CRU Simulation and Verification

Ola Grøttvik – August 23rd, 2017

- Currently 3 simulation setups
  - cru_misc
  - gbt_wrapper
  - datapath_wrapper
- Usage of Bitvis UVVM for randomization and concurrency
  - Especially useful for testing multiple interfaces at once
  - Produces log with all necessary information
  - Control of verbosity
- Current version reflects internal release v20170820 + bugfix

- Simulation folder must be placed with the core-folder to function
- All simulation runs are automated by tcl-scripts
- Howto for run is found in each simulations README.md
- Updating core versions must be reflected in common.ini which serves as the basis for all created prerequisites for the simulations

```
[structure]
cores_folder=pcoresng
[core_versions]
cru_misc=cru_misc_v2_00_a
clkmeas=clkmeas_v2_01_a
datapath_wrapper=datapath_wrapper_v1_03_a
gbt_core=gbt_core_v5_00_a
gbt_link=gbt_link_v5_00_a
gbt_sc=gbt_sc_v3_00_a
gbt_wrapper=gbt_wrapper_v5_00_a
```

```
├── pcoresng
│   ├── bsp_generic_v2_02_b
│   ├── clkmeas_v2_01_a
│   ├── cru_dummy_userlogic_v2_02_a
│   ├── cru_i2c_v1_04_c
│   ├── cru_misc_v2_00_a
│   ├── ctpemu_v1_00_a
│   ├── datapath_wrapper_v1_02_a
│   ├── datapath_wrapper_v1_03_a
│   ├── gbt_core_v5_00_a
│   ├── gbt_link_v5_00_a
│   ├── gbt_sc_v3_00_a
│   ├── gbt_wrapper_v5_00_a
│   ├── onu_core_v1_02_b
│   ├── onu_user_logic_v1_00_c
│   ├── pciedma_v3_00_i
│   ├── ttcpon_wrapper_10ax_v2_04_a
│   └── wfplayer_v1_01_a
└── simulation
    ├── cru_misc
    │   ├── script
    │   ├── sim
    │   └── tb
    ├── datapath_wrapper
    │   ├── gbt_packet_vvc
    │   ├── script
    │   ├── sim
    │   └── tb
    ├── gbt_wrapper
    │   ├── script
    │   ├── sim
    │   └── tb
    └── libraries
        └── UVVM
```

- cru_misc
  - Targets the Avalon IC Module
  - Multiple Avalon VVC for concurrency
  - Multiple levels of ICs and slaves
  - Default checks. Read/write access.
  - Randomization of data
  - Randomization of timing

cru_misc from README.md

$ cd ./sim

$ vsim -do ../script/compile_all_and_sim_cru_misc_tb.do

Example log message

```
ID_BFM   125447.5 ns  AVALON_MM_VVC,0   avalon_mm_check_response(A:x"9005000E",x"7A6CA16A")=>
OK, received data = x"7A6CA16A". 'Check reg'
```

- gbt_wrapper
  - Targets the gbt_wrapper module
  - Register read/write via Avalon VVC
  - Check GBT data transmission
    - GBT frame
    - Wide frame
    - Raw
  - Currently only one link, but will be upgraded

gbt_wrapper from README.md

```
$ cd ./sim
$ quartus_sh -t ../script/generate_ips.tcl
$ vsim -do ../script/compile_all_and_sim_gbt_wrapper_tb.do
```

- datapath_wrapper
  - Check packetized GBT data transmissions to the datapath_wrapper
  - Register read/write via Avalon VVC
  - Multiple concurrent links via GBT Packet VVCs
  - Randomized packet length, data, idle words
  - Check for link ID, packet length, packet drops, etc

datapath_wrapper from README.md

```
$ cd ./sim
$ quartus_sh -t ../script/generate_ips.tcl
$ vsim -do ../script/compile_all_and_sim_tb.do
```

- Status: Work in progress
- Currently in private gitlab repository
  - https://gitlab.cern.ch/ogrottvi/cru_simulation
- Request access: ola.slettevoll.grottvik@cern.ch

UNIVERSITY OF BERGEN

Eco-Lighthouse®