a source of problems, since multiple error-prone situations can occur; first, software can contain bugs that are harder to identify in comparison with a digital design, described in VHDL code. Besides, non-deterministic situations can be provoked as consequence of several microprocessor-related mechanisms (stack, interrupt controllers, exception handlers). All those facts speak in favor of developing a MPEG-2 TS multiplexer using FPGA rather than a microprocessor.

The MPEG-2 TS multiplexer we designed performs in accordance with standard /5/, hence it is fully compatible with hard and/or soft players available on both consumer electronics and professional equipment market.

Upgrade to supporting multiple H.264 streams is possible and effortless. If other content than Network Abstraction Layer (NAL) bit stream shall be multiplexed, only minor modifications in the sampling logic before storage in First-In-First-Out (FIFO) have to be made. The limitations represent the chosen FPGA and the outgoing communication interface, in our case Gigabit Ethernet.

As benefit of this FPGA-based design, use in Application-Specific Integrated Circuit (ASIC) technology could be applied. To make that possible, some improvements should be implemented. Since all stream data (program tables, stream parameters) is stored in FPGA an external interface to an Electrically Erasable Programmable Read-Only Memory (EEPROM) should be considered. The data could, then, be externally programmed, which could lead to a far more flexible system and after all, reduce costs in large quantities of the application.

Acknowledgments

The authors of this paper thank *Laboratory of Biocybernetics* (University of Ljubljana, Slovenia) for Xilinx ML507 Evaluation Board and *Manzanita Systems, Ltd.* for software evaluation.

References

- /1/ Cisco, *Cisco Visual Networking Index: Forecast and Methodology, 2008–2013*, 2009.
- /2/ D. Colbeck, "Communications Convergence," in 6th IEE Conference on Telecommunications, Edinburgh, UK, 1998, pp. 258-262.
- /3/ Y. Zhang, H. Wei, and Y. Qiao, "A multi-channel HD video streaming in digital community network: Video rate adaptation and playback adjustment," *IEEE TRANSACTIONS ON CONSUMER ELECTRONICS*, vol. 53, no. 4, pp. 1449-1455, NOV 2007, 2007.
- /4/ E. Heine, J. Hogenbirk, P. Jansweijer *et al.*, "Using VDSL2 over copper in the vertical string," *NUCLEAR INSTRUMENTS & METHODS IN PHYSICS RESEARCH SECTION A-ACCELERATORS SPECTROMETERS DETECTORS AND ASSOCIATED EQUIPMENT*, vol. 602, no. 1, pp. 155-158, APR 11 2009, 2009.
- /5/ ISO/IEC, "13818-1, Information technology - Generic coding of moving pictures and associated audio information: Systems," ISO Copyright Office, 2000.
- /6/ J. Arnold, M. Frater, and M. Pickering, *Digital Television*, New Jersey, USA: John Wiley & Sons, Inc., 2007.
- /7/ ISO/IEC, "14496-10, Information Technology - Coding of Audio-Visual Objects, Part 10: Advanced Video Coding," ISO Copyright Office, 2004.
- /8/ Xilinx. "LocalLink Interface," http://www.xilinx.com/products/ipcenter/LocalLink_UserInterface.htm.
- /9/ Xilinx, "Virtex-5 FPGA Users Guide (UG190)," Xilinx, 2009. http://www.xilinx.com/support/documentation/user_guides/ug190.pdf
- /10/ Xilinx. "Virtex-5 FXT FPGA ML507 Evaluation Platform," <http://www.xilinx.com/products/devkits/HW-V5-ML507-UNI-G.htm>.
- /11/ IEEE. "802.3 Ethernet Working Group," <http://www.ieee802.org/3/>.
- /12/ Wireshark Foundation. "Wireshark: The World's Foremost Network Protocol Analyzer," <http://www.wireshark.org/>.
- /13/ Manzanita Systems. "MP2TSAE - MPEG-2 Transport Stream Analyzer Enhanced," <http://www.manzanitasystems.com/mp2tsa.html>.
- /14/ Elecard. "Stream Analyzer," <http://www.elecard.com/products/products-pc/professional/stream-analyzer/>.
- /15/ Xilinx. "Design Tools," <http://www.xilinx.com/tools/designtools.htm>.

Denis Pavliha, B.S.
University of Ljubljana, Faculty of Electrical Engineering
Tržaška c. 25, 1000 Ljubljana, Slovenija
Phone: +386 (0)1 4768 771; Fax:+386 (0)1 4264 658
Email: denis.pavliha@fe.uni-lj.si

Andrej Trost, PhD.
University of Ljubljana, Faculty of Electrical Engineering
Tržaška c. 25, 1000 Ljubljana, Slovenija
Phone: +386 (0)1 4768 350; Fax:+386 (0)1 4264 630
Email: andrej.trost@fe.uni-lj.si