

In this thesis, I designed and implemented a compiler which performs optimizations that reduce the number of low-level floating point operations necessary for a specific task; this involves the optimization of chains of floating point operations as well as the implementation of a “fixed” point data type that allows some floating point operations to be simulated with integer arithmetic. The source language of the compiler is a subset of C, and the destination language is assembly language for a micro-floating point CPU. An instruction-level simulator of the CPU was written to allow testing of the code. A series of test pieces of code was compiled, both with and without optimization, to determine how effective these optimizations were. appa.tex appb.tex biblio.tex chap1.tex contents.tex cover.tex main.tex propcover.tex signature.tex