SERENA

D5.3

Interface between the modules and the signal processing platform

Project number:	779305
Project acronym:	SERENA
Project title:	gan-on-Silicon Efficient mm-wave euRopean systEm iNtegration plAtform
Start date of the project:	1st January, 2018
Duration:	42 months
Programme:	H2020-ICT-2017-1

Deliverable type:	Demonstrator
Deliverable reference number:	ICT-31-779305 / D5.3/ V1.0
Work package contributing to the deliverable:	WP5
Due date:	December 2020 – M36
Actual submission date:	14th January 2021

Responsible organisation:	TUB
Editor:	Thomas Kühne
Dissemination level:	PU
Revision:	V1.0

Abstract:	This deliverable report describes the interface between the SERENA mmWave hardware and the TUB CommIT massive MIMO SDR. The report explains the implementation of the hardware and the digital control of the interface.
Keywords:	demonstrator, interface, signal processing platform, front-end

The project SERENA has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 779305.

Editor

Thomas Kühne (TUB)

Contributors (ordered according to beneficiary numbers)

Kristoffer Andersson (EAB)

Reviewer

Christian Fager (Chalmers)

Disclaimer

The information in this document is provided “as is”, and no guarantee or warranty is given that the information is fit for any particular purpose. The content of this document reflects only the author`s view – the European Commission is not responsible for any use that may be made of the information it contains. The users use the information at their sole risk and liability.

Executive Summary

This deliverable report describes the interface between the SERENA millimeter-wave (mmWave) hardware and the TUB CommIT massive MIMO software defined radio (SDR) testbed. The report explains the implementation, from the specifications to the final design. The interface is necessary for the future work to build the proof-of-concept platform. The interface consists of coaxial cables for the data signal connection in the IF domain, a PCB as an interconnect for the control signals for the beamforming, and a digital system to control the beamforming from a host PC and the signal processing. The digital control system was tested using evaluation boards for the main components of the SERENA hardware and found to be fully functional.

Contents

List of Figures

List of Tables

Chapter 1 Introduction

The interface between the SERENA millimeter-wave (mmWave) modules and the signal processing platform is part of the 39 GHz proof-of-concept-demonstrator of the SERENA project. The interface consists of hardware and a digital control system. This report describes the specifications and implementation details of the interface. The interface connects the TUB CommIT Massive MIMO software defined radio (SDR) and the system board developed in the SERENA project. The SERENA architecture is specified in D1.1 [1] and the design of the demonstrator is reported in D6.1 [2]. Due to the nature of an interface, it is dependent on the two systems it connects. Hence, the design had to be delayed until the design of the system board of the demonstrator (currently developed in task T6.1) achieved the required level of completeness.

The extra interface printed circuit board (PCB) described in this deliverable is necessary due to the use of the TUB CommIT Massive MIMO SDR platform in the SERENA 39 GHz proof-of-concept system demonstrator. The TUB CommIT Massive MIMO SDR is the basis for the signal generation and processing of the demonstrator. It was developed outside of the SERENA project and is existing hardware. The project can leverage it to enable the demonstration of wireless data transmission. It is used for the implementation of the signal processing part of the hybrid beamforming architecture underlying the SERENA system. Without the reuse of the SDR this would be infeasible. The presented interface will adapt the SDR and its digital control system to provide the required data and control signals to the SERENA hardware.

The interface is a required step towards the proof-of-concept demonstrator but is scientifically not challenging. In the following, this report focuses on the engineering details describing the work done in SERENA to facilitate the scientifically interesting final demonstration. The report is organized as follows: Chapter 2 gives an overview of the proof-of-concept demonstrator and the requirements of the interface. Chapter 3 describes the developed hardware and Chapter 4 the implemented digital control system. Chapter 5 contains concluding remarks.

Chapter 2 System Overview

The interface described in this report connects the 39 GHz mmWave system developed in the SERENA project with the TUB CommIT massive MIMO SDR. Figure 1 shows the architecture of the proof-of-concept platform. The red circle visualizes the location of the physical interface between the SDR and the mmWave system. The interface has two main tasks. The first is to link the data signals of the system board and the SDR. The second task is to provide a control system to the signal processing to setup the mmWave hardware and the beamforming. The black lines labelled *data* in Figure 1 are the analog intermediate frequency (IF) data signals. The blue lines are the control signals to control the beamforming of the integrated modules. The specifications and requirements of the interface are given by the system board and the SDR and listed in the following section. This chapter further describes the location of the interface in the proof-of-concept demonstrator and lists its specifications.

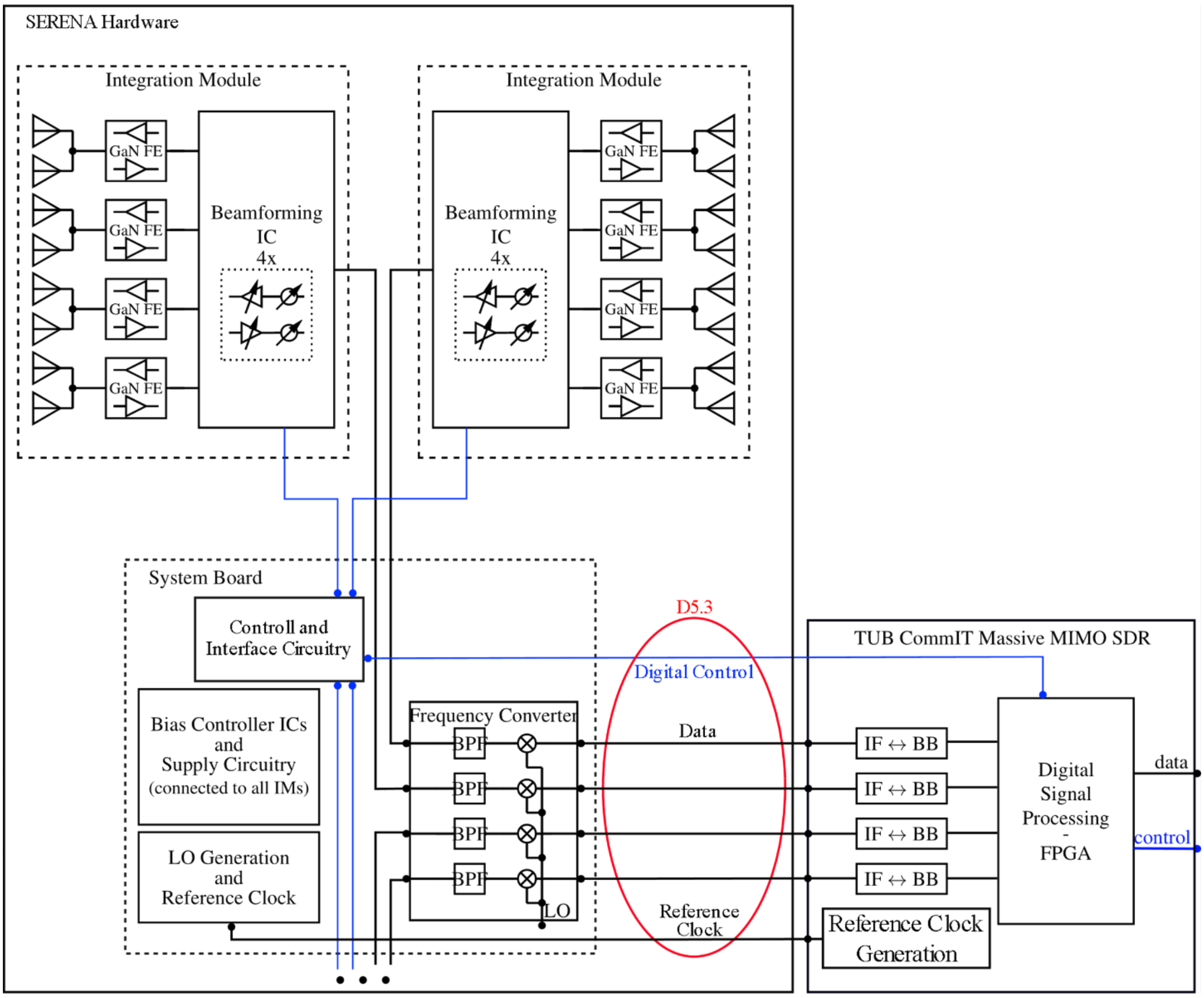

SERENA Hardware
Integration Module
Integration Module
GaN FE
Beamforming IC 4x
Beamforming IC 4x
System Board
Controll and Interface Circuitry
Bias Controller ICs and Supply Circuitry (connected to all IMs)
LO Generation and Reference Clock
Frequency Converter
BPF
LO
D5.3
Digital Control
Data
Reference Clock
TUB CommIT Massive MIMO SDR
IF ↔ BB
Digital Signal Processing - FPGA
Reference Clock Generation
data
control

Figure 1 Proof-of-concept-platform architecture, the red circle is the interface described in this report

The main component of the proof-of-concept demonstrator is the so-called system board (developed in task T6.1). The system board is comprising of multiple integrated beamformer modules (developed in task T5.2), frequency up- / downconverters, LO generation, bias and supply circuitry, and a small control system. Functionally, it is an analog mmWave antenna array with a controllable beamforming function. It does not integrate any signal processing capability. An additional signal processing platform is required to demonstrate wireless data transmission and the full hybrid beamforming potential. The proof-of-concept platform uses the TUB CommIT massive MIMO SDR (see Figure 2) as the signal processing platform.

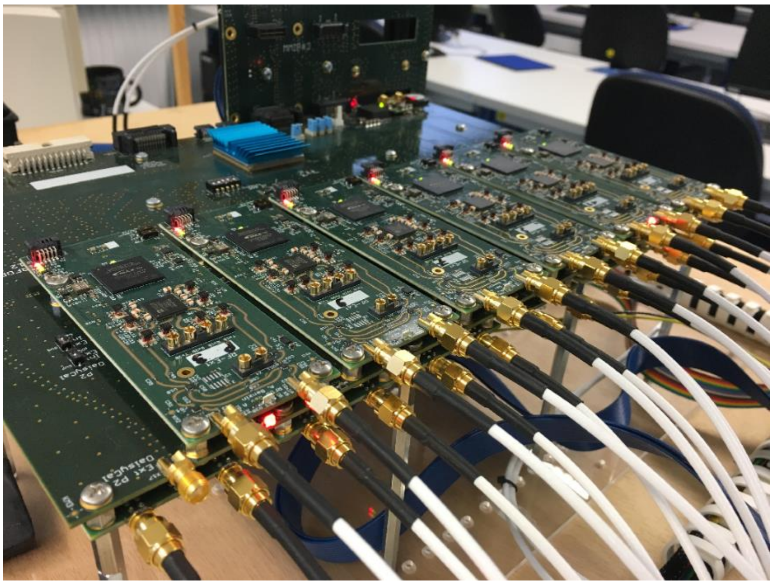


Figure 2 Photo of the TUB CommIT massive MIMO SDR

2.1 Interface Specifications

The specifications and requirements of the interface from both the system board and the TUB CommIT massive MIMO SDR were derived in WP1 D1.1 [1] and WP6 D6.1 [2]. Some of the requirements were refined during the current work on the system board design in task T6.1. The following tables list the specifications of the interface hardware and the digital control system.

Specification	Value
Data signal IF connection	Single ended coaxial cable
IF frequency	3 GHz
Impedance	50 Ohm
Number of signals	4
Connector style system board	SMP
Connector style SDR	SMA
Typical required TX power level	-20 dBm

Table 1 Data signal interface specifications

Specification	Value
Control signal type	• Serial peripheral interface (SPI) • Logic signals (enable, reset, ready)
Required control signals	• serial in, serial out, serial clock, two chip selects • TX enable, RX enable • Reset • Ready indication
TX / RX control	Two enable logic signals (TX_EN, RX_EN)
Control clock frequency	Below 100 MHz
System board connector	Samtec HSEC8-120-01-S-DV-A-K PCB edge connector
SDR connector	24 pin header 2.54 mm pitch
System board voltage level	1.8 V
SDR voltage level	2.5 V

Table 2 Digital control interface specifications

Specification	Value
Reference clock frequency	122.8 MHz
Reference clock input signal type, system board	AC coupled differential 100 Ohm, min 200 mV peak to peak, e.g., LVPECL required on the PCB edge connector
Reference clock signal type, SDR	AC coupled differential LVPECL provided on 2 SMA coaxial cable connectors

Table 3 Reference clock specifications

2.2 Interface Overview

The interface consists of hardware and a digital control system. The interface hardware is located at the position of the red circle in Figure 1. It connects physically the proof-of-concept system components. It converts between different signal types e.g., voltage levels and clock formats. The programmable digital hardware of the TUB CommIT massive MIMO SDR is used with an adopted firmware as the digital control system for the interface. The digital part is running on an Intel field-programmable gate array (FPGA). The structure of the SDR is shown in Figure 3. The two tasks, the data signal connection (green in Figure 3) and the control connection (blue in Figure 3), of the interface are handled by different parts of the platform.

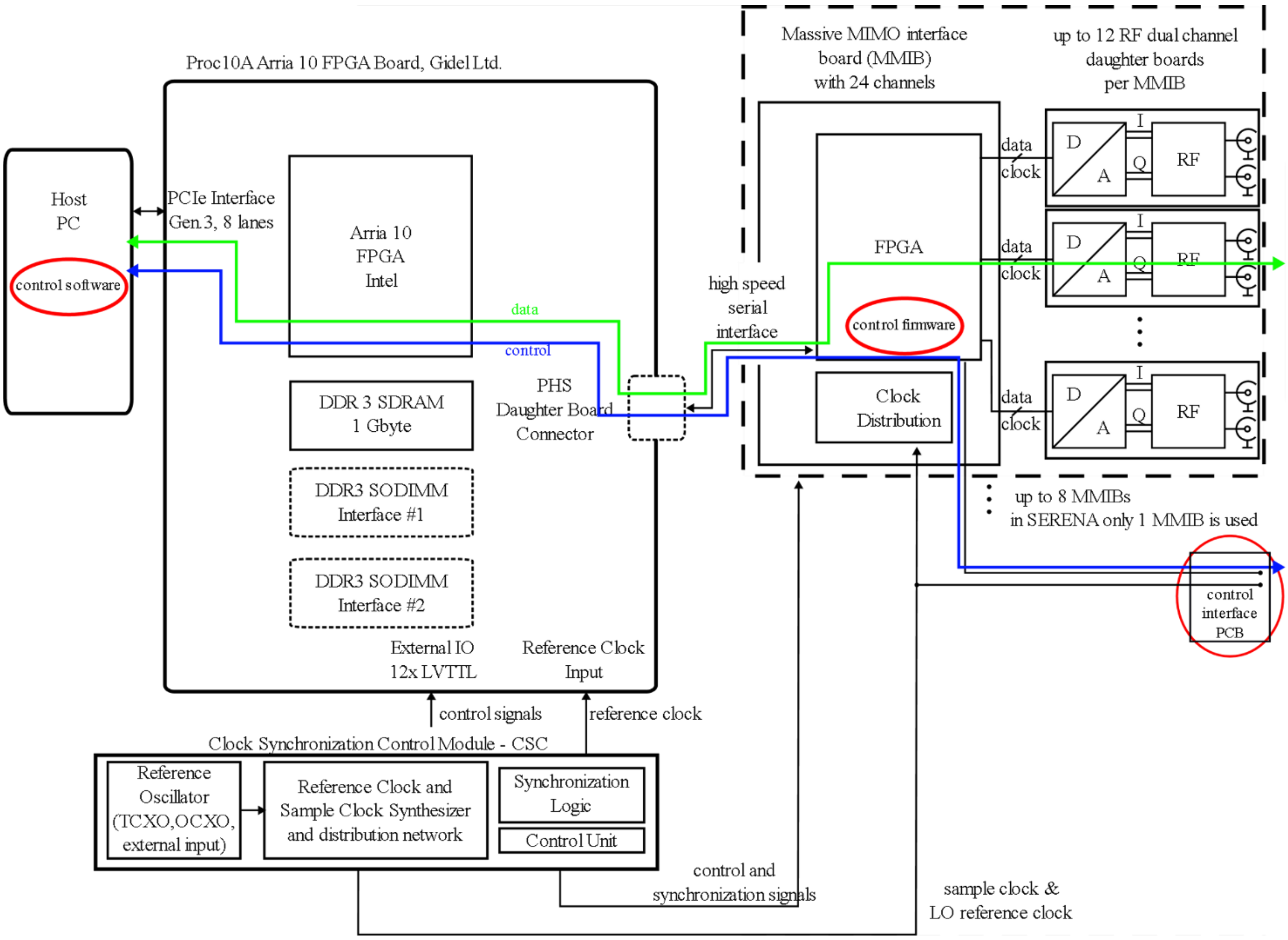

Proc10A Arria 10 FPGA Board, Gidel Ltd.
Host PC
control software
PCIe Interface Gen.3, 8 lanes
Arria 10 FPGA Intel
data
control
DDR 3 SDRAM 1 Gbyte
PHS Daughter Board Connector
DDR3 SODIMM Interface #1
DDR3 SODIMM Interface #2
External IO 12x LVTTL
Reference Clock Input
control signals
reference clock
Clock Synchronization Control Module - CSC
Reference Oscillator (TCXO,OCXO, external input)
Reference Clock and Sample Clock Synthesizer and distribution network
Synchronization Logic
Control Unit
control and synchronization signals
sample clock & LO reference clock
Massive MIMO interface board (MMIB) with 24 channels
up to 12 RF dual channel daughter boards per MMIB
high speed serial interface
FPGA
control firmware
Clock Distribution
data
clock
D
A
I
Q
RF
up to 8 MMIBs
in SERENA only 1 MMIB is used
control interface PCB

Figure 3 Block diagram of the TUB CommIT massive MIMO SDR

The data signals are connected as analog single-ended IF signals to the system board. As listed in Table 1, the data signals are directly wired with coaxial cables between the system board and the SDR. Each IF signal uses one cable. The system board and the SDR use different connectors which can be connected by cables and adapters. The signal specifications, for example the impedance and power levels, of the system board and the SDR match very well. On the SDR side, the SDR RF ports (connected to the white cables in Figure 2) are used to generate / receive these signals (see Figure 3). The signals are converted to the digital baseband domain and forwarded to a central signal processing FPGA (Intel Arria 10). This FPGA is in turn connected to a host PC. The data signals use the existing data path through the SDR without the need for an adoption in SERENA. Additional information on the SDR can be found in [3]. Since the SDR is a controllable radio by itself (but in the frequency range below 6 GHz) its RF ports are very versatile. Hence, the SDR can provide sufficient output power and gain for the 3 GHz IF signal. Please note, that the signal bandwidth of the SDR is much smaller than the supported bandwidth of the SERENA mmWave hardware. The SDR is only used to demonstrate a data transmission and is required for a fast and feasible implementation of the hybrid signal processing for the duration of the project. The data rate will be much lower than what a final mmWave system could achieve. Nonetheless, the demonstrator results can be used to evaluate the concept and the performance through measuring the achieved SINR and mathematical up-scaling.

The second task of the interface is to provide a control of the beamforming integrated circuits (the IFAT BEAM39PA) to the signal processing / host PC. Since this is a new task for the SDR, an additional digital control system had to be implemented. This system is realized on a FPGA on the so-called massive MIMO interface board (MMIB) component of the SDR (see the red circle in Figure 3). A detailed description follows in Chapter 4. The digital control system is wired to a general-purpose input-output connector which is used as interface on the SDR to the system board. On the system board, the interface uses a PCB edge connector. The specifications of the control interface are listed in Table 2. The connectors and voltage levels are not directly compatible. Therefore, an interface PCB (see Chapter 3) converts the control signals according to the specifications. The control interface consists of a serial peripheral interface (SPI) and some additional logic signals, e.g., TX / RX enable and reset signals. The system board is an SPI peripheral providing a register-based access to the beamforming. This access protocol is equivalent to the SPI access to the Infineon BEAM39PA used in the SERENA modules. The registers and the SPI format are described in the datasheet of the Infineon beamforming integrated circuit. The digital control system in the MMIB FPGA acts as a SPI master controlling the beamforming. The system itself receives higher level commands from the host PC or the central signal processing FPGA. It converts these commands into register settings of the system board.

In addition to the two tasks above, the SDR and the system board need a common reference clock. The reference clock input of the system board is on the PCB edge connector. The SDR has an internal clock module which can generate reference clocks and is used by all components in the SDR. A spare output is connected via coaxial cables to the interface PCB, which forwards the reference clock to the PCB edge connector of the system board. The voltage levels (see Table 3) of the SDR clock module and the system board are compatible, hence no additional conversion circuitry is needed.

Chapter 3 Interface Hardware

The additional hardware parts used for the interface between the SERENA system board and the TUB CommIT massive MIMO SDR are SMA coaxial cables (and adapters) and a self-developed PCB. The coaxial cables and adapters are standard commercially available parts. The control interface PCB was designed to be an interconnect between the two different connectors on the SDR and the system board. The SPI control signals have a data rate of up to 100 Mbps. The voltage level difference (see Table 2) is handled by standard commercially available integrated circuits. The edge of the PCB will be inserted into the connector on the system board. The reference clock is fed by coaxial cables through the connectors X1 and X2 from the SDR. The control signals are connected by a ribbon cable to the connector SV1. The power is supplied from the SDR connection. Figure 5 shows the full schematic.

The PCB layout is shown in Figure 4. The PCB is a 4-layer construction although the two inner layers are GND layers. This construction reduces the interference between the different signals. All signals are low power and low data rate (< 100 Mbps) and, hence, standard commercially available FR4 material and processes are used.

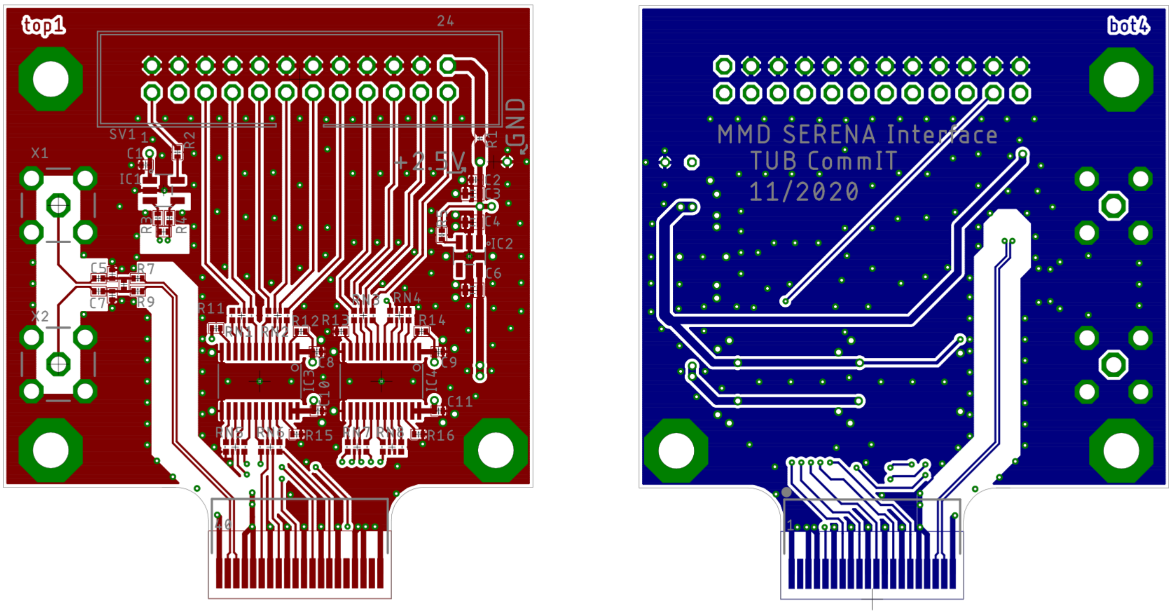


Figure 4 PCB layout of the interface hardware; the two GND middle layers are not shown; scaled to 150%

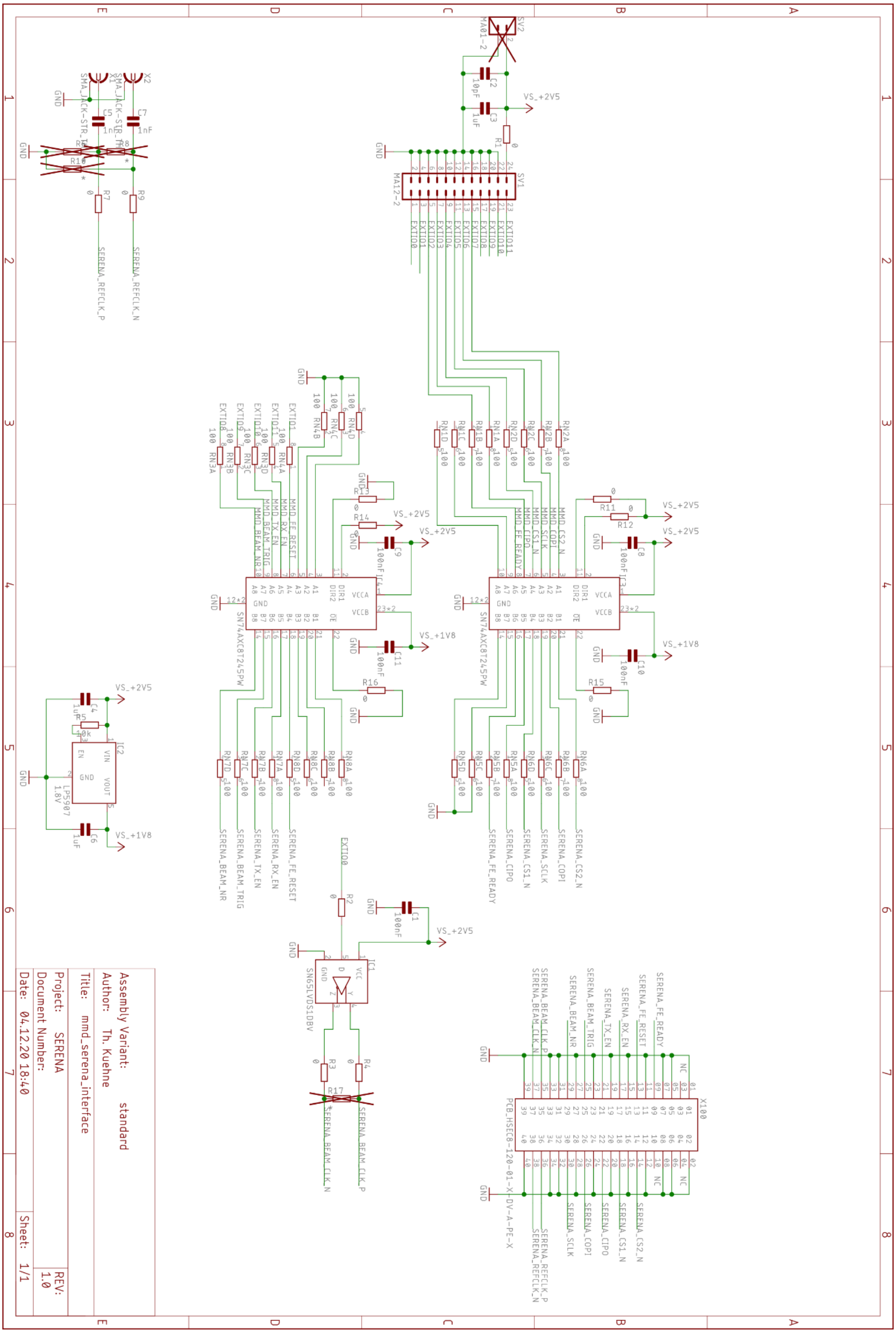


Figure 5 Schematic of the interface PCB

Chapter 4 Interface Digital Control System

The task of the digital control system is to create an interface for the signal processing and host PC to control the SERENA beamforming module. It is implemented as a firmware for the TUB CommIT massive MIMO SDR. This includes additions to the FPGA design of the SDR and a firmware program running on a microcontroller. This microcontroller is also implemented on the FPGA. As shown in Figure 3, the FPGA on the MMIB board of the SDR is used. A ribbon cable connects the interface PCB to a connector on the MMIB board. This connector is wired to the MMIB FPGA.

The design reuses the control system of the TUB SDR. This system is based on a command protocol with defined commands which can address different parts of the SDR. The beamforming control interface is a new address in this system. The command packets can be sent over different byte oriented digital interfaces e.g., over the SDR internal data path (named high speed serial interface in Figure 3) or an external USB connection. The source of the commands can be either a host PC running the control software or the signal processing core in the central FPGA. The microcontroller receives the commands and interprets the sent parameters (depending on the specific command). For the beamforming interface, the microcontroller converts the commands to register settings for the BEAM39PA and the corresponding series of SPI packets.

4.1 FPGA Design

The main control firmware of the SDR MMIB boards runs on a Nios II microcontroller. The Nios II is an IP core available by Intel. It is a 32-bit general-purpose microcontroller which is synthesized in the FPGA. The microcontroller core can be extended by different components to support peripheral devices and other parts. For example, in the MMIB design the Nios II is extended by multiple general-purpose input-output ports, UART transceiver blocks and others. Nios II systems are designed using the Intel software called Qsys. Appendix A shows the design with all the required components for the interface in Qsys. The firmware program receives commands over the UART block and controls the system board using the SPI block and the general-purpose input and output ports (named PIO in Appendix A). The actual MMIB Nios II system is too complex to be added to this report and does not add valuable information related to the beamforming interface. The final Qsys system is embedded in the full FPGA design. The MMIB design was implemented using the block design entry from the Intel Quartus software. The design for the control interface does include mainly the Nios II sub-system and some pin assignment structures. It can be seen in Appendix B.

As mentioned before, the control of the BEAM39PA uses a SPI. The SPI packet structure of the BEAM39PA is different compared with a typical SPI. The Intel IP library includes a SPI master component for Nios II systems. Unfortunately, due to the non-standard SPI protocol of the BEAM39PA, this SPI master cannot be used to control BEAM39PAs. Therefore, an extra SPI master core was developed. This core is adopted to the BEAM39PA SPI format and supports all required SPI packet types. This includes reset packets, short packets (2 bytes), normal packets (4 bytes), and long packets (15 bytes). The SPI IP core was developed with the Intel DSP-builder software. The design is shown in Appendix C.

All required Nios system components of the control system can be configured and used from the firmware program.

4.2 Firmware and Software

The firmware program running on the Nios II microcontroller is a state machine converting the received commands from the host PC to the control sequences for the system board / BEAM39PAs. It is designed to support multiple BEAM39PAs per system board as there are multiple integration modules on one system board. The firmware is designed in a way to also support multiple system boards, although the current connection between the MMIB and the SERENA system supports only a single system board. This was implemented to support future extensions of the system.

The main tasks of the control system are:

1. Initialization and reset of the SERENA hardware (the system board controller and the BEAM39PAs)
2. Control of all TDD switches (TX / RX control)
3. Direct control of the beamforming of the BEAM39PAs
4. Direct control of the power / gain settings
5. Uploading of a pre-calculated beam table to the memory of the BEAM39PAs
6. Control of the beamforming using the beam table in the BEAM39PAs.

Each of these tasks can be executed by a high-level command from the host PC interface. The high-level commands are converted by the firmware to lower-level SPI packets or general-purpose input-output port settings. This yields to an easier-to-use command protocol and a higher configuration speed. Additionally, the control system can forward arbitrary register write / read commands from the host PC to the BEAM39PAs. This lower-level access can be used to support BEAM39PA functions which are not time critical and are not used during the data transmission operation. An example is the test and calibration system of the BEAM39PA.

Table 4 lists all commands available to the host PC / signal processing core to control the firmware.

Function Name	Type	Description
MMD_SERENA_reset	Main task 1.	Resets the SERENA system (incl. BEAM39PAs and system board).
MMD_SERENA_init	Main task 1.	Initialize the SERENA system (incl. BEAM39PAs and system board).
MMD_SERENA_write_sram	Main task 5.	Write data to the SRAM of a certain BEAM39PA.
MMD_SERENA_write_rfctrl	Main task 2., 3. and 4.	Write data to the RFCTRL register of a certain BEAM39PA. This sets up the phase and gain of the 4 channels.
MMD_SERENA_set_beam	Main task 2. and 6.	Set a beam using the BEAM39PA SRAM values.
MMD_SERENA_set_power	Main task 4.	Set the power state (power on/off) of a SERENA module (incl. BEAM39PAs).
MMD_SERENA_spi_reset	Support command	Sends a special SPI command to reset the BEAM39PA SPI part.
MMD_SERENA_spi_extra_sclk	Support command	Sends extra SCLK pulses to the BEAM39PA.
MMD_SERENA_get_stat	Support command	Get the content of the status register of a BEAM39PA.

Function Name	Type	Description
MMD_SERENA_clear_stat	Support command	Clear the status register of a BEAM39PA.
MMD_SERENA_write_reg_direct	Secondary task	Write directly (direct address) to a register of a certain BEAM39PA.
MMD_SERENA_write_reg	Secondary task	Write a register of a certain BEAM39PA using the firmware variable representing all beamer ICs.
MMD_SERENA_read_reg_direct	Secondary task	Read directly (direct address) from a register of a certain BEAM39PA.
MMD_SERENA_read_reg	Secondary task	Read a register of a certain BEAM39PA using the firmware variable representing all beamer ICs.

Table 4: Implemented Firmware Commands

Beside the firmware, also a software library running on the host PC was developed. This software runs in Mathworks Matlab. It implements the commands on the host PC side and has an even higher-level interface to control the SERENA system. This library can be incorporated into the demonstration and signal processing software.

Chapter 5 Summary, Conclusion and Challenges

This report presented the details of the interface between the SERENA mmWave hardware (more precisely the system board) and the TUB CommIT massive MIMO SDR. Both parts together make the 39 GHz proof-of-concept SERENA demonstrator. The interface is necessary for the future work to build the proof-of-concept platform. The tasks of the interface and the corresponding parts have been described. The details of the implementation of all parts are shown. The interface consists of coaxial cables for the data signal connection in the IF domain, a PCB as an interconnect for the control signals for the beamforming, and the firmware to control the beamforming from a host PC and the signal processing. The PCB functions as an adapter between the different connectors and the different voltage levels. The digital control part is implemented as an FPGA design and a firmware running on a microcontroller in the FPGA.

The finalized specifications derived from the work on the system board in T6.1 are listed. Methodically, the interface development is a hardware-software co-design including specification work, PCB design, FPGA logic design and microcontroller firmware programming. This report describes the development of the main parts of the interface. The challenges solved are mainly engineering problems. The interface in general supports the tasks T6.3 and T6.4 to make the scientific investigation of the SERENA hardware and hybrid concept possible using the proof-of-concept platform

By the time this report was submitted, the system board was not yet finally designed and manufactured. Since the interface connects to the system board, it cannot be fully tested. Especially the hardware cannot be tested without the system board. Nonetheless, there are no major problems anticipated with the hardware due to the simple design. The digital control system running on a FPGA was tested using an evaluation board of the beamforming integrated circuit. The BEAM39PA on the evaluation platform is equivalent to the circuits embedded in the integrated modules, which are mounted on the system board. The evaluation boards were provided by the partner IFAT to the partner TUB. To simplify the tests, instead of the large SDR a smaller FPGA evaluation board was used. The FPGA on this board and the design environment is similar to the one used for the firmware in the SDR. The tested firmware design can directly be employed in the SDR. Figure 6 shows the test setup.

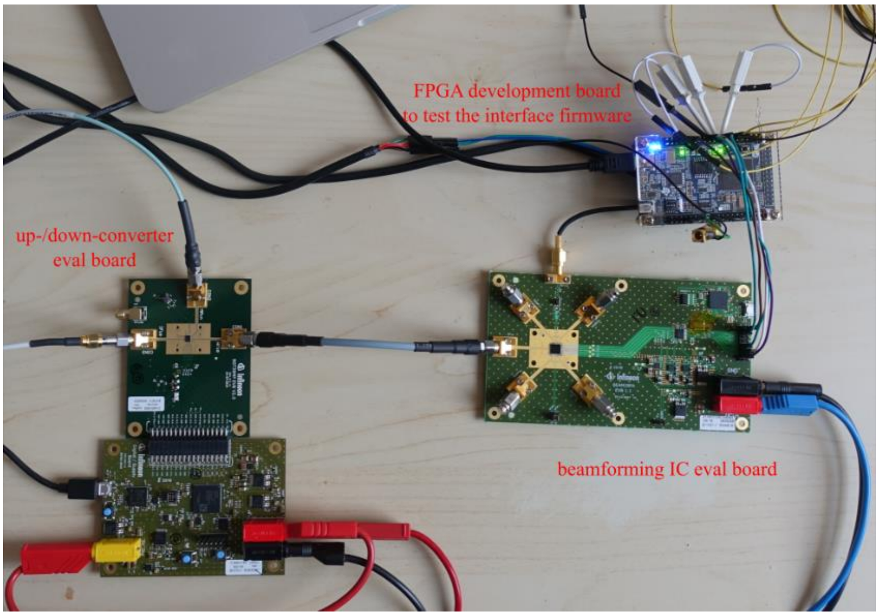


Figure 6 Evaluation-board-based tests

The control interface described in Section 4.2 was successful tested using the evaluation boards and a host PC. The beamforming control is verified, and all functions work as described in the BEAM39PA datasheet. Please note that the RF characteristics were not tested as they are not part of this deliverable.

An additional advantage of this test solution in the future is that it can be used to verify the function of the integrated modules during the manufacturing. The test setup can run the so-called built-in test equipment (BITE) tests of the beamforming circuit to verify proper function. The test setup can connect with a small adapter PCB to the SERENA integrated module without being mounted on the system board. These tests will be described in the deliverable D5.5 and help to mitigate risks during the manufacturing.

Chapter 6 List of Abbreviations

Abbreviation	Translation
BITE	Built-in test equipment
FPGA	Field-programmable gate array
GND	Ground
IF	Intermediate frequency
MMD	Massive MIMO demonstrator
MMIB	Massive MIMO interface board
mmWave	Millimeter-Wave, frequency range from 30-300 GHz
MIMO	Multiple input multiple output
PCB	Printed circuit board
RF	Radio frequency
RX	Receiver
SDR	Software defined radio
SPI	Serial peripheral interface
TX	Transmitter

Chapter 7 Bibliography

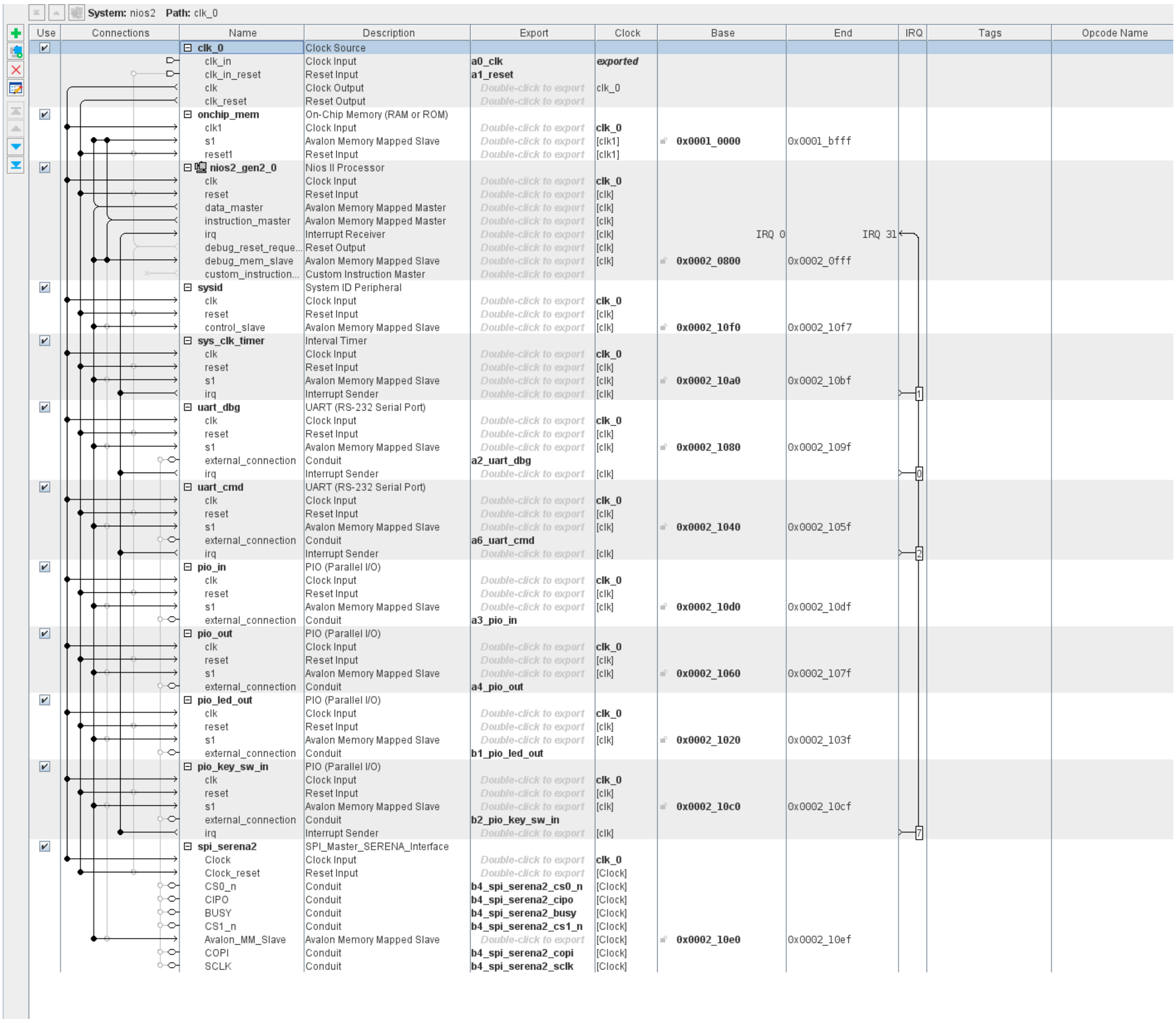
[1] Serena Project. June 2018. “39 GHz proof-of-concept platform and front-end specifications - D1.1.”

[2] Serena Project. June 2019. “39 GHz Proof-of-concept Platform design – D6.1.”

[3] A. Benzin, D. Osterland, M. Dill and G. Caire, “Centralized Single FPGA Real Time Zero Forcing Massive MIMO 5G Basestation Hardware and Gateware,” 2020 IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Atlanta, GA, USA, 2020, pp. 1-5, doi: 10.1109/SPAWC48557.2020.9154248.

Appendix

A. Qsys design of a Nios II example

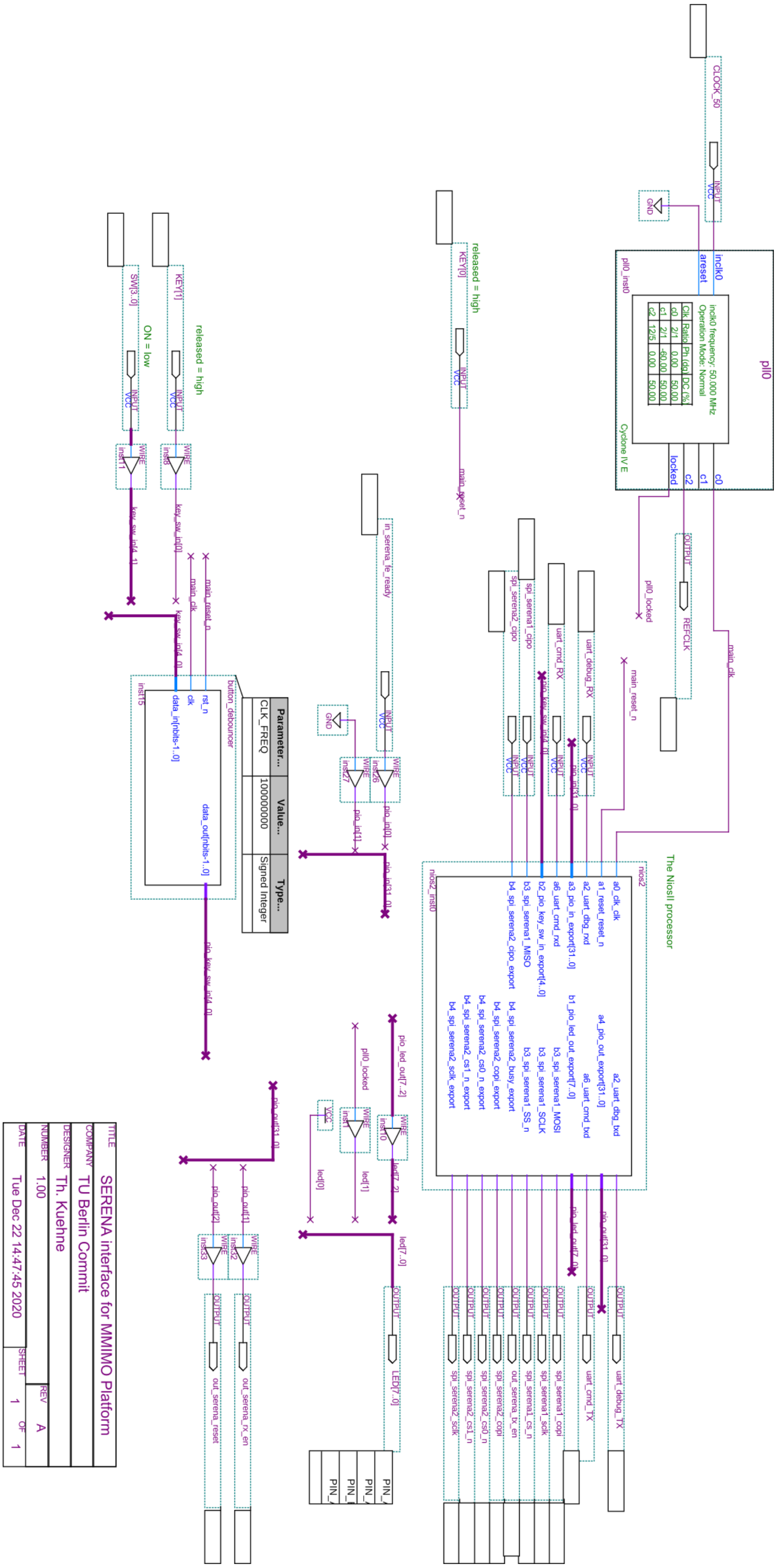


System: nios2 **Path:** clk_0

Use	Connections	Name	Description	Export	Clock	Base	End	IRQ	Tags	Opcode Name
☑		⊟ **clk_0**	Clock Source							
		clk_in	Clock Input	**a0_clk**	***exported***					
		clk_in_reset	Reset Input	**a1_reset**						
		clk	Clock Output	*Double-click to export*	clk_0					
		clk_reset	Reset Output	*Double-click to export*						
☑		⊟ **onchip_mem**	On-Chip Memory (RAM or ROM)							
		clk1	Clock Input	*Double-click to export*	**clk_0**					
		s1	Avalon Memory Mapped Slave	*Double-click to export*	[clk1]	**0x0001_0000**	0x0001_bfff			
		reset1	Reset Input	*Double-click to export*	[clk1]					
☑		⊟ **nios2_gen2_0**	Nios II Processor							
		clk	Clock Input	*Double-click to export*	**clk_0**					
		reset	Reset Input	*Double-click to export*	[clk]					
		data_master	Avalon Memory Mapped Master	*Double-click to export*	[clk]					
		instruction_master	Avalon Memory Mapped Master	*Double-click to export*	[clk]					
		irq	Interrupt Receiver	*Double-click to export*	[clk]	IRQ 0	IRQ 31			
		debug_reset_reque...	Reset Output	*Double-click to export*	[clk]					
		debug_mem_slave	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	**0x0002_0800**	0x0002_0fff			
		custom_instruction...	Custom Instruction Master	*Double-click to export*						
☑		⊟ **sysid**	System ID Peripheral							
		clk	Clock Input	*Double-click to export*	**clk_0**					
		reset	Reset Input	*Double-click to export*	[clk]					
		control_slave	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	**0x0002_10f0**	0x0002_10f7			
☑		⊟ **sys_clk_timer**	Interval Timer							
		clk	Clock Input	*Double-click to export*	**clk_0**					
		reset	Reset Input	*Double-click to export*	[clk]					
		s1	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	**0x0002_10a0**	0x0002_10bf			
		irq	Interrupt Sender	*Double-click to export*	[clk]			1		
☑		⊟ **uart_dbg**	UART (RS-232 Serial Port)							
		clk	Clock Input	*Double-click to export*	**clk_0**					
		reset	Reset Input	*Double-click to export*	[clk]					
		s1	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	**0x0002_1080**	0x0002_109f			
		external_connection	Conduit	**a2_uart_dbg**						
		irq	Interrupt Sender	*Double-click to export*	[clk]			0		
☑		⊟ **uart_cmd**	UART (RS-232 Serial Port)							
		clk	Clock Input	*Double-click to export*	**clk_0**					
		reset	Reset Input	*Double-click to export*	[clk]					
		s1	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	**0x0002_1040**	0x0002_105f			
		external_connection	Conduit	**a6_uart_cmd**						
		irq	Interrupt Sender	*Double-click to export*	[clk]			2		
☑		⊟ **pio_in**	PIO (Parallel I/O)							
		clk	Clock Input	*Double-click to export*	**clk_0**					
		reset	Reset Input	*Double-click to export*	[clk]					
		s1	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	**0x0002_10d0**	0x0002_10df			
		external_connection	Conduit	**a3_pio_in**						
☑		⊟ **pio_out**	PIO (Parallel I/O)							
		clk	Clock Input	*Double-click to export*	**clk_0**					
		reset	Reset Input	*Double-click to export*	[clk]					
		s1	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	**0x0002_1060**	0x0002_107f			
		external_connection	Conduit	**a4_pio_out**						
☑		⊟ **pio_led_out**	PIO (Parallel I/O)							
		clk	Clock Input	*Double-click to export*	**clk_0**					
		reset	Reset Input	*Double-click to export*	[clk]					
		s1	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	**0x0002_1020**	0x0002_103f			
		external_connection	Conduit	**b1_pio_led_out**						
☑		⊟ **pio_key_sw_in**	PIO (Parallel I/O)							
		clk	Clock Input	*Double-click to export*	**clk_0**					
		reset	Reset Input	*Double-click to export*	[clk]					
		s1	Avalon Memory Mapped Slave	*Double-click to export*	[clk]	**0x0002_10c0**	0x0002_10cf			
		external_connection	Conduit	**b2_pio_key_sw_in**						
		irq	Interrupt Sender	*Double-click to export*	[clk]			7		
☑		⊟ **spi_serena2**	SPI_Master_SERENA_Interface							
		Clock	Clock Input	*Double-click to export*	**clk_0**					
		Clock_reset	Reset Input	*Double-click to export*	[Clock]					
		CS0_n	Conduit	**b4_spi_serena2_cs0_n**	[Clock]					
		CIPO	Conduit	**b4_spi_serena2_cipo**	[Clock]					
		BUSY	Conduit	**b4_spi_serena2_busy**	[Clock]					
		CS1_n	Conduit	**b4_spi_serena2_cs1_n**	[Clock]					
		Avalon_MM_Slave	Avalon Memory Mapped Slave	*Double-click to export*	[Clock]	**0x0002_10e0**	0x0002_10ef			
		COPI	Conduit	**b4_spi_serena2_copi**	[Clock]					
		SCLK	Conduit	**b4_spi_serena2_sclk**	[Clock]					

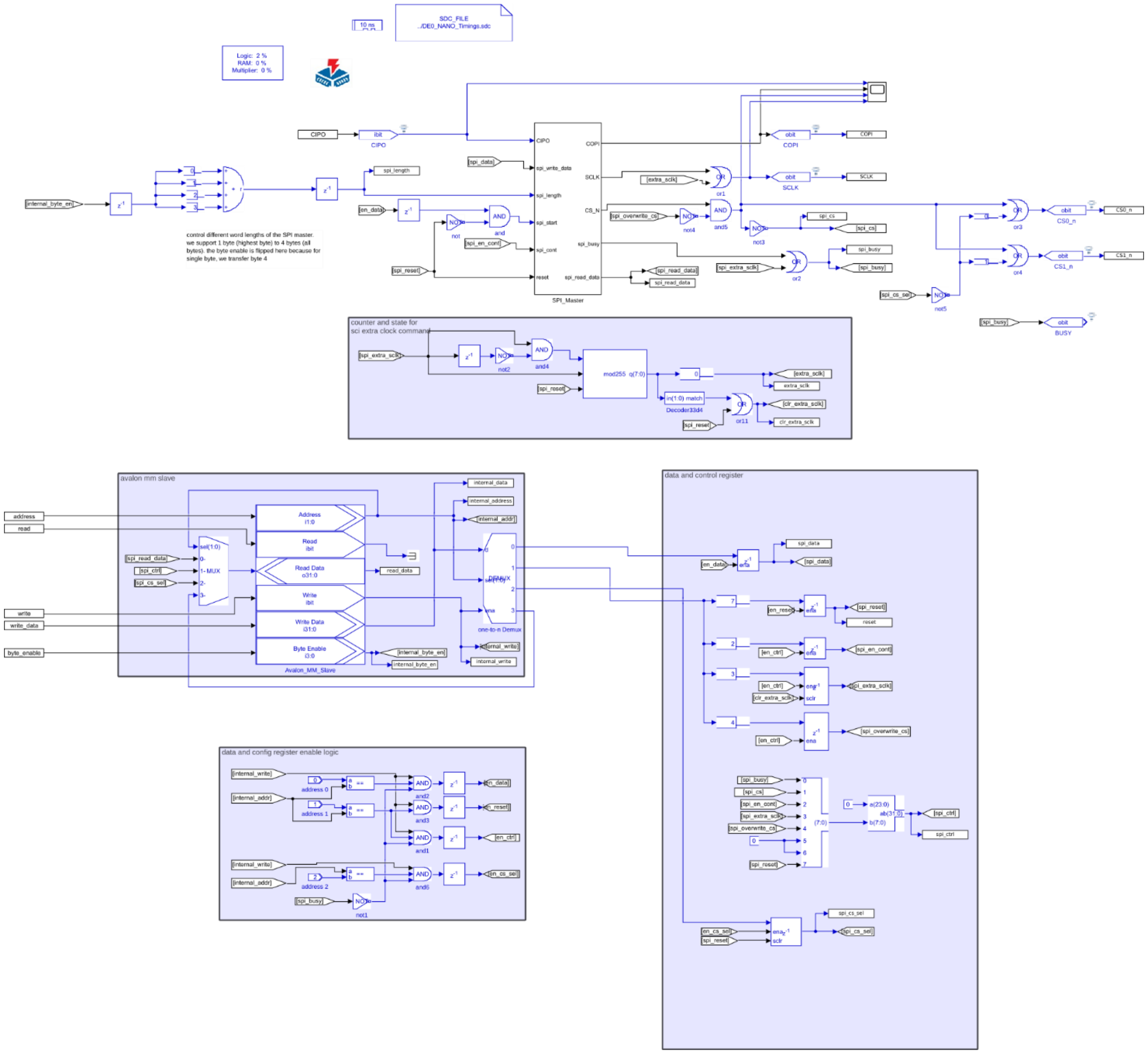
Qsys design of the FPGA evaluation-board-based Nios II design

B. Top level Quartus design of the FPGA

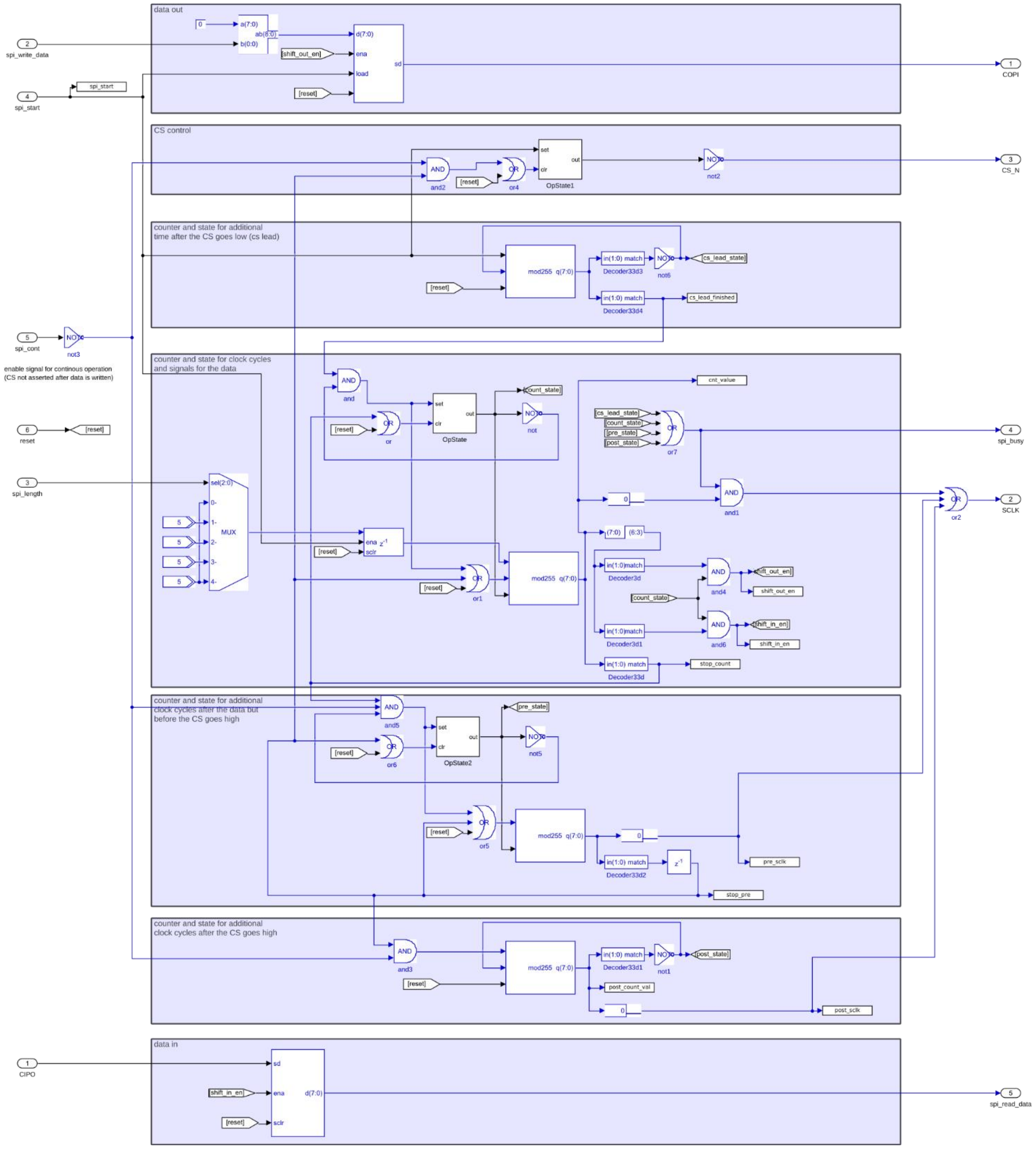


Quartus top-level design of the FPGA evaluation-board-based system

C. SPI Master Intel DSP-Builder Design



DSP-builder top-level design



SPI Master block