Facilitating software construction for nano- and microscale measurement

Juha Röning, Ville Tuhkanen, Risto Sipola, and Tero Vallius

The combination of modular architecture and a four-part control framework simplifies and quickens instrument design for nanotechnology systems.

Nanotechnology research frequently necessitates small-scale measurements and manipulations. Such endeavors require precise sensors and actuators, and to date have been expensive and limited to specific applications such as gauging tiny adhesion forces and moving microscale objects. The most commonly used tools to date have been atomic force microscopes, and relevant technologies based on them. Now, however, thanks to advances in technology, off-the-shelf components are available for realizing measurements much more easily.

Nanomanipulation devices have many components in common: most contain an actuator, an end-effector (such as a spot welder, spray gun, or gripper), and a controller. Standard operations include manipulation simultaneous to measuring a quantity of interest, which requires controlling software for support. Like the devices themselves, the software programs share common features. As a result, programs for reusing design and development can be constructed employing patterns or models known as frameworks. These provide a universal skeleton that can be expanded to generate an application-specific program, an example of which we report here.

Our nanotechnology platform (software plus hardware) consists of a haptic (i.e., touch-based) user control interface, a PC, and a piezoelectric actuator (see Figure 1). The system can be used with a variety of tools and sensors, depending on the application. It is based on a generalized modular architecture for both instrumental hardware and our Scalable Modular Control Architecture (SMCA) software. The latter enables swift changing of actuators, sensors, and tools with minimal effort. The system thus offers an ideal frame for various applications, such as adhesion-patterning or cell-manipulation techniques, which are important in microbiology. We have tested the system on a number of real-life cases. We can also ease and hasten software development for measuring technologies using our Scalable Modular Control Framework (SMCF) software.

The SMCF consists of four parts (see Figure 2): a module stack that is responsible for controlling actuators and other measuring equipment, a data-handling component, a configuration-management component, and separate user interface (UI) elements that exploit the framework’s functionality.

The StackFrame object forms the basis of SMCA implementation. It executes and monitors the system architecture’s control loop. Application-specific functionality, such as technology interfaces, signal processing, and simple behaviors, can be achieved by adding modules from different layers of the architecture to the module stack included in the StackFrame object. The only restriction on the modules is that they must inherit (i.e., use the same features as) a module base class. The StackFrame object automatically places two modules in the stack. These remove old information from the so-called blackboard object of the SMCA, and monitor the time required to perform one...
control-loop cycle. The user can override the functionality offered by these modules.

Handling and presenting measurement data is one of the key functions of a measurement application and ordinarily takes up a significant portion of the total time required for technological development. To reduce this time cost, the SMCA framework includes a component for handling and storing information during measurements. Information is transferred from the modules in a module stack to various UI components of the specific technology. It ensures that all acquired data is available to the components by referring to it through variable names that are assigned to the data-gathering modules. This enables the module stack structure to be hidden from the UI. The component also allows the data to be split into measurements that can be saved for later use. Data handling is implemented by the DataBroker class, and saving by the DataSaver class.

The measurement systems that the SMCA framework targets are typically prototypes. This means that the functions offered by the software should be easy to modify. To achieve this goal, the framework includes a component for handling application settings to effect easy loading and storage of settings. It also includes a settings window, the content of which is automatically generated from modules registered to the component. Furthermore, it provides a component that allows the user to create measurement tasks at runtime by selecting and combining basic operations from a list offered by the sequencer layer of the application. Finally, data can be saved during tasks using the DataBroker class.

Generally, in micromanipulation applications, the system actuator must be under manual control. In contrast, the SMCA framework includes UI components that also enable autonomous operation. All of the components input their control feed through a generic controller module, located in the module stack. This allows the applications UI to be created without in-depth knowledge of the control architecture.

In summary, we have described a framework for designing nano- and microscale measurement applications. The SMCA consists of a module stack for controlling the actuator and other measurement equipment, and components for data handling, configuration management, and UI. Two test cases were implemented (measuring the thickness of an organic film and the adhesion force of microfibers) that showed the platform to be well suited to designing measurement systems software and effective in reducing the work required for software development. As a next step, we will investigate ways of applying the SMCA framework to other areas of application.

The author thanks Infotech Oulu, as well as Marko Pudas, Marko Kauppinen, and Olli Pitkänen for their contributions. This work was supported by the Provincial State Office of Oulu and the European Regional Development Fund (NanoResolution Tools for Interdisciplinary Applications).

Author Information

Juha Röning, Ville Tuhkanen, Risto Sipola, and Tero Vallius
Department of Electrical and Information Engineering
University of Oulu
Oulu, Finland

Juha Röning is professor of embedded systems and head of the Department of Electrical and Information Engineering. He has two patents and has published more than 250 papers. His

Continued on next page
primary research interest is intelligent systems, especially mobile robots, machine vision, and software security.

Ville Tuhkanen is an MSc student.

Risto Sipola is an MSc student in the Department of Electrical and Information Engineering.

Tero Vallius received an MSc in information engineering, with honors, from the University of Oulu (2001). He is currently working in the robotics group there as a postgraduate student. Aside from robotics, his research interests also include embedded systems, which he gained experience in while working at ABB Corporate Research and Nokia Networks.

References