Title	動的モデルObTSの大規模組込みシステム記述に関する研究
Author(s)	久保秋，真
Citation	
Issue Date	1998-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1137
Rights	
Description	Supervisor:片山 卓也，情報科学研究科，修士

Study of Large Scale System Description based on Dynamic Model ObTS

Shin Kuboaki

School of Information Science,
Japan Advanced Institute of Science and Technology

February 13, 1998

Keywords: object-oriented, reactive system, ObTS, ObCL, ObML.

Recently, embedded systems are used in a great variety of electrical equipments. Complexity and scale of embedded systems are steadily growing as electronic devices like microprocessors are evolved. Then, specifications of many products on embedded systems fields are affected complexity of multi-functions and more flexible user interface. To make matters worse, life cycle of products is going shorter. Therefore, it is important success point to keep a market share that products are developed a short space of time and released timely. In the present situation, designers who develop embedded systems want to reuse both specifications and implementation codes already written, to give and verify detailed description of specifications. State transition diagrams and state transition tables are widely used to specify reactive systems before the great fever of Object-Oriented Methodologies. This fact is a good indication that a dynamic model is important to apply OO methods on developing fields of embedded systems.

Harel's Statecharts are introduced by Booch's methods and Rumbaugh's OMT to describe a dynamic model. Statecharts extend finite-state machine notation; there are three extensions: hierarchy, concurrency, and broadcast communication. There are some other related work: Itou's ObTS, Coleman's Objectcharts, Grangopadhyay's ObjChart, Sane's Object-Oriented State Machines, etc.

ObTS based on Statecharts' concepts is proposed as a dynamic model for OO methods. ObTS is characterized that system specifications are composed by hierarchy of objects, which relates dynamic model with structures of object model. System behavior is specified state transitions, functional evaluation of attributes, and broadcast communications of attributed event. Objects of ObTS include attributes and a state transition chart. Inner objects are delegated some parts of behaviors of parent object. Inner objects include attributes and a state transition chart, too. Locality of behaviors and attributes are increased, because every object has a state transition chart and attributes.

OO methods are not used widely on embedded systems fields. It is a solution to give a opportunity of using OO methods that a way to describe large scale systems and a computer aided environment to examine functional specifications. The research purpose in this thesis is to make availability of ObTS clear on large scale specification by implementing computer aided environment based on ObTS and using it to study on a large scale specification of embedded system.

First, we propose specification description language *ObCL* based on ObTS and introduce simulator *ObML*. ObCL defines concrete syntax for ObTS computation model. There are some extensions to reuse ObCL codes for large scale codes: class,inheritance of them for reusing codes, field for multicast communication, event class for encapsulating attributes of events. ObCL codes are converted into ObML codes. ObML codes can be simulated on ObML which is simulation environment based on ObTS computation model. Simulator called ObML is an interactive environment constructed on Standard ML. ObML includes ObTS computation model as ML functions work with simulation functions. Some simple examples using ObCL/ObML will be shown. And, we will mention ObML environment expansion which supports design process itself and integrated ObCL environment.

Next, a case study using ObCL/ObML to analyze virtual copying machine's operation panel as a large scale embedded system are examined. Scale of the machine is as large as real midrange copying machine. Jacobson's OOSE is used to analyze it and problem domain object model, analysis object model, interaction diagrams, ObTS model diagrams, ObCL codes, ObML codes are gained as results. And we will show availability of ObTS to simulate large scale codes of the result on ObML environment.

Finally, evaluations of ObCL/ObML from the result of case study are given. There are some advantages of readability on a large scale codes, because ObCL codes are well corresponded to ObTS diagrams and interaction diagrams. There is less advantage to reuse existing descriptions, because inheritance in ObCL is able to be used only restricted case.

We found that ObCL and ObML are adaptable for large scale problems though they have any problems.