EECS 222: Embedded System Modeling Lecture 8

Rainer Dömer

doemer@uci.edu

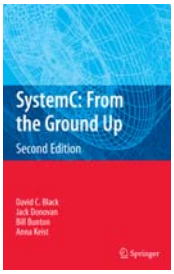
The Henry Samueli School of Engineering
Electrical Engineering and Computer Science
University of California, Irvine

Lecture 8: Overview

- SystemC: From the Ground Up (Part 2)
  - Excerpts from SystemC Training at DAC '15
  - by David Black
- Project Assignment 4
  - SLDL Model of the Canny Edge Detector

The SystemC Language

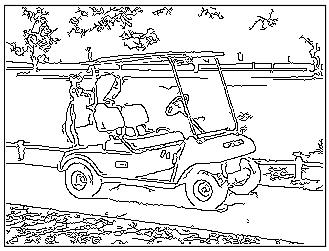
- *"SystemC: From the Ground Up"*, 2nd edition
  - Springer 2010
- SystemC Training Day at DAC 2015
  - *"The Definitive Guide to SystemC: The SystemC Language"*, by David Black, Doulos
  - Core Concepts and Syntax
    - Elaboration and simulation
    - Channels and interfaces
    - Ports and exports
  - Bus Modeling
    - Master and slave interfaces
    - Blocking versus non-blocking



EECS 222 Project

- Application Example: Canny Edge Detector
  - Embedded system model for image processing: Automatic Edge Detection in a Digital Camera

golfcart.pgm



golfcart.pgm_s_0.60_l_0.30_h_0.80.pgm

  - Application Source and Documentation:
    - http://marathon.csee.usf.edu/edge/edge_detection.html
    - http://en.wikipedia.org/wiki/Canny_edge_detector

Review: Project Assignment 1

- Task: Introduction to Application Example
  - Canny Edge Detector
  - Algorithm for edge detection in digital images
- Steps
  1. Setup your Linux programming environment
  2. Download, adjust, and compile the application C code with the GNU C compiler (**`gcc`**)
  3. Study the application
  4. Fix a bug and clean-up the source code
- Deliverables
  - Source code and text file: **`canny.c`**, **`canny.txt`**
- Due
  - Next week: January 16, 2019, 6pm

Project Assignment 4

- Task: SLDL Model of the Canny Edge Detector
  - Convert ANSI-C source code into SLDL model
  - Choose either SpecC or SystemC for simulation
- Steps
  1. Prepare clean SLDL source code without compiler warnings
  2. Fix configuration parameters to compile-time constants
  3. Remove or replace dynamic memory allocation
     - No calls to **`malloc()`**, **`calloc()`**, and **`free()`** in the model
- Deliverables
  - **`Canny.sc`** or **`Canny.cpp`** (choose one!)
  - **`Canny.txt`**
- Due
  - Next week: February 6, 2019, 6pm