OBSAI RP3-01 INTERFACE IMPLEMENTED ON FPGA

Cristian ANGHEL[1], Remus CACOVEANU[2]

Lucrarea prezintă pe scurt interfaţa OBSAI RP3-01 existentă între unitatea radio externă şi modulul în banda de bază ale unei staţii de bază cu arhitectură distribuită. Este propusă o implementare bazată pe niveluri pentru o staţie de bază cu arhitectură distribuită cu capabilităţi MIMO. Sunt furnizate rezultate ale implementării realizate pe un cip XC4VFX60.

Unei perechi Nivel Aplicaţie şi Nivel Transport îi corespund două perechi Nivel al Legăturii de Date şi Nivel Fizic. În acest mod, proprietăţile interfeţei asigură suportul implementării tehnicilor de diversitate spaţială şi de multiplexare spaţială, precum şi suportul pentru redundanţă.

Abstract—The paper presents briefly the OBSAI RP3-01 interface existing between the Remote Radio Unit (RRU) and the Base band module (BBM) of a Base Station (BS) with a split architecture. It proposes a layer based implementation model for BS split architecture with MIMO capabilities and it presents some implementation results obtained when targeting the XC4VFX60 device.

One pair Application Layer and Transport Layer is used for two pairs Data Link Layer and Physical Layer. This way the interface features are suitable for spatial diversity and spatial multiplexing schemes and redundancy support.

Keywords: OBSAI RP3-01 Interface, base station, MIMO, FPGA

1. Introduction

More and more companies try to provide full solutions when it comes to wireless telecommunications systems. But unfortunately this concept, called ecosystem, is not always an easy task to realize. The costs are quite large and, due to system high complexity, the development time is also very long. That means it is very possible that parts of a system can be made by different vendors. The interconnection between different parts should be made by standard interfaces. The usage of third party intellectual property solution reduces the compatibility area. This is the reason why standards as Common Public Radio Interface (CPRI) [1] and Open Base Station Architecture Initiative (OBSAI) were developed. The OBSAI RP3-01 interface permits the transport of data corresponding to different

[1] PhD Assist., Department of Telecommunications, ETTI, University POLITEHNICA of Bucharest, Romania, e-mail: cristian.anghel@upb.ro
[2] Reader, Department of Telecommunications, ETTI, University POLITEHNICA of Bucharest, Romania

communications standards, such as WCDMA, GSM/EDGE, CDMA2000 and 802.16.

The OBSAI RP3-01 interface [2] represents an extension of the Reference Point 3 protocol for remote radio unit use. The BS can support multiple RRUs connected in chain, ring and tree-and-branch topologies, which makes the interface very flexible. Also, in order to minimize the number of connections to RRUs, the RP1 management plan [3], which includes Ethernet and frame clock bursts, is mapped into RP3 messages. This solution is an alternative to the design in which the radio module collocates with the BBM. Although in such a case the interface between the radio unit and the BBM becomes less complex, the transmitter power should be increased in order to compensate the feeder loss.

This paper is organized as following: Section II describes the RP3-01 protocol stack and the corresponding parameters; Section III presents the RP3-01 synchronization procedure; Section IV describes the proposed BS split architecture and the implementation solutions for the most important interface layers components. It also provides implementation results obtained when targeting the FPGA (Field Programmable Gate Array) XC4VFX60. Finally, Section V presents the conclusions and the future perspective of the study.

2. RP3-01 Protocol Stack

The RP3-01 interface is a high speed serial interface for both up link and down link data and control transfer. The protocol stack is based on a packet concept using a layered protocol with fixed length messages.

2.1 Physical Layer

The transmitter Physical Layer is responsible for the 8b10b encoding, which provides a mechanism for clock recovery, and data serialization. At receiver, the mirrored functions are applied. The Physical Layer can be implemented by a dedicated device, such as an XGMII transceiver [4], or by an internal FPGA component, such as RocketIO transceiver from Xilinx Virtex 5 family [5], when a hardware implementation is considered. The supported rates are 768 Mbps, 1536 Mbps, 3072 Mbps and 6144Mbps. The 6144 Mbps rate does not concern this study.

2.2 Data Link Layer

The Data Link Layer contains the frame builder and the link synchronization unit. The frame builder receives from superior layers the data and control messages and generates, according to the transmitter rate, the RP3 frame. The data or control message has a fixed 19 bytes length. The message format contains 4 fields. The first one is the Address field on 13 bits, the second one is

the Type field on 5 bits, the third one is the Time Stamp field on 6 bits and the last one is the Payload field on 128 bits.

The duration of RP3-01 Master Frame (MF) is fixed to 10 ms. This length corresponds to i x N_MG Master Groups (MG), where i is selected accordingly to the transfer rate, i.e. 1 for 768 Mbps, 2 for 1536 Mbps and 4 for 3072 Mbps. A MG consists of M_MG messages and K_MG idle bytes (special characters). Fig. 1 presents an example of MF corresponding to 802.16 standard [6], when 768 Mbps rate is used. Also one can observe the values of the specified parameters.

For the example described in Fig. 1, the transfer rate can be computed as follows: first we compute the maximum number of bytes per frame, and then, having in mind the 8b10b line encoder and the MF duration, we calculate the transfer rate, as in (1). In order to generate the MF, the frame builder uses two counters: for data messages and for control messages.

$$Rate = i \cdot N\_MG(M\_MG \cdot 19 + K\_MG \cdot 1) \cdot 1000 \quad (1)$$

The link synchronization unit contains transmit and receive Finite State Machine (FSM). Both the transmitter and receiver FSMs contribute to physical and logical link synchronization. The physical synchronization is based on special characters, K28.5 and K28.7, which mark the end of message groups and MF, while the logical synchronization is based on fixed MF structure.

2.3 Transport Layer

The Transport Layer is responsible for the end-to-end delivery of the messages, which could be simply routing of messages. The routing is based on the first 13 bits of the message, which represent the address field. The first 8 bits of the address represent the Node address and the other 5 bits represent the Sub-node address. These fields are used in a hierarchical addressing scheme where the first field identifies the bus nod and the second one selects the corresponding module from the device. The message routing is made based on a Routing Table which

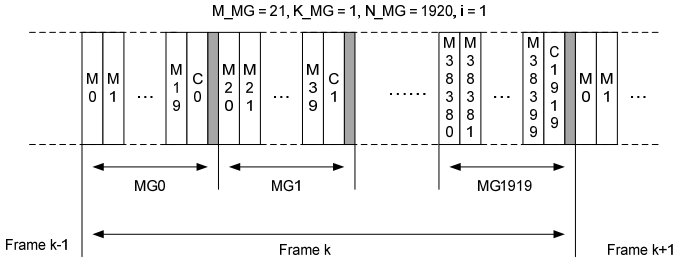

Fig. 1 MF for 802.16 air interface standard for the 768Mbps line rate

indicates the correspondence between the used addresses and the node ports. The table content is defined by the initial configuration procedure of the interface.

Another block of the Transport Layer is the Message Multiplexer/ Demultiplexer. It performs time interleaving/ deinterleaving of messages from N RP3 input links into one RP3 output link. Several multiplexing/ demultiplexing tables are defined as functions of the number of input links and their corresponding rates and the rate of the output link.

2.4 Application Layer

The Application Layer builds the data messages. It maps data and control information into the message payload and attaches the message header. The payload is represented by concatenation of signal samples in the baseband. For the case of the 802.16 air interface, the format of a data message payload field is shown in Fig. 2.

All transfers over the bus are performed over paths. A path represents a connection between a source node and a destination node. The connection is made by a set of bus links defined by the routing tables. Physically, a path consists of a set of message slots per MF. Paths are defined before bus initialization and they remain fixed during operation, i.e. the transfer between two nodes is made on the same bus links using the same message slots. For each path, a message transmission rule is applied. There are two types of rules: mandatory low level rules, using modulo computation over message slot counters and optionally high level rules, using the Bit Map (BM) concept. In the second case each MG is formed as groups of messages from base band channels. Each group may contain Ethernet frames as well.

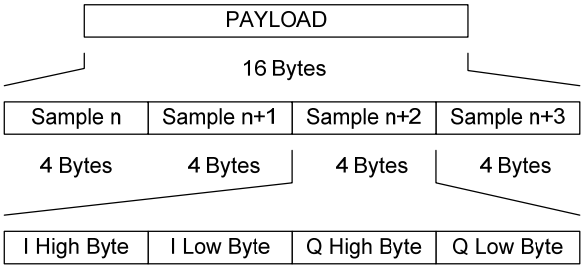

Fig. 2 802.16 payload mapping

3. RP3-01 synchronization procedure

In frame based communications systems, appropriate frequency/ time synchronization is critically important. In order to avoid inter-cell interference, all base stations must use the same frequency/ time reference. The transmission over

the radio channel, both on downlink and uplink, should be synchronized with the same reference. The RP1 synchronization burst generator, located at Clock Control Manager (CCM) level, shall provide frame timing and time stamping for each of the air interface standard independently, based on that reference. The PR1 Frame Clock Bursts (FCB) is mapped into RP3 messages.

The synchronization algorithm uses several inputs and is realized based on information collected both at BBM and RRU. The first one is the propagation delay (PD) between BBM and RRU. This time interval is measured on BBM side using special message transmission called Round Trip Time (RTT) measurement message. The second input is the receiving time of the RP1 FCB. This interval is measured from the beginning of the last MF until the last bit of the FCB using a counter called *C1*. After measuring this time interval, the Frame Clock Synchronization (FCS) message is generated. FCS contains information from FCB and the C1 value. The last input is the detection time of a FCS inside a MF. This time interval is measured by RRU using a counter called *C2*. Having all this information, RRU computes the buffering time (*Brru*). Fig. 3 describes such an example. The time intervals are not at real scale. They are expressed in Time Units (TU). One can observe that at RRU side, the end of recovered FCB corresponds to the beginning of RF(k+4).

Using the formulas from [2], we obtain the *Bruu* as in (2), where k equals to 2 from some computing conditions.

$$Brru = k \cdot RFd + (C1 - PD) - FCBd = 11\text{TU} \quad (2)$$

4. Proposed implementation scheme

4.1 BS split architecture

The proposed BS split architecture is presented in Fig. 4. A BBM is connected to the two RRUs in order to have multiple transmit/ receive antennas for MIMO capabilities [7]. The connection between the two RRUs is realized using a chain topology. In order to obtain a single point failure redundancy scheme, a second BBM connected to the two RRUs is required. Only one BBM will be active at the time. Fig. 4 depicts also the OBSAI RP3-01 Interfaces required for blocks interconnection. Note that on OBSAI RP3-01 interface of each RRU the same Transport and Application Layers serves the both Physical and Data Link Layers.

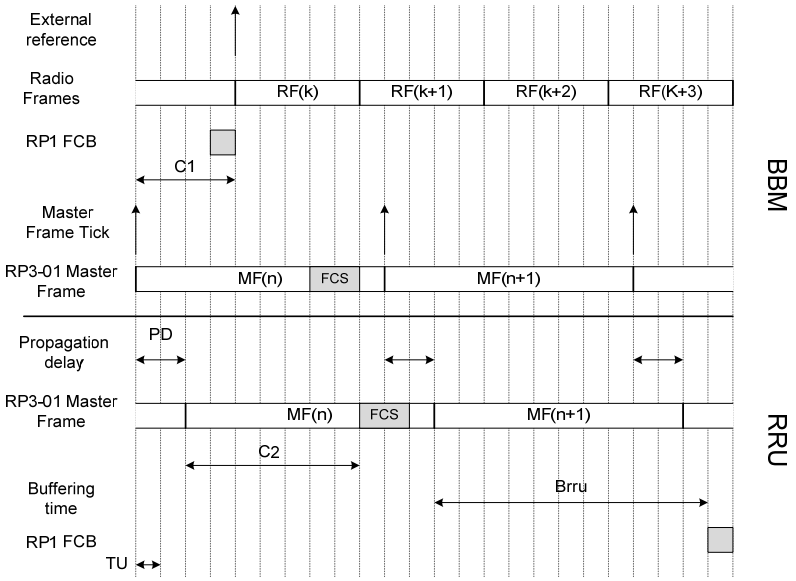

Fig. 3 Timing principle in RP1 frame clock burst transfer

The RP3-01 connections between each BBM and RRU or between RRUs are bidirectional. On downlink (DL) direction (from BBM to RRU), the data stream from a BBM can contain multiple data streams interleaved/ multiplexed for the two Transceivers (e. g. in order to provide Space Time Coding or MIMO) or can contain data streams only for one Transceiver. By selecting the right node/ sub-node address, the Application Layer from BBM OBSAI RP3-01 Interface selects the desired RRU. The Transport Layer from RRU1 OBSAI RP3-01 Interface directs the data streams to its own Application Layer when RRU1's address is used, otherwise forwards the data streams to the second Data Link Layer from the OBSAI RP3-01 Interface. In uplink (UL) direction (from RRU to BBM) the procedure is similar. Both receivers can be used (e. g. receive diversity or collaborative MIMO) or only one receiver can be active. In addition to these data and control streams that should be treated by the OBSAI RP3-01 Interface as RP3 streams, an Ethernet stream will be also transmitted between BBM and RRUs in order to connect the corresponding Control & Management (CM) modules to RRU. This stream should be treated by the OBSAI RP3-01 Interface as RP1 stream.

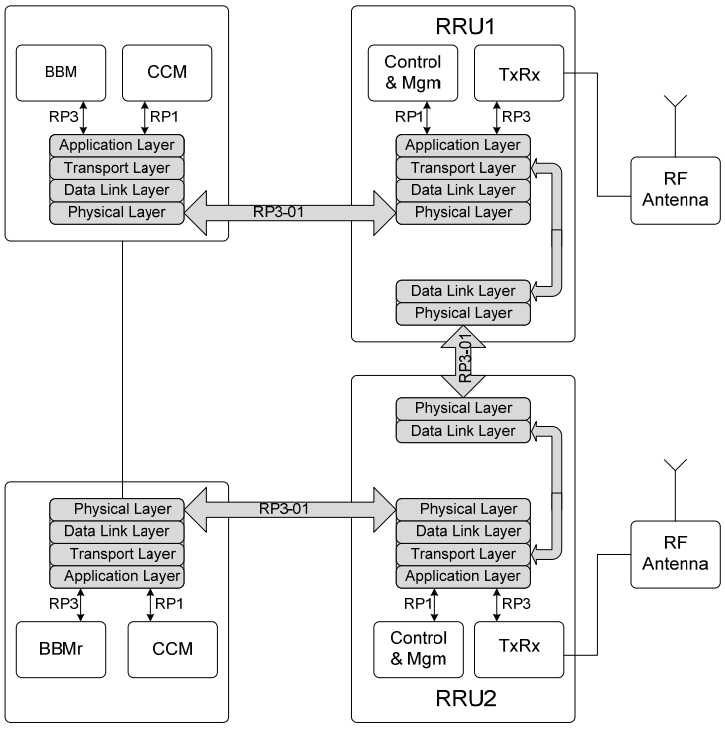

Fig. 4. Proposed BS split architecture with interface protocol stack

4.2 BBM and RRU interface nodes

Fig. 5 presents the generation of RP3-01 stream at BBM side, while Fig. 6 corresponds for RRU1 side. Each RP3-01 node has its own address known from the initial configuration procedure. The DL node addresses is represented with normal fonts and with bold italic fonts the UL node addresses. For the proposed system architecture, in DL direction the RP3-01 link is used to connect a source node with two destination nodes, so two paths exist over DL connection. Even if only one RRU transmitter is used at the time, the two paths will exist and the Transport Layer from BBM OBSAI RP3-01 Interface will place its messages over the message slots corresponding to selected RRU (consider twice the rate on link between BBM and RRU1 comparing with the one between RRU1 and RRU2). In the downlink direction, a point-to-point message transfer is applied, while in uplink direction, the same message is multicast to the two BBMs. Only the active BBM uses the received information.

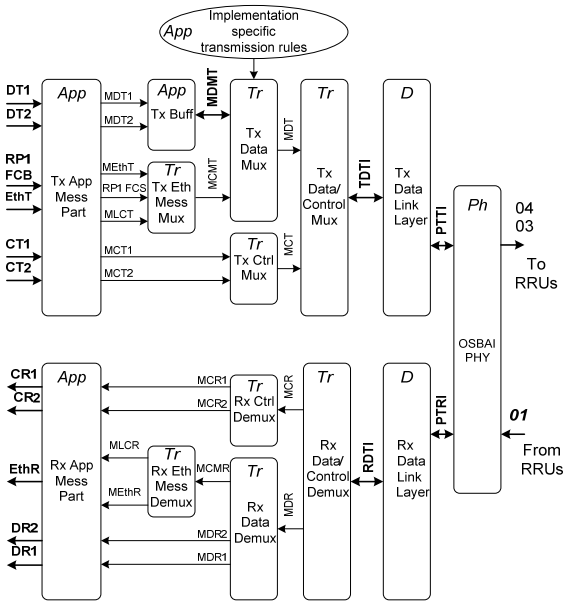

Fig. 5 Block scheme for BBM RP3-01 interface

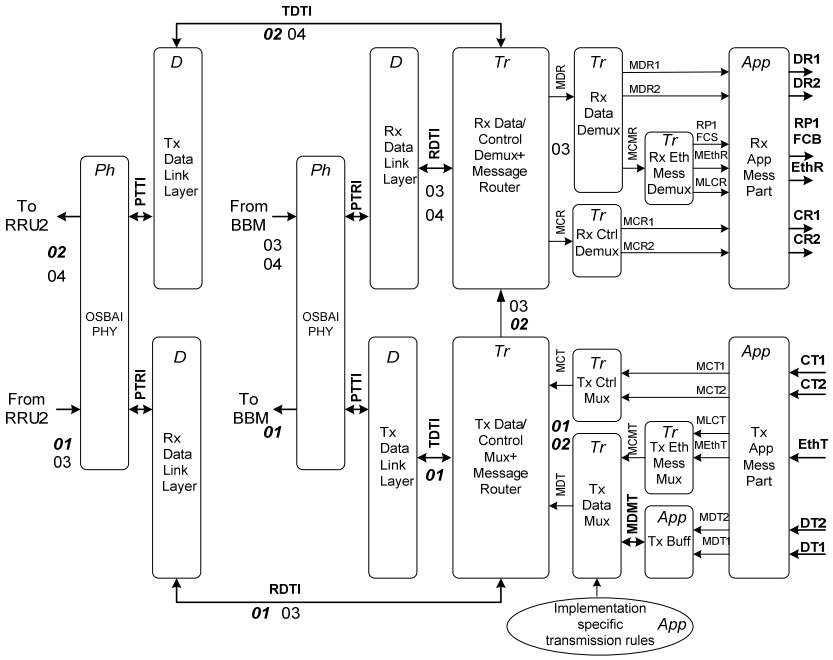

Fig. 6 Block scheme for RRU1 RP3-01 interface

Each path will be considered having a data path and a control path. Bus manager will provide separate message transmission rules for the paths utilizing data and control message slots. We will explain the notations used in Fig. 5 and Fig. 6 and we will describe the steps made for RP3-01 interface generation.

In DL direction, on BBM side, the Application Layer receives two data streams for the two RRUs, DT1 and DT2 and two control streams, CT1 and CT2. Also, an Ethernet stream for management called EthT and the RP1 FCBs are received. The Application Layer generates the corresponding messages steams, i.e. MDT1, MDT2, MCT1, MCT2, MEthT and RP1 FCS. Also specific RP3-01 link control messages are generated. This stream is called MLCT. Beside the message generator function, Application Layer is responsible for buffering the data paths. A buffer is required for each 802.16 signal (antenna-carrier) in order to compensate the jitter caused by message transmission. Finally, the Application Layer has to provide to Transport Layer the implementation specific message transmission rules. These rules could include the lower layer message transmission rules and/ or extra rules for mapping the RP1 traffic to RP3 data message slots.

The Transport Layer has four blocks with interleaving/ multiplexing function. First, the management messages, including RP1 FCS, MLCT and MEthT are put on the same flow called MCMT, based on a priority list. The two control messages streams MCT1 and MCT2 are multiplexed in the MCT flow. The data flow, called MDT is obtained by interleaving/ multiplexing the data messages from buffers and the management messages from MCMT stream. Finally, the Transport Layer multiplexes the MDT and the MCT streams based on the TDTI interface with the Transport Layer. Using this interface, the frame builder from Transport Layer requires data or control messages and increments the corresponding counters for each successful transfer. The generated RP3-01 frame, including also the special characters, is transferred to Physical Layer on PTTI interface. The DL continues on RRU side with the receiving chain form Fig. 6. The Transport Layer works out the RP3-01 flow, first on data stream MDR and control stream MCR, and then the data path is split into data streams MDR1 and MDR2, respectively management messages RP1 FCS, MEthR and MLCR, while the control path is split into MCR1 stream and MCR2 stream. The Application Layer receives all these messages and extracts the corresponding payload. One interesting observation is that at RRU, the Transport Layer has also the message router function, as one can see from Fig. 6.

In UL direction the procedure follows the same steps as the one described for DL.

4.3 Implemented diagrams

Fig. 7 depicts the transmission part of the interface, while Fig. 8 describes the receiving one. One can observe:

- The Synchronization modules. Based on a Finite State Machine (FSM) they provide physical and logical synchronization
- The Frame Builder and The Frame Unpacking. They implement the data and the control counters. These modules are connected to the Transport Layer through data and control buffers.
- The Switched Memories. These blocks are used for implementing the message multiplexing/ demultiplexing procedures. There are used two BRAMs with 19x4 locations on 8 bits. 4 messages can be stored in such a memory. On transmission side, the bytes from the input line *i*, where $i \in \{0,1,2,3\}$ is the number of valid input lines, are written to locations with address $19i + k, k \in \{0..18\}$. When a message was stored for each valid input line, the writing procedure starts on the second memory. In the same time the old memory is read using a counter from (*i*-1)19 to (19*i*-1) for each line. On receiving side the procedure is similar.
- The buffer zone between the Application Layer and the Transport Layer. It provides a simple interface that can be used when integrating the interface module into the user design.

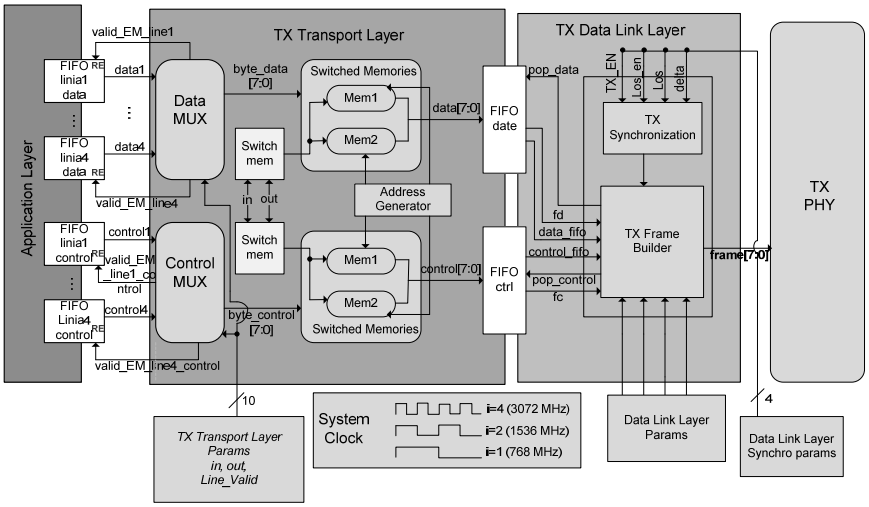

Fig. 7 Implemented TX chain

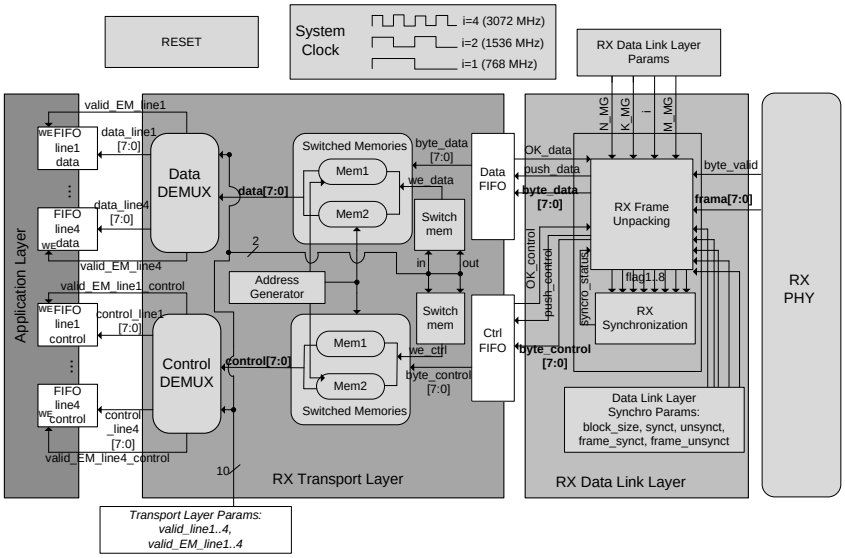

Fig. 8 Implemented RX chain

4.4 Implementation results

For implementation we considered a XC4VFX60 device from Xilinx Virtex 5 family. The functional tests were made using ModelSIM 6.2g and the synthesis results were obtained using Xilinx ISE 9.2i. From the proposed architecture depicted in Fig. 4 one can see that the RRU Interface contains two Data Link and two Physical Layers. The implementation results obtained for the Data Link Layer are critical for the global resources, while the Physical Layer implementation cost reflects in the number of used RocketIO Transceivers.

The area and speed reports are presented in Table 1. They can be compared with the ones presented in [8, 9].

Table 1

Area and speed results

Component	No. LUTs (max 28800)	No. FFs (max 28800)	No. Slices (from 7200)	No. BRAMs (from 96 of 18Kb)	Speed (MHz)
TX Transport Layer	494 (1.72%)	312 (1.08%)	192 (2.67%)	14 (14.6%)	407
TX Data Link Layer					194
RX Transport Layer	750 (2.60%)	388 (1.35%)	317 (4.4%)	14 (14.6%)	407
RX Data Link Layer					185
TOTAL	1244(4.32%)	700(2.43%)	509(7.07%)	28(29.16%)	185

5. Conclusions

This paper presented an overview of an OBSAI RP3-01 Interface implementation. There were described the main functions of the interface layers and there were presented interface block schemes for both BBM and RRU sides based on the protocol stack. Some examples were made for 802.16 air interface standard without reducing the generality of presentation. The authors proposed a base station split architecture, with support for redundancy and multiple transmit and receive antennas. The implementation solutions were proposed and presented for Transport Layer and Data Link Layer. Simple interfaces for connecting the implemented modules with the Physical Layer and the Application Layer were defined. It was shown, based on the reduced number of occupied resources and on the set of supported features, that this implementation can represent a solution for the real world telecommunication systems.

R E F E R E N C E S

[1] "CPRI Specification v4.0", June 2008, http://www.cpri.info/downloads/CPRI_v_4_0_2008-06-30.pdf
[2] "OBSAI Reference Point 3 Specification", version 4.1, July 2008 http://www.obsai.com/obsai/documents/public_documents/download_specifications
[3] "OBSAI Reference Point 1 Specification", version 1.0, October 2003
[4] "PMC Sierra PM8358 QuadPHY 10GX" datasheet, March 2005 http://www.pmc-sierra.com/products/details/pm8358/
[5] "Xilinx Virtex 5 FPGA RocketIO GTP Transceiver", user guide, June 2009 http://www.xilinx.com/support/documentation/user_guides/ug196.pdf
[6] IEEE Standard 802.16e, 2005, "Broadband Wireless Metropolitan Area Network"
[7] *Ezio Biglieri, Robert Calderbank, Anthony Constantinides, Andrea Goldsmith, Arogyaswami Paulraj* and *H. Vincent Poor*, "MIMO wireless communications", Cambridge University Press, 2007
[8] *Christian F. Lanzani*, "OBSAI RP3-01 6.144 Gbps Interface Implementation", www.obsai.org/obsai/content
[9] "Xilinx OBSAI v3.1" datasheet, April 2009 http://china.xilinx.com/support/documentation/ip_documentation/obsai_ds612.pdf