TÉCNICO LISBOA

Computer Electronics

Lecture 9: Putting it all together

The iob-timer design

Verilog/iob-lib: what we have learned

- Create registers with
  - `REG
  - `REG_R
  - `REG_E
  - `REG_AR
  - `REG_ARE
- Create counters with
  - `COUNTER_R / `WRAPCNT_R
  - `COUNTER_RE / `WRAPCNT_RE
  - `COUNTER_ARE / `WRAPCNT_ARE
- Create combinational circuits with `COMB
  - Use blocking assignment '='
  - Missing "if" and "for" and "while", etc

New iob-timer design

Or

How you will prepare your design

New iob-timer design

- Temporarily integrated in iob-soc branch devel

  >git remote add iob-soc git@github.com:IObundle/iob-soc.git #assumed you are in a clone of your fork

  >git fetch iob-soc #downloads all branches from the original iob-soc

  >git checkout iob-soc/devel #replaces master with devel tree

  >git submodule update –init –recursive #get timer, etc

- Iob-soc has very little changes
- It all happens in submodule TIMER
- Run system simulation

  >cp submodules/TIMER/software/example_firmware.c software/firmware/firmware.c #copy example firmware to system

  >make sim #INIT_MEM=1 is now default

- Submodule TIMER to be removed from iob-soc/devel

lob-timer: dissecting from printout

- What causes the printout?
  - A: the printfs in the firmware
- Analyse the firmware
  - Top-level file is software/firmeware/firmware.c
- Remember it was copied from
  - submodules/TIMER/software/example_firmware.c
- Let’s visit and explain everything from there

iob-timer: software tour itinerary

Make sure you can describe each place you visited:

- firmware.c #top-level file
- iob_timer.h #timer header file
- iob_timer.c #timer software driver file
- interconnect.h #read/write to interconnected devices
- software.mk #establish include dirs, defines, header files for sw build
- core.mk #peripheral definitions file
- **sw_reg.v** #description of Software Accessible Registers (SAR)
- mkregs.py #script for producing hw and sw for SAR automatically

Iob-timer: hardware tour itinerary

Make sure you can describe each place you visited:

- iob_lib.vh # hardware macros for easy Verilog use, note Software Accessible Registers (SAR)!
- **sw_reg.v** #declaration of SAR
- sw_reg_gen.v #automatically created SAR
- sw_reg_w.v #autimatically generated containing the widths of SARs
- hardware.mk #hardware definitions file
- iob-timer.v #hardware Verilog description of CPU interface
- timer.v #timer core Verilog file, where the design is
- timer_tb.v #Verilog testbench of timer.v

lob-timer: document tour itinerary

Make sure you can describe each place you visited:

- document/ug/Makefile #Makefile building the user guide
- ug.tex #User guide top text file
- document.mk #makefile segment for documents
- **sw_reg.v** #declaration of SAR
- sw_reg_tab.tex #table of SAR
- bd_tab.tex #table of main hardware blocks in block diagram
- cpu_nat_s_is_tab.tex #table of CPU interface signals
- gen_is_tab.tex #table of general interface signals (clk and rst)