FP8-17: Software Programmable Signal Processing Platform Analysis Exercises for Episode 2

Andrzej Wąsowski

Monday, 12 February 2007

Exercise 2.1 This exercise demonstrates the compilation environment using the GNU Compiler Collection (gcc) and the Texas Instruments compiler. Feel free to skip it if you feel comfortable with this issues from your previous programming experience.

Download `hello.c` from the website and try to run manually various toolchain components (preprocessor, compiler, assembler, linker). Inspect the files produced in all stages. Use the tools from the GNU suite for your host, or the TI tools for 67xx.

Exercise 2.2 Write a simple (incorrect) C program that exhibits a lexical error. Test it either with gcc or cl6x.

Exercise 2.3 Write a simple (incorrect) C program that exhibits a syntactic error (having all tokens correct). Test either with gcc or cl6x.

Exercise 2.4 [Appel, p.35] Write regular expressions for each of the following:

a. Strings over the alphabet $\{a, b, c\}$ where the first a precedes the first b.

b. Strings over the alphabet $\{a, b, c\}$ with an even number of a's.

c. Binary numbers that are multiples of four.

d. Binary numbers that are greater than 101001.

f. The language of nonnegative integer constants in C, where numbers begining with 0 are octal constants and other numbers are decimal constants.

Exercise 2.5 [Appel Ex. 3.1 p.83] Translate each of these regular expressions into a context free grammar:

a. $((xy^*x)|(yx^*y))$

b. $((0|1)^+"."(0|1)^*)|((0|1)^*"."(0|1)^+)$

Exercise 2.6 Study the grammar of straight line programs, the derivation and the tree for the example (lecture slides).