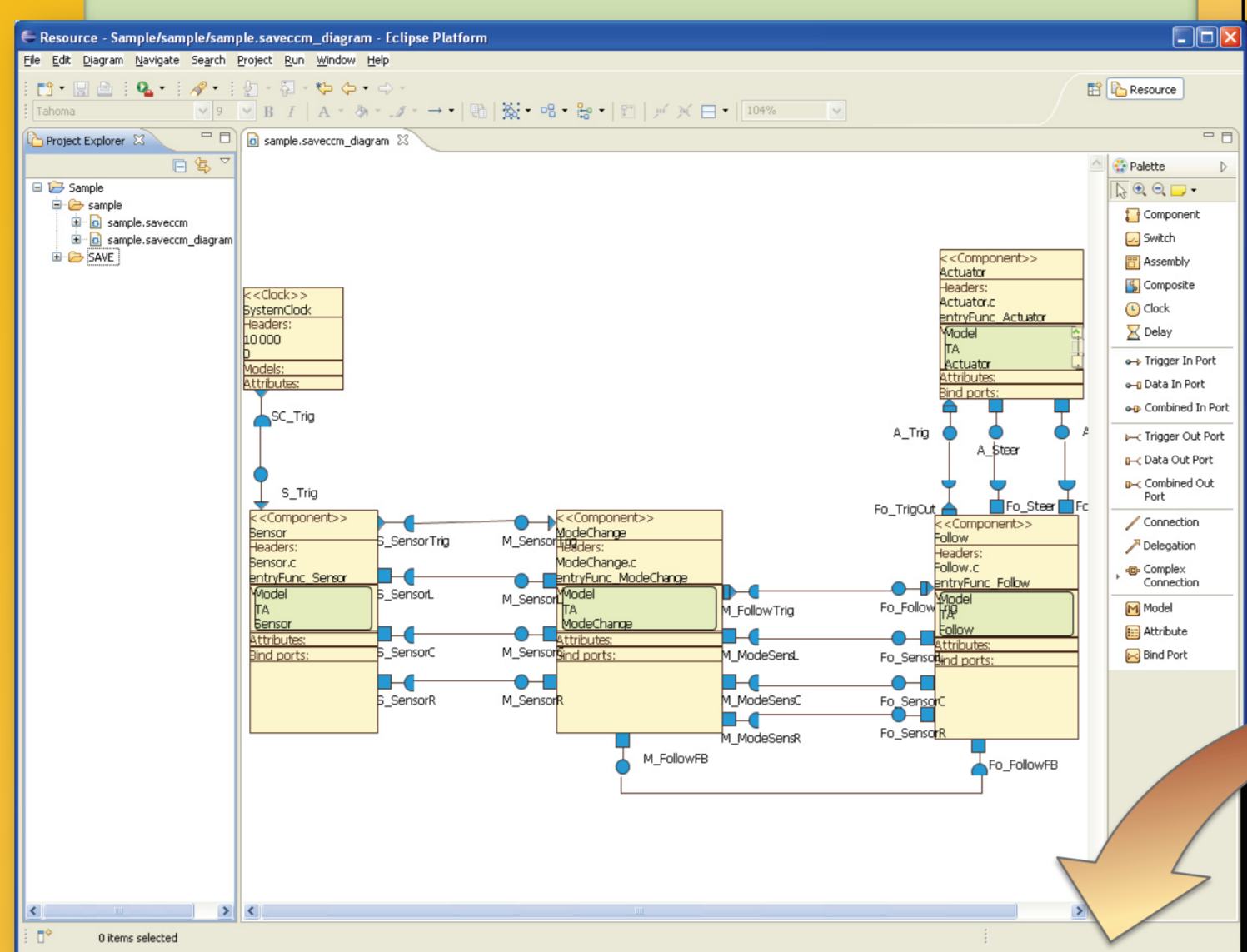


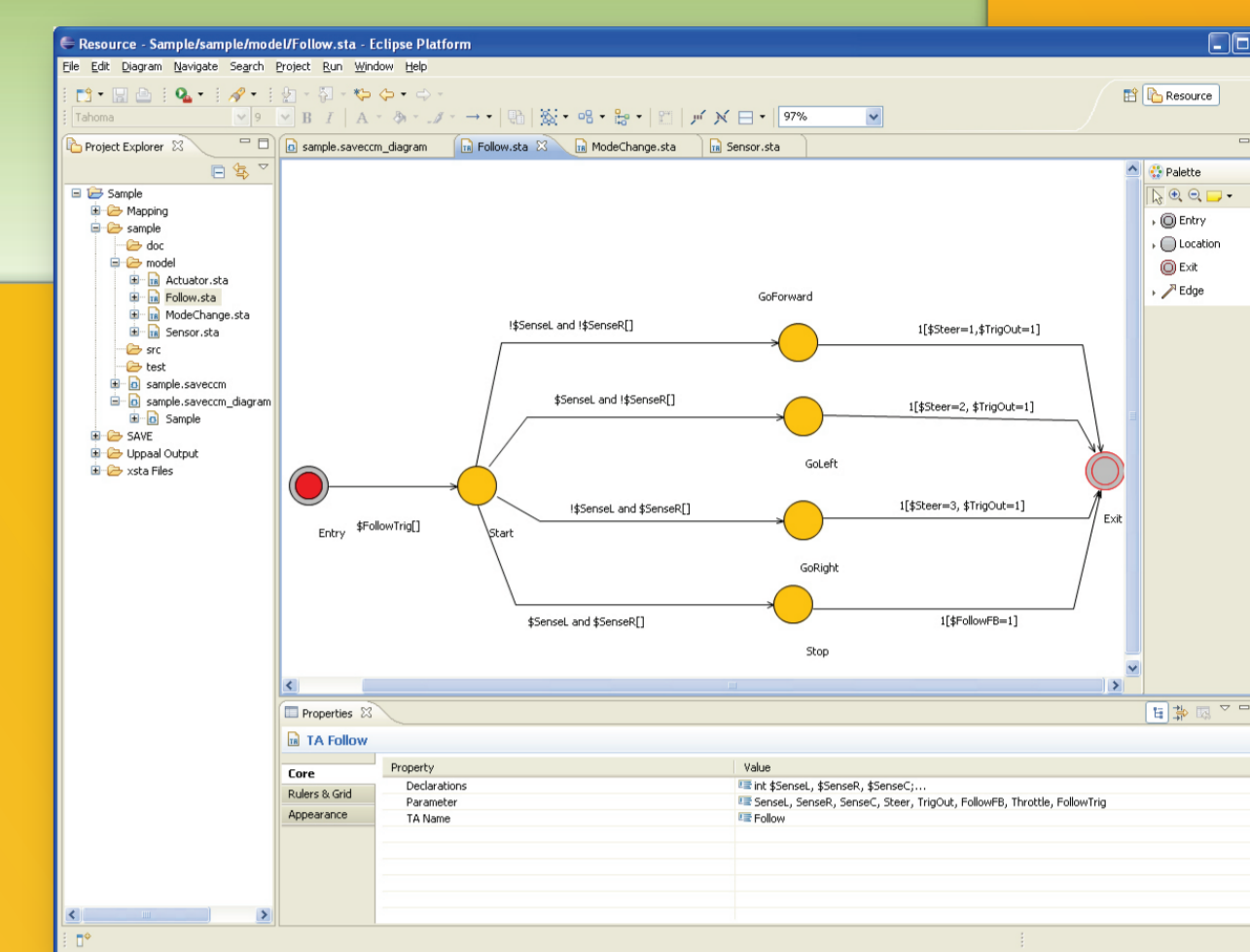
Architecture Editor

- GUI to design a system compliant with SaveCCM component model
 - Component, composite, assembly, switch, ports
- Top-Down decomposition



Timed-Automata Editor

- GUI to create formal behavioural models of components
- Timed automaton but with distinct end location



Behaviour Model Merger

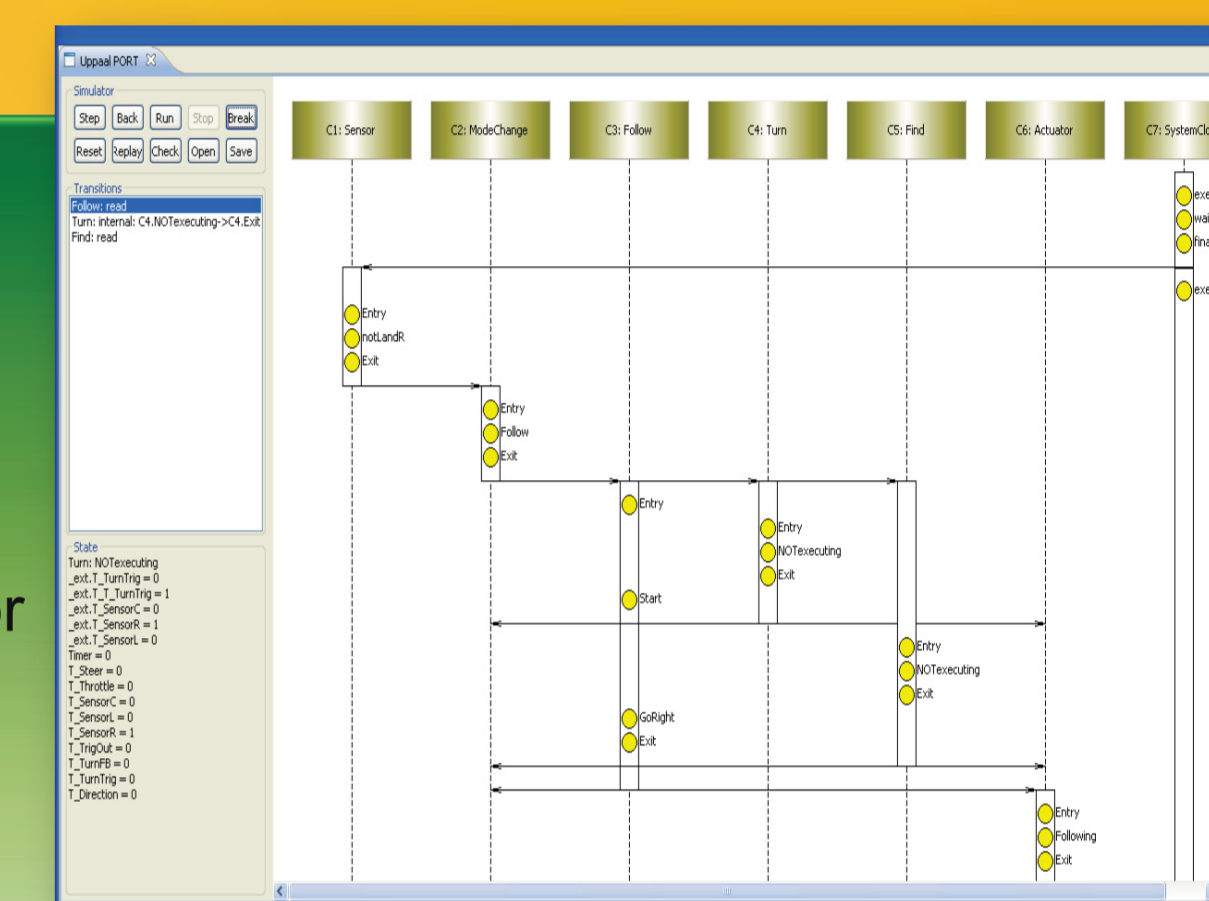
- Links the timed-automata model to the corresponding component (Mapping between port names and TA variables)

Validation & Verification (timing/safety/reachability)

automated

UPPAAL PORT

- GUI for simulation and formal verification
- Simulator:
 - Exploration of dynamic behavior of SaveCCM system in early development phases (prior to implementation)
- Verifier:
 - Model check formal system requirements specified in the logic Timed CTL
 - Partial order reduction exploring system architectural structure



Code Generator

- Generates C-file templates based on design properties

```

#include "generated_model.h"
/* Start of user code of entryFunc_Follow_1 */
/* Please enter your user includes and user globals here. */

/* End of user code */

result_t entryFunc_Follow(Fo_FollowTrig_combinedtype Fo_FollowTrig,
int Fo_SensorL, int Fo_SensorC, int Fo_SensorR, int* Fo_FollowFB,
Fo_TrigOut_combinedtype* Fo_TrigOut, int* Fo_Steer, int* Fo_Throttle) {

/* Start of user code of entryFunc_Follow_2 */
/* Please enter your user declarations here. */

/* End of user code */

/* Start of user code of entryFunc_Follow_3 */
/* Please enter your code here. */

/* End of user code */

return OK;
}
  
```

Synthesis

- Transformation from component model into execution model
 - Tasks, periods, priorities
- Constructs a set of trees based on the application triggers
- Independent of the run-time environment: SaveOS

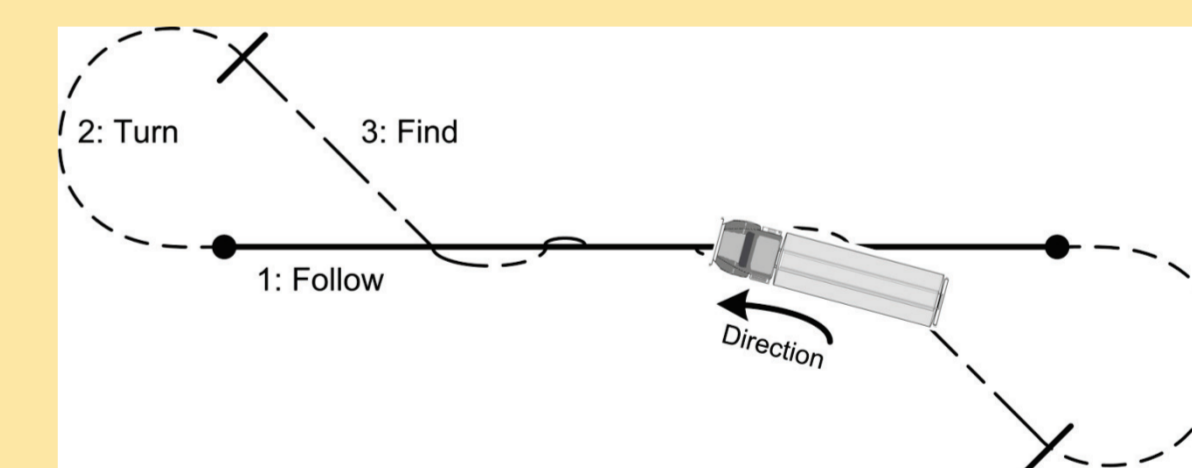
Compiler

Download

Abstract

Save-IDE supports efficient development of dependable embedded systems by providing tools for design of embedded software systems using a dedicated component model, formal specification and analysis of component and system behaviors already in early development phases, and a fully automated transformation of the system of components into an executable image.

The Autonomous Truck Example



Purpose:

Illustrate the use of the Save-IDE in the development process of a component-based real-time embedded systems and the benefit of formal analysis to obtain early predictions of the behaviour of the final system.

Constituents:

- A truck
- A surface with a straight black line with two filled black circles on each end
- An embedded control-program developed in the Save-IDE

Operational Mode:

1- Follow mode

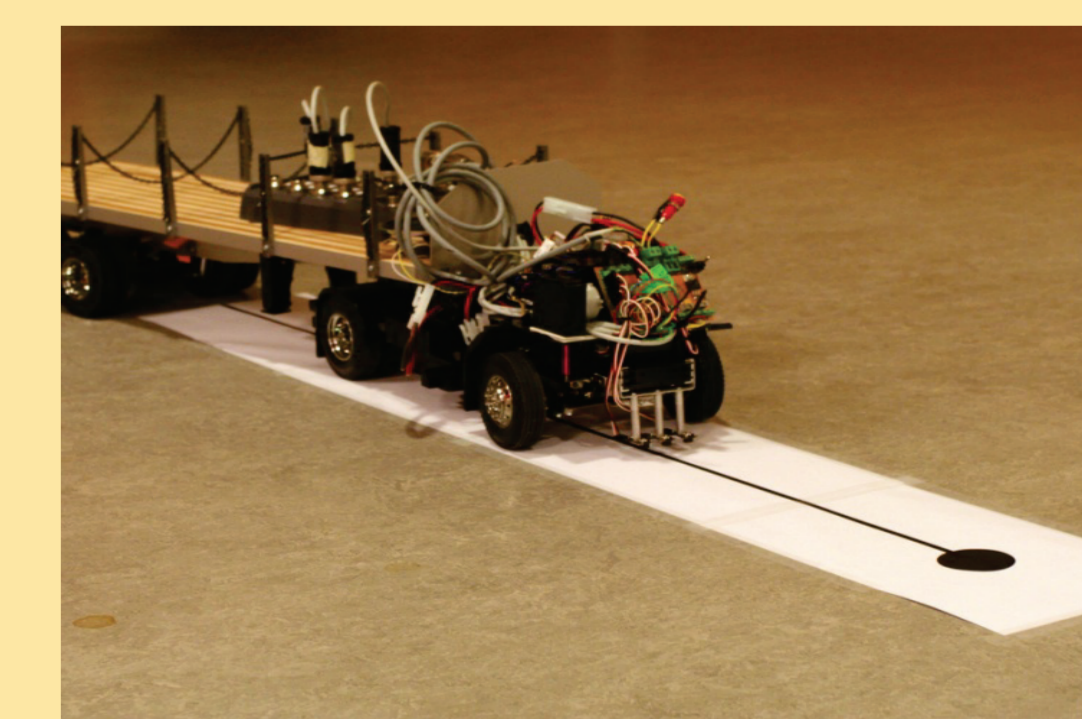
The truck follows the black lines using its light sensors. When the truck detects the end of the line, it changes to Turn mode

2- Turn mode

The truck turns without guidance from any line until it reaches a state where it is suppose to be able to find the line again. Upon completion, the truck changes to Find mode

3- Find Mode

The truck turns without guidance from any line until it reaches a state where it is suppose to be able to find the line again. Upon completion, the truck changes to Find mode



Project Overview

The Save-IDE was developed within the Save project (2002-2008). The aim of the project was to:

“Establish a software engineering discipline for safety-critical real-time embedded systems with the main focus on the vehicular domain.”

The main research outcomes from Save:

- A component model (SaveCCM)
- Various analysis techniques
- Various synthesis techniques

The Save-IDE will be further developed within the PROGRESS project at Mälardalen Real-Time Research Centre (MRTC), and is available for download as open source at:

<http://sourceforge.net/projects/save-ide/>