

FPGA Implementation of Video Transmission System Based on LTE

Yan Lu*, Qianlong Zhang & An Tang

College of Information Science and Technology, Beijing University of Chemical Technology, Beijing, China

ABSTRACT: In order to support high-definition video transmission, an implementation of video transmission system based on Long Term Evolution is designed. This system is developed on Xilinx Virtex-6 FPGA ML605 Evaluation Board. The paper elaborates the features of baseband link designed in Xilinx ISE and protocol stack designed in Xilinx SDK, and introduces the process of setting up hardware and software platform in Xilinx XPS. According to test, this system consumes less hardware resource and is able to transmit bidirectional video clearly and stably.

Keywords: FPGA, Long Term Evolution, video transmission

1 INTRODUCTION

The Long Term Evolution (LTE) is standardized by the 3GPP, as the successor of the Universal Mobile Telecommunication System (UMTS), in order to ensure a high speed data transmission with mobility for mobile communication [1]. LTE improves on the 3GPP air interface, and applies Orthogonal Frequency Division Multiplexing (OFDM) and Multiple Input Multiple Output (MIMO) to improve the performance of wireless system. LTE downlink transmission rate can be up to 100Mbps, and uplink transmission rate can be up to 50Mbps [2-3].

With the increase of the pace of modern life, people are eager for high-speed wireless access services at all times and places. This makes LTE get mature quickly. Now the researches of LTE mainly focus on PHY, and the studies of software protocol stack are quite few. Reference [4] only shows the design of LTE baseband link. Reference [5] shows the simulation of physical link. Reference [6] mentions the implementation of hardware platform for LTE baseband link, but doesn't refer to the design of protocol stack. Reference [7] implements data transmission on system layer, but doesn't mention about the development of hardware platform.

In this paper, we propose a bidirectional video

transmission system based on LTE. First, we set up system platform on Virtex-6 FPGA in Xilinx XPS environment to support protocol stack and baseband link. Second, Protocol stack is designed in Xilinx SDK environment to conduct data transmission between digital video server (DVS) and baseband link. Finally, baseband link is designed in Xilinx ISE environment to conduct data transmitting and receiving.

This paper is organized as follows. In the next section, an overview of the system is described. In Section 3, the development of the system platform is described. In Section 4, the process of protocol stack is shown, including Ethernet protocol stack and LTE protocol stack. In Section 5, the design of baseband link IP core is shown. In Section 6, some test results of our system are analyzed. Finally, conclusions are drawn in Section 7.

2 AN OVERVIEW OF SYSTEM

Architecture of the video transmission system is shown in Figure 1. This system consists of camera, DVS, Xilinx ML605 board, radio frequency (RF) module, decoder and display. We set up hardware and software platform on the Virtex-6 FPGA on ML605 board, and then develop software protocol stack and

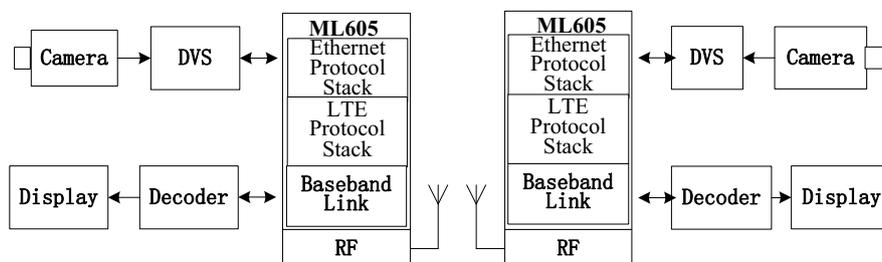


Figure 1. Architecture of video transmission system

*Corresponding author: luyankuku@126.com

This is an Open Access article distributed under the terms of the Creative Commons Attribution License 4.0, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

baseband link IP core based on the platform. This system is able to conduct bidirectional video transmission. Camera captures video and transmits video signal to DVS. DVS conducts video encoding according to pre-set compression algorithm, and then transmits code stream to local decoder and transmitter of baseband link via software protocol stack. Transmitter transforms code stream into OFDM symbols, and then transmits them via RF module. Receiver of the other ML605 board transforms the OFDM symbols received into code stream, and then transmits it to local decoder via software protocol stack. Decoder conducts video decoding and transmits video signal to display. In this way, we can see both local and received video on one display.

3 SYSTEM PLATFORM

To support protocol stack and baseband link, we should set up hardware and software platform on Virtex-6 FPGA first. ML605 board is connected to decoder and DVS via two RJ-45 interfaces. The VITA-57 FMC HPC connector on board can support extra features that developer needs. In this paper, this connector is used to connect ML605 board with RF module. In this section, a general description of the system platform which is implemented on ML605 board will be provided, including hardware platform and software platform.

3.1 Hardware platform

The hardware platform is designed in Xilinx XPS environment. We select necessary IP cores as outlined in Table 1, then connect the IP cores to specified bus, and finally generate ports and address for these IP cores. By this means we complete the construction of hardware platform.

Table 1. Selected IP cores

IP cores	Function
LMB_BRAM	Stores instructions and data. High access rate.
ILMB_CNTRLR	Accesses instructions in LMB_BRAM
DLMB_CNTRLR	Accesses data in LMB_BRAM
MicroBlaze	CPU of the hardware platform. Executes instructions and processes data.
XPS_INTC	Interrupt controller
RS232_UART	Transmits data via serial port.
DDR3_SDRAM	Stores instructions and data. Large storage space. Lower access rate than LMB_BRAM
XPS_TIMER	Offers software timing function.
HARD_MAC_FIFO	Data buffer of baseband link IP core

HARD_ETHERNET_MAC Offers RJ-45 interfaces driver

The connection of selected IP cores is shown in Figure 2. MicroBlaze supports four kinds of bus protocols [8]: Local Memory Bus (LMB), the IBM Processor Local Bus (PLB), the AMBA® AXI4 interface (AXI4) and Xilinx CacheLink (XCL). We apply LMB, PLB and XCL in hardware platform. LMB is used by MicroBlaze to access block ram on-chip. XCL is used to improve the access rate of DDR3_SDRAM. PLB is used to connect IP cores and RAMs. Baseband link will be connected to PLB as an IP core after implementation. Then hardware platform is set up completely. We generate bit stream file in XPS environment and export to SDK environment for setting up software platform.

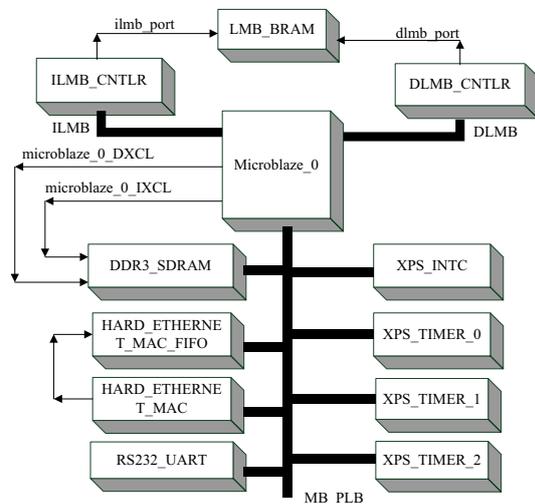


Figure 2. Architecture of hardware platform

Microblaze is based on RISC instruction set. It can support 3-stage and 5-stage pipeline architecture. 5-stage pipeline architecture consumes more hardware resource but has better performance. We choose 5-stage pipeline architecture in this paper, and the 5 stages are configured to instruction fetch, decoding, executing, memory access and writing back.

3.2 Software platform

In SDK environment, we generate Board Support Package (BSP) file first. BSP is a miniature operating system. Protocol stack is designed and also run based on this operating system. BSP contains two kinds of inner cores: single-threaded “Standalone” and multi-thread “Xilkernel”. In this paper, we choose and configure the more powerful “Xilkernel” to support bidirectional video transmission.

Third party libraries are necessary for software platform to design protocol stack. “Lwip130” and “Xilnfs” are two build-in libraries of SDK. “Lwip130”

is a lightweight TCP/IP network library, and “Xilnfs” provides inner core with memory filesystem. These two libraries are chosen and configured in SDK

4 PROTOCOL STACK

Protocol stack is designed in SDK, and it is made up by two parts: Ethernet protocol stack and LTE protocol stack. Ethernet protocol stack is designed to support DVS and decoder. LTE protocol stack is used to implement data exchange between Ethernet protocol stack and baseband link.

4.1 Ethernet protocol stack

Ethernet protocol stack conduct data routing by using the build-in “LWIP” module of SDK. First, DVS broadcasts the Address Resolution Protocol (ARP) request messages to the local network. The local ML605 board, as an intermediate device, resolves the ARP request to detect the destination IP address. If the destination IP address is the same as the IP address of ML605 board, ML605 board would response the ARP request, and DVS would start to transmit code stream to ML605 board. Otherwise, ML605 board would discard the request, and DVS would broadcast the ARP request message repeatedly.

The transmission of code stream needs to use two cores of “LWIP”. One is for packet buffer, the other one is for data receiving and transmitting. When receiving code stream, Ethernet protocol stack stores it in packet buffers. These buffers are from packet buffer pool which is pre-allocated when system starts. The use of packet buffer pool avoids memory allocation which is time-consuming, when interrupt handler respond to an interrupt request from network card. In this way interrupt handler can respond rapidly.

Before transmitting, Ethernet protocol stack fetches code stream to form service data unit (SDU) according to LTE protocol.

4.2 LTE protocol stack

LTE protocol stack is designed based on LTE RLC layer. In transmitter, RLC entity of LTE protocol stack received the SDU from Ethernet protocol stack via service access point (SAP). Then it processes the SDU and makes up protocol data unit (PDU). Finally, the PDU is transmitted to baseband link via logic channel. In receiver, RLC entity receives the PDU from baseband link via logic channel and makes up SDU. The SDU is sent to LTE protocol via SAP. This process is shown in Figure 3.

Three kinds of RLC modes are applied in LTE protocol stack to ensure transmission efficiency and reliability. In transparent mode (TM), RLC entity completes transmission without processing SDU and PDU. In unacknowledged mode (UM), RLC entity in transmitter splits SDU, and conducts cascading and RLC

header adding to make up PDU; RLC entity in receiver regroup SDU on the contrary. Comparing with UM, RLC entity in acknowledged mode (AM) is able to conduct automatic retransmission. AM loses some efficiency but improves reliability.

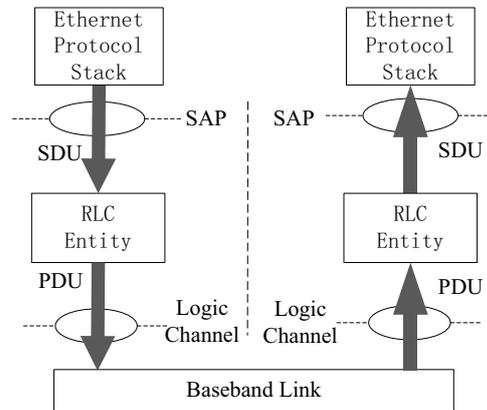


Figure 3. Process of LTE protocol stack

5 BASEBAND LINK

The architecture of baseband link is illustrated in Figure 4, and it is developed in Xilinx ISE environment. In order to support 2*2 MIMO configuration, and baseband link applies two-layer architecture. The channel encoder applies Turbo code to encode data, and the scrambler randomizes data. These improve the system reliability. QAM modulator changes bit data into IQ data. MIMO diversity is adopted by MIMO encoder to support multiple antennas. MIMO encoder applies space-frequency block code, and it encodes IQ data into space-frequency block. Resource map module maps IQ data, reference signals, primary synchronization signals and secondary synchronization signals to subcarriers according to frame structure of baseband link, and then input these data and signal to FFT IP core to conduct IFFT. Synchronization module is used to establish link between transmitter and receiver. Resource de-mapping module is used to conduct FFT and separate IQ data and reference signals. Channel estimator utilizes the reference signals received to estimate channel parameters for MIMO decoder. MIMO decoder uses channel parameters and IQ data received to rebuild original IQ data. LLR module, descrambler and channel decoder are the inverse processes of QAM module, scrambler and channel encoder. The clock frequency baseband link used in FPGA design is up to 200MHz to decrease time delay during data processing.

5.1 Baseband link simulation

As illustrated in Figure 5, the adopted frame structure for baseband link is type 2, which is use for FDD-LTE

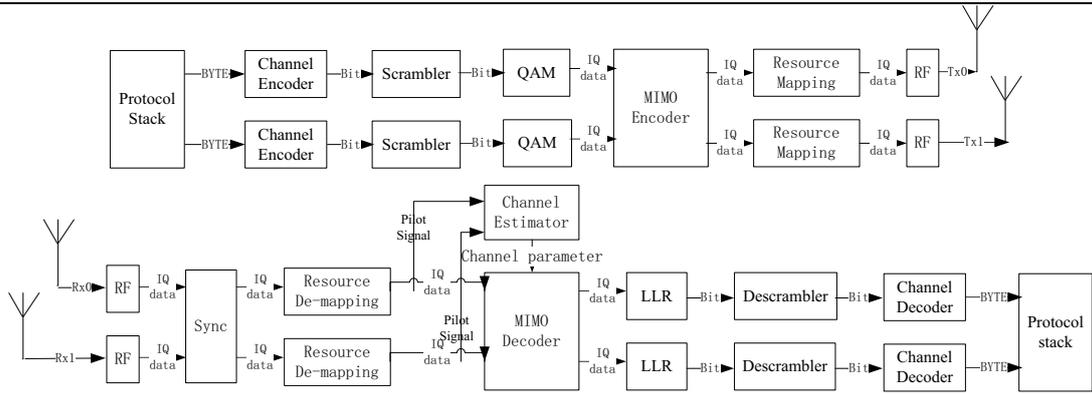


Figure 4. An overview of baseband link

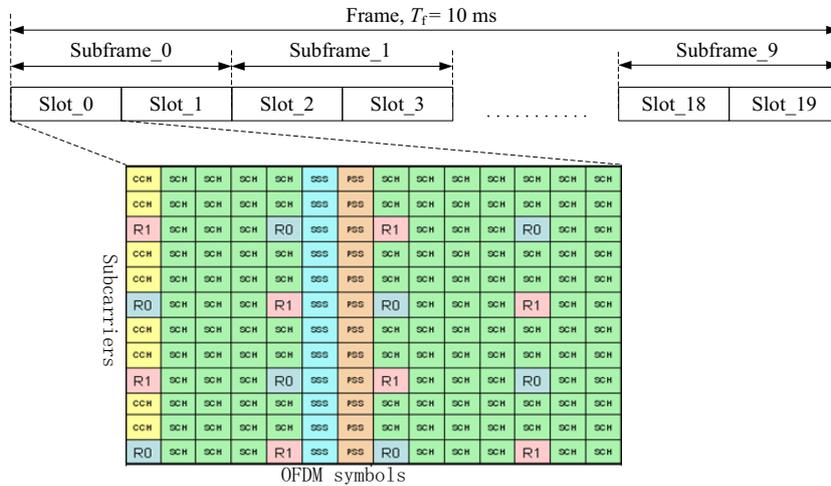


Figure 5. Frame structure for baseband link

system [9-10]. R0 and R1 are reference signals transmitted by Tx0 and Tx1. Reference signal is mapped to every OFDM symbol 0 and symbol 4, and used by channel estimator to estimate channel parameter. CCH is physical control channel. It is mapped to the first OFDM symbol in every sub-frame, and used to carry control signal. PSS is primary synchronization signal and SSS is secondary synchronization signal. They are mapped to the last two OFDM symbols. They are used to establish link between transmitter and receiver, and also used to indicate cell ID. SCH is physical shared channel. It is mapped to the rest of resource elements.

The main parameters of baseband link are outlined in Table 2. Least square algorithm and linear interpolation is used by channel estimator to restrain interference indoors. We simulate the baseband link in gauss and multi-path channels. The bit error rate (BER) performance is described in Figure 6.

Table 2. Baseband link parameters

Parameters	Value
Bandwidth	10MHz
Number of subcarriers	600
IFFT length	1024
MIMO configuration	2Tx2R
Modulation type	16QAM
CP length	80 or 72
Multipath model	Time delay(us) [0,2,4,6,8,12]
(6 paths)	Power ratio (dB) [-3,0,-2,-6,-8,-10]

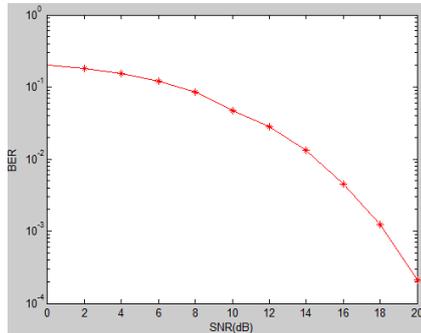


Figure 6. BER performance of baseband link

5.2 Module design

Baseband link adopts modular design. Simplified AXI-stream interface is applied to every module. FIFO with AXI-stream interface is not only used to connect master and slave module, but also buffer data from master module. General structure of AXI-stream FIFO is shown in Figure 7. Ready signal indicates weather the module is ready for transmission. Valid signal indicates weather data signal is valid. According to AXI-stream, if ready signal and valid signal are both 1 when clock rising edge comes, data transmission completes once.

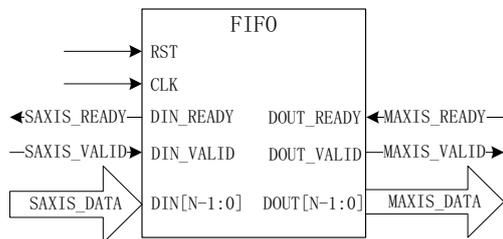


Figure 7. General Structure of AXI-stream FIFO

State machine decomposition is applied to module design. In the top module, state machine has 4 states: start, receiving, processing and transmitting. Start state is used for module reset. Receiving state sets ready signal to 1 when it begins. When receiving enough data, it sets ready signal to 0, and then state machine is moved from receiving to processing. In processing state, sub-state machine start to work to conduct data processing. Sub-state machine also can be divided into smaller state machine. In this way, we divide long logic delay path into lots of short paths to improve performance of clock. When completing data processing, state machine is moved to transmitting. Transmitting state sets valid signal to 1 when it starts. When data transmitting completes, valid signal is set to 0 and state machine is moved to receiving state

5.3 RF module

Figure 8 shows the connection between RF module and ML605 board. RF module contains D/A chips, A/D chips, clock distribution chip, and so on. RF module must be configured before data transmission. The VITA-57 FMC HPC connector is used for not only data transmission, but also RF module configuration. When the system is turned on, baseband link starts to configure RF module. First, clock distribution chip is configured to provide ADC and DAC module differential clock. Second, ADF4531 chips are configured to provide the local oscillators. Finally, A/D and D/A chips are configured to conduct ADC and DAC.

The sampling frequency of baseband link is 15.36MHz when the system bandwidth is 10MHz. However, the clock frequency of baseband link is up to 200MHz in order to decrease system delay. Therefore a FIFO which can work in different clock domains is needed for data buffer both in transmitter and receiver. In transmitter, before conducting IFFT resource mapping module will first check the data occupation of the FIFO to ensure that there is enough space for IFFT IP core to output 1104 or 1096 data in a row. In receiver, after removing cyclic prefix (CP), resource demapping module will check the data counter of the FIFO. If there are 1024 data stored in the FIFO, resource demapping module starts to input 1024 data in a row from the FIFO to conduct FFT.

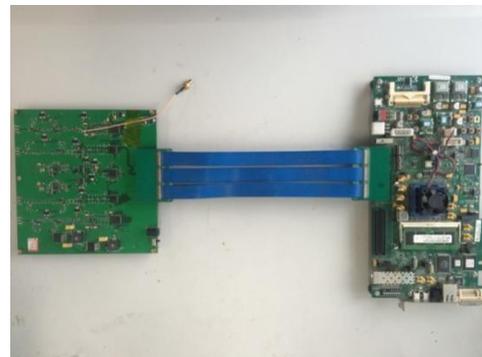


Figure 8. Connection of ML605 board and RF module

5.4 Hardware consumption

After synthesizing, the total hardware consumption of baseband link can be seen in ISE. As is shown in Table 3, the Virtex-6 FPGA still remains lots of logic resources for further development. The maximum clock frequency can reach 283.046MHz, meanwhile the maximum clock frequency ML605 can provide is only 200MHz. this means baseband link can run accurately in ML605 even if the maximum clock frequency is provided. Then we export the baseband link into XPS and connect it to PLB as an IP core. This IP core contains both transmitter and receiver.

Table 3. Hardware consumption

Logic utilization	Used	Utilization
Slice Registers	80090	26%
Slice LUTs	68345	45%
fully used LUT-FF pairs	51702	53%
Block RAM/FIFO	196	47%
BUFG/BUFGCTRLs	1	3%
DSP48E1s	212	28%

6 VIDEO TRANSMISSION TEST

We generate bit stream file in XPS, and download it to the Virtex-6 FPGA. Two sets of the equipment are fixed in the neighboring rooms during video transmission test. Figure 9 represents the real-time video transmission pictures on one display. One picture is captured by local camera, the other one is from receiver of baseband link. The test verifies that our bi-directional video transmission system works well.



Figure 9. Video transmission test

7 CONCLUSIONS

In this paper, a FPGA implement of video transmission system based on LTE has been presented. We have set up system platform on the Virtex-6 FPGA, programmed protocol stack in SDK, and designed baseband link in ISE. Test results obtained in indoor environment show that our system can conduct stable video transmission. In addition, the consumption of FPGA logic resource is not too much. Our future work is to apply more complex algorithms to our baseband link.

ACKNOWLEDGEMENTS

This research is funded by National Natural Science Foundation of China (No. 50975019). The authors would like to express thanks for the support.

REFERENCES

- [1] Bechir, N., Nasreddine, M., Messaoud, E. & Mahmoud, A. 2014. Performance evaluation of downlink LTE system. In *Advanced Technologies for Signal and Image Processing, 2014 1st International Conference on*. IEEE. pp: 492-496.
- [2] Sesia, S., Toufik, I. & Baker, M. 2009. *LTE: The UMTS Long Term Evolution*. New York: John Wiley & Sons.
- [3] Ghosh, A., Zhang, J., Andrews, J. G. & Muhamed, R. 2010. *Fundamentals of LTE*. Pearson Education.
- [4] E-UTRA and E-UTRAN Overall description; 3GPP Technical Specification TS 36.300 V9.6.0, Dec. 2010. [Online]. Available: <http://www.3gpp.org>.
- [5] Berkmann, J., Carbonelli, C., Dietrich, F., Drewes, C., & Xu, W. 2008. On 3G LTE terminal implementation-standard, algorithms, complexities and challenges. In *Wireless Communications and Mobile Computing Conference, 2008. IWCMC'08. International*. IEEE. pp: 970-975.
- [6] Mehlführer, C., Wrulich, M., Ikuno, J. C., Bosanska, D., & Rupp, M. 2009. Simulating the long term evolution physical layer. In *Proc. of the 17th European Signal Processing Conference (EUSIPCO 2009), Glasgow, Scotland, 27*: 124.
- [7] Hessel, S., Szczesny, D., Traboulsi, S., Bilgic, A., & Hausner, J. 2009. On the design of a suitable hardware platform for protocol stack processing in LTE terminals. In *Computational Science and Engineering, 2009. CSE'09. International Conference on*. IEEE. 2: 1-8.
- [8] Holmark, R., Johansson, A., & Kumar, S. 2004. On connecting cores to packet switched on-chip networks: A case study with microblaze processor cores. In *7th IEEE workshop DDECS, 4*.
- [9] Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and Modulation; 3GPP Technical Specification TS 36.211 V9.1.0, March. 2010. [Online]. Available:<http://www.3gpp.org>.
- [10] Evolved Universal Terrestrial Radio Access (E-UTRA); LTE Physical Layer; General Description; 3GPP Technical Specification TS 36.201 V9.1.0, March. 2010. [Online]. Available: <http://www.3gpp.org>.