MEDS: Mockup Electronic Data Sheets for Automated Testing of Cyber-Physical Systems Using Digital Mockups

Bailey Miller
Dept. of Computer Science and Eng.
University of California, Riverside
bmiller@cs.ucr.edu

Frank Vahid
Dept. of Computer Science and Eng.
University of California, Riverside
vahid@cs.ucr.edu

Tony Givargis
Center for Embedded Computer Systems
University of California, Irvine
givargis@uci.edu

Abstract—Cyber-physical systems have become more difficult to test as hardware and software complexity grows. The increased integration between computing devices and physical phenomena demands new techniques for ensuring correct operation of devices across a broad range of operating conditions. Manual test methods, which involve test personnel, require much effort and expense and lengthen a device's time to market. We describe a method for test automation of devices wherein a device is connected to a digital mockup of the physical environment, where both the device and the digital mockup are managed by PC-based software. A digital mockup consists of a behavioral model of the interacting environment, such as a medical ventilator device connected to a digital mockup of human lungs. We introduce Mockup Electronic Data Sheets (MEDS) as a method for embedding model information into the digital mockup, allowing PC software to automatically detect configurable model parameters and facilitate test automation. We summarize a case study showing the effectiveness of digital mockups and MEDS as a framework for test automation on a medical ventilator, resulting in 5x less time spent testing compared to methods requiring test personnel.

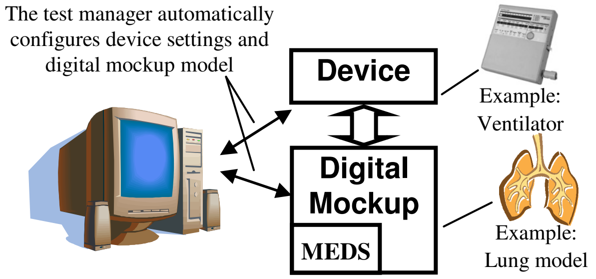
I. INTRODUCTION

Cyber-physical systems possess increasingly complex device software and hardware. Systems like modern automotive electronics or medical equipment consist of heterogeneous processors executing hundreds of thousands of lines of code, coupled with complex transducers such as sensors or actuators. The growing software and hardware complexity of cyber-physical systems introduces new problems for developers, including more time spent testing. Reducing testing time is thus an important goal.

Testing a cyber-physical system with the real environment, such as testing a car on a highway or medical equipment on a human, is cost prohibitive and dangerous. Instead, a common testing method for cyber-physical systems uses *physical mockups*, wherein the compute device is connected to a mechanical analog of the physical environment. For example, a mockup of a human lung may be achieved using a balloon consisting of some elasticity and flow resistance. However, physical mockups commonly lack the ability to represent more complex physical scenarios, such as diseased or coughing lungs. Another method uses *digital mockups*, wherein a mathematical model of the environment interacts directly with the device's computers, thus bypassing transducers. Digital mockups can represent highly-complex phenomena using sophisticated models [13], such as representing diseased lungs. A drawback is not including the transducers in the testing. Hybrid physical/digital mockups seek a compromise solution, having mechanical analogs that interact with transducers, but controlling the mechanical parts with computers, as in a lung mockup having an inflatable chamber whose volume is governed by a computer-controlled piston [5]. The work in this paper applies to both digital and hybrid mockups.

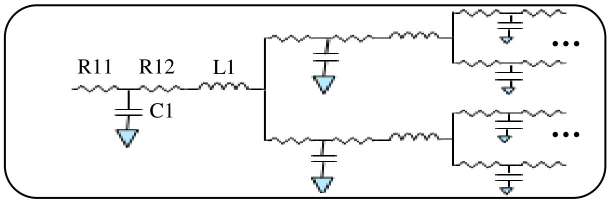
Digital mockups enable a new capability of test automation due to being configurable automatically by PC-based test manager software to vary physical scenarios. In developing test manager software for a broad range of cyber-physical systems rather than just a single device, we encountered the problem of different digital mockups having different configurable items such as settable parameters or watchable variables, requiring time-consuming and error-prone manual setup of the test manager based on each mockup's datasheet. We introduce Mockup Electronic Data Sheets (MEDS) as a method for embedding model-specific information into digital mockups. MEDS provides mechanisms for allowing test manager software to perform configuration to model parameters, obtain a list of available maneuvers (such as simulating the obstruction of a ventilator patient's airway), and obtain internal model state information for debugging. Digital mockups plus MEDS thus enable fully-automated cyber-physical system testing. Figure 1 shows a device connected to a digital mockup hosting a behavioral model of the environment, and a MEDS component on the digital mockup facilitates test automation with the test manager software on a PC. While the rest of this paper focuses on medical devices and physiological systems, the concepts extend to general cyber-physical systems.

Figure 1: Testing of device software using a digital mockup. The test manager on a PC configures the device and digital mockup environment to automate tests. Model information is embedded in the MEDS component.



978-3-9810801-8-6/DATE12/©2012 EDAA

Figure 2: Example model: Two generations of a Weibel lung model.



II. RELATED WORK

Physical systems modeling can be used to create better-quality cyber-physical systems. For medical equipment, Lee [9] describes the need for the accurate modeling of patients to deliver higher-confidence medical devices. Arney [1] uses a patient model consisting of drug absorption levels and patient vitals to test a closed-loop system of medical devices including a PCA pump and pulse oximeter. Lee presents a closed-loop artificial pancreas design [8], which is validated through the use of a human diabetic subject simulator [7]. Sirowy [14] introduced a technique for bypassing transducers to enable direct interaction between environmental models and cyber devices. The Virtual Heart Model [6] presents interacting electrocardiography and pacemaker models for testing pacing algorithms.

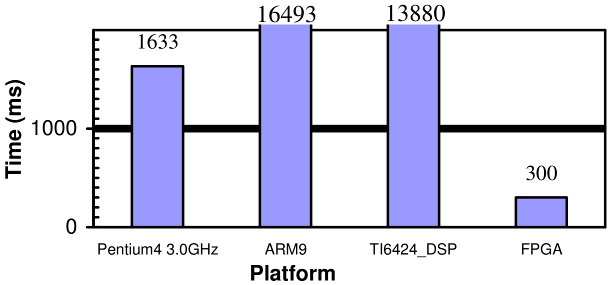
MEDS was inspired by Transducer Electronic Data Sheets (TEDS)—IEEE 1451.4 [11]—which embeds transducer calibration and identification information inside a transducer to allow for external software access. Automated testing is a well-researched area [3], but the use of digital mockups to facilitate test automation with cyber-physical devices has yet to be investigated. Digital mockups and MEDS provide a general method for use with any cyber-physical device and environmental models, not just this paper's targeted ventilator device.

III. DIGITAL MOCKUPS

Digital mockups utilize a computing platform to execute complex models in real-time and facilitate communication with the device under test. Figure 2 depicts the first few branches of a lung model based on Weibel morphology [15]. The lung system is characterized by a tree of bifurcating branches of 23 generations where the root node represents the trachea and leaf nodes represent respiratory zones where gas exchange occurs.

Field Programmable Gate Arrays (FPGAs) are especially well-suited platforms for executing physical models. Physical models generally consist of hundreds to thousands of differential equations, depending on the complexity of the behavior being modeled. Complex models often can not meet real-time constraints using common simulation techniques like the popular Simulink software or custom-written C code. FPGAs can be configured to take advantage of the inherent parallelism and local communication of physical systems models, resulting in simulations orders of magnitude faster than processor-based implementations.

Figure 3: Execution times for one second of 11-generation branching Weibel lung model featuring 4000+ ODEs.

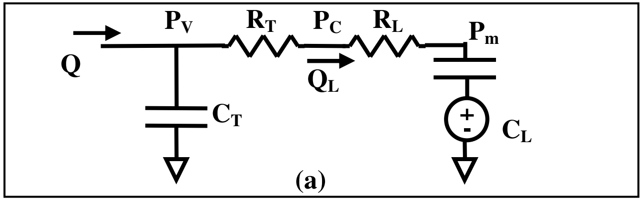


We created an 11-generation Weibel lung model consisting of over 4000 differential equations and implemented the equations on various platforms. Figure 3 displays execution times for one second of execution time of the model on a Pentium IV processor, ARM9 processor, Texas Instruments DSP, and a FPGA. The horizontal line indicates the real-time constraint, where the model is running in real time on a platform if the execution time falls below the line. The FPGA implementation consists of a network of processing elements designed to solve differential equations [4], and can run the model 3x faster than real-time because of the ability to exploit the parallel characteristics of the model, whereas other platforms cannot meet the constraint. We target FPGAs because of their ability to compute even the most complex models in real-time; however, digital mockups may also be hosted on desktop CPUs or GPUs.

IV. MOCKUP ELECTRONIC DATA SHEETS

MEDS consists of model-specific information described in XML format, hosted within a digital mockup to enable outside software to automate test procedures more effectively. Each mockup behavioral model is accompanied by a MEDS component that details specifics about modifiable parameters, available maneuvers (such as simulating disconnected pressure sense lines on a ventilator), and watchable state variables (such as pressures of a lung model). Because the testing of a cyber-physical device can include many types of scenarios, the flexibility to swap models easily is desired. For example, a ventilator may need to ensure that appropriate pressures are delivered to a patient using a respiratory mechanics model, but later testing may require considering blood oxygen levels using a gas-exchange model. Using simpler models requiring only a processor may be desirable during early testing phases, while more complex models requiring coprocessor acceleration may be used during later phases. MEDS encourages such flexibility by providing a concise and full specification of a model, which external software may utilize to gain insight into the model's behavior.

Figure 4: MEDS: (a) Resistor/capacitor circuit for modeling patient airway and lung behavior, (b) corresponding XML contents of MEDS for a digital mockup hosting the resistor/capacitor model.



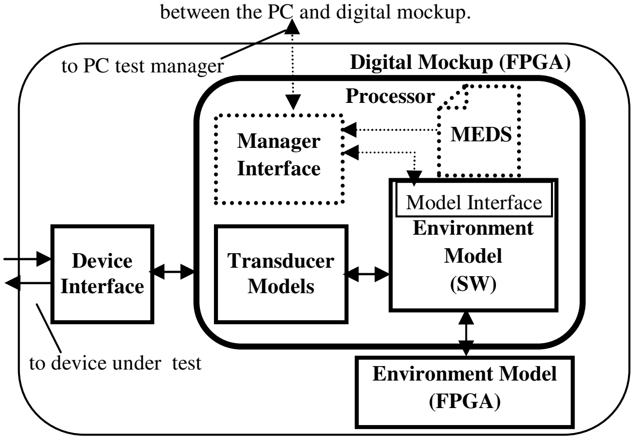
```
<?xml version="1.0"?>
<meds>
 <model  name="rrcc_lung" id="0">
  <parameter name="Rl" units="cmH20*Min/L" min="0.01"
          max="0.5" def="0.3" id="0" />
  <parameter name="Rt" units="cmH20*Min/L" min="0.0001"
          max="0.75" def="0.008" id="1" />
  <parameter name="Cl" units="L/cmH2O" min="0.0001"
          max="0.25" def="0.0005" id="2" />
  <parameter name="Ct" units="L/cmH2O" min="0.0001"
          max="0.1" def="0.0001" id="3" />
  <variable name="F" units="Liters/Min" id="4" />
  <variable name="Pv" units="cmH20" id="5" />
  <variable name="Pc" units="cmH20" id="6" />
  <variable name="Pl" units="cmH20" id="7" />
  <maneuver name="Disconnect Pressure Sense Line" id="8"/>
 </model>
</meds>
```

(b)

A. *MEDS Content*

MEDS describes a model through an XML description of the various configurable parameters, watchable variables, and available maneuvers. Figure 4(b) details MEDS contents for a lung model based on a resistor/capacitor circuit (RRCC) model described by Borello [2], as shown in Figure 4(a). <model> elements are used to indicate the presence of a model within MEDS contents. Multiple <model> elements may be to indicate that a digital mockup contains more than a single model available for use, which allows for swapping between models to test for different functionality. Supporting multiple models has the added benefit of reducing the amount of synthesis if large models must be mapped to the FPGA reconfigurable fabric. The children of each model element are either <parameter>, <variable>, or <maneuver> elements. <parameter> elements describe the configurable parts of a model. Parameters for the RRCC model include the resistor and capacitor components. The behavior of the model can be altered via changing the values of these parameters. For example, assigning high values of lung resistance may simulate a patient with emphysema or other respiratory illness. Elements marked with <variable> denote internal model values which may be traced. Variables within the RRCC model are the pressures and flow: Pv, Pc, Pl, and Q. Variables are the output of the model, and as such are read-only and may not be altered by external software. <maneuver> elements denote a special action that a model may execute.

Figure 5: Architecture of a digital mockup utilizing MEDS to facilitate test automation. A manager interface module handles communication between the PC and digital mockup.



B. *Digital Mockup + MEDS Architecture*

A digital mockup architecture with support for the MEDS component is depicted in Figure 5. Digital Mockup architecture has been previously proposed [12]. The MEDS content is stored within a memory block on the FPGA. The MEDS XML description is loaded into the memory through a JTAG interface. A manager interface component is instantiated to facilitate communication between the environment model and external PC test manager software. The environment model is also augmented to host a model-specific interface which exports an API for reading and writing of parameters, watching of variables, and maneuver execution.

The communication between the test manager PC and the manager interface consists of a packet-based message-passing serial protocol. A 6-byte packet consists of a 2 bits ‘type’ field, 14 bits ‘id’ field, and 4-byte ‘data’ field. 14 bits of ‘id’ allows for a model to possess over 16000 unique parameters or variables. The protocol contains 4 possible values for the ‘type’ field: *LoadMEDS, WriteRequest*, *ReadRequest*, and *SwapModel*.

The environment model must be augmented to contain a model-specific interface that hosts an API for the manager interface to utilize. The interface is model-specific because the implementations of models can vary, and the methods for performing operations such as changing parameter values are dependent upon the design of the model. A model may be implemented purely on a processor, in which case a parameter value might be changed by writing a global variable. On the other hand, a model may be implemented as a circuit coprocessor on the FPGA reconfigurable fabric, and thus the writing of a parameter depends on the structural configuration of the design—perhaps involving a controller to write to embedded block memories. The implementation details of models can be abstracted away by exporting a consistent API for the manager interface to use.

V. CASE STUDY

The usefulness of digital mockups and MEDS as a framework for test automation is demonstrated in a case study. The test is hosted on a Pentium IV, 3.0 GHz machine. The FPGA used is a Xilinx XC5VLX110T, hosted on an ML505 evaluation platform. The digital mockup system uses a MicroBlaze soft-core processor to host an execution kernel for the models. Communication between the digital mockup and test manager software is performed through a serial connection. The ventilator device was obtained through collaboration with a medical device company. The communication between the ventilator and digital mockup is an implementation of the device’s internal protocol used to communicate between command processor and transducers, which allows the digital mockup to intercept packets utilizing the transducer bypass method [14]. Communication between the ventilator and manager software is facilitated by an onboard serial debugging port

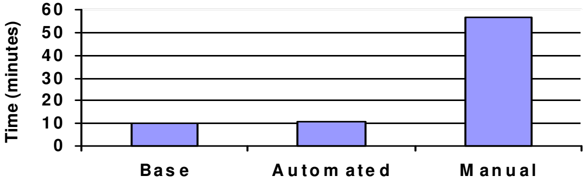
The digital mockup was loaded with the RRCC model. The MEDS content is loaded with definitions of the model parameters, variables, and available maneuvers. A test which verifies device functionality across a range of lung resistances and compliances is defined in the test software. The test software configures and runs the test 20 times, once for each combination of parameters. Each test requires 30 seconds recording time, with less than 3 seconds on average of overhead to calculate the result of the test and configure the settings.

To compare to the automated framework method, we performed the test procedure using the ventilator connected to a physical mockup of a lung (i.e., a balloon-type device). Parameters of the physical lung mockup are altered by changing physical restrictor plates, thus requiring a human in the loop to swap components. Data is recorded on an oscilloscope for 30 seconds. Each visible breath is observed and it must be determined whether each breath is within the target threshold of +/-5% of the target pressure. We performed these tests manually, after having become familiar with the procedure as a real test engineer might be, and determined that the time to perform the test procedure for 20 combinations of parameters would take approximately 1 hour. The time difference when compared to the automated test results is a factor of the overhead required to swap components on the physical mockup, record data, and manually perform the calculations. The automated digital mockup framework can quickly record and analyze data, while a trained human takes considerably more time to perform the same operations. Figure 6 depicts the differences in testing time between automated, manual, and a base case where no overhead is required.

VI. CONCLUSION

We presented an approach for automated testing of cyber-physical systems with digital mockups and introduced Mockup Electronic Data Sheets (MEDS). Model information is embedded within the digital mockup to allow external software access and facilitate automation. A case study was performed using a commercial ventilator which yielded time savings of up to 5x on test procedures as compared to a manual approach.

Figure 6: Comparison of testing times for automated and manual methods of performing a target pressure test. The base category shows minimum amount of time required to record data with no overhead.



ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation (CNS1016792, CPS1136146), the Semiconductor Research Corporation (GRC 2143.001), and a U.S. Department of Education GAANN fellowship.

REFERENCES

[1] Arney, D., et al. Toward patient safety in closed-loop medical device systems. Proceedings of the 1st ACM/IEEE International Conference on Cyber-Physical Systems.

[2] Borrello, M. 2005. Modeling and control of systems for critical care ventilation. American Control Conference.

[3] Dustin, E. 1999. Automated software testing. Addison-Wesley.

[4] Huang, C., Vahid, F., and Givargis, T., A Custom FPGA Processor for Physical Model Ordinary Differential Equation Solving, IEEE Embedded Systems Letters, Fall 2011 (to appear).

[5] Ingmar Medical. http://www.ingmarmed.com.

[6] Jiang, Z., Pajic, M., Connolly, A., Dixit, S., and Mangharam, R. 2010. Real-Time Heart Model for Implantable Cardiac Device Validation and Verification. ECRTS, 239-248.

[7] Kovatchev, B.P., Breton, M., Dalla Man, C., and Cobelli, C. 2009. In Silico Preclinical Trials: A Proof of Concept in Closed-Loop Control of Type 1 Diabetes. Diabetes Science and Technology.

[8] Lee, H., Buckingham, B.A., Wilson, D.M., and Bequette, B.W. 2009. A Closed-Loop Artificial Pancreas Using Model Predictive Control and a Sliding Meal Size Estimator. Journal of Diabetes Science and Techonology.

[9] Lee, I., Pappas, et al. 2006. High-confidence medical device software and systems. Computer.

[10] Lee, I., and Sokolsky, O. 2010. Medical Cyber Physical Systems. Design Automation Conference (DAC ’10).

[11] Licht, T.R. 2001. The IEEE 1451.4 proposed standard. Instrumentation & Measurement Magazine, IEEE.

[12] Miller, B., Vahid, F., and Givargis, T. Application-Specific Codesign Platform Generation for Digital Mockups in Cyber-Physical Systems. ESLsyn, June 2011, pp 1-6.

[13] Physiome Project. www.physiome.org.

[14] Sirowy, S., Givargis, T., and Vahid, F. 2009. Digitally-Bypassed Transducers: Interfacing Digital Mockups to Real-Time Medical Equipment. Engineering in Medicine and Biology Society.

[15] Weibel, E.R.. Morphometry of the human lung. 1963., Academic Press. New York.