Project : contract n° 20041/06/NL/CP

Europeanisation of Mil-STD-1553B Data Bus Products

Task 1 – Digital Remote Terminal ASIC

RT53eur IP Core User Manual

	Name and Function	Date	Signature
Prepared by	**Marc Souyri** *ASIC expert*	12 Nov. 2009	
Verified by	**Arnaud Wagner** *1553B Expert*	12 Nov. 2009	A Wagner
Authorized by	**Nicolas BEHOT** *1553 product Manager*		

Document type	**N° WBS**	**Key words :**

SUMMARY

This document is written in the frame of ESA R&D AO/1-5052/06/NL/CP Europeanisation of "MIL STD 1553B Data Bus".

It is part of task 1 related to the development of the RT53eur IP

This document is the User Manual of RT53eur IP that is a MIL STD 1553B Remote Terminal

DOCUMENT CHANGE LOG

Issue/ Revision	Date	Modified pages	Observations
00/00	29 Oct 2009		
00/01	2 Dec 2009	all	new version of the synchronous IP model IP version is 1.2 – add a source version chapter – add Xilinx synthesis results

PAGE ISSUE RECORD

Issue of this document comprises the following pages at the issue shown

Page	Issue/ Rev.	Page	Issue/ Rev.	Page	Issue/ Rev.	Page	Issue/ Rev.	Page	Issue/ Rev.	Page	Issue/ Rev.
All	00.00										

TABLE OF CONTENTS

1. INTRODUCTION

This document is written in the frame of ESA R&D AO/1-5052/06/NL/CP Europeanization of "MIL STD 1553B Data Bus". It is part of task 1 related to the development of a Digital RT IP performing a 1553B Remote Terminal. This document is the User Manual of the IP core.

The IP is called RT53eur .

- RT stands for Remote Terminal
- 53 is a reminder for 1553B Data Bus
- EUR reminds that the IP has been developed within the " Europeanisation of Mil-STD-1553B Data Bus Products". Moreover the IP core includes some of the new features defined within the ECSS 5013 "MIL STD 1553 B extension" working group.

This document describes the usage of the IP database. The functionality of the IP is described in [AD10].

The version of the IP tagged in CVS is V1.2

2. DOCUMENTS, ACRONYMS AND CONVENTIONS

2.1 APPLICABLE DOCUMENTS

AD-2:	ECSS Q60-02: Final Draft ASIC/FPGA Development Standard, http://www.estec.esa.nl/microelectronics/asic/Final-Draft-ECSS-Q-60-02.pdf
AD-6	MIL-STD-1553B, Notice 1-4, 15 January 1996 "Digital Time Division Command/Response Multiplex Data Bus".
AD-7	MIL-HDBK-1553A, "Multiplex applications Handbook", 1 Nov 1988 http://ams.aeroflex.com/ProductFiles/AppNotes/milhbk1553a.pdf
AD-8	SAE AS4112, January 1989 "Production Test Plan For the Digital Time Division Command/Response Multiplex Data Bus Remote Terminals
AD-9:	SAE AS4111, Issue 1998-10 "Validation Test Plan For the Digital Time Division Command/Response Multiplex Data Bus Remote Terminals
AD-10	RT53eur IP core data sheet – Astrium – reference R&D.E53.RP.01192.ASTR
AD-14	ASP54 Transceiver ASIC Specification - Astrium – R&D.E53.NT.00461.V.ASTR

2.2 ACRONYMS

AD	Applicable Document
ASIC	Application Specific Integrated Circuit
ASSP	Application Specific Standard Product
CDR	Critical Design Review
CPPA	Central Part Procurement Agency
CPU	Central Processor Unit
DDR	Detailed Design Review
DFF	D-Type Flip Flop
DRC	Design Rule Check
DSP	Digital Signal Processor
EDAC	Error Detection And Correction
ESA	European Space Agency
ESTEC	European Space Research and Technology Centre
FDIR	Failure Detection Isolation and Recovery
FPGA	Field Programmable Gate Array
GEO	Geosynchronous Equatorial Orbit
I/O	Input/Output
ID	Identification
IDR	Initial Design Review = Specification Requirement Review
IEEE	Institute of Electrical and Electronics Engineers
IP	Intellectual Property
ITT	Invitation To Tender
JTAG	Joint Test Action Group (refer to IEEE std 1149.1)
LEO	Low Earth Orbit
LET	Linear Energy Transfer
LVS	Layout Versus Schematic
PCB	Printed Circuit Board
PDF	Portable Document Format
PDR	Preliminary Design Review
PID	Process Identification Document
PRT	Packet Remote Terminal
RD	Reference Document
RT	1533 Remote Terminal
SCC	Space Components Co-ordination group
SCoC	Spacecraft Controller on a Chip
SEE	Single Event Effect (or SEP Single Event Phenomena)
SEL	Single Event Latch up
SEP	see SEE
SET	Single Event Transient
SEU	Single Event Upset
SOC	System-On-a-Chip
SRAM	Static Random Access Memory
SRR	Specification Requirement Review = Initial Design Review
SRT	Simple Remote Terminal
TC	TeleCommand
TID	Total Integrated Dose
TM	TeleMetry
TRP	Technological Research Programme
VHDL	VHSIC Hardware Description Language, IEEE standard 1076
VLSI	Very Large Scale Integration (component)
VTP	Validation Test Plan (Standard 1553B Test Plan for Remote Terminal)
WP	Work Package
WWW	World Wide Web

2.3 DEFINITION OF TERMS

Bus Controller (BC)	The Bus Controller's task is to initiate and control all data transfer on the Mil Std 1553B Data Bus. It is the sole device allowed to transmit Command Words.
Remote terminal (RT)	The Remote Terminal is used to interface the subsystems to the Mil Std 1553B Data Bus system and perform data transmissions on the Mil Std 1553B Data Bus as controlled by the Bus Controller.
Bus Monitor	The Bus Monitor's task is to listen to the Mil Std 1553B Data Bus traffic and to extract selected information to be used at a later time.
Mil Std 1553 Data Bus	All the hardware including cables, isolation resistors, transformers etc. required to provide a data path between the Bus Controller and all the associated Remote Terminals.
Message	A single Message is the transmission of a Command Word, Status and Data Words (if they are specified).
Mode Command	Command Word in which the Subaddress Field is set to '00000' or ´11111´and a Mode Code is transferred from the Bus Controller to the Remote Terminal.
Receive Command	Message with a data flow from the Bus Controller to the Remote Terminal.
Transmit Command	Command Word in which the Subaddress Field is set different to '00000' or ´11111´and the subsequent transfer of Data Word(s) from the Remote Terminal to he Bus Controller is initiated.
Transmit Message	Message with a data flow from the Remote Terminal to the Bus Controller.
Vector Word	Data Word provided by the Remote Terminal subsequent to the Status Word in answer to a Mode Command Transmit Vector Word.
Bit Order	The most significant bit shall be transmitted first with the less significant bits following in descending order of value in the Data Word.
Redundant Bus	The redundant data bus implements a particular approach for obtaining multiple data paths to improve message arrival probability.
Bus Switching	When an RT is receiving or operating on a message on one bus, and another valid, legal command to the RT occurs on the opposite bus later in time.
Superseeding Command	When an RT is processing a command, a second valid command word sent to an RT shall take precedence over the previous command. After the minimum intermessage gap time has been exceeded, the RT shall respond to the second valid command when it is not transmitting on that data bus.

2.4 CONVENTIONS

2.4.1 BIT numbering

Bit n° 0 refers to LEAST Significant Bit (LSB)
Bit n° 15 refers to MOST Significant Bit (MSB)
For 1553 interface, as given in RD4, the Bit 15 (MSB) in registers corresponds to the 4th bit of a 1553 word.

2.4.2 DATA entity

“**Word**” refers to 16 Bits data entity
“**Byte**” refers to 8 Bits data entity
“BITi..BIT0” indicates value binary coded in BITi TO BIT0, BITi being the most significant bit

“0XNIBi..NIB0” indicates value hexadecimally coded in NIBi TO NIB0, NIBi being the most significant nibble

For instance, ‘1111000010101010’ and 0xF0AA indicates the same word value.

2.4.3 BIT definition

A bit in a register is **set** when its value is 1
reset when its value is 0

2.4.4 Word numbering in the 1553 frame

The first data word transmitted in a 1553B message shall be numbered as DW 1, the second transmitted shall be numbered as DW 2 and so on.

3. INSTALLING THE IP MODEL

3.1 INSTALLATION OF THE DELIVERY

The IP database is provided as a single file that has been generated on a unix workstation with a tar command followed by a gzip command.

The user must create a specific directory on unix and then type the following commands:

```
> gunzip delivery.gz
```

```
> tar xvf delivery
```

The tar command should create the directories described in the next paragraph.

The database can be simulated using Modelsim, and synthetized using Synplify for FPGA and Synopsys DC compiler for ASIC. Other tools could be used but no command files are provided.

The user shall then modify the modelsim.ini file of the simenv1 directory according to his CAD environment for :

std = standard library

ieee = IEEE library

synopsys = Synopsys libraries – std_arith std_unsigned and std_signed

std_developerskit = Mentor Std_developerskit provided with Modelsim

synplify = Synplify package containing synplify attributes

unisim = unisim Xilinx package

simprim = simprim Virtex package

modelsim_lib = Mentor modelsim_lib (that contains signal_spy etc…)

axcelerator = compiled ACTEL AX library

xilinxcorelib = compiled XILINX corelib library

The user has to find the std_developerskit, the Actel and the Xilinx libraries that cannot be distributed with the IP.

If the user wants to synthetize the IP using an ATMEL libray as ATC18RHA, then the .synopsys_dc_setup file located in synop_atc18rha has to be modified to take into account the location of the ATMEL libraries in the user environment.

If the installation is correctly made, typing ../bin/MAKE in the simenv1 directory should execute correctly the compilation of the IP model.

3.2 STRUCTURE OF THE DELIVERY

The delivery contains the following files:

- The bin directory containing procedures for running simulations.
- The source directory contains the VHDL files for the top level, the VHDL models for the memories and the netlists for the IP core.
- The tbe directory contains the source code of the testbench of the IP.
- The scripte directory contains the scripts read by the testbench of the IP.
- The simenv1 directory is used for simulating the IP.
- The gen_lib directory contains the source code of the generic library used to generate the generic netlist.
- The synplify_ax2000 provides an example of synthesis of the IP in ACTEL AX technology
- The synplify_virtex provides an example of synthesis of the IP in Xilinx Virtex technology
- The synop_atc18rha provides an example of the synthesis of the IP in ATMEL ATC18RHA technology.
- The gen_ram_actel is provided for information, it is the result of the generation of the ACTEL AX ram by smartgen.
- The gen_ram_xilinx is provided for information, it is the result of the generation of the XILINX ram by coregen.

4. DESCRIPTION OF RT53EUR IP

4.1 ENVIRONMENT

The RT53eur IP is a 1553B Remote Terminal. When used in nominal and redundant mode, it is connected to the 1553B bus with the following elements, as depicted in Figure 1:

- Two single 1553B Transceivers, or a dual 1553B Transceiver
- Two 1553B Transformers
- A stable oscillator that delivers the IP clock frequency

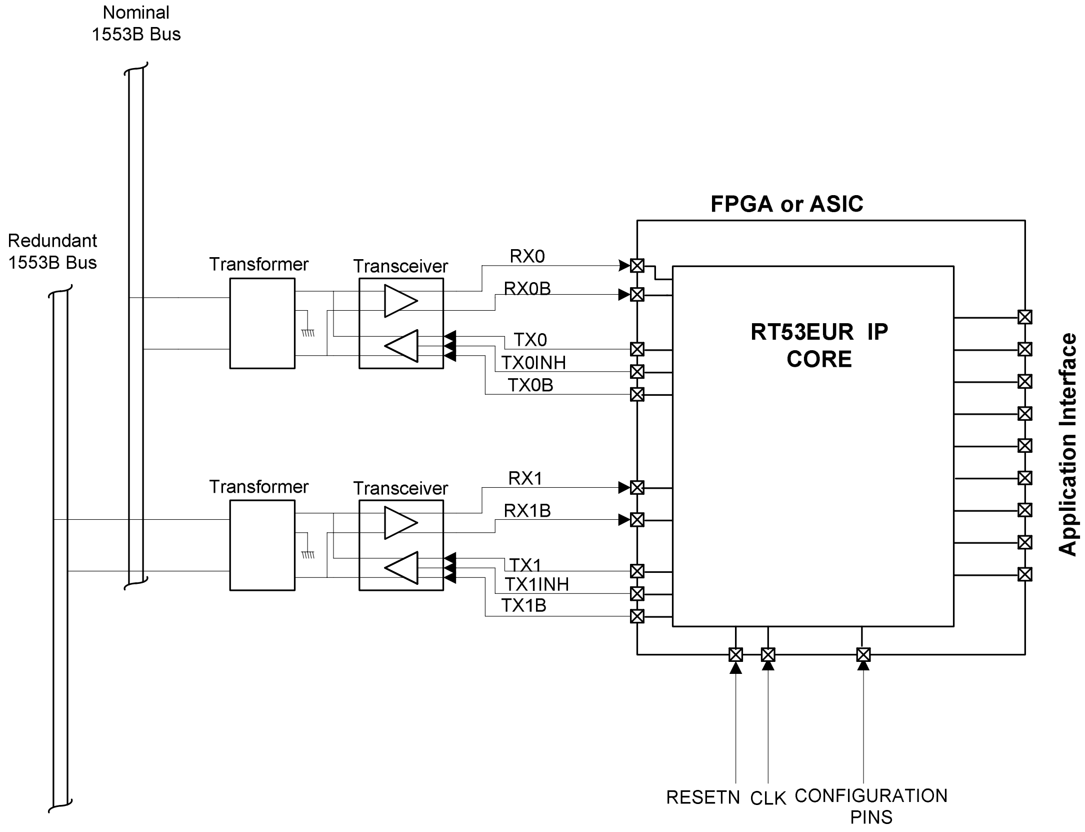

Figure 1 : RT53eur IP functional description

The RT53eur IP block diagram is depicted in **Erreur ! Source du renvoi introuvable.**. It can be functionally divided in 2 parts: the 1553B interface and the Application Interface. The RT53eur IP contains the following functions:

- A nominal and a redundant Manchester decoder
- A nominal and a redundant Manchester encoder
- A redundancy management function
- A protocol management function
- An application management function

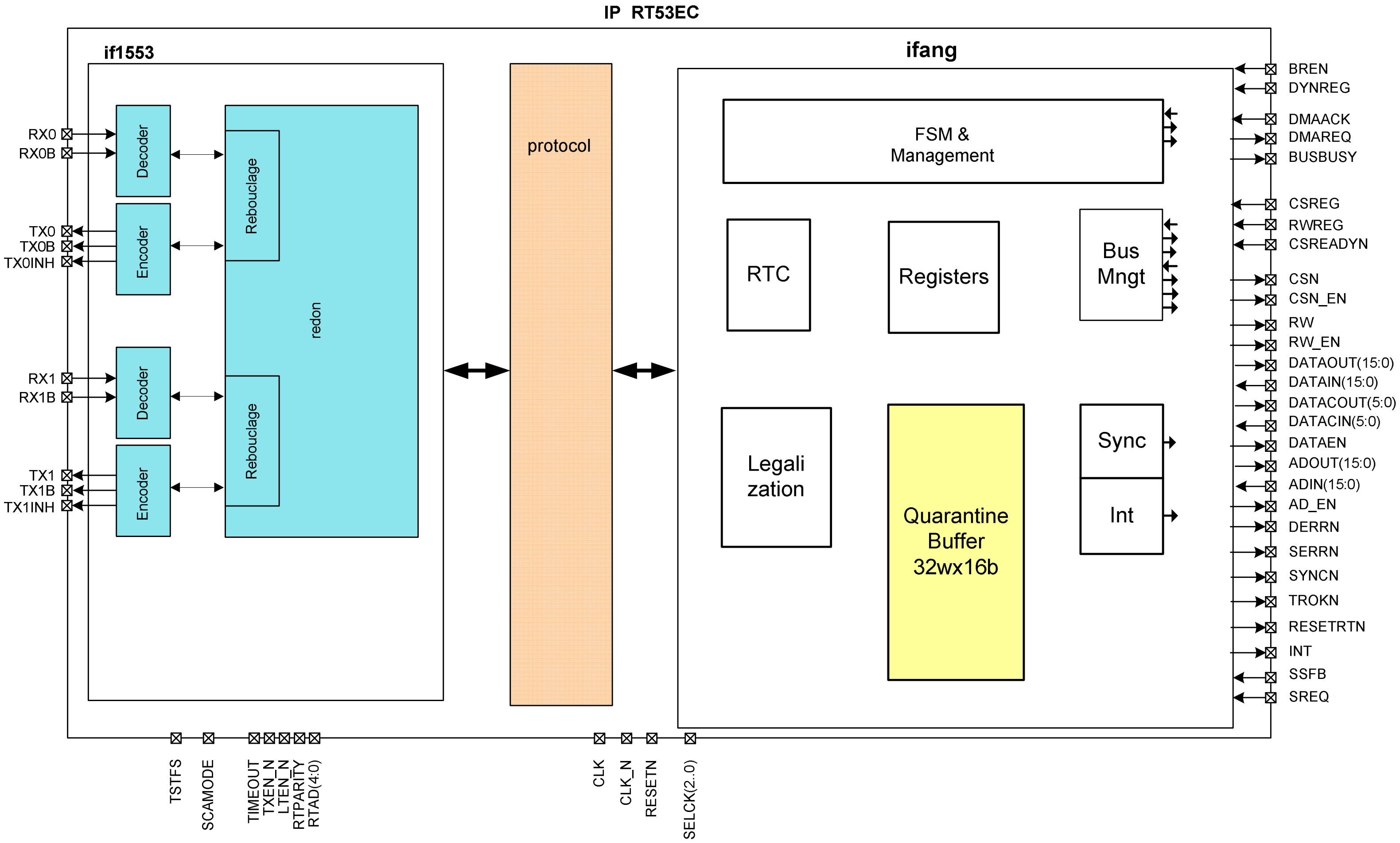

Figure 2 : RT53eur IP block diagram

4.2 THE IP MODEL DESCRIPTION

The IP model structure is depicted in Figure 3.

The user has to define the RT53eur core by selecting the following options:

- The RT53eur core that can have a synchronous or an asynchronous reset
- The quarantine buffer. which can be made with flip-flops, ACTEL AX memories or XILINX Virtex memories.

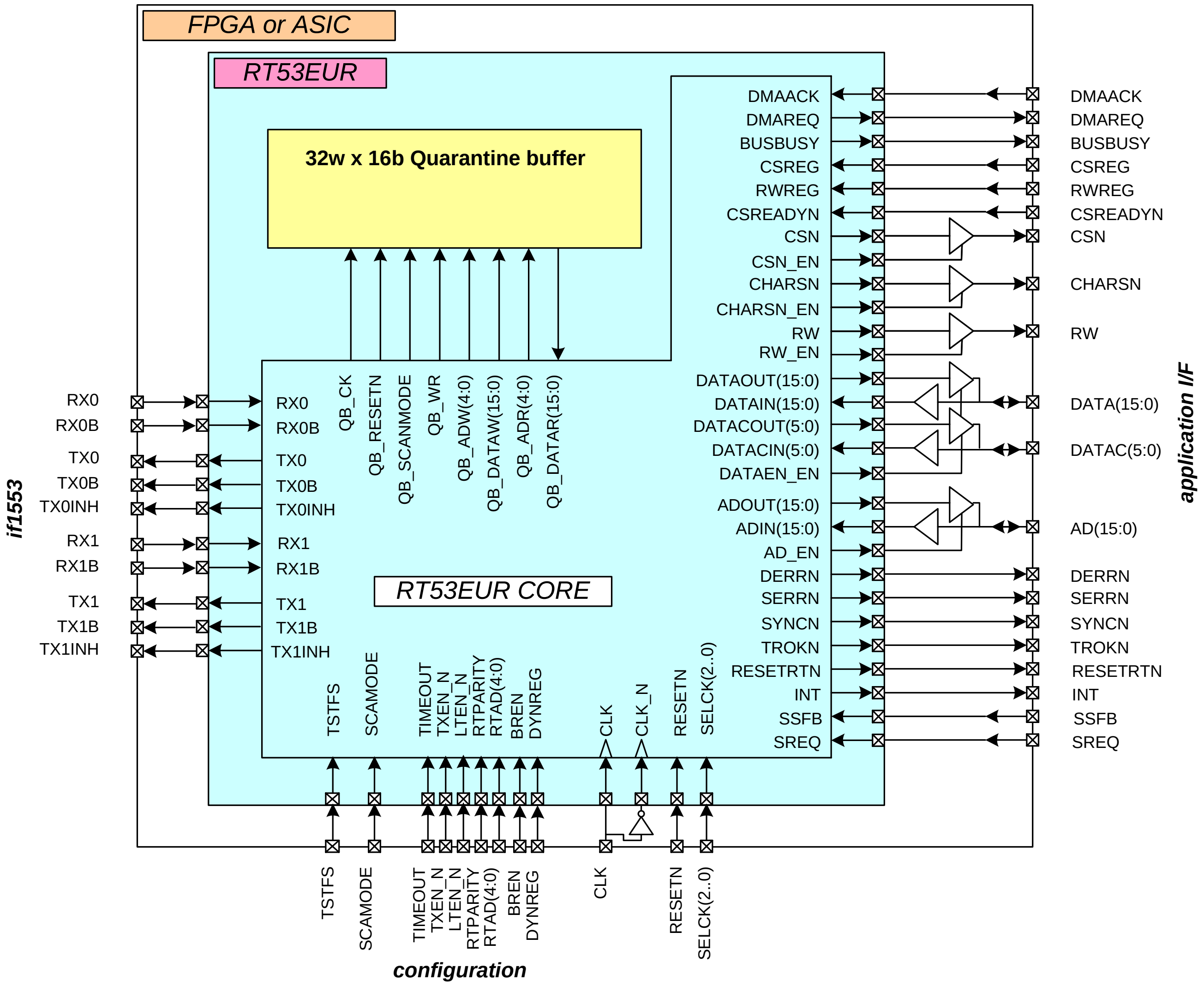

Figure 3 : Structure and Usage of the IP core

The RT53eur core is provided as a generic netlist. Two versions of this netlist are provided:

- A version with a synchronous reset in the file named: source/core/rt53core_sync.vhd.
- A version with a synchronous reset in the file named: source/core /rt53core_async.vhd.

The user has to select which core to use. The following recommendation, that has to be verified for each project, can be given:

- for an ATMEL ASIC in 0,35 um or 0,18 um it is recommended to have a synchronous reset, since an SET can impact the asynchronous reset.
- for an ACTEL RTAX or SX the asynchronous reset tree is protected against SET and thus the core with the asynchronous reset could be used.
- no recommendations are given for Xilinx in flight.

The quarantine buffer is a 32-word of 16 bits internal memory used to keep the received words in the direct and indirect modes before delivering them to the application. According to the technology the buffer can be :

- a buffer made with flip-flops that is coded in VHDL RTL.
- a buffer made using the memories of the ACTEL AX family. It is generated with smartgen.
- a buffer made using the memories of the XILINX libraries. It is generated with coregen.

The following recommendation can be given

- In ACTEL RT54SU there is no internal memory. The buffer has to be made with flip-flops.
- In ACTEL RT5AX the buffer can be made with flip-flops or by using the internal memory of the ACTEL.
- In ATMEL 0,35 um pure gate array, there is no internal memory, thus the buffer can be made with flip-flops.
- In ATMEL 0,35 um composite array or in ATMEL 0,18 um ATC18RHA, it is not really recommended to create a hard block of memory of only 32 words of 16 bits, thus the buffer can be made with flip-flops.

5. USING THE IP CORE

5.1 INSTANTIATING THE IP CORE

The IP core is easy to instantiate. An example of instantiation of the IP core is also depicted in Figure 3.

At first two generics have to be defined:

```
generic(
  GSYNC_RST  : boolean := false;  -- asynchronous reset(false), synchronous reset(true)
  RAMTYPE    : natural := 1       -- 1 flip flops, 2 Actel AX, 3 Xilinx
);
```

RAMTYPE selects the type of RAM used to make the quarantine buffer.

GSYN_RST selects the type of reset of the IP core.

The CLK and CLK_N pins shall be connected to clock and inverted clock signals. There are very few flip-flops of the IP core that are controlled by the falling clock edge. The inversion of the clock has been made outside the IP, since it is sometimes required to separate the clock trees (for example, in case of scan insertion with some CAD tools). The user shall add an inverted before the CLK_N pin as depicted.

The 1553 IF shall be connected to the input and output buffers of the ASIC or the FPGA driving the interface to the 1553 transceiver. According to the type of your Transceiver (Schmit or Harris), the BUSLEVEL pin shall be defined (refer to [AD10]).

In order to have a symmetric signal it is important to route identically the TXi and TXBi signals to get almost identical rise and fall times. Refer to your Transceiver data sheet.

The application interface of the RT53eur IP is provided with split busses: tri-state output buffers have their enable signal provided, and bidirectional pins are split in 3 (in out, and control). The user shall connect the tri-state and bidirectional buffers are shown in Figure 3.

5.2 CONFIGURING THE IP CORE

A set of configuration signals is used to configure the IP. They are described in the RT53eur IP Core DataSheet [AD10]. The configurations signals must be static either driven by external pins, or forced in the VHDL code. But they shall be defined at reset as explained in [AD10].

These signals are:

Name	Definition
SELCK<2..0>	Clock Frequency Selection
RTAD<4..0>	Address of the RT
RTPARITY	Odd parity of the RTAD<4..0> address
BUSLEVEL	select an Harris or Schmitt Transceiver
BREN	Broadcast Enable, when set to "1" broadcast is allowed (can also be configured by using RT53eur IP register)
TIMEOUT	Time Out selection for an RT to RT exchange: • "0": 31 µs • "1": 14 µs
LTEN_N	Signal for loop function inhibition
TXEN_N	Signal for encoder output inhibition
TSTFS	Selection of Fail-Safe test mode i.e. test of the 800 us inhibition timeout. shall be set to zero.
DYNREG	Enable the dynamic load configuration register (by the 1553 bus)
SCANMODE	Shall be set to zero – or possibly connected to the SCANMODE signal of the ASIC

Figure 4 : configuration signals of RT53eur IP (see [AD10])

6. THE TESTBENCH OF THE RT53EUR IP CORE

6.1 TESTBENCH OVERVIEW

The ASIC testbench is written in full VHDL 93. It is depicted in Figure 5. It is composed of the following modules either used for stimuli's generation or for checking:

- The RT53eur IP core embedded in a wrapper that includes the IO buffers.
- Two 1553B emulators modelizing two 1553B Bus controller that are used to generate commands, to generate data words, and to check transmitted data and status.
- A memory model connected to the application interface of RT53eur IP. This memory can be initialized by reading a file.
- An "Environment" emulator that is connected to the application bus as the memory. It contains an arbiter that is able to grant the bus to the IP. It is able to read and check the ASIC register and to read and check memory content.

All the emulators are able to read a script file that contains generation and check commands.

The memory of the application interface can be initialized by a file, which is needed to define the characterization data for example.

The testbench is auto-verifier. By searching ERROR string in the log file, the correctness of the response can be verified.

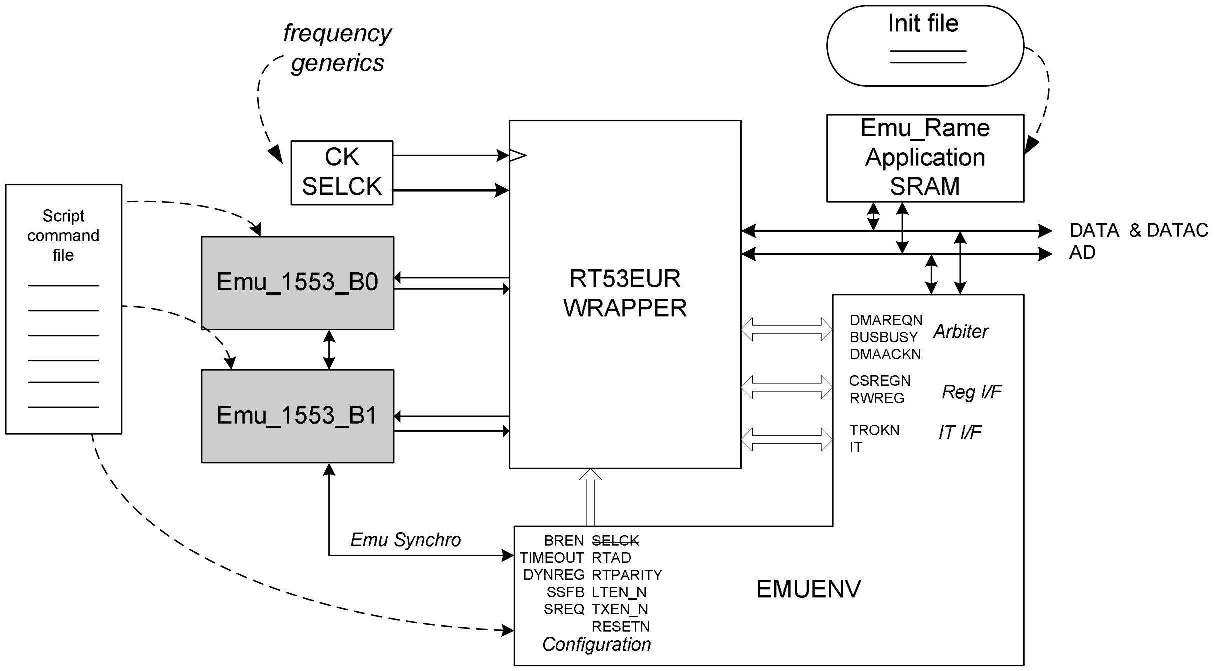

Figure 5 : RTEUR53 testbench structure

6.2 THE EMULATOR COMMANDS

6.2.1 Introduction

Each emulator has a set of commands allowing to define and check signals.

The commands are listed in the present section. The scripts located in the scripte directory provide many examples of their usage.

6.2.2 Synchronisation commands

The following commands are used to synchronize the 3 emulators and to visualize signals.

WAIT_T time
Function : wait for time before reading the next command
syntax : WAIT_T time

PAUSE
Function : wait all other emulators
syntax : PAUSE

TRACE dec
Function : Assign the signal TRACE that allows a visualisation of simulation progress in the waveform window
syntax : TRACE dec

6.2.3 Commands of the 1553 Emulator

SEND_COM RTAD E/R SA CWC
Function : Fill array in order to Send a command word on 1553 bus
RTAD : Address of the remote terminal (0..30), or Broadcast Address (31)
E/R : E or R for Transmit or Receive
SA : Sub Address (1..30), or Mode Command (31)
CWC : Number of Data Words, or Value of the Mode Command
syntax : SEND_COM dec str dec dec

SEND_CC RTAD E/R CODE
Function : Fill array in order to Send a Mode command word on 1553 bus
RTAD : Address of the remote terminal (0..30),
or Broadcast Address (31) E/R : E or R for Transmit or Receive
CODE : Value of the Mode Command
syntax : SEND_CC dec str dec

SEND_DATA NBDATA DATA1 DATA2 .. DATA8

Function : Fill array in order to send Data words on 1553 bus

NBDATA : Number of data words

DATA1, DATA2, ..., DATA8 : hexadecimal value of the data words.

8 values shall be entered, even if only e.g. 3 words are used.

The remaining ones shall be set to 0000.

Sending 5 data words can be done by issuing 5 SEND_DATA commands with one valid word or by sending only one SEND_DATA command with 5 valid words

Sending 32 words cane be done by sending 4 SEND_DATA commands with 8 valid words

syntax : SEND_DATA dec hex hex hex hex hex hex hex hex

SEND_BUS

Function : effectively send the command word and data (if any) on the bus

syntax : SEND_BUS

CHECK_STATUS RTAD ERR DIFF OCC STATUS

Function : Receive and check of the status word

RTAD : Address of the remote terminal (0..30), or Broadcast Address (31)

ERR : OK or ERROR

DIFF : BROADCAST or SINGLE

OCC : BUSY or FREE

STATUS : The 8 bits of the reserved field of the status

syntax :CHECK_STATUS dec str str str str

READ_DATA NBDATA

Function : Receive without checking NBDATA

NBDATA : Number of data words

syntax : READ_DATA dec

CHECK_DATA NBDATA DATA1 DATA2 .. DATA8

Function : Receive and check of NBDATA data words

NBDATA : Number of data words

DATA1, DATA2, ..., DATA8 : hexadecimal value of the data words.

8 values shall be entered, even if only e.g. 3 words are received.

The remaining ones shall be set to 0000 and are not checked.

syntax : CHECK_DATA dec hex hex hex hex hex hex hex hex

ECHO MESSAGE

Function : print a message in the log file of the simulator

MESSAGE : string to print

syntax : ECHO str

6.2.4 Commands of the Emuenv Emulator

===================== Application Interface - Register and RAM Access ============

REG_READ address exp_data

Function : performs a read in an ASIC register

read data shall be equal to exp_data

syntax : REG_READ hex hex

REG_WRITE address data

Function : performs a write in an ASIC register

syntax : REG_WRITE hex hex

RAM_READ address exp_data

Function : performs a read in an the application memory

read data shall be equal to exp_data

syntax : RAM_READ hex hex

RAM_WRITE address data

Function : performs a write in the application memory

syntax : RAM_WRITE hex hex

=====================Application Interface - check IT======================

CHECK_TROKN tmax

check that TROKN signal rise before tmax

syntax : CHECK_TROKN time

=====================Configuration commands======================

VALUE SIGNAL VAL

Function : Set the value of the signal SIGNAL at VAL (hex)

syntax : VALUE str hex

VALUE_AFTER SIGNAL VAL TIME

Function : Set the value of the signal SIGNAL at VAL (hex) after time TIME

syntax : VALUE_AFTER str hex time

ECHO MESSAGE

Function : print a message in the log file of the simulator

MESSAGE : string to print

syntax : ECHO str

7. THE SCRIPT FILES DELIVERED WITH THE IP

7.1 INTRODUCTION

The whole simulation plan of the RT53eur IP core is not delivered since it requires a more complex testbench to be used. The IP is provided with 4 scripts that are:

- msg.nominal.direct : basic exchanges
- allwc.direct : all word count in direct mode
- allwc.indirect : all word count in indirect mode
- allwcc.stacked : all word cound in stacked mode.

7.2 DECRIPTION OF THE SCRIPTS

7.2.1 msg.nominal.direct

Name	**msg.nominal.direct**
Object	**Test of the ASIC nominal T/R functionalities according to 1553 messages**

[General configuration of the test]

- Current testbench
- Characterization word legalizes subaddress tested

Frame 10 : Receive message in DIRECT mode

- Direct mode is defined in characterization word
- Send a 1553 message with reception of 2 data
- Check the status emission

Frame 20 : Transmit message in DIRECT mode

- Direct mode is defined in characterization word
- Send a 1553 message with transmission of 2 data
- Check the status emission

Frame 30 : Receive message in INDIRECT mode

- Direct mode is defined in characterization word
- Send a 1553 message with reception of 2 data
- Check the status emission

Frame 40 : Transmit message in INDIRECT mode

- Direct mode is defined in characterization word
- Send a 1553 message with transmission of 2 data

- Check the status emission

<u>Frame 50 : Receive message in STACKED mode</u>

- Direct mode is defined in characterization word
- Send a 1553 message with reception of 2 data
- Check the status emission

<u>Frame 60 : Transmit message in STACKED mode</u>

- Direct mode is defined in characterization word
- Send a 1553 message with transmission of 2 data
- Check the status emission

7.2.2 test allwc.direct

Name	**allwc.direct**
Object	**Test of the all word count in Rec Tr Broadcast in direct mode**

<u>[General configuration of the test]</u>

- All IT unmasked except Endmsg
- RTC disabled
- All exchanges declared valid by setting ILLENAB in configuration register 1
- All characterization word are initialized in direct mode
- For each message, a pseudo aleatory subaddress ranging from 1 to 30 is used.

<u>Frames R : send 32 receive messages</u>

- Send 32 messages with word count from 1 to 32 but not in sequence
- Check correctness of each exchange

<u>Frames T : send 32 transmit messages</u>

- Send 32 messages with word count from 1 to 32 but not in sequence
- Check correctness of each exchange

<u>Frames B : send 32 broadcast messages</u>

- Send 32 messages with word count from 1 to 32 but not in sequence
- Check correctness of each exchange

7.2.3 test allwc.indirect

Name	**allwc.indirect**
Object	**Test of the all word count in Rec Tr Broadcast in indirect mode**

[General configuration of the test]

- All IT unmasked except Tiover and Endmsg
- All exchanges declared valid by setting ILLENAB in configuration register 1
- All characterization word are initialized in indirect mode
- For each message, a pseudo aleatory subaddress ranging from 1 to 30 is used.

Same tests as allwc.direct but in indirect mode

7.2.4 test allwc.stacked

Name	**allwc.stacked**
Object	**Test of the all word count in Rec Tr Broadcast in stacked mode**

[General configuration of the test]

- All IT unmasked except Tiover and Endmsg
- All exchanges declared valid by setting ILLENAB in configuration register 1
- All characterization word are initialized in stacked mode
- For each message, a pseudo aleatory subaddress ranging from 1 to 30 is used.

Same tests as allwc.direct but in stacked mode

8. COMPILING AND SIMULATING THE IP

8.1 PROCEDURE

Simulation and compilation are only given for Modelsim simulator.

The compilation of the IP is made in the simenv1 directory. The compilation of the IP is made by executing:

../bin/MAKE

../bin/MAKETB

Warnings are issued when compiling the xilinx models. They are normal since the configuration provided by Xilinx does not contain the entity declaration.

```
** Warning: [8] ../source/sram_xilinx.vhd(121): (vcom-1155) Formal port "dinb" declared in entity "blkmemdp_v6_3" is not in the component declaration.
** Warning: [8] ../source/sram_xilinx.vhd(121): (vcom-1155) Formal port "ena" declared in entity "blkmemdp_v6_3" is not in the component declaration.
** Warning: [8] ../source/sram_xilinx.vhd(121): (vcom-1155) Formal port "enb" declared in entity "blkmemdp_v6_3" is not in the component declaration.
** Warning: [8] ../source/sram_xilinx.vhd(121): (vcom-1155) Formal port "web" declared in entity "blkmemdp_v6_3" is not in the component declaration.
…. etc.…
** Warning: [8] ../source/sram_xilinx.vhd(121): (vcom-1155) Formal port "rfda" declared in entity "blkmemdp_v6_3" is not in the component declaration.
** Warning: [8] ../source/sram_xilinx.vhd(121): (vcom-1155) Formal port "rfdb" declared in entity "blkmemdp_v6_3" is not in the component declaration.
** Warning: [8] ../source/sram_xilinx.vhd(121): (vcom-1155) Formal port "douta" declared in entity "blkmemdp_v6_3" is not in the component declaration.
```

A warning is also issued when running MAKETB. It is normal.

```
** Warning: [8] ../tbe/pack_emu_gene.vhd(292): (vcom-1009) Implicit "&" operator uses 1993 rules when computing actual for formal "s" of subprogram "print".
```

The simulation of the IP is made in the simenv1 directory. It is run with the SIME perl procedure.

```
# ../bin/SIME [options] [reference_testbench reference_script configuration]
#
# options  :
#   -help           : detailed help
#   -ref             : reference simulations
#                     batch mode and save the log files in ./zzlog_ref
#   -f=xxx           : force IP frequency to 10 12 ...24 MHz
#                     if omitted the frequency used for the previous simulation is used since
#                          it is stored in the .freq file
#   -ramax             : Quarantine buffer is made with ACTEL AX memories (default flip_flop)
#   -ramxi             : Quarantine buffer is made with XILINX VIRTEX memories (default flip_flop)
#   -syncrst           : Synchronous reset (default asynchronous)
# arguments :
#   reference_script    : name of the script or ALL
```

for example :

```
../bin/SIME msg.nominal1 –f=24
        Simulate IP with msg.nominal1 script at 24 MHz  in interactive mode
        IP has an asynchronous reset and buffer made with flip-flops
../bin/SIME msg.nominal1 –ramax –syncrt –f=10
        Simulate IP with msg.nominal1 script at 10 MHz  in interactive mode
        IP has a synchronous reset and buffer made with ACTEL AX memories
../bin/SIME all –f=24 -ref
        Simulate IP with all the scripts at 24 MHz  in batch mode and save the log files in zzlog_ref
        IP has an asynchronous reset and buffer made with flip-flops
../bin/SIME all –f=24
        Simulate IP with all the scripts at 24 MHz  in batch mode
        IP has an asynchronous reset and buffer made with flip-flops
```

8.2 RESULTS

The IP is provided with the simulation results for the 3 buffer types, the 2 reset types, the 4 script files and 2 frequencies (10 and 24 MHz) generating 48 log files stored in zzlog_ref directory.
The name of a log file is built as described below :

"Script_name"_[ax|xi|dff]_[|s]_"frequency".log

The BATCH command located in simenv1 allows to run all the simulations described above.
A chkvslog perl procedure is provided in ../bin and checks the errors and warnings in the log files. The initial error at 0 ps are removed for the ACTEL AX Ram. An AA_errorr_summary.txt file is created that gives the number of errors.
No errors are reported in the provided files.

9. SYNTHESIS OF THE RT53EUR IP TO DIFFERENT TECHNOLOGY

9.1 METHOD

The core of RT53eur IP is provided as a generic netlist. This generic netlist is based on a generic library provided in the gen_lib directory. The library that contains all the cells required to synthesize the VHDL code to the IP core. These cells are used.

ADD	Adder
AND2	AND2 cell
AND3	AND3 cell
AND4	AND4 cell
AOI22	And Or cell
BUF2X	buffer with a drive of 2
BUF4X	buffer with a drive of 4
DFF	HDFF
DFFR	HDFFR
DFFSB	HDFFSB
INV	INV
INV2X	INV2X
INV4X	INV4X
MUX21	Multiplexer
NAND2	NAND2 cell
NAND3	NAND3 cell
NAND4	NAND4 cell
NOR2	NOR2 cell
NOR3	NOR3 cell
OAI22	Or And cell
OR2	OR2 cell
OR3	OR3 cell
RLAT	RLAT
XNOR2	XNOR2 gate
XOR2	XOR2 gate

Figure 6 : the generic library cells

The RT53eur IP can be synthetized in any technology. Example of synthesis are given for :

- ACTEL RTAX2000 component using Synplify synthesis.
- ATMEL ATC18RHA using Synopsys DC compiler.

9.2 SYNTHESIS IN ACTEL RTAX2000

An example of synsthesis in ACTEL RTAX2000 is given in the synplify_ax2000 directory.

The synthesis tool used is Synplify issue 9.6.2.

The .prj file is provided. The top level of the IP has been copied locally to modify the generics: quarantine buffer made with ACTEL memory and asynchronous reset.

The number of cells of the synthesis is provided hereafter:

```
Resource Usage Report of RT53eur
Combinational Cells:    2782 of 21496 (13%)
Sequential Cells:     960 of 10752 (9%)
Total Cells:          3742 of 32248 (12%)
Clock Buffers:        3
IO Cells:             129

Details:
   and2:               191     comb:1
   and2a:              186     comb:1
   and2b:              74 comb:1
   and3:               35 comb:1
   and3a:              52 comb:1
   and3b:              34 comb:1
   and3c:              26 comb:1
   and4:               12 comb:1
   and4a:              32 comb:1
   and4b:              35 comb:1
   and4c:              13 comb:1
   buff:               9  comb:1
   cm8:                1616    comb:1
   cm8inv:             876
   dfp1b:              51 seq:1
   inv:                14 comb:1
   or2:                36 comb:1
   or2a:               100     comb:1
   or2b:               87 comb:1
   or3:                16 comb:1
   or3a:               28 comb:1
   or3b:               34 comb:1
   or3c:               11 comb:1
   or4:                2  comb:1
   or4a:               19 comb:1
   or4b:               16 comb:1
   or4c:               19 comb:1
   or4d:               9  comb:1
   xa1:                12 comb:1
   xnor2:              25 comb:1
   xor2:               39 comb:1

   df1:                40 seq:1
   dfc1b:              767     seq:1
   dfe3c:              99 seq:1
   dfe4f:           3  seq:1

   clkbuf:          2  clock buffer
   inbuf:           67
   outbuf:          60

   clkint:          1  clock buffer

   RAM64K36:        1
   XOR4:            18
   false:           3016
   true:            3016

RAM/ROM Usage Summary
Block Rams : 1 of 64 (1%)

The critical paths are provided hereafter


      Requested     Estimated     Requested     Estimated
Clock   Frequency   Frequency      Period        Period         Slack
--------------------------------------------------------------------------
CK       24.0 MHz     55.5 MHz      41.667        17.973         17.205
CK_N     24.0 MHz     55.5 MHz      41.667        17.973         12.298
=========================================
Worst slack in design: 12.298
```

9.3 SYNTHESIS IN ATMEL ATC18RHA

An example of sysnthesis in ATMEL ATC18RHA is given in the synop_atc18rha directory.

The synthesis tool used is Synopsys DC Compiler issue 2007.03-SP5-1.

The SYN.tcl file is provided. The top level of the IP has been copied locally to modify the generics: quarantine buffer made with flip-flops and synchronous reset.

The number of cells of the synthesis is provided hereafter, the result is given in um² since it is a standard cell technology.

The critical path is also provided. The IP runs at about 58 MHz, but a derating of about 30% has to be taken for routing. In any case, the maximum frequency of the IP that is 24 MHz is largely exceeded.

```
*****************************************
Report : reference
Design : RT53eur
Version: Z-2007.03-SP5-1
Date   : Mon Nov  9 14:07:24 2009
****************************************

Attributes:
    b - black box (unknown)
   bo - allows boundary optimization
```

```
    d - dont_touch
   mo - map_only
    h - hierarchical
    n - noncombinational
    r - removable
    s - synthetic operator
    u - contains unmapped logic

Reference      Unit Area    Count    Total Area    Attributes
----------------------------------------------------------

Reference               Library          Unit  Area     Count      Total  Area
Attributes
------------------------------------------------------------------------
an02d1                15.68      94   1473.92
an03d1                18.81      47    884.35
an04d1                25.08      13    326.14
aoi21d1               15.68      96   1505.28
aoi21d2               31.36       2     62.72
aoi22d1               18.81      48    903.16
aoi31d1               18.81      63   1185.40
aoi211d1              18.81      45    846.71
aoi211d2              34.49       2     68.99
aoi221d1              25.08      38    953.34
aoi222d1              31.36       9    282.24
aoi311d1              21.95      13    285.37
aoi321d1              31.36       9    282.24
aoi322d1              34.49       8    275.96
aoim21d1              18.81      14    263.42
aoim22d1              21.95      41    900.03
aoim31d1              21.95      11    241.47
aoim211d1             21.95       7    153.66
aon211d1              18.81     102   1919.23
aor21d1               21.95       4     87.80
aor22d1               25.08      18    451.58
aor31d1               21.95      15    329.27
aor31d2               25.08       1     25.08
aor211d1              25.08       1     25.08
aor221d1              31.36       2     62.72
aor311d1              25.08       7    175.61
buffd1                12.54     140   1756.15
buffd3                15.68       1     15.68
cg01d1                28.22       2     56.44
hdfcrb1              254.01      27   6858.43 n
hdfnrb1              235.19    1394 327868.76 n
hdfprb1              254.01      15   3810.24 n
inv0d0                 6.27     358   2245.37
inv0d1                 9.40     130   1223.03
inv0d2                12.54       7     87.80
nd02d0                12.54      10    125.43
nd02d1                12.54     292   3662.84
nd02d2                18.81      33    620.92
nd03d0                15.68       1     15.68
nd03d1                15.68     110   1724.80
nd03d2                28.22      13    366.91
nd04d1                18.81      98   1843.96
```

```
nd04d2                34.49      12   413.95
nd12d0                15.68       1    15.68
nd12d1                15.68      32   501.76
nd12d2                25.08       5   125.43
nd13d1                21.95      11   241.47
nr02d0                12.54      14   175.61
nr02d1                12.54     209  2621.69
nr02d2                18.81       3    56.44
nr03d1                15.68     113  1771.84
nr03d2                31.36       7   219.52
nr04d1                18.81     111  2088.57
nr04d2                34.49      17   586.43
nr13d1                21.95       5   109.75
oai21d1               15.68     155  2430.40
oai21d2               31.36       5   156.80
oai22d1               18.81    1139 21431.42
oai31d1               21.95       7   153.66
oai31d2               34.49       7   241.47
oai211d1              18.81      78  1467.64
oai221d1              28.22      21   592.70
oai222d1              28.22     216  6096.38
oai311d1              25.08       3    75.26
oai321d1              31.36      13   407.68
oai322d1              34.49      16   551.93
oaim21d1              18.81      22   413.95
oaim21d2              34.49       1    34.49
oaim22d1              25.08      22   551.93
oaim31d1              25.08       2    50.17
oaim211d1             25.08       3    75.26
oan211d1              18.81      83  1561.72
or02d1                15.68      24   376.32
or03d1                21.95       5   109.75
or04d1                25.08       6   150.52
ora31d1               25.08       1    25.08
ora211d1              25.08       2    50.17
ora311d1              28.22       1    28.22
xn02d1                34.49      14   482.94
xr02d1                31.36     161  5048.96
xr03d1                53.31      36  1919.23
-----------------------------------------------
Total 81 references                419665.78
```

1

```
Startpoint: core/ifa_edacc_DataInR_reg_2/Q_reg
              (rising edge-triggered flip-flop clocked by CK)
  Endpoint: core/ifa_ctl_FirstStackPointer_reg_0/Q_reg
            (rising edge-triggered flip-flop clocked by CK)
  Path Group: CK
  Path Type: max

  Des/Clust/Port     Wire Load Model       Library
  ------------------------------------------------
  RT53eur            46KG                  ATC18RHA_CELL_slow_1p65v_145c

  Point                                                   Incr       Path
  -------------------------------------------------------------------------
-
  clock CK (rise edge)                                    0.00       0.00
  clock network delay (propagated)                        0.00       0.00
  core/ifa_edacc_DataInR_reg_2/Q_reg/cp (hdfcrb1)         0.00 #     0.00 r
  core/ifa_edacc_DataInR_reg_2/Q_reg/qn (hdfcrb1)         1.12       1.12 r
  U3374/z (xr02d1)                                        0.56       1.69 r
  U5619/z (xr03d1)                                        0.55       2.23 r
  U5618/z (xr03d1)                                        0.58       2.81 r
  U4986/zn (nr02d1)                                       0.35       3.16 f
  U3324/zn (aoi21d1)                                      0.51       3.67 r
  U4782/zn (inv0d0)                                       0.67       4.34 f
  U5022/zn (nr03d1)                                       0.70       5.04 r
  U3321/zn (aoi321d1)                                     0.40       5.45 f
  U5024/zn (oan211d1)                                     0.51       5.95 r
  U4974/zn (inv0d0)                                       0.47       6.42 f
  U3319/zn (aon211d1)                                     0.29       6.71 r
  U4787/zn (inv0d0)                                       0.66       7.38 f
  U4568/zn (aoi311d1)                                     1.08       8.46 r
  U4783/zn (inv0d1)                                       0.72       9.18 f
  U3184/zn (oai211d1)                                     0.59       9.77 r
  U4784/zn (inv0d0)                                       0.63      10.41 f
  U4796/zn (aoi311d1)                                     0.53      10.94 r
  U3177/zn (oai21d1)                                      0.32      11.26 f
  U4781/zn (inv0d0)                                       0.98      12.24 r
  U5005/z (an03d1)                                        1.20      13.44 r
  U4582/zn (inv0d1)                                       1.12      14.56 f
  U2731/zn (oai22d1)                                      0.50      15.05 r
  core/ifa_ctl_FirstStackPointer_reg_0/Q_reg/d (hdfcrb1)
                                                          0.00      15.05 r
  data arrival time                                                 15.05

  clock CK (rise edge)                                   32.00      32.00
  clock network delay (propagated)                        0.00      32.00
  core/ifa_ctl_FirstStackPointer_reg_0/Q_reg/cp (hdfcrb1)
                                                          0.00      32.00 r
  library setup time                                     -0.58      31.42
  data required time                                                31.42
  -------------------------------------------------------------------------
  data required time                                                31.42
  data arrival time                                                -15.05
  -------------------------------------------------------------------------
  slack (MET)                                                     16.37 ns
```

9.4 SYNTHESIS IN VIRTEX4

An example of synthesis of the IP in Virtex 4 is also provided. The Synplify synthesis tool does not recognize the Xilinx memory that will have to be provided during the place and route as an edif file. All the files of the Xilinx memory are provided in the gen_ram_xilinx directory.

```
---------------------------------------
Resource Usage Report for rt53eur

Mapping to part: xc4vlx100ff1148-10
Cell usage:
FD              40 uses
FDC             746 uses
FDCE            63 uses
FDP             48 uses
FDPE            3 uses
MUXF5           1 use
sram_xilinx     1 use
LUT1            32 uses
LUT2            426 uses
LUT3            891 uses
LUT4            1209 uses

I/O ports: 129
I/O primitives: 127
IBUF           67 uses
OBUF           60 uses

BUFGP          2 uses

I/O Register bits:                  0
Register bits not including I/Os:   900 (0%)

Global Clock Buffers: 2 of 32 (6%)

Total load per clock:
   CK: 896
   CK_N: 6

Mapping Summary:
Total  LUTs: 2558 (2%)



Worst slack in design: 18.666




      Requested     Estimated     Requested     Estimated
Clock   Frequency   Frequency      Period       Period        Slack
-------------------------------------------------------------------------
CK      24.0 MHz      72.6 MHz      41.667        13.773        18.737
CK_N    24.0 MHz     173.3 MHz      41.667         5.769        18.666
============================================
Worst slack in design: 12.298
```

10. CAO TOOLS CONFIGURATION

The configuration of the tools used by Astrium to develop the spacewire core is the following:

- VHDL simulator : Mentor Modelsim version 5.8d
- Synthesis tool : Synplify version 9.6.2
- Synopsys DC compiler : .2007.03

11. VERSION OF THE SOURCE FILES

```
--========================================================
-- cvs status at Wed Dec  2 14:37:04 CET 2009
-- directory     /home/rt53eur/rt53eur/rt53eur_ip/source
--========================================================
    packrt53eurcmp.vhd  Up-to-date work_rev=1.1.1.1 rep_rev=1.1.1.1

    rt53eur.vhd         Up-to-date work_rev=1.1.1.1 rep_rev=1.1.1.1

    rt53eur_wrapper.vhd Up-to-date work_rev=1.1.1.1 rep_rev=1.1.1.1

    sram_ax.vhd         Up-to-date work_rev=1.1.1.1 rep_rev=1.1.1.1

    sram_dff.vhd        Up-to-date work_rev=1.1.1.1 rep_rev=1.1.1.1

    sram_xilinx.vhd     Up-to-date work_rev=1.1.1.1 rep_rev=1.1.1.1

    directory core

    rt53core_async.v    Up-to-date work_rev=1.2 rep_rev=1.2

    rt53core_async.vhd  Up-to-date work_rev=1.2 rep_rev=1.2

    rt53core_sync.v     Up-to-date work_rev=1.2 rep_rev=1.2

    rt53core_sync.vhd   Up-to-date work_rev=1.2 rep_rev=1.2

  > end of cvs status
```

-----%-----%-----%-----%-----

DISTRIBUTION LIST

Alberto BOETTI	ESA		
Kostas MARINIS	ESA		
Arnaud WAGNER	Astrium		
Marc SOUYRI	Astrium		
Jean Marc TAINE	Astrium		
Marc Daniel WEBER	Astrium		
Nicolas BEHOT	Astrium		