NAME

cec − The Columbia Esterel Compiler

SYNOPSIS

cec [**−c** | **−−blif** | **−−verilog** | **−−sm**] [*options...*] *file***.strl**

cec [**−−blif** | **−−verilog**] [*options...*] *file***.sm**

DESCRIPTION

The **cec** command invokes the compiler, translating Esterel source code (suffix **.strl**) into either a C program or a circuit representation in either the BLIF format accepted by the **sis**(1) logic optimizer from Berkeley or the Verilog language. In the most basic form,

cec myfile.strl

compiles the Esterel source file **myfile.strl** into the C file **myfile.c**. A gate-level logic circuit, in the form of a BLIF file named **myfile.blif** can be generated by providing an option:

cec −−blif myfile.strl

Similarly, a Verilog file named **myfile.v** may also be generated:

cec −−verilog myfile.strl

In circuit-generation mode, all local state machines are given a default encoding that can be manually overriden by supplying a **.sm** file. This is done by asking **cec** to generate an **.sm** file, editing it, then asking **cec** to finish its job, e.g.,

cec −−sm myfile.strl

Edit the file **myfile.sm**

cec −−blif myfile.sm

This generates a **.blif** file. Verilog may also be generated:

cec −−verilog myfile.sm

OPTIONS

Output language options

−c Generate a **.c** file as output. This is the default.

−−blif Generate a **.blif** file as output.

−−verilog
Generate a **.v** (Verilog) file as output.

−−sm Generate a **.sm** (state machine) file as output. This file may be edited and later resupplied to **cec** to complete the compilation process and produce either Verilog or BLIF.

Circuit-generation options

−−sis Invoke the **sis**(1) logic synthesis system as part of the compilation process. By default, this runs the standard **script.rugged** optimization script, but this may be changed using the following option.

−−sis−script *script*
Specify the script invoked by **sis**(1) during optimization. This implies **−−sis**

−−pdgblifargs *args*
Specify additional arguments to be passed directly to the **pdgblif** pass. Useful mostly to developers.

C generation options

−a Generate ANSI-compliant C code. This is the default.

−g Generate C code that uses the GCC computed-goto extension. This code is generally faster, but will not compile with compilers that do not support this extension.

Output control options

−B *basename*
Specify the basename of generated files. This normally defaults to the basename of the given **.strl** file, but may be specified explicitly using this option.

−D *directory*
Specify the destination directory for generated files. This defaults to the current working directory.

−K Keep all intermediate files; by default all are deleted after compilation is complete. Most are in XML format. This is useful mostly to developers.

−−keep *extension*
Keep the intermediate file with the given extension. Invoking **cec −−keep** alone lists the possible extensions. Multiple **−−keep** directives may be issued.

Miscellaneous options

−h

−−help Print a usage summary.

−−version
Print the version of the compiler.

−v

−−verbose
Enable verbose mode. Report each internal command as it is executed.

−−logfile *file*
Generate a log file with the given name. This contains version information, the command line that invoked the compiler, and the list of commands invoked during the compilation process.

−−eachcmd *cmd*
Specify a command to precede each command invoked by the compiler. For example, **−−eachcmd time** times each internally-executed command. Useful mostly to developers.

BUGS

Cec does not support certain parts of the Esterel V5 language, including the *pre* operator and tasks. In addition, circuit generation mode does not support variables, external types, functions, and procedures.

The generated circuit can always be improved.

AUTHORS

— Stephen A. Edwards (sedwards@cs.columbia.edu)

— Cristian Soviani (soviani@cs.columbia.edu)

— Jia Zeng (jia@cs.columbia.edu)

SEE ALSO

<http://www.cs.columbia.edu/~sedwards/>

The CMA/INRIA Esterel V5 compiler, available from <http://www.esterel−techologies.com/>

The Esterel V5 Language Primer

Gerard Berry, *The Constructive Semantics of Esterel*

The Icarus Verilog simulator/synthesizer <http://www.icarus.com/eda/verilog/>

The **sis**(1) logic synthesis system.