Octopus

TCL Packages for Your Daily Work

Octavian Petre

April 17, 2013

Get the latest presentation from Octopus Repository

Outline

What is Octopus()?

- Collection of TCL packages that should help the development of TCL scripts
  - A TCL package is just a collection of useful procedures.
- Available packages
  - octopus
    TCL package.
  - octopusRC
    RTL Compiler package. Depending on octopus.
  - octopusDS
    DesignSync package. Depending on octopus.
  - octopusNC
    NCsim package. Depending on octopus.
- Other packages will be developed as needed.

WHY?

- I am unable to easily parse many nested procedures for which arguments are parsed by position. No clear API.
  Thus, code readability.
- Inability to enhance procedures without breaking all calls, or use tricks such as: default values or **args** variable.
  Thus, portability.
- Lack of documentation of procedures (RC scripts, test benches, ncsim.... etc.)
  Thus, documentation.
- Structural reuse as opposed to adhoc.
  Thus, traceability.

HOW?
1

- code readability & portability
- Instead of positional arguments

~~`set_attribute_recursive dont_touch true [find * ... ] up`~~

- Use options for calling procedures/scripts:

```
::octopusRC::set_attribute_recursive \
    --attribute dont_touch true \
    --objects [find * ... ] \
    --direction up
```

HOW?
2

- portability
  Easy to extend procedure without breaking previous calls

```
::octopusRC::set_attribute_recursive \
    --attribute dont_touch true \
    --objects [find * ... ] \
    --direction up \
    --ignore_clocks # THIS IS A NEW OPTION!!!
```

HOW?
3

- documentation
  Built in. Serves as comments as well.

```
set var_array(10,attribute) ..."Specify the attribute to be applied. Format is: attribute <true|false>."]
set var_array(30,objects)   ..."Specify the objects for which the attributes will beapplied. e.g. instaces
set var_array(40,direction) ..."Specify the direction of recursion. up: all parents will get the attribute

set help_tail {
        puts "More information:"
        puts "    --objects:   While pins, or other objects, can be specified, it makes no sense, since re
}
```

Quick Start

- Add OCTOPUS_INSTALL_PATH variable to the shell environment. For bash:

```
export OCTOPUS_INSTALL_PATH=<directory location of octopus.tcl>
```

- Add to your TCL script:

```
lappend auto_path $env(OCTOPUS_INSTALL_PATH)
package require octopus   0.1
# using procedures without ::octopus:: prefix[1]
namespace import ::octopus::*
```

[1] Don't use it for OctopusRC. Conflicts with RC procedures.

Installation

- Latest version available via git repository

  `git clone https://github.com/octavsly/octopus.git`[1]

- Subsequent updates

  `git pull`

[1] git available via cadenv

Documentation

- Octopus Wiki Page available
  Wiki behind the development scripts.
- Help available from every procedure using --help option.
- Standalone utility to display help of all/any/many procedures:

```
joe@moon> export OCTOPUS_INSTALL_PATH=<location of octopus.tcl>
joe@moon> ./info_procedures.tcl
```

Octopus

- Procedures/Scripts argument parsing.
- Messages handling.
- Flow Control.
- Colours, etc.

```
::octopus::extract_check_options_data
::octopus::display_message
::octopus::summary_of_messages

::octopus::set_octopus_color
::octopus::abort_on
::octopus::parse_file_set
::octopus::debug_variables
```

::octopus::extract_check_options_data
1

The Core Procedure.

Can parse procedures and/or script arguments.

```
proc ::octopusRC::define_dft_test_signals args {
    set help_head {
        ::octopus::display_message none "Extracts DfT constraints from SDC set_case_analysis statements"
    }
    # Procedure options parsing
    set var_array(10,timing-modes) [list "--timing-modes" "<none>" "string" "1" "infinity" "$timing_modes"
        "The timing mode(s) the set_case_analysis will be extracted from" ]
      ...
    extract_check_options_data
```

::octopus::extract_check_options_data
2

```
set var_array(10,timing-modes) [list "--timing-modes" "<none>" "string" "1" "infinity" "T1 T2"  "The timi
```

--timing-modes Command line/Procedure option.
⟨*orphaned*⟩ can be used for arguments without command line option.

⟨*none*⟩ Keyword specifying there is no default argument/value for this option. Thus specifying this --timing-modes option is compulsory.

string Keyword specifying that a string is expected as a value to --timing-modes option.
Other possible values are: number, boolean and two more.

::octopus::extract_check_options_data
3

```
set var_array(10,timing-modes) [list "--timing-modes" "<none>" "string" "1" "infinity" "T1 T2" \
    "The timing mode(s) the set_case_analysis will be extracted from" ]
```

1	Minimum number of arguments(values).
infinity	Maximum number of arguments(values).
T1 T2	Allowed values. Empty string means anything.
The timi...	Help associated with the option and displayed when --help is called.
timing_modes	variable will contain all strings/arguments specified after --timing-modes, timing_modes will have a minimum length of 1 and be one of T1 and/or T2.

::octopus::display_message / ::octopus::summary_of_messages

```
::octopus::display_message none \
    "Extracts DfT constraints from SDC set_case_analysis stateme
```

none is message type.
Other types include error, warning, workaround, info, fixme, tip, debug.

```
summary_of_messages infos errors
```

- Displays a user selected list of recorded messages during the flow. Useful at the end of a run.

OctopusRC
Novel procedures

```
::octopusRC::read_dft_abstract_model
::octopusRC::define_dft_test_clocks
::octopusRC::define_dft_test_signals
::octopusRC::constraints_from_tcbs

::octopusRC::generate_list_of_clock_inverters_for_dft
::octopusRC::rec_grouping
::octopusRC::report_power_over_area
```

OctopusRC
Potential Useful

```
::octopusRC::report_attributes
::octopusRC::read_hdl
::octopusRC::output_driver
::octopusRC::set_attribute_recursive

::octopusRC::fan_hierarchical
::octopusRC::advanced_recursive_grouping

::octopusRC::modules_under
```

OctopusRC
Put Some Structure

```
::octopusRC::synthesize
::octopusRC::read_cpf
::octopusRC::write
::octopusRC::elaborate

::octopusRC::set_design_maturity_level
::octopusRC::report_timing
```

::octopusRC::read_dft_abstract_model

```
WARNING:   OBSOLETE PROCEDURE ::octopusRC::read_dft_abstract_model:
WARNING:   ::octopusRC::read_dft_abstract_model will dissapear in the future since RC 12.1 is natively support

Usage:
  ::octopusRC::read_dft_abstract_model  --ctl <string>...<string> [--assume-connected-shift-enable] --module <s

Options:
  --ctl  <string>...<string>       : Specify the CTL file(s) to be read in. If --module option is used then onl
  --assume-connected-shift-enable  : Specify this option if the shift enable is already connected for all CTL i
  --module  <string>               : Specify the module/library cell associated with the ctl file. If missing
  --instance  <string>             : Specify the instance associated with the ctl file. If missing 'Environmen
  --boundary-opto                  : By default, boundary optimization is switched off for modules with CTL ass

General Options:
  --no-colour                      : Turns off colourful output (not recommended). Default value is false.
  --debug-level  <#>               : Displays more debug information during the run. Default value is the calli
  --help                           : This help message. Default value is false.

Note:
  --instance option supports only one instance. Thus, the user is strongly encouraged to use native RC commands
  for defining chains on instances.
```

::octopusRC::constraints_from_tcbs

```
          Extracts the TCB values in a specific mode and writes out constraints

Usage:
  ::octopusRC::constraints_from_tcbs  --tcb-td-file <string>...<string> --mode <string> --exclude-ports <string
                                      --ports <string>...<string> [--no-false-paths] --constraint-file <string
                                      [--append] --design <string> [--no-colour] [--debug-level <#>] [--help]

Options:
    --tcb-td-file  <string>...<string>   : TCB test data file(s).
    --mode  <string>                     : TCB mode for which constant values are extracted.
    --exclude-ports  <string>...<string> : Skip the specified TCB port(s) completely.
    --ports  <string>...<string>         : Only this ports are considered. For the rest a false path constrain
    --no-false-paths                     : No false paths are generated for the unconstrained TCB signals. Def
    --constraint-file  <string>          : The name of the file where the constraints are written into.
    --append                             : Appends into <constraint-file> instead of truncating it. Default va
    --design  <string>                   : Top-Level design.

General Options:
    --no-colour                          : Turns off colourful output (not recommended). Default value is fals
    --debug-level  <#>                   : Displays more debug information during the run. Default value is th
    --help                               : This help message. Default value is false.

          Note:
          --ports option is compulsory for design maturity higher than bronze, if the design is not in applic
                 The reason is that you might hide valid timing paths
```

::octopusRC::define_dft_test_signals

```
Usage:
  ::octopusRC::define_dft_test_signals  --timing-modes <string>...<string> [--skip-signals <string>...<string>]
                                        [--add-signals <string>...<string>] [--test-mode shift|capture] \
                                        [--no-colour] [--debug-level <#>] [--help]

Options:
    --timing-modes  <string>...<string> : The timing mode(s) the set_case_analysis will be extracted from.
    --skip-signals  <string>...<string> : Skip the signal specified in the constraints (not recommended). Defa
    --add-signals  <string>...<string>  : Add more signals then the one specified in the constraints. (not rec
    --test-mode  shift|capture          : The DfT mode the timing mode is associated with. Default value is sh

General Options:
    --no-colour                         : Turns off colourful output (not recommended). Default value is false
    --debug-level  <#>                  : Displays more debug information during the run. Default value is the
    --help                              : This help message. Default value is false.
```

::octopusRC::define_dft_test_clocks

```
Usage:
  ::octopusRC::define_dft_test_clocks  --timing-modes <string>...<string> [--skip-clocks <string>...<string>]

Options:
    --timing-modes  <string>...<string> : The timing mode(s) the clocks will be extracted from.
    --skip-clocks  <string>...<string>  : Skip the clocks specified in the constraints. Default value is false
    --add-clocks  <string>...<string>   : Add more clocks then the one specified in the constraints. Default va

General Options:
    --no-colour                         : Turns off colourful output (not recommended). Default value is false
    --debug-level  <#>                  : Displays more debug information during the run. Default value is the
    --help                              : This help message. Default value is false.
```

Contact & Contribute

- Octavian Petre
- Use git
  git clone https://github.com/octavsly/octopus.git
  Send me the path of your repository to pull your changes from

Usages

Future presentation about:

DieHardUS - TCL Utilities and Scripts for your daily work

- temposync script using native temposync commands and several type of file lists
- nccoex compilation script using irun.
- tempoflow2TCL file list (alpha stage)
- rtlcompiler scripts (will be discontinued due to Cadence reference flow)

Questions

?