CGEN LLVM-IR
Design Document

Leonardo Arcari
Politecnico di Milano

February 2018

Contents

1 Introduction

1.1 Scope

This document is meant to provide a resource to those who are going to work with GNU CGEN and my extension to it: CGEN LLVM-IR. The purpose of this paper is to introduce the reader first to GNU CGEN from a code perspective, as GNU CGEN already provides a user guide. The reader will find in this document a code analysis, with a, possibly more clear, description of the main classes in Scheme source code in order to use them effectively.

In second place, I will provide a similar description of the code that I wrote in order to extend GNU CGEN to allow the generation of C++ programs capable of translating binary programs into a semantically equivalent representation in LLVM-IR language.

1.2 Out of scope

In this paper I am not going to describe several topics related to GNU CGEN

- How to run GNU CGEN. There is a manual online for it.[1]
- What are the features of GNU CGEN. There is a manual online for it.[2]
- What is CGEN RTL and what each language feature does. There is a manual online for it.[3]
- How to write a CGEN application to define your CPU architecture in RTL. Guess what? There's a manual online for it.[4]

1.3 Project History

CGEN LLVM-IR generator is part of the project I was assigned to while taking the *Code Transformation and Optimization* course held by Professor G. Agosta[5] in the A.Y. 2017/2018. The idea of extending GNU CGEN, in order to generate C++ translators capable of producing a semantically-equivalent representation in LLVM-IR of a binary for a given architecture, is from Alessandro Di Federico, PhD[6].

[1] https://sourceware.org/cgen/docs/cgen_2.html
[2] https://sourceware.org/cgen/docs/cgen_1.html
[3] https://sourceware.org/cgen/docs/cgen_3.html
[4] https://sourceware.org/cgen/docs/cgen_8.html
[5] https://home.deib.polimi.it/agosta
[6] https://clearmind.me/

2 GNU CGEN

2.1 Introduction to CGEN

2.2 CGEN RTL classes

2.3 Code Analysis

2.3.1 Entry Point

2.3.2 RTL-C Generator

3 CGEN LLVM-IR

3.1 CGEN-IR common

3.2 IR-Gen registers

3.3 IR-Gen decoder

3.4 RTL-CPP Generator