SPARK

Quick Reference 4

Proof Guide

Prerequisites

Reading and reference material

- SPARK Book (in particular Chapter 11 “Verification”)
- "Generation of VCs" manual (in the "Reference" folder)
- "Generation of RTCs" manual (in the "Reference" folder)
- Examiner User Manual
- Simplifier User Manual
- POGS User Manual
- Black-Belt SPARK Training course material, in particular the sections
  - "Understanding VCs"
  - "Using the tools for Proof"
  - "The Proof Cycle"

Basic SPARK analysis OK
No semantic errors
No dataflow errors or warnings

Config file OK and selected [see section 4.5 Examiner User Manual]

Base-type assertions in place for all integer types (and for floating types if needed) [see Generation of RTCs Manual, section 5]

SPARK_QR4 (v1.2)
For use with SPARK toolset 8.1 and above

Basics

4 Phases

a. Generate VCs
b. Simplify VCs
c. Run POGS
d. Review results

Phase A – Generate VCs

Basic command line to analyse a single package body:

```
spark -vcg -conf=mycomp my_package.adb
```

OR to analyse all units in a meta-file:

```
spark -vcg -conf=mycomp @all
```

If you already have -config in your spark.sw file, then it's not needed again.

Phase B – Simplify VCs

SPARKSimp is the main command

Example:

```
sparksimp -a –l –p=2
```

Simplifies ALL files, and collects Simplifier Log files, running 2 simplifiers in parallel [see section 8 Simplifier User Manual]

Phase C – Run POGS

POGS collates and summarises proof status

Usage:

```
    pogs
```

generates <cwd>.sum where <cwd> is the name of the current working directory.

See POGS User Manual for more details and command-line options.

Phase D - Review Results

Look at the bottom of POGS Output

1. Check how many VCs undischarged.
   - Well written code should yield <5% undischarged.
2. Look for "VCs Proven False" section - are there any?
   - Yes - these are a definite defect.
3. Review which subprograms/packages have the most undischarged VCs.
   - Start with a "bottom up" view of the programs' call tree.

Review the "Understanding VCs" material from the training course.

Using "Plain Output" mode

Useful for regression analysis to see what's changed from a baseline set of results. Specify the “plain” switch to all three tools:

```
spark -vcg -conf=mycomp -plain @all
sparksimp -a -l -sargs -plain
pogs -i -p
```