Exercise 8: SystemC and Virtual Prototyping

Exercise on TLM Payload Extensions

Matthias Jung
WS 2017/2018

The source code to start this execise is available here:
`https://github.com/TUK-SCVP/SCVP.Exercise8`

Task 1

Setup of TLM AT Model

Setup a TLM AT simulation model of the system shown in Figure 1. The cpu modles and the memory models should use *Simple Sockets*. The target has an input buffer with the size of 8 and should implement backpressure. The bus should use *Multipasstrough Sockets* and should use *Payload Extensions* for the routing.

To make things easier for you, you can use the code, which is provided in the github URL above.

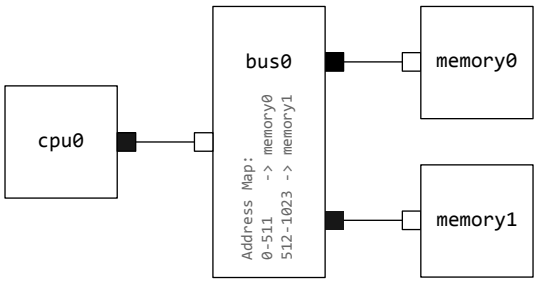

Fig. 1: Initiators Interconnect and Target

The initiators of these system model the usual behavior of the application that we execute on them. If you execute the model you can observe how the buffers of the targets are behaving on the simulation output. By running the simulation with the following command, we can estimate the maximum number of transactions in the target buffer.

```
$ ./tlm_advanced | grep Buffer | awk '{print $9}' | sort | tail -n 1
```

Try to understand this bash scripting.

Task 2

Designspace Exploration of the Buffer Size

So it seams that the targetbuffersize is over engineered. The target buffer size can be reduced to 6 without any problems like this:

```
Target * memory1    = new Target("M1", 6);
Target * memory2    = new Target("M2", 6);
```

Please verify that the simulation time has not changed. However, the greedy management of our company forces us to save even more resources. So we have to find a tradeoff between execution speed and buffer size.

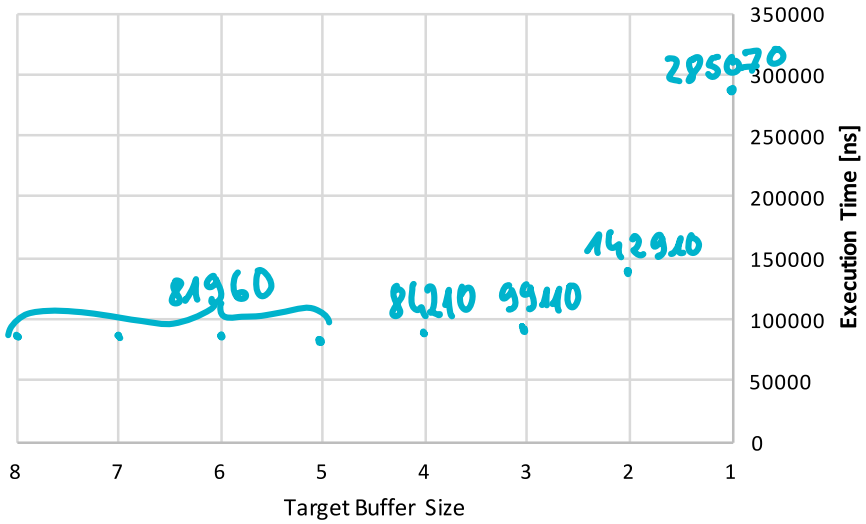

Fig. 2: Tradeoff Estimation

Very the buffersize from 8 to 1 and document the simulation time i.e. the execution time of the CPU in Figure 2.

Which buffersize is reasonable to fulfill the request of the management?

Task 3

Wallclock Time vs. cout

Our simulation has a lot of `cout` statements. Measure the simulation time by running:

```
$ time ./tlm_advanced
```

Now, comment out all cout statements in the simulation models and measure the wallclock time again.

How much difference do you see?

with cout

real 0m10.209s
user 0m0.468s
sys 0m0.244s

without cout

real 0m0.063s
user 0m0.048s
sys 0m0.000s