

# Real Time Systems

## Uppsala University – Autumn 2015

### Report for Lab 2 by Team 4

Martin Kjellin      Pei-Chun Chen      Jai-Ying Lin

September 28, 2015

## Part 2: Event-Driven Scheduling

In this part of the lab, we were asked to write a program using event-driven scheduling. The program should make a Lego car drive forward as long as a touch sensor is pressed and a light sensor indicates that there is a table underneath the car.

We have defined four different events. `TouchOnEvent` and `TouchOffEvent` indicate that the touch sensor is pressed or released, respectively. `OnTableEvent` and `OffTableEvent` indicate that the reading of the light sensor goes below or above 700, respectively. This threshold was experimentally determined to be a good indicator of the light sensor being above the table or not. The light sensor is mounted in the front of the car, and the car will therefore stop before going over the edge of the table.

According to the instructions, our program includes two tasks. The `EventdispatcherTask` generates the appropriate events when the touch sensor is pressed or released or the light sensor reading goes below or above the threshold. The `MotorcontrolTask` repeatedly first waits for a `TouchOnEvent` and an `OnTableEvent` before starting the motors and then waits for a `TouchOffEvent` or an `OffTableEvent` before stopping the motors.

The code can be found in the files `eventdriven.c` and `eventdriven.oil`.