

SoC D/M Exercises 09/10

These exercises are allocated marks at Tripos examination level, with 20 marks making a full exam question. Example answers are available to supervisors.

1 RTL Exercises

- RTL1. Give a brief definition of RTL and Synthesisable RTL. Name two example languages. [4 Marks]
- RTL2. Explain Verilog's blocking and non-blocking assignment statements. Show how to exchange the contents of two registers using non-blocking assignment. Show the same using blocking assignment. [6 Marks]
- RTL3. Synthesisable RTL standards require that a variable is updated by at most one thread: is this strictly necessary ?
- RTL4. Explain the terms 'structural hazard' and 'non-fully pipelined'. [4 Marks]
- RTL5. Give a fragment of RTL that implements a counter that wraps after seven clock ticks. [3 Marks]
- RTL6. Give a fragment of RTL that uses two multiply operators but where only one multiplier is needed in the generated hardware. Sketch the output circuit. [3 Marks]
- RTL7. Show an example piece of synchronous RTL before and after inserting an additional pipeline stage. [4 Marks]
- RTL8. Give a concise abstract syntax for an RTL module that uses the 'synthesisable' subset of Verilog or VHDL (structural hierarchy may be ignored). [6 Marks]
- RTL9. Describe possible sources of non-determinism that may arise in synthesisable RTL. [4 Marks]
- RTL10. Give an RTL design for a component that accepts a five-bit input, a clock and a reset and gives a single-bit output that holds when the running sum of the five bit input exceeds 511. [6 Marks]
- RTL11. Give a schematic (circuit) diagram for the design of RTL10. Use adders and/or ALU blocks rather than giving full circuits for an such components. [7 Marks]
- RTL12. Summarise the main differences between synthesisable RTL and general multi-threaded software in terms of programming style and paradigms. [20 Marks].

2 SystemC Components Exercises

- SYSC1. Describe the principle features of SystemC. [5 Marks]
- SYSC2. How is an RTL-style non-blocking assignment achieved in SystemC ? [5 Marks]
- SYSC3. How is design module heirarchy expressed in SystemC and what sorts of 'channels' are supported between modules ? [8 Marks]
- SYSC4. Why adapt a general-purpose language like C++ for hardware use when special hardware languages exist ? [2 Marks]
- SYSC5. To what level of detail can a gate-level design be modelled using SystemC, would one ever want to do this and what simulation performance might be achieved ? [5 Marks]
- SYSC6. Give a fragment of SystemC or RTL that relies on its kernel scheduler to correctly implement non-blocking updates (avoiding shoot-through) and then give an equivalent fragment of pure C that has the same behaviour but which does not need support from a scheduler or other library. [10 Marks]
- SYSC7. Show how a user-defined abstract datatype can be passed along a SystemC 2.0 channel by sketching the code for a packet switch, router or demultiplexer. *This was lectured but is partly a legacy question because today we would rather pass such data using TLM 2.0 convenience sockets.* [7 Marks]
- SYSC8. How does SystemC model registers that have widths not native to the C language ? [4 Marks]

- SYSC9. Give synthesisable SystemC for a five-bit synchronous counter that counts up or down dependent on an input signal. *You should sketch C code that looks roughly like RTL rather than worrying about a precise definition of synthesisable.* [5 Marks]
- SYSC10. Give the SystemC synthesisable equivalent design for the design of RTL10. *You should sketch C code that looks roughly like RTL rather than worrying about a precise definition of synthesisable.* [7 Marks]
- SYSC11. Repeat the previous exercise using a slightly more complex design: e.g. a long division component.
- SYSC12. Define suitable nets for a simplex interface that transfers bytes using a four phase-handshake. Describe the protocol. *Answer this part using RTL or natural language. You may assume a suitably high-frequency clock is available that will not alias the protocol.* [5 Marks]
- SYSC13. Sketch RTL for a counter module that writes its output to the four-phase interface of SYSC12.. *Precise syntax and operational details are unimportant, but a sensible answer would be a Verilog module that increments once for each output operation and wraps after decimal 255 back to zero.* [5 Marks]
- SYSC14. Sketch code for a blocking transactor that writes to a four-phase, net-level interface and also some client code for it that, when the two are combined, gives it equivalent functionality to the module of SYSC13. *Answer this time using basic C-like syntax: later (ESL29.) you are asked to use a TLM library.* [6 Marks]
- SYSC15. Extend SYSC14. with further code for a transactor that owns its own thread and is a net-level client for the four phase handshake that makes an upcall to a user-provided function for each byte received. [4 Marks]

3 SoC Components Exercises

- SOC1. What is meant by polled I/O and how does it compare with interrupt driven I/O ? [4 Marks]
- SOC2. Sketch a set of typical macro definitions in C suitable for making low-level hardware access to a UART or similar device that contains status, control and data registers. [4 Marks]
- SOC3. Give a pair of short subroutines in C that perform polled-mode, blocking read and write operations using your macros of SOC2.. [4 Marks]
- SOC4. Sketch the RTL or SystemC code for an interrupt arbiter that stores eight vectors with individual interrupt enable flags. The arbiter monitors eight interrupt inputs and presents the highest-priority, non-masked interrupt vector to the processor when the processor asserts an interrupt acknowledge signal. *Fine details will vary from answer to answer. Syntactic accuracy would not be expected in examination answers.* [10 Marks]
- SOC5. How does the processor set up the interrupt arbiter device of SOC4. and what must it do after servicing an interrupt ? [4 Marks]
- SOC6. How would you make an interrupt arbiter that shares work over two CPUs ? Is this always a good idea ? [6 Marks]
- SOC7. Give a programming model for a simple DMA controller with one control/status register and three operand registers for block length and source and destination addresses. *The DMA (direct memory access) controller, when active, becomes a bus master and copies a block of data from one area to another, generating an interrupt on completion.* [4 Marks]
Answer: There's not much to say here: Just a gentle introduction to the
 b) Sketch a full implementation of such a DMA controller that includes provision for slave access to the programmable registers, active bus mastership and interrupt generation. Memory access should use a high-level modelling style that ignores bus arbitration. *Answer preferably using SystemC syntax, or pseudocode at the same level of abstraction. Use RTL if and where needed or preferred.* [7 Marks]
- SOC8. Say with justification whether your SystemC DMA controller could be synthesised into RTL for use in a real SoC ? [3 Marks]

SOC9. Bus Bridge.

- a) What is the function of a bus bridge in a SoC ? [2 Marks]
- b) What typical address translation semantics might a bus bridge implement ? [4 Marks]
- c) How might internal queue structure vary between bus bridge designs ? [3 Marks]
- d) How might arbitration policy vary between bus bridge designs ? [3 Marks]

SOC10. Input and Output to a Network Controller

- a) Sketch the structural schematic symbol for a generic network block that is bus target only, giving full details and descriptions of the signals used to connect to a typical system bus. *The network type or internal structure does not matter, it could be Ethernet, USB, Firewire etc..* [6 Marks]
- b) What advantages are there to giving the network block the capability of being a bus master? [2 Marks]
- c) Describe the additional signals needed to make the network block a bus master. [6 Marks]
- d) Assuming the device can be a bus master, sketch the code for a typical device driver. [6 Marks]

SOC11. Define a feasible serial interface to an audio output DAC that conveys a pair of stereo channels of 16 bit precision at 44.1 ksps. *Hint: Three nets are normally used.* [4 Marks]

SOC12. Sketch the block diagram or RTL for a simple audio output controller that uses DMA to send a serial audio data-stream to a DAC. Include the full programmers' model. [12 Marks]

SOC13. Structural Hazards.

- a) Why might a memory cause structural hazards and how does the number of ports on the memory affect the problem? [5 Marks]
- b) Compare the structural hazards and other relative merits arising from on-chip RAM and off-chip DRAM. [5 Marks]
- c) How is an on-chip RAM tested and what effect does this have on user-level RTL ? [5 Marks]
- d) Compare a register-file synthesised from flip-flops with an on-chip SRAM macrocell ? [5 Marks]

SOC14. Clock Domain Crossing.

- a) List basic principles used in the design of a reliable clock-domain crossing bridge to avoid metastability problems and achieve reliable transfer of data ? [6 Marks]
- b) Sketch the RTL or block diagram for a simplex clock crossing bridge that internally uses one parallel data bus and four-phase handshake ? *If giving RTL, only the receiving side logic is needed.* [6 Marks]
- c) What constraints exist for simplex protocols that cross clock domains ? [6 Marks]
- d) What constraints exist for duplex protocols that span clock domains ? [2 Marks]

SOC15. Single-Bit DAC/ADC

Single-Bit DAC/ADC was not lectured in 09/10.

- a) What is the advantage of a 'single-bit' digital to analog converter over older techniques ? [5 Marks]
- b) Give either the circuit for or RTL design of a pulse density modulator that accepts a five-bit input word. [5 Marks]
- c) Give a lower bound on the word rate at the input to five-bit modulator for CD-quality audio (44.1 ksps, 16 bits). [5 Marks]
- d) Give and explain the block diagram for a CD-quality delta-sigma analog to digital convertor. [5 Marks]

4 ESL (Electronic System Level) Exercises

ESL1. Define a transaction in Computer Science. How does the ESL use of this term differ ? [5 Marks]

ESL2. What is the difference between a blocking and non-blocking transaction in terms of implementation, efficiency and callability? [6 Marks]

- ESL3. Sketch SystemC code for a shim function that converts a transactional port from blocking to non-blocking, or vice versa. [5 Marks]
- ESL4. Add a simple transactional entry point to the five-bit counter of SYSC9. that allows a remote client to make a five-bit, asynchronous parallel load of a value using a TLM call. [4 Marks]
- ESL5. Restructure your answer of ESL4. so that the five-bit counter has a hardware-style parallel load and remains synthesisable and use a separate transactor to convert the TLM parallel load into a net-level parallel load. (You may ignore contention with other, simultaneous net-level operations on the counter.) [7 Marks]
- ESL6. Give two ways that timing annotations embedded in a transactional level call can be synchronised with system global time ? [5 Marks]
- ESL7. Sketch a templated TLM SystemC model for a basic FIFO with capacity 8 items. [8 Marks]
- ESL8. Sketch code that will join two such TLM FIFOs together to make a longer FIFO. [5 Marks]
- ESL9. Sketch synthesisable SystemC or RTL-like code for such a FIFO (using either a circular buffer in a RAM or else based on a multi-stage structure). *This is rather straightforward exercise, but it is useful preparation for the next one!* [5 Marks]
- ESL10. Sketch code for a transactor (one of several possible) that enables interworking between the TLM and Synthesisable FIFOs of ESL7. and ESL9.. [5 Marks]
- ESL11. Sketch a SystemC model of a bus bridge and say what arbitration, queuing and address translation policies it implements. *Hint: a high-level model will likely lead to the shortest answer. Syntax details are unimportant and pseudocode is acceptable.* [8 Marks]
- ESL12. Sketch a block diagram for a SoC containing at least two identical processor cores, a DRAM controller and some amount of on-chip SRAM. Mark each end of each connection with a suitable port style to be used as part of a TLM model (eg. blocking, non-blocking, initiator, target). [10 Marks]
- ESL13. Roughly estimate (order of magnitude) how many workstation instructions are used when modelling each access to the DRAM. [5 Marks]
- ESL14. Consider how back-door access to a DRAM (or other RAM model) might be implemented, whereby bus cycles for certain traffic, such as instruction fetch, are modelled with less detail. [5 Marks]
- ESL15. What is an ISS (instruction set simulator or emulator) ? [2 Marks]
- ESL16. Consider what simulation performance an ISS might give and can it ever be faster than real time ? (Perhaps mention 'JIT' mode). [5 Marks]
- ESL17. Describe ways that caches can be modelled in conjunction with an ISS. [5 Marks]
- ESL18. Describe a feasible high-level or TLM model of the subsystem of SOC12., whereby the sound can come out of the sound card on the modelling workstation. What problems might arise ? *hint: There is a TLM example of a music playing system, with TLM DAC model, in the additional material on the course web site.* [4 Marks]
- ESL19. a) When an ISS is embedded in a SoC design, what differences can we expect to see when compared with a cycle-accurate model ? [5 Marks]
 b) Why might embedded firmware be cross-compiled to native code for a workstation ? [5 Marks]
 b) Give two or more ways hardware device access can be modelled when firmware is compiled for the modelling platform (i.e. in a mixed-abstraction model). [5 Marks]
 d) What issues of endianness might arise ? How can they be overcome ? [5 Marks]
- ESL20. What problems might arise when using high-level models of systems that use dynamic code loading and self-modifying code ? [5 Marks]
- ESL21. Give alternative definitions of the blocking calls of SOC3. to produce a high-level C/C++ model of a UART device (that just does console or file I/O rather than implementing a full serial port). [4 Marks]

- ESL22. Explain how firmware can be conditionally compiled to either direct calls through the code of SOC3. or instead call the code of ESL21.. (Note, there are two answers to the latter half, where the the bus interface between the components is either modelled or not) [10 Marks]
- ESL23. Briefly describe each of: cycle-accurate, approximately-timed, loosely-timed, untimed. [8 Marks]
- ESL24. Why might a transactional system exhibit different behaviour on the different models ? Is this good or bad ? [2 Marks]
- ESL25. What is the purpose and effect of the timing quantum in the loosely-timed model? [5 Marks]
- ESL26. Explain how different timing models can be used (e.g. loose, approximate, cycle-accurate) in conjunction with your answer to the DMA question (SOC7.) and what bugs in the system architecture might be exposed by each form. [6 Marks]
- ESL27. How can contention for a resource be modelled ?
- ESL28. Sketch code that would measure traffic load and the number of transactions per millisecond at a contention point in an ESL model.
- ESL29. (*Non-examinable*) Re-answer SYSC14. using the full TLM 2.0 syntax with convenience sockets based on the additional material on the course web page.

5 ABD: Assertion-Based Design.

ABD1. : Assertion-based design.

- What is the difference between a safety and liveness assertion over the behaviour of a system. [4 Marks]
- How does a declarative safety assertion differ from an imperative assert statement ? [4 Marks]
- How can safety and liveness assertions be used in dynamic validation ? [5 Marks]
- Give a short segment of RTL or pseudocode that contains an imperative assertion that holds and give also a pair of valid safety and liveness assertions that hold for your code. [7 Marks]

ABD2. : General ABD.

- What are the benefits of the assertion-based design (ABD) methodology ? [5 Marks]
 - Illustrate how a regular expression can be used as part of a safety assertion ? [5 Marks]
 - Using three or more *modelling layers*, describe the PSL reference model. [5 Marks]
- In PSL next-cycle suffix implication uses `|=>` and same-cycle suffix implication uses `|->`.
- Use these two different forms to give a pair of PSL expressions that have identical meaning. [6 Marks]
- See http://www.esperan.com/tutorial/psl_simple.html

ABD3. : Four-phase handshake.

- Give a temporal logic expression that defines a four-phase handshake in PSL-like syntax. [5 Marks]
- Give the synthesisable RTL or circuit for a monitor that checks operation of a four-phase handshake. You may assume a high-frequency clock is available that does not alias any transitions. [5 Marks]
- Discuss whether your PSL specification or circuit are equivalent or in any sense complete. [5 Marks]
- What is automated stimulus generation and can it be usefully applied to interfaces such as the FP H/S? [5 Marks]

ABD4. : Bus Monitors.

- a) What is meant by the formal specification of a protocol and what is a bus monitor ? [5 Marks]
- b) How are bus monitors used in ABD and what sort of error might be detected (safety of liveness etc.) ? [5 Marks]
- c) How can a bus monitor be used to generate simulation stimulus ? What coverage might be possible ? [5 Marks]
- d) What statistics might a bus monitor collect ? [5 Marks]

ABD5. : PSL Operators and Algorithm.

- a) Why is it recommended to always use a PSL SERES as part of a suffix implication ? [5 Marks]
- b) Describe five infix operators defined in PSL. [5 Marks]
- c) Outline an algorithm for synthesising a pattern detecting automaton from a PSL SERES. *This was not lectured in 0910 but is briefly included in the additional material.*[5 Marks]
- d) How can pattern detectors be combined with suffix implication operators ? [5 Marks]

ABD6. : ABD Methodology.

- a) What is meant by ‘Assertion Based Design’ ? [5 Marks]
- b) Compare the use of assertions and yes/no test wrappers in regression testing ? [5 Marks]
- c) Explain how certain assertions can be re-used at different layers of modelling abstraction (and others not). For example, some might be used for TLM modelling as well as for pre-synthesis and post-synthesis forms of an RTL design. [5 Marks]
- d) What is meant in testing by the term ‘coverage’ and can this be applied to set of assertions ? [5 Marks]

ABD7. : Sequential Equivalence Checker (SEC).

- a) What is the combinational equivalence problem ? What is the role of don’t cares in it ? [5 Marks]
- b) What is meant by sequential equivalence and strong and weak/stuttering bi-simulation ? [5 Marks]
- c) Why might sequential equivalence be violated in a design flow (i.e. SEC gives a negative result)? [5 Marks]
- d) Why might we see false negatives from a SEC ? [5 Marks]

6 SS: SoC Bus and NoC Structures.

SS1. : Cell Library.

- a) Give a short list of logic cells to be found in a standard cell library. [5 Marks]
- b) List five types of information that should be stored about each cell. [5 Marks]
- c) How can an algorithm that chooses an assembler instruction from an instruction set in the back end of a compiler be used for choosing a cell from a cell library have in the back end of a logic synthesiser ? *Non-examinable.* [5 Marks]
- d) Name several illustrative, specialist VLSI structures or components that cannot readily be made out of standard logic cells and explain why custom design is needed. [5 Marks]

SS2. : JTAG Port.

Not lectured in 0910

- a) Why do ASICs commonly support special test modes? [4 Marks]
- b) Define and compare boundary scan with full scan test path [4 Marks]
- c) Briefly describe the structure and operation of the JTAG test port used on many chips. [4 Marks]
- d) How can JTAG ports be combined and is this a good idea within a single SoC ? [4 Marks]
- e) What other uses can the JTAG port frequently be put to ? [4 Marks]

SS3. : SoC Structure.

- a) Sketch the block diagram for a SoC with one processor, one SRAM, one ROM, one Counter/Timer block and one PIO section, all connected to a single bus without any bus bridges. [5 Marks]
- b) List the (main) signals that make up the BVIC port (or an IP block interface of similar functionality) and explain the protocol. [6 Marks]
- c) Is DMA supported or used in the SoC of part *a* and how might it be added ? [3 Marks]
- d) How are interrupt signals routed in the SoC of part *a* ? [3 Marks]
- e) What modifications are needed if a second processor core were to be added ? Is a second bus a good idea ? [3 Marks]

SS4. : Multiple Busses With Bridges.

- a) In SoC terms, what is a bus and how does it compare with the 1980's concept of a motherboard bus (such as the ISA or PCI bus) ? [2 Marks]
- b) How might the destination port for a transaction over such a bus be decided ? [2 Marks]
- c) What is a bus bridge, what transactions might it support and what internal operations might it implement ? [4 Marks]
- d) If a SoC is designed with a number of bridged busses, what are the main aspects that determine the allocation of initiators and targets to the busses ? [3 Marks]
- e) Is there no real difference between a Network On Chip and a set of bus bridges ? [3 Marks]
- f) What form of bus protocol is needed for good performance on a SoC that uses a number of bridges busses or clock domains ? [3 Marks]
- g) How is contention for destinations handled in a SoC that uses a number of bridges busses compared with a NoC (network on chip) ? [3 Marks]

SS5. : Network-On-Chip (NoC).

NB: Detailed NoC material to answer this exercise is not lectured/examinable in 09/10.

- a) What is meant by the term Network-on-Chip and what are the main two differences between using a number of bus bridges and a network fabric? [5 Marks]
- b) Describe two buffering techniques that might be used in a NoC ? [2 Marks]
- c) Describe two flow control techniques used in a NoC ? [2 Marks]
- d) What can be done to avoid NoC deadlock ? How can it be detected ? What should be done when it is detected ? [6 Marks]
- e) What is the flattened-butterfly NoC topology and why is it considered ? [5 Marks]

SS6. : DRAM and Cache.

- What are the main features of DRAM and why is it not commonly integrated as part of a SoC ? [5 Marks]
- Why should out-of-order read responses ideally be supported by a SoC Bus or NoC ? [5 Marks]
- Using a system clock of 400 MHz, a 32 bit MIPS/ARM-like CPU is served without a cache by a 16-bit DRAM system with the following parameters

Operation	Clock cycles	Function
RAS	3	Sending row address,
CAS	1	Read or write 16 bits in current row,
RAS precharge	2	Write back time when finished with row.

Making some assumptions about the pattern of access that the processor will make of the memory, calculate its performance in terms of instructions per second. [5 Marks]

- If all instructions for inner loops are copied to a 32-bit wide on-chip SRAM (that provides true random access at 400 MHz) at code start, what is the performance now. [5 Marks]
- If a cache structure with 98 percent instruction and 80 percent data hit rate is applied, what processor performance is now achieved ? [5 Marks]

7 ST: SoC Tools, Technology and Engineering

TTE1. : Power Consumption

- What are the main components of power consumption in a laptop computer? [5 Marks]
- How does clock frequency affect power consumption ? [5 Marks]
- How might clock frequency be controller in a laptop and for what reasons ? [5 Marks]
- When viewing a DVD (including moving video and audio) on a laptop, what is the best clock frequency policy? [5 Marks]

TTE2. : VLSI Energy Use.

For this question, use the following figures:

Parameter	Value	Unit
Drawn Gate Length	0.08	μm
Metal Layers	6 to 9	layers
Gate Density	400K	gates/ mm^2
Track Width	0.25	μm
Track Spacing	0.25	μm
Gate Output Capacitance	0.06	fF
Gate Input Capacitance	0.03	fF
Tracking Capacitance	1	fF/mm
Core Supply Voltage	0.9 to 1.4	V
FO4 Delay	51	ps
Leakage current	21	nA/gate

A processor core in the above technology uses 200k gates, excluding cache memories. It has two operating conditions: 100 MHz at 0.9 volts or 400 MHz at 1.4 volts. The average net activity ratio during halt is negligible and 0.3 when running.

Give all working and intermediate results. State any additional assumptions you need or use.

- Estimate the area of the processor. [2 Marks]
- Compute the power consumed per gate at each operating condition when driving a tracks of 0 mm and 1 mm. [2 Marks]
- Estimate the power consumption of the processor core when halted and running for each operating condition. [3 Marks]

- d)* Compared with having the processor running at full performance all the time, how much power is saved just by halting the processor when it is idle ? [2 Marks]
- e)* How much power is saved by dynamic frequency scaling ? [2 Marks]
- f)* How does dynamic frequency scaling compare with halting ? [2 Marks]
- g)* How much power is saved by combined dynamic voltage and frequency scaling ? [2 Marks]
- h)* How much power might be saved by power gating (i.e. power isolation) ? [2 Marks]
- i)* Estimate the relative costs of performing a 32 bit addition and sending the 32 bit result 1 mm over the chip [3 Marks]

TTE3. : Dynamic Voltage and Frequency Scaling.

- a)* Give a formula for the power dissipation associated with a net on a silicon chip. [3 Marks]
- b)* What is meant by coarse-grained and fine-grained clock gating ? [3 Marks]
- c)* For a fixed supply voltage, quantify the power benefits of frequency scaling. In other words, compare computing quickly and halting with computing more-slowly and finishing just in time. [3 Marks]
- d)* Give two ways that the supply voltage to a region may be varied? [3 Marks]
- e)* Using variable supply voltages, quantify the power benefits of frequency scaling. [3 Marks]
- f)* Sketch the architecture of an ASIC (or part of) that uses all of these techniques. [5 Marks]

TTE4. : Design Partition

- a)* What are the major costs and risks in SoC development ? [5 Marks]
- b)* What factors commonly influence the choice between using standard parts and an ASIC or SoC ? [5 Marks]
- c)* What factors tend to make a hardware implementation preferable to a software implementation? Give an example of each approach. [5 Marks]
- d)* When is a standard processor preferable to a custom processor ? [5 Marks]

TTE5. : FPGA

- a)* What are the principal differences between an FPGA and a masked ASIC for implementation of a SoC ? [5 Marks]
- b)* How can a SoC design team use FPGAs to prototype their product before SoC fabrication ? [5 Marks]
- c)* When would it be sensible to ship an FPGA instead of a masked ASIC in production runs ? [5 Marks]

TTE6. : Delay and Power

- a)* It is necessary to send a one-bit value a distance of 11 mm over the surface of a silicon chip where the clock available is 300 MHz. Determine how many D-types should be re-used in the path of the signal based on the maximum spacing in millimetres they should have? State any assumptions made. [5 Marks]
- b)* Consider sending a 32-bit value the same distance over the same chip. Compare serial and parallel transmission of the data in terms of latency, throughput and power consumption. [15 Marks]

TTE7. : Static Timing Analysers

- a)* Draw a gate-level circuit for a divide-by-eight synchronous counter. Annotate the timing delays relative to the master clock of each net for a technology that has the following properties: [8 Marks]

Gate	Parameter	Value
AND	propagation delay	0.1 ns
OR	propagation delay	0.1 ns
INV	propagation delay	0.05 ns
XOR	propagation delay	0.15 ns
D-type	clock-to-q time	0.2 ns
D-type	set up time	0.05 ns

b) Describe the algorithm for a static timing analyser and show its operation on your circuit, giving the maximum clock frequency. [7 Marks]

c) Draw a circuit where a static timing analyser will give an overly poor answer. [3 Marks]

TTE8. : : Dynamic Clock Gating.

a) What is dynamic clock gating and why is it used ? [4 Marks]

b) Compare coarse-grained manual and fine-grained automatic clock gating. [4 Marks]

c) Describe some common clock-gate insertion transformations. [6 Marks]

d) Compare dynamic clock gating with power isolation in terms of automation, scale and functionality. [6 Marks]

TTE9. : : Memory Macrocell Generator (RAM Compiler).

a) What input parameters might we expect to give to a generator program that creates multi-ported SRAM memories for use in a System on Chip ? [5 Marks]

b) What output files might we expect from the memory generator program ? [5 Marks]

c) Sketch either a TLM-style or RTL-style simulation model in RTL or SystemC code for a SRAM memory with two read ports and one write port. [5 Marks]

d) What differences in terms of timing and contention might we see if a model of a memory subsystem is populated with TLM-style models of the RAMs compared with RTL-style models. [5 Marks]

Bonus: What problems might there be if the simulation model from part c were fed into a logic synthesiser for use on an actual ASIC or FPGA ?

8 HLS: High-Level Synthesis

This section is not examinable in 09/10 and only parts may be lectured.

HLS1. System Verilog.

a) How does SystemVerilog extend Verilog ? [5 Marks]

b) Does System Verilog promote or discourage higher-level expression of designs ? [10 Marks]

c) In what ways are System Verilog designs different from C-to-gates designs ? [5 Marks]

HLS2. Bluespec System Verilog.

a) What is a Bluespec rule (guarded atomic transaction) ? [5 Marks]

b) How is the programmer's mental model of parallel programming expressed in Bluespec Verilog compared with conventional RTL ? [5 Marks]

c) What is the code explosion problem in high-level synthesis and how does Bluespec avoid it ? [5 Marks]

d) How does higher-level design expression potentially help with timing closure ? [5 Marks]

HLS3. Kiwi Project (C#-to-gates via .net).

- a) What is a **generate** statement, as found in VHDL and Verilog, and how is the same effect achieved in Kiwi ? [5 Marks]
- b) What do parallel programming and hardware design have in common ? [5 Marks]
- c) How have the designers of the Kiwi system exploited this ? [5 Marks]
- d) What might be the interpretation of a thread fork and join in general C-to-gates flows and in Kiwi in particular ? [5 Marks]

HLS4. UML For VLSI Design (Marte Project).

- a) What are the basic roles of a graphical cockpit, such as Eclipse or another GUI, in SoC design ? [5 Marks]
- b) Give an overview of the Marte Project and explain the potential roles of UML in SoC design ? [5 Marks]
- c) How does UML differ from IP-XACT as used in SoC design ? [5 Marks]
- d) List several visualisation tools could usefully be offered in an integrated development environment for SoC design, debugging and evaluation ? [5 Marks]

HLS5. Glue Logic Synthesis.

- a) What is the data conservation principle in interface design ? [2 Marks]
- b) List the major steps in the product method for glue logic synthesis. [5 Marks]
- c) Give four (or more) commonly appearing component joining paradigms. [8 Marks]
- d) Using example fragments of RTL or SystemC-like glue code that joins a pair of interfaces, explain what the user needs to define and what might be synthesised. (*You might choose, as your example, a duplex mailbox that offers blocking target ports on both sides.*) [5 Marks]

HLS6. Transactor Synthesis.

- a) What is a transactor, as used when mixing different modelling styles in ESL ? [5 Marks]
- b) Explain why it might be useful for a common protocol specification to be used both to synthesise bus monitors and to synthesise transactors. [5 Marks]
- c) Name three typical transactor configurations (and explain why the obvious fourth is potentially useless). [5 Marks]
- d) Can glue logic be synthesised from transactor definitions ? [5 Marks]

HLS7. : IP-XACT

- a) What is the purpose of the IP-XACT specification ? [5 Marks]
- b) How can device driver register definitions be kept in step with RTL implementations ? [5 Marks]
- c) What alternatives to IP-XACT might be considered for structural netlists ? [5 Marks]
- d) How might IP-XACT be used in conjunction with transactor synthesis ? [5 Marks]

9 Additional Material

The additional material is not examinable, except for specific examples of TLM modelling that were presented in detail in lectures.

Additional Material: Multipliers and Adders

MA1. Adder Synthesis

a) Implement a function that accepts a pair of lists of nets, least significant net first, and outputs the net lists for an adder with fast carry, in a similar, list form. (NB a fast carry uses gates with many inputs as compared with a ripple carry that uses gates of limited fan-in). [10 Marks]

b) Outline the modifications needed to your function to make it output a subtractor. [2 Marks]

c) Explain how your subtractor can easily implement all six common integer comparison predicates. [2 Marks]

d) (*Lengthy!*) Outline the modifications needed to instead generate a Kogge-Stone adder and say what differences in adder performance this leads to. [6 Marks]

http://www.acsel-lab.com/Projects/fast_adder/adder_designs.htm

MA2. Barrel Rotator

a) In the additional material, the ML code for generating a bit level barrel shifter net list is provided. Last year it was on

<http://www.cl.cam.ac.uk/teaching/0809/SysOnChip/additional/lg1-rtl/barrel.txt>

Modify the provided code to generate a barrel rotator instead and show the output from an example. A rotator re-inserts the bits lost at the other end of the word. Also, modify the provided code to generate an arithmetic shift instead of a logical shift.

MA3. Sketch a micro-architecture (data path) suitable for long multiplication of 32 bit unsigned operands (using Booth's method if you recall it). *If you have the lecture notes to hand, do not copy out the answer verbatim, but instead design the controller for the micro-architecture.* [6 Marks]

N.b. Booth's algorithm for base four is similar, except two bits of **b** are looked at each time and the operands are shifted two bits.

MA4. How many clock cycles does the design of MA3. use on average and in the worst case? [5 Marks]

MA5. If a synthesisable RTL program uses asterisks for multiplication, what is typically placed on an ASIC or on an FPGA and what problems might there be ? [5 Marks]

Additional Material: Compilation and Simulation Algorithms

CSA1. RTL Syntax and Semantics.

a) Outline an algorithm for converting a set of threads in your abstract syntax into a form where each register or net is assigned from exactly one expression. (Full marks will be awarded for answers that only consider one of the following types of assignment: signal, variable, blocking and non-blocking). [6 Marks]

b) Give an example where your algorithm may fail to resolve name aliases. [4 Marks]

CSA2. Outline an algorithm to convert an abstract syntax representation (tree) of a Verilog continuous assignment into gates. [8 Marks]

CSA3. What would the an algorithm like the one you gave for CSA2. do for the problem of RTL6. ? [3 Marks]

CSA4. Give three or more reasons why the basic algorithm from CSA2. may not always be appropriate. [3 Marks]

CSA5. Compute/Commit Cycle.

a) In VHDL (and SystemC), why are both signals and variables provided and what is their difference ? [5 Marks]

b) Describe the compute/commit evaluation paradigm used with signals (also used by Verilog's non-blocking assignments). [5 Marks]

c) What are the consequences of using signals instead of variables in clock distribution trees ? [5 Marks]

d) What is meant by a 'delta cycle' in an event-driven hardware simulator (EDS). Discuss whether they are a good or bad thing to have? [5 Marks]

CSA6. Communication Styles.

- a) Explain when each of the following are used to communicate between components in a SoC simulation: event, variable, net, signal, transaction. [2 Marks Each]
- b) Using a bus bridge as an example, explain which model of communication is best for which situations. [7 Marks]
- c) Explain why it is uncommon for a SoC to have a uniform memory architecture, even if it has a single logical address space. [3 Marks]

Additional Material: Engineering and Physical Considerations.

EP1. : Logical Effort

NB: Detailed material to answer this question is unlikely to be lectured this year.

- a) When sending a signal a long distance over a chip, compare using powerful drivers with a repeater arrangement that uses a larger number of less-powerful drivers. [5 Marks]
- b) When building a multi-stage logic circuit, what arrangement gives least area ? [5 Marks]
- c) When building a multi-stage logic circuit, what arrangement gives least power ? [5 Marks]
- d) When building a multi-stage logic circuit, what arrangement gives lowest delay ? [5 Marks]

EP2. : Information Flux.

NB: Detailed material to answer this question may not have been lectured in 09/10.

- a) How many signal nets per square micron can be routed in a vertical plane in modern VLSI ? [5 Marks]
- b) How does the power required to drive a signal net vary with its planar density and length ? [5 Marks]
- c) What is the maximum information flux feasible in a modern silicon chip ? [5 Marks]
- d) How might we use replicated computation to ameliorate this situation ? [5 Marks]

EP3. : Technology/Scaling.

- a) What is meant by the term *feature size* in VLSI ? Give typical values. [5 Marks]
- b) What are the main consequences of moving to a smaller feature size in VLSI fabrication ? [5 Marks]
- c) What happens to the relative costs of computation and communication as features get smaller ? [5 Marks]
- d) Why has parallel computation become more important than ever before ? [5 Marks]

EP4. : Cost and Power

- a) Summarise the historical trends that affect the relative merits of FPGA and custom silicon in consumer, professional and military, mains-powered applications [5 Marks].
- b) How does the argument differ for battery-powered devices ? [5 Marks]
- c) What are the main power consuming components in FPGA, embedded processors, custom silicon and programmable core silicon ? [5 Marks]
- d) Discuss whether multi-core processor chips can/should take over from FPGA and custom silicon in various applications. Consider Picochip, XMOS and ARC if you are familiar with them. [5 Marks]