Proceedings of the Peradeniya University Research Sessions, Sri Lanka, Vol. 15, 16th December 2010

DEVELOPMENT OF AN ELEVATOR PROTOTYPE FOR MULTI-DISCIPLINARY RESEARCH IN COMPLEX REACTIVE SYSTEMS

S.D. Dewasurendra[1] S.G. Abayaratne[2] and A. Vidanapathirana[3]

[1] *Department of Computer Engineering, Faculty of Engineering, University of Peradeniya*
[2] *Department of Electrical & Electronic Engineering, Faculty of Engineering, University of Peradeniya*
[3] *Ceylon Chocolates Ltd., Kundasale, Sri Lanka*

Introduction

Complex reactive systems (CRS) with safety-critical aspects require design techniques that facilitate continued verification and validation of control software/hardware in their life-cycles. There are no integrated laboratory set-ups for these (Drusinsky et al., 2008). The objectives of this research project have been (1) the development of such a platform, (2) to facilitate research in reconfigurable controller design for the target systems on the set-up, (3) use the it to demonstrate how different disciplines of engineering can be brought together for energy saving and (4) demonstrate different operational modes of motors, their control issues and safety considerations in a real-scale complex system. An elevator prototype was built for the purpose. A modelling framework based on statecharts was developed to facilitate experimentation. The current Control implementation is on a Programmable Logic Controller (PLC) and this is being ported to a Field programmable Gate Array (FPGA).

Methodology

For objective 1: a generic CRS was characterised (Dewasurendra, 2006a) and the verification/validation tests developed following (Bringmann and Kramer, 2008) 5 level design:Model-in-the-Loop (MiL), Software-in-the-Loop (SiL), Processor-in-the-Loop (PiL), Hardware-in-the-Loop (HiL) and Real system.

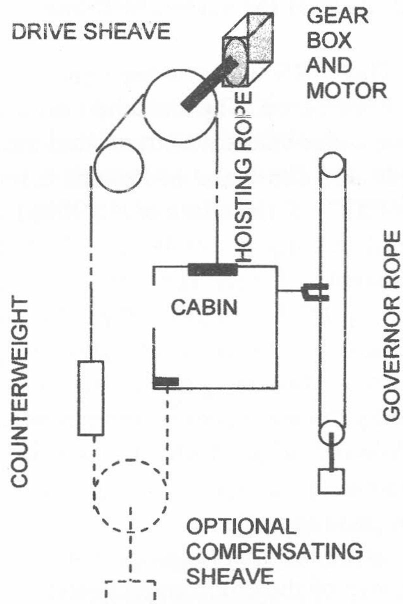

Fig. 1. Simplified physical model.

MiL: functional testing of CRS on simulation environment/s (Simulink-stateflow). The modular verification methodology for statecharts of (Dewasurendra, 2006b) is used: the statechart is decomposed into a set of language generators (LG) communicating through port structures (PS) and controllability and liveness checks done for each LG-PS pair;

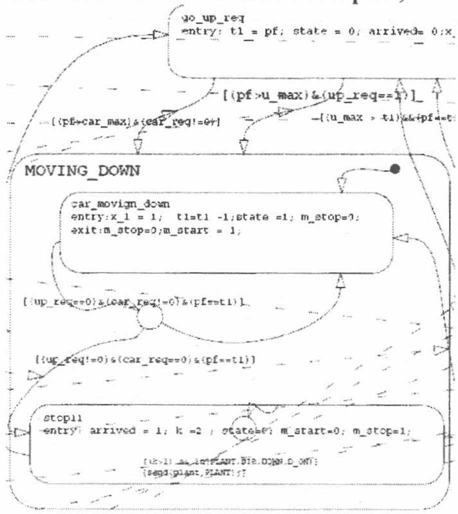

Fig. 2. Part of the stateflow model

SiL: The CRS is software tested in closed/open loop. The statechart model for two super-states was translated into C--code and simulated in Proteus 6 for PIC16F877 C (Bandara et al., 2006);
PiL: PiL testing is similar to SiL, but the control software runs on a target board (FPGA) with the target processor or on a target processor emulator. This is yet to be done. Presently the controller is implemented on a telemecanique TSX PLC which is programmed using SFC (Grafcet) which permits analysis;
HiL: HiL testing requires real-time behaviour of the environment model to ensure that the communication with the controller is the same as in the real application. This is yet to be done;
Real system: The final controller runs in the real system (Elevator).

For objective 2 (reconfigurability): the communicating language generators and the port structures provide an effective method of component-based design. The need for the design objects (the components) to come under version control is addressed by the strategy of local verification of each language generator (which models a component) only with its port structures. This is a useful compositionality result in our design method.

For objective 3 (demonstrating energy saving): a stochastic performance model is to be built from the statechart. The elevator prototype is used to validate the performance model by direct measurement: yet to be done.
Finally for objective4: suitable supervisory controllers are developed as outlined in (Dewasurendra, 2006a). Modular supervisors with full/partial observability are facilitated by component based design: in progress. Speed reference pattern generation and position control strategy for the elevator involve dynamic modelling of the physical system with the drive. Using the elevator rig for validating these serves as a multi-disciplinary pedagogic tool: work is in preparation.

Results

MiL: construction of communicating language generators (LG's) and port structures (PS'). The super-state MOVING_DOWN (MD) (Fig. 2) with two sub-states car_moving_down (CMD) and stop11 (S11) results in the following sub-tree of LG's:

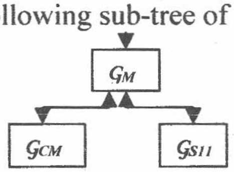

Fig. 3. Statechart decomposition

The port alphabet for communication between G_{MD} and G_{CMD} is then $\Sigma^P = \Sigma^{MD} \cap \Sigma^{CMD}$ = *{enteredCMD, leftCMD, leaveCMD}*, the intersection of their respective event alphabets. The port automaton, ζ_P, is constructed by masking the transitions corresponding to events **not** in Σ^P from G_{MD} and G_{CMD} and taking the union of the two resulting LG's. Controllability check under prioritised-synchronous composition between ζ_P and G_{MD} and test for non-blocking behaviour succeed immediately. Same is true between ζ_P and G_{CMD}. Broadcast communication of events generated on state entry and state transition is treated as value updates of global variables of which each current update can be read multiple times before it is written over by a new value. This is the asynchronous mode. In synchronous mode these variables are represented by FIFO queues of update values. The value at the head is consumed after a synchronous read by all receptive transitions. Port structures for these communications are built and tested in an analogous manner. Non-blocking simulation run of the simulink-statechart model validated these results. The other results will be reported in a later communication.

Discussion and Conclusions

The task of designing and installing an integrated research facility for developing critical control technologies needed in the life-cycle of complex reactive systems/automated utility systems was undertaken and accomplished successfully. Validation of control design methodology with respect to reconfigurability and formal verification has then been done at a first level and continues.

Acknowledgements

We thank the University of Peradeniya for Research Grant RG/2006/32/E and civil work/hardware support from CECB and Schneider Electric Co. Ltd.

References

Bandara, K.M.K.S., Abeyaratne, S.G. and Dewasurendra, S.D. (2006) Statechart based elevator controller and its verification, Pp. 88-92. In Proc. ICIIS2006: First Int. Conf. on Industrial and Information Systems, Peradeniya, Sri Lanka, August 8-11, 2006.

Bringmann, E. and Krämer, A. (2008). Model-based testing of automotive systems. Pp. 485-493. In Proc. ICST2008: First Int. Conf. on Software Testing, Verification and Validation, Lillehammer, Norway, April 9-11, 2008.

Dewasurendra, S.D. (2006a) Statecharts for reconfigurable modular control of complex reactive systems: need for formal verification and associated problems, Pp. 268-273. In Proc. ICIIS2006, August 8-11, 2006.

Dewasurendra, S.D. (2006b) Statecharts for reconfigurable modular control of complex reactive systems: a new formal verification methodology, Pp. 274-278. In Proc.ICIIS2006, August 8-11, 2006.

Drusinsky, D., Michael, J. B. and Man-Tak, S. (2008). A visual tradeoff space for formal verification and validation techniques. IEEE Systems Journal, 2(4): 513-519.