COMPSYS 723 ASSIGNMENT 2 (ESTEREL) REPORT

Group 11 - Parie Malhotra & Riley Fromont

Department of Electrical and Computer Engineering
University of Auckland, Auckland, New Zealand

Abstract

This is the final report for COMPSYS 723 Assignment 2 which requires a car cruise controller to be implemented in Esterel. The overall structure as well as each individual module of the cruise controller will be discussed here in detail.

1. Introduction

The second assignment for COMPSYS 723 requires a car cruise controller to be implemented in Esterel. The car cruise controller needs to be able to monitor the speed of the car and the state of the cruise controller. As the inputs change, the outputs must also change to reflect the current status of the system. As the system must be both synchronous and reactive, Esterel is the language of choice. In this report, the design and structure of our implementation is discussed.

2. Overall Structure

When designing our cruise controller, we decided to split our system up into separate modules. Each module controls one aspect of the overarching system. The highest level module is the "topLevel" module which runs all of the other modules and emits the expected outputs. Aside from that, there are 4 sub-modules – "drivingControl", "speedManagement", "pedalDetection" and "CruiseControllerState". Each of these submodules are described in the following sections in detail but in general they each perform a specific task within the larger cruise controller system.

2.1. Context Diagram

Below is the context diagram of the cruise controller system. Here the inputs and outputs of the system can be seen denoted by the arrows. Any arrows pointing inwards denote inputs into the system. Conversely, any arrows which point out from the circle denote system outputs.

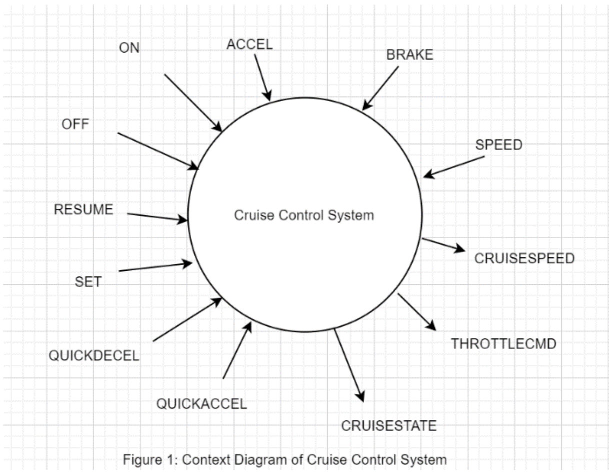

Figure 1: Context Diagram of Cruise Controller System

2.2. Control and Data Flow Diagram

Below is the diagram separating the control and data flow of the cruise controller to show what inputs and outputs the control block consists of and what inputs and outputs the data dominated block consists of as well as how they connect.

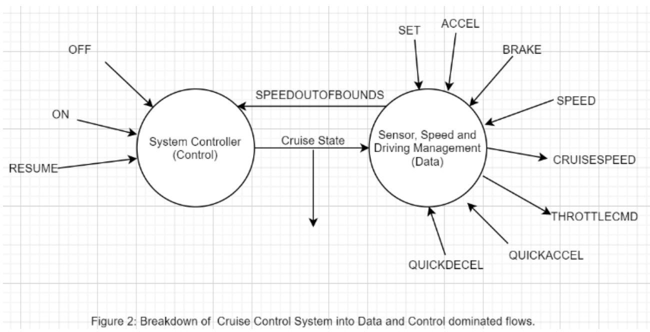

Figure 2: Cruise Control System separated into data and control dominated flow

3. Sub-Modules

3.1. Sensor Environment Control

This submodule is used to control the current state of the vehicle environment. It checks for pedal input from both the accelerator and brake as well as checking if the speed is out of bounds or not. It does all of this by taking in 3 inputs; Accel, Brake and Speed. Using these inputs it computes the output and then emits them. There are 3 outputs; BrakePressed, AccelPressed and SpeedOutOfBounds. This submodule is part of the data dominated side of the cruise controller system, with one of the outputs – SpeedOutOfBounds – being the connection between the data driven side and the control driven side. A diagram of the submodules can be seen below.

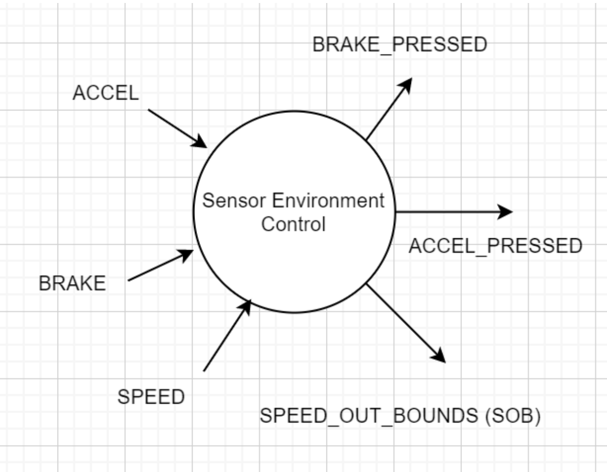

Figure 3: Sensor Environment Control Module Diagram

3.2. Cruise Controller State Machine

This submodule is used to determine the current state of the cruise controller based on several inputs received. In this submodule, 5 inputs are required; On, Off, AccelPressed, BrakePressed and Resume. Some of these inputs come directly from the environment such as On, Off and Reset. Whereas others, such as AccelPressed and BrakePressed, come from other submodules which in this case is the Sensor Environment Control submodule. This submodule only has one output which is CruiseState. This output belongs to the control dominated side of the system and acts not only as a link between and the control and data flow but also as an output back to the external environment. The CruiseState output feeds back information on which state the cruise controller is currently in; On, Off, Standby or Disabled. A diagram of the submodules can be seen below.

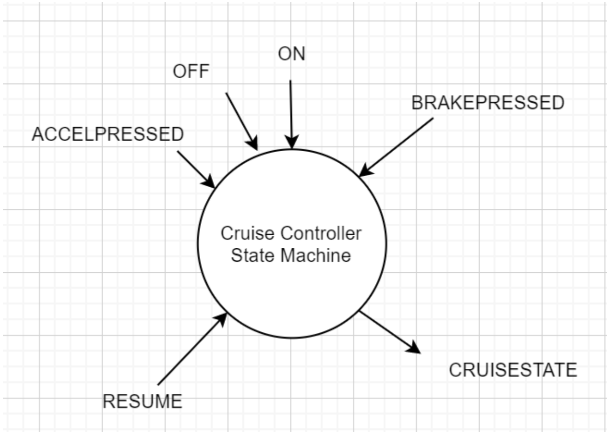

Figure 4: Cruise Controller State Machine Module Diagram

3.3. Car Driving Control

This submodule is used to determine the throttle command of the vehicle. This submodule depends upon 4 inputs (Accel, Speed, CruiseState and CruiseSpeed) to create one output (ThrottleCmd). Depending on the state of the cruise control (On or Off), the throttle command varies. It can become saturated at a predetermined value called ThrottleSatMax when the cruise controller is on. When the cruise controller is off however, the output is largely determined by the accelerator input. A diagram of the submodules can be seen below.

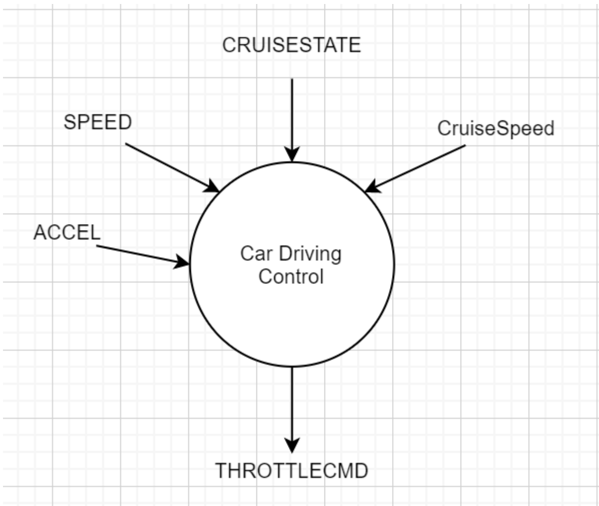

Figure 5: Car Driving Control Module Diagram

3.4. Cruise Speed Management

This submodule is used for controlling the speed of the vehicle. This submodule takes in 5 inputs; Speed, Set, QuickAccel, QuickDecel and CruiseState and produces one output – CruiseSpeed. The output is a float type and varies depending on the particular combination of inputs received. The cruise speed management submodule requires the cruise controller to not be off. Provided this is the case, the CruiseSpeed output is determined by what the current speed of the vehicle is (Speed input) and then which if the actions the cruise controllers wishes to perform (either accelerate – QuickAccel input, decelerate – QuickDecel input or remain at the current speed – Set input). A diagram of the submodules can be seen below.

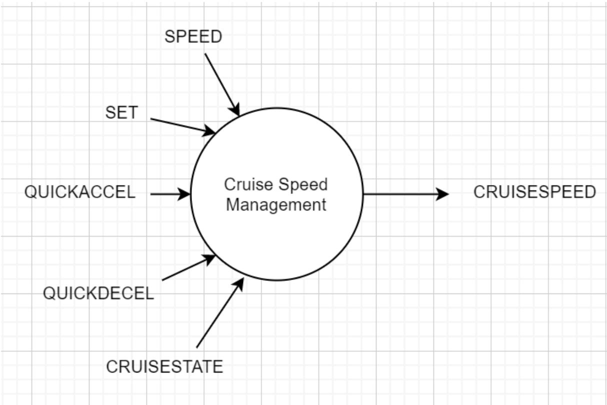

Figure 6: Cruise Speed Management Module Diagram

4. Conclusions

In this report, we have discussed the module design and structure of our cruise controller system implementation. Diagrams have been provided alongside written explanations to ensure a clear understanding of our system is enabled.

5. Work Distribution

The type of work done in this project can be divided up into three sections, Design, Implementation and Report Writing. The time spent by each person on these three sections are as follows: -

Riley:

Design: 4 hours
Implementation: 6 hours
Report Writing: 2 hours

Parie:

Design: 2 hours
Implementation: 4 hours
Report Writing: 6 hours

Appendix A

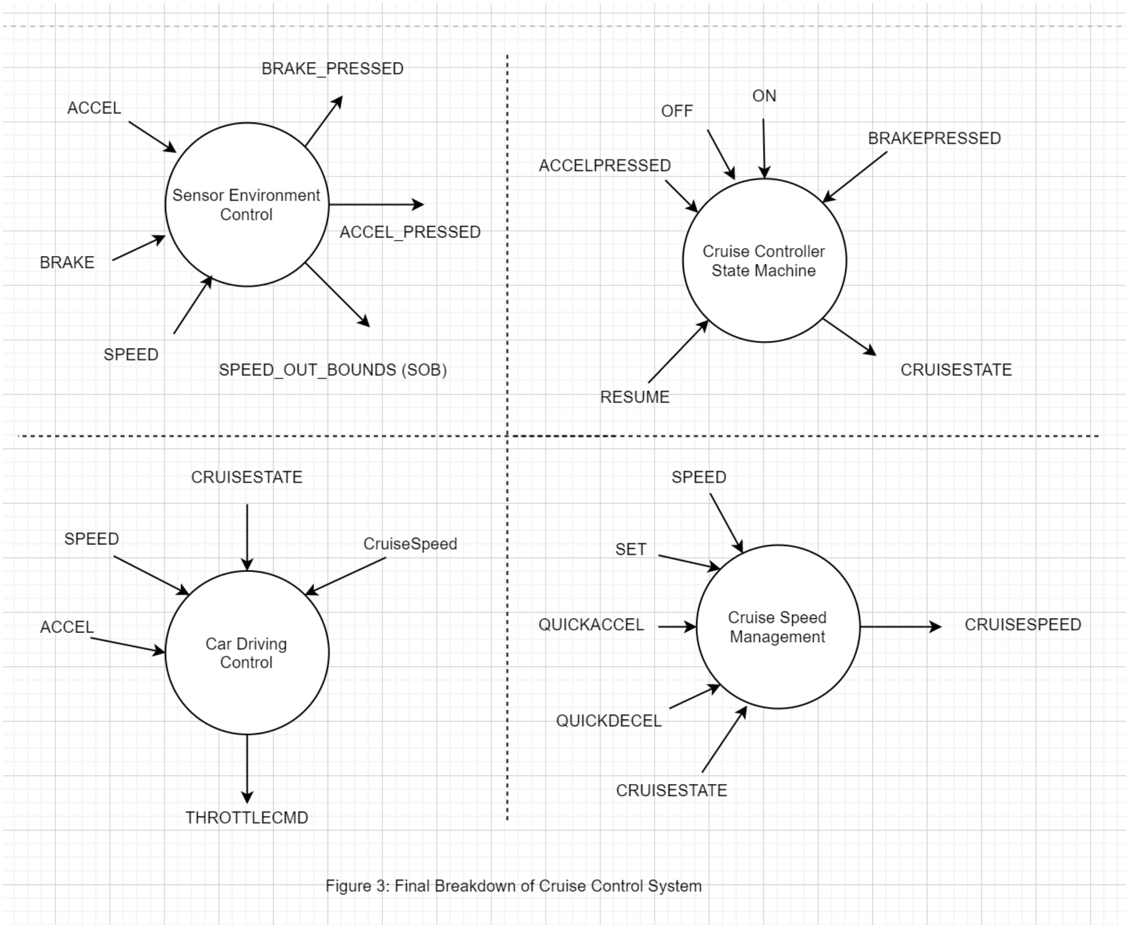

Figure 7: Full Breakdown of Cruise Control System Sub-Modules

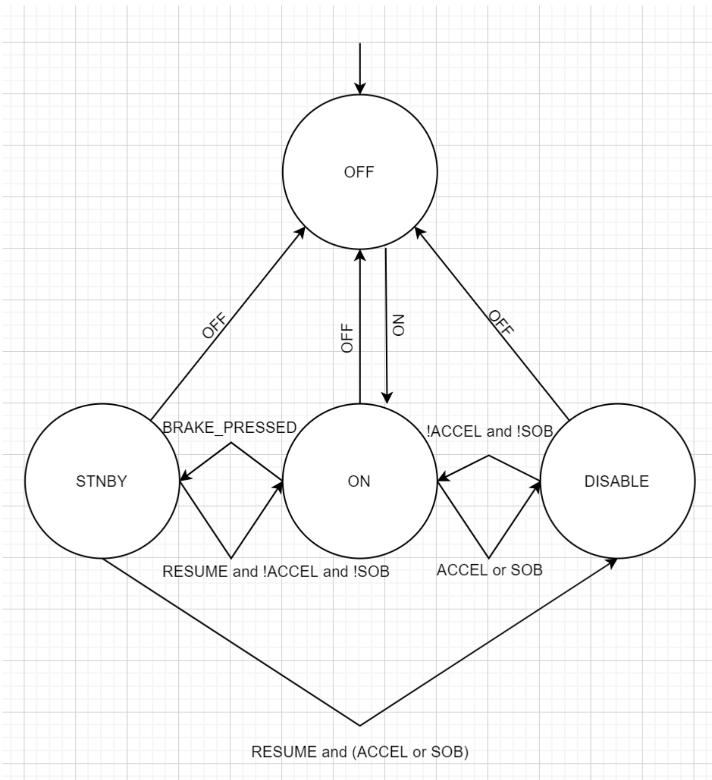

Figure 8: Full FSM of the Cruise Controller