Review of Scientific Instruments

RESEARCH ARTICLE | JANUARY 31 2007

WSXM: A software for scanning probe microscopy and a tool for nanotechnology

I. Horcas; R. Fernández; J. M. Gómez-Rodríguez; J. Colchero; J. Gómez-Herrero; A. M. Baro

Rev. Sci. Instrum. 78, 013705 (2007)
https://doi.org/10.1063/1.2432410

CrossMark

View Online

Export Citation

Articles You May Be Interested In

Investigation on chemical instability and optical absorption of ion bombarded Si surfaces

AIP Conference Proceedings (September 2021)

Using GeoGebra for constructing the intersections of two surfaces of revolution

AIP Conference Proceedings (September 2022)

03 October 2023 07:38:31

WSXM: A software for scanning probe microscopy and a tool for nanotechnology

I. Horcas and R. Fernández
Nanotec Electrónica S.L., Centro Empresarial Euronova 3, Ronda de Poniente 2, Edificio 2-1[a] Planta Oficina A, 28760 Tres Cantos, E-28049 Madrid, Spain

J. M. Gómez-Rodríguez
Laboratorio de Nuevas Microscopías, Departamento de Física de la Materia Condensada, Universidad Autonoma de Madrid, E-28049 Madrid, Spain

J. Colchero
Departamento de Física, Facultad de Quimica. Universidad de Murcia, E-30100 Murcia, Spain

J. Gómez-Herrero
Laboratorio de Nuevas Microscopías, Departamento de Física de la Materia Condensada, Universidad Autonoma de Madrid, E-28049 Madrid, Spain

A. M. Baro
Instituto de Ciencia de Materiales de Madrid-CSIC, Campus de Cantoblanco, 28049 Madrid, Spain

(Received 14 October 2006; accepted 18 December 2006; published online 31 January 2007)

In this work we briefly describe the most relevant features of WSXM, a freeware scanning probe microscopy software based on MS-Windows. The article is structured in three different sections: The introduction is a perspective on the importance of software on scanning probe microscopy. The second section is devoted to describe the general structure of the application; in this section the capabilities of WSXM to read third party files are stressed. Finally, a detailed discussion of some relevant procedures of the software is carried out. © *2007 American Institute of Physics.*
[DOI: 10.1063/1.2432410]

03 October 2023 07:38:31

I. INTRODUCTION

The final aim of nanotechnology is the control of the matter at the atomic and molecular levels. Since the introduction of scanning tunneling microscopy (STM) by Binnig *et al.*,[1] scanning probe microscopy (SPM) has proven its ability to be "the eyes and the fingers" of nanotechnology. Both STM and scanning force microscopy (SFM) have clearly demonstrated impressive performances resolving structures with subnanometer details;[2] in addition, using SPM the controlled manipulation of individual atoms and molecules has been demonstrated.[3]

The growth of SPM has been parallel to the revolution of computer technologies. In fact, computers have played a central role in the development of this technique significantly improving the data acquisition, control, image processing, and data analysis.

The most relevant application of a SPM is to acquire images; but this is not the only use of SPM. Other important features are, as mentioned above, the manipulation of matter as well as the measurement of the tip-sample interaction. From the measurement of the interaction, different material properties can be inferred, making SPM an important analytical tool for nanotechnology. The same hardware equipment can thus be operated in a great variety of operation modes, which makes SPM an extremely versatile technique. A computer controls nowadays almost every SPM and the different modes are implemented by specific subroutines of the corresponding SPM program. In fact, a very sophisticated feature such as digital control in SPM (Ref. 4) was used for the first time by Piner and Reifenberger in the very early days of STM. The importance of the software has been recognized by different research groups and private companies, which have therefore implemented their own SPM software[5] and file formats during the last 20 years.

II. OVERVIEW OF WSXM

In light of the situation we decided to introduce WSXM as a program that can be widely used by the SPM community. According to this idea we have left WSXM as a freeware application that is able to read most of the SPM file formats.[6] Moreover, we are continuously adding new file formats under users' requirement. In addition we have developed a tool that we call *heuristic open* that opens files with unknown format in a semiautomated way, including those files that due to the lack of a common identifier are not suitable to be recognized as a specific file format. If we have no information at all about the file that we are trying to open we can simply drag and drop the file from the Windows explorer into the "heuristic open" window and WSXM will suggest us some parameter/s to open the image using an expert system. The algorithm tries to determine parameters such as the number of points, header size, and number of images in the files based only on the file size. Almost every known information can be fed to the system to improve the reading capabilities including fixed positions for the most important fields in the file header such as number of rows/columns or the real axis

 © 2007 American Institute of Physics

TABLE I. Third party files read by WSXM.

Format	2D plots	Multiple image file
Burleigh™	No	No
Digital Instruments™ (Nanoscope™)	I/V, I/Z, F/Z, dCdV/Z	Yes
DME™	No	No
Free University of Berlin	No	No
Fritz-Haber (4 K STM-UHV)	No	No
Fritz-Haber Hippo	No	No
ImageSXM	No	Yes
Iso surface LDOS file generated by SIESTA	No	No
LASUAM	No	No
McAllister Instruments™	No	No
Molecular Imaging™	No	Yes
Morita's laboratory	No	No
Nanosurf™	No	Yes
NT-MDT™	No	Yes
Omicron™	No	Yes
Oxford Instruments™	No	No
Park Instruments™	No	No
Quesant™	No	No
RHK™	I/V, I/Z	Yes
SPMLab™	No	No
Standard grayscale TIFF	No	No
TopoMetrix™	No	No

amplitudes. Table I summarizes the file formats read by our software up to date. The aim of this work is to briefly describe the structure and performances of WSXM by remarking some of its most outstanding features. Figure 1 shows a general scheme of the possibilities of the software.

WSXM has been designed to work under MS-Windows; hence all the common features of this operating system are naturally included in this software: multiple windows, clipboard copy and paste, memory management, etc. WSXM is divided into two clearly different parts: image processing and data acquisition (see Fig. 1 for an overview); while the first part runs in all Windows versions since MS-Windows 95, the second part only works under MS-Windows NT, 2000, and XP; in this respect we have sacrificed generality to gain stability in data acquisition where system crash must be avoided as much as possible. WSXM code has been written in C++ using Microsoft Foundation Classes, a C++ library that allows easy handling of the complex Windows programming. C++ is a well-known object oriented programming language with clear advantages in large codes (WSXM has now about half a million code lines).

A. Data acquisition

Figure 2 shows the general flowchart of the hardware implementation commonly used by the WSXM program. The main philosophy of this implementation is oriented towards a multitasking environment based on a digital signal processor (DSP) board integrated in the main computer.[7] The DSP is mainly in charge of maintaining a digital feedback loop, and providing all the scanning signals (X,Y,Z) supplied to the microscope through a dedicated electronics that accom-

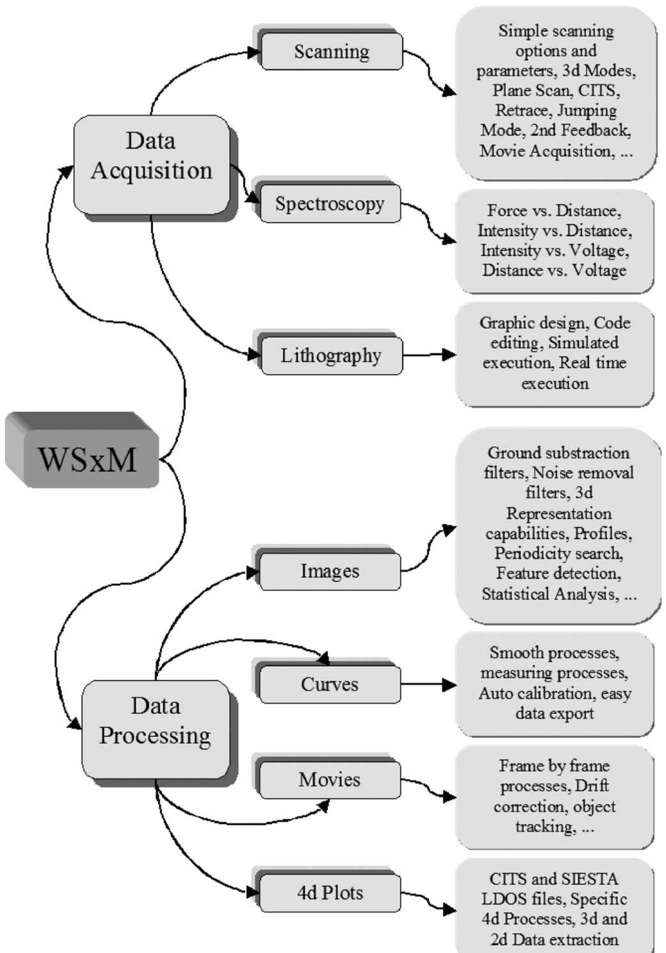


FIG. 1. General scheme of WSXM showing the most representative process.

plishes all the required signal conditioning and amplification processes.

When starting a data acquisition session in WSXM the user can select between STM, SFM, scanning near optical microscopy (SNOM), tuning fork, and interferometer; the toolbar and menus will be automatically adapted to the selected microscope type. The user can also select different types of coarse approach before initializing the data acquisition session. The approach dialog box will be also automatically modified according to the current approximation.

Data acquisition in WSXM is built around a multichannel environment. WSXM can simultaneously measure and display up to 17 different top view images in real time. In addition to the top view representation the data can be also displayed as two-dimensional plots [similar to oscilloscope traces in both $Y(t)$ or XY formats]. Finally, several real time filters such as line subtraction, derivative, fast Fourier transform (FFT), etc., can be applied to the data. Any of the displayed channels can be selected to be saved; each of them will create an individual file with an ASCII header containing general information about the image and then the image points saved as binary data (currently WSXM uses 2 bytes per image point).

Besides the regular scanning images WSXM allows us to measure different plots on a fixed surface point: force versus distance (tip-sample distance), current versus distance, cur-

03 October 2023 07:38:31

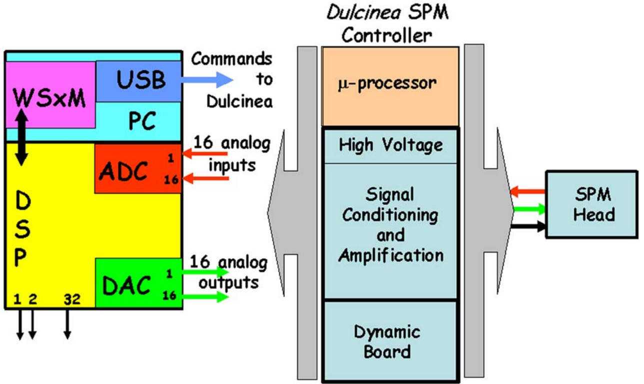


FIG. 2. (Color online) Block diagram of the hardware system controlled by WSXM.

rent versus bias voltage, and distance versus tip-sample bias voltage. WSXM can also perform spatially resolved spectroscopy; from the classical method called current imaging tunneling spectroscopy[8] (CITS) to three-dimensional (3D) modes,[9] a tool to measure tip-sample interactions. Jumping mode[10] also provides spectroscopy information since it is able to obtain adhesion maps of the studied surface. Adhesion is a magnitude that depends on the water meniscus formed between tip and sample, the tip-sample geometry (for instance, larger tips always give higer adehesion), and the chemical composition of both the sample and the tip. WSXM can also measure long-range interactions such as magnetic or electric fields using three different methods: retrace (also known as lift mode), constant plane scanning, and Kelvin probe.

The motion of individual atoms on a surface can be also studied with SPM.[11] In order to carry out investigations on surface dynamics we have implemented a *movie* facility in WSXM. The simplest way to obtain a movie is to acquire a sequence of images (frames) that can be subsequently displayed at faster speed. However, if the dynamics is slow enough a reasonable movie can last several hours. Hence, even a small thermal drift can be totally unacceptable. In order to avoid this problem, we have implemented a tracking method that compares the first image of the movie with subsequent images by calculating the cross correlation between them.[12] The relative position of the correlation maximum with respect to the correlation image center is a good estimation of the displacement between images that can be compensated by elongating the *X*-*Y* piezoelectric scanners. Drift compensation has been successfully applied to study the thermal motion of individual Pb atoms on Si(111)7×7; in these experiments it was possible to track small surface spots of about $15\text{m}\times 15\ \text{nm}^2$ for more than 8 h at room temperature.[12] This drift correction technique has been applied also very recently to the continuous observation in real space of surface phase transitions by STM:[13] exactly the same surface regions were observed with atomic resolution while varying the sample temperature from 40 to 200 K.

B. Image processing

The most outstanding feature of any SPM is its capability to produce high-resolution images of a surface. Hence, image processing is a central issue for any SPM software. An operating system such as MS-Windows is particularly well suited for image processing since it is naturally oriented to produce high quality graphics in many different windows that can simultaneously coexist in the computer memory. WSXM uses these features to create a new window for any finished process applied to the active window. In order to explain what we mean by finished let us take an example: WSXM provides a number of processes to remove high frequency noise. One of them is called "Gaussian smooth" and it works by averaging image points with a Gaussian weight. The number of neighbors that enter in the average can be selected through a Windows dialog that also shows a preview of the image. Each time the user changes the number of neighbors the preview can be updated according to this new value. When the user considers that the result is satisfactory a new window can be created in the mainframe; this new window will display a "finished" process and it can be considered as a *child* of the original window (the *father* window) that will also persist in the mainframe. WSXM considers *father* and *child* windows as totally independent; i.e., any process applied to the *child* window will not affect the *father* and vice versa. Of course there are many other image filters that immediately create a child window without the necessity of any Windows dialog. In both cases the new created window is set as the active window.

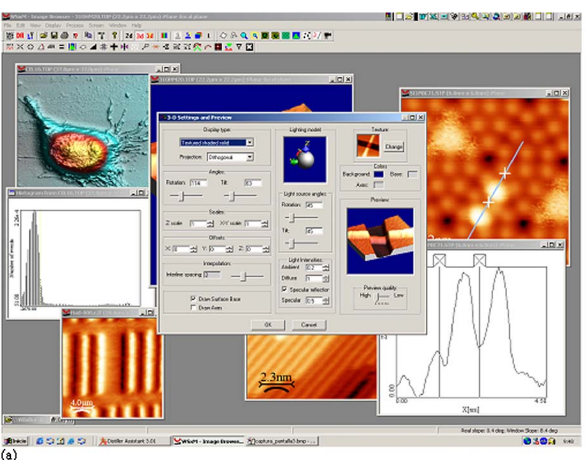
A significant difference between conventional microscopies that use waves to produce images, and SPM that keeps constant a given interaction between a tip and a sample, is that while the first mainly obtains projected images, the second obtains three-dimensional images. WSXM can display the images by assigning a different color to a range of heights according to a color palette previously selected by the user: pseudocolor images, constant height curves, and three-dimensional projected views with illumination. In addition, WSXM can combine these three different modes. While the most usual representation is top view images, WSXM provides an extensive dialog box allowing different options in projective views: image rotation, lightning, and textures.

Three-dimensional images, in any of the rendering modes, are very useful to recognize structure and patterns; however, when accurate measurements of distance and height are required two-dimensional profiles (cross sections) are the option of choice. WSXM can carry out single and multiple profiles: profiles along a given line that are simultaneously calculated in several related top view images previously selected by the user. As in the case of images each profile is displayed in a new window. Profiles can be then exported via clipboard as images or data to other Windows applications. Once a profile is created the menu and profile toolbar appear allowing us to measure distances along the profile. While measuring is active two crosses in the corresponding top view image are drawn indicating the *X*-*Y* coordinate that are being measured in the profile window. The user can edit profiles in the top view window: once scaled, erased, moved, etc., the profile window is automatically updated. The profile toolbar presents other options such as smoothing, derivative, power spectrum, etc.

03 October 2023 07:38:31

In many cases SPM images present a clear two-dimensional periodicity (for example, images that show atomic resolution). WSXM provides two different methods to analyze these images. The first one is called “lattice” and it works in the real space by superimposing a lattice to the top view images. WSXM has a set of predefined lattices that usually appear in SPM. Lattice can be scaled, rotated, and squashed; besides, the number of elements of the lattice can be also modified. The second method uses a two-dimensional fast Fourier transform that automatically gives the periodicity of the image. Filtering in the reciprocal space is provided through a variety of spectral windows: rectangular, Hamming, Blackman, and Blackman-Harris.[14]

When periodicity is not present statistical values such as roughness are relevant. Three different methods have been implemented to calculate these magnitudes. The first one is based on calculating the height distribution of an image. Then assuming that the height follows a Gaussian distribution the roughness is estimated as the rms value of the distribution. The second method to calculate roughness uses again the reciprocal space. After evaluating the FFT of the image it is angle integrated to produce the so-called power spectrum density. From this plot the roughness of the surface as a function of the wavelength is directly obtained. This method is also used to estimate the fractal dimension, an important parameter in system dynamics and, in particular, in surface growth. A third method to calculate statistical quantities from an image is “flooding.” As the name indicates this method is based on setting all the image values above or below a given threshold value selected by the user to a constant value. The resulting image resembles a picture of a flooded region of land. Once the process is finished WSXM calculates the flooded volume, the surface and perimeter of the islands, the center of mass of the islands, and the distance between centers. This method also provides a variety of histograms such as the number of islands versus islands’ area or the number of islands versus islands’ perimeter. Besides, using this method the fractal dimension can also be calculated.[15] Figure 3 illustrates and summarize the image processing capabilities of WSXM.



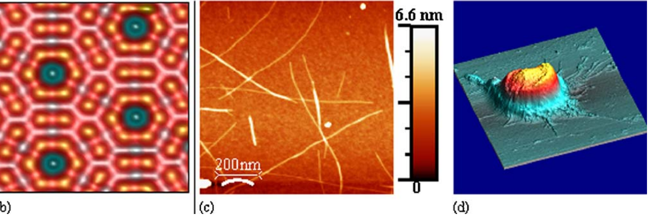


FIG. 3. (Color online) (a) Screenshot of the WSXM main process window. Only one of the many available process windows. (b) Silicon (111) 7×7 reconstruction rendered using the three-dimensional package integrated in WSXM. (c) Single walled carbon nanotubes imaged with a scanning force microscope. (d) 3D rendering of a tumor cell.

III. DETAILED DISCUSSION OF RELEVANT ISSUES

A. Multiple feedbacks

One of the unique features of WSXM data acquisition is the possibility to run a large number of simultaneous digital-controlled feedback loops. As it is well known,[16] SPM operation requires, at least, one feedback loop system in order to maintain constant a given parameter (tunneling current in STM, cantilever deflection in contact mode SFM, etc.) to a predefined value (the so-called set point value) by means of variations of the tip-sample distance. While in early STM designs this feedback system was completely analog, the advantages of computer-controlled digital feedbacks[4] were soon realized and most SPM controllers use at present digital controlled systems. In WSXM this task is performed by an interrupt driven subroutine running in the DSP. A built-in timer of the DSP board provides the timebase for the interrupt, ensuring a well-defined and consistent timing (with a period Δt) for the control subroutines.[17] The feedback algorithm implements a simple proportional-integral (PI) controller that has been successfully used for SPM in the past.[18] Explictly it reads

$$Z_t = Z_{t-1} + a_1\varepsilon_t + a_2\varepsilon_{t-1}$$

with

$$a_1 = P + I,$$

$$a_2 = -P + I,$$

where Z_t is the Z scanner position at a time t (in timer period units), ε_t is the error signal at time t (i.e., the difference between the measured value of the control parameter at time t and the set point value), and P and I are the proportional and integral gains, respectively.

Up to seven independent feedback loops can be driven simultaneously with WSXM. A first feedback loop is in charge of the SPM main control parameter (e.g., tunneling current in STM or cantilever vibration amplitude in intermitent-contact SFM). This feedback loop is linearized in the case of STM by calculating the error signal as $\varepsilon_t = \ln(I_t/I_{\text{setpoint}})$, where I_t is the tunneling current at time t. Besides this main digital feedback loop, the DSP can process three independent feedbacks for piezolinearization purposes: with the help of extensiometric gauges, the X, Y, and Z piezos can be operated in closed loop. Three additional general feedback loops are available for miscellaneous purposes. In particular, they can be used in frequency modulation noncontact mode SFM.

03 October 2023 07:38:31

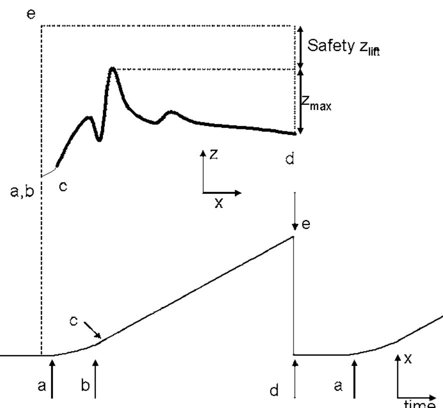


FIG. 4. (Upper) One directional line scan. The most representative point has been labeled. (Lower) The corresponding $V(t)$ signal applied to the piezoelectric has been drawn for the sake of clarity.

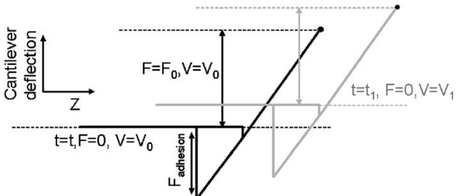


FIG. 5. Force vs distance curve. The zero force level has been initially set to V_0 (usually $V_0=0$). This level fluctuates with time and at $t=t_1$ the new voltage value is V_1.

B. Scanning

One of the drawbacks of SPM is the scanning speed. WSXM integrates a few simple, but effective, algorithms to improve the scanning rate. First, the scanning ramps are rounded at the turning point to avoid high frequencies as much as possible. This feature is implemented as follows: assuming that we are not using the nonlinear piezocorrection, the voltages applied to the *X*-*Y* piezoelectrics are linear ramps with opposite slopes for the forward and backward ways. Therefore there is a discontinuity on the derivative at the turning point. This discontinuity is avoided by smoothing this peak with a simple algorithm in the DSP that keeps the first derivative continuous. In our discrete and cyclic system we have implemented an iterative algorithm to round the scan, using constant acceleration. The rounding magnitude can be selected as a percentage of the total line scan time and this parameter fix the acceleration. The data are only represented on the constant slope periods to avoid image distorsions.

The other relevant feature related with the scan is shown in Fig. 4. In many cases, the backward (forward) image is not necessary and therefore it is possible to move the sample (assuming that we are scanning with the sample) in a short time to the left (or right) side of the image. Figure 4 shows an arbitrary topography line scan that is related with the voltage applied to the scanner by the labels. Let assume that at (*a*) the sample is stopped on the left side of the scan area. At point (*b*) the sample starts moving to the right; in order to avoid mechanical resonances a short round period is chosen (the corresponding topography trace has been drawn as a thin line in Fig. 4; this topography region is not stored in the images). At point (*c*) the linear ramp starts and from this point to point (*d*) the topography is acquired and represented on the computer screen. At point (*d*) the maximum *x* displacement of the sample is reached. Then, the highest z_{max} of the line scan is calculated in the DSP, the feedback loop is opened, and the tip-sample distance is increased from its current position to that in z_{max} plus a user defined safe distance (safety z_{lift}) to avoid tip and sample damage. The sample is then moved very fast back to the left side of the image (point *e*) and the feedback is closed again. A recover time is then executed (time elapsed between points *a* and *b*) to allow system stabilization. Both the recover time and the safety z_{lif} are user parameters that can be tuned depending on the imaging conditions. Obviously this scan feature saves almost a half of the acquisition time for an image at the price of loosing one of the scan directions.

C. Jumping mode

Jumping mode (JM) is an atomic force microscopy (AFM) scanning option that plays an important role in WSXM. JM was developed as a midpoint between classical contact mode and dynamic mode. While the physics of JM have been thoroughly described elsewhere;[10] the basic algorithm for JM has not been published. JM, with a working principle very similar to pulsed force microscopy,[19] was originally designed as a scanning mode to minimize shear forces. The difference between pulsed force microscopy and JM is that the first is implemented electronically whereas JM is just a software method running in the digital signal processor memory. JM mode operation can be described as a cycle repeated at each image point with the following steps: (i) tip-sample separation, (ii) lateral tip motion at the furthest tip sample distance, (iii) tip-sample approach, and (iv) feedback, which is generally performed on the cantilever deflection. This method is particularly valuable in liquids[20] where the low Q dramatically reduces the sensitivity of dynamic mode. The key point to obtain good quality topography images using JM is the precise control of the normal force applied by the tip. Most of the AFM heads use optical beam deflection to detect the cantilever bending. This system is simple, robust, and reliable; however, it does not measure an absolute value for the zero force level that is adjusted, when the tip is far away from the sample, to an arbitrary voltage (usually zero) that is basically the voltage difference between the two vertical sectors of a photodiode. In contact mode the set point is a voltage value that is also related with the difference between the two vertical sectors of the photodiode. Unfortunately, the zero force level changes with time (see Fig. 5) for many reasons, including motion of the laser spot from its initial position due to thermal expansions, laser intensity variation, etc. Consequently the force used as a set point to acquire topographic images in contact mode may arbitrarily change with time. In other words, the contact mode images, which are supposed to be taken at constant force, are in reality taken at an unknown force that fluctuates with time. JM offers a simple remedy to this problem that is to measure the voltage corresponding to the zero force level when the tip-sample distance is maximum and recalculate

03 October 2023 07:38:31

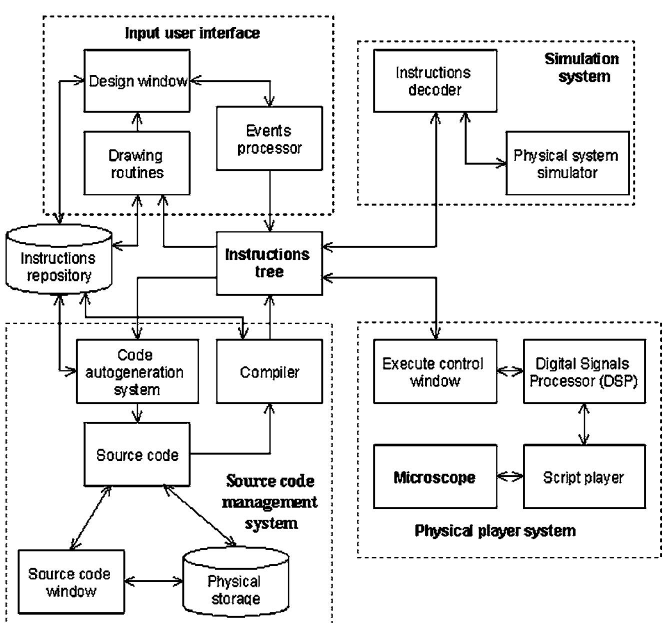


FIG. 6. Schematic of the lithography section of WSXM.

the set point to correct the zero force level fluctuations. The JM dialog in WSXM allows choosing between three different options: set point is not recalculated, it is recalculated exclusively at one single point at the end of each scanning line or it is recalculated at each point of the image. In liquids the drift in the zero force level is even more accused than in air ambient and the third option is usually the best. Another remarkable feature of JM is that in addition to the topography image, it allows us to measure adhesion. This is performed by storing the cantilever deflection during steps (i) and (iii) within a vector in the DSP memory. The adhesion is measured as the minimum value of this vector. This value is sent to the personal computer (PC) memory and represented as an adhesion map in the computer screen.

D. Nanolithography

The control of the tip motion by the user is a clear goal for any SPM software. We have implemented this feature in WSXM through the combination of a computer-aided design (CAD) application and a specially designed compiler. From the point of view of the end user, the working basis of this lithography mode is quite simple: using the CAD interface, the user draws a pattern that the tip must follow. In addition to the drawing instructions, different parameters such as bias voltage, set point, feedback state, tip speed, etc., are fed by the user. The drawings performed in the CAD window generate code that can be seen and modified in the code window. This code can be simulated before running it to avoid some logical errors. Finally, the user opens a data acquisition session and begins to execute the predefined tip motion by executing the CAD generated (or user written) code.

We have developed a new programming language with some fixed rules and a continuously growing set of primitives, allowing more and more capabilities for controlling the system. The capabilities of the lithography module depend on the *Instructions Repository* module (see Fig. 6 for more details of the different lithography modules implemented). Most of the basic instructions can be easily drawn using the *design window* and automatically translated to source code by the *code autogeneration system*. This is a structured language with forced, declared, and typed variable support, allowing conditionals, loops, and blocks as the main flow control instructions, which provides a powerful interface to control the microscope. Comments in the C++ format are also allowed to clarify the source code. We have included optional arguments to increase the flexibility and allow the same instruction to be simple but accurate if required. This code is compiled in the PC, sent to the DSP board, and then interpreted in real time. In this way we obtain the compiling efficiency of a desktop computer and the low response time of a DSP. Bias pulses as fast as 60 μs long can be done using the lithography. Of course longer pulses can also be programed.

The general idea of the lithography module is to be powerful enough to provide low level access and let the user implement the algorithms, avoiding internal extra checking that could decrease the source code efficiency and limit the control power of the user over the machine. Some of the provided primitives are as simple as read/write some input/output in any system channel. In addition some complex processes need to be done, such as acquiring curves that need an output to be modified while input data are taken or acquiring some input channels while the tip is moving over the sample. We have implemented these two cases and allow the acquisition of generic curves (any output channel such as Z, bias, etc., and any input channel such as normal force, current, lateral force, amplitude, etc.) and the acquisition and visualization of any set of channels while *drawing* lines. This visualization is very useful while trying to move atoms as the resulting curve can tell us if the attempt was successful, while the first feature allows the user to acquire any common kind of curve (IV, ZV, FZ, and IZ) and any other specialized one.

We have referred several times to draw something in the surface. This drawing can be performed in several ways depending on the particular experiment and most of the times it consists on moving the tip over the sample with some specific configuration different from the scanning one. Higher set point, higher bias voltage, or constant height motions are only some of the options widely used. To facilitate the task we have added what we call *start drawing configuration and finish drawing configuration*, which are two sets of actions to be performed before starting drawing and finishing drawing. These two sets are managed in the same way than the main source code, compiled in the PC main processor, and run in the DSP board.

Another important issue is to be easily hardware interacted. In this way we have developed some bit and byte level signals and allow turning on the execution using a configurable bit to increase the synchronization capabilities with external systems. We use the transistor transistor logic (TTL) standard for these signals. Mixing this feature with other developments such as the external scan control we can turn our whole system from master to slave mode and vice versa in only some milliseconds using TLL signals. The user

03 October 2023 07:38:31

should program the external device which will control the microscope and the synchronization protocol used in WSXM will let the control of the system as soon as it receives this communication from the external device. The slave operating mode can mean deactivate some of the WSXM control routines and goes from turning off the feedback loop to let every low-level signal controlled by the user, if the appropriate hardware is used. Almost every feature can be enabled/disabled individually, so it is allowed to turn off the main feedback loop and *XY* positioning but let active the closed loop scanners linearization. In this case we can solve the lacks of other systems.

Focusing in the compiling process we have developed both a compiler and a precompiler (or preprocessor) as most of the widely used languages and compilers do. The main functions of this preprocessor are removing the comments, tokenizing the source code, replacing the constant values, and generating the symbols table. It also has to store information about the original source code and check the syntax of the code outputting warning and errors messages to help the developer to fix them including references to the line number of the wrong/suspicious instruction. The generated symbols table stores information about the variables used in the program, not only the value in execution time but also some control flags. We use a tagged format for the arguments in the same way that XML and their derived languages (like the commonly used in the web HTML) do. In this way the order of the arguments is free and it is easier to add optional arguments providing a very valuable flexible and easy of use language. The compiler itself uses the tokenized code and the symbols table and generates some instructions trees. This is a set of three tree structures implemented in C++ classes using pointers for the edges and a primitive lithography instruction in each node. The radix of the trees will be the program's main block, or the drawing configuration main blocks and every node will store the information about the arguments, including a flag for optional arguments to know if the argument was present. Optional arguments are not assigned a default value as Java/C++, they are marked as "not present" and the system will take different decisions depending on the single primitive and argument. These trees are then serialized and size optimized and transmitted to the DSP board, which stores them and runs the instructions on demand. The user can choose between running the whole program, only a single instruction or a complete block, and pause it while it is running. In the DSP side there is an interpreter that loads the compiled and serialized code and runs it. A complete communication protocol is in charge of the proper synchronization between PC and DSP and between the internal routines and the user interface. The user can stay in front of the computer and the drawn instructions will change their color when executed. This is very useful when running conditional code as we can know the paths followed by the running thread and therefore which instructions were actually run. In all the process the compiled code is hidden so there is no need to deal with objects or executable files. The user only has to worry about source code, saving and loading both main program and drawing configurations if desired.

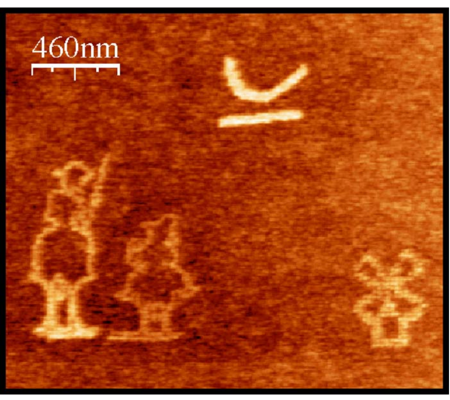


FIG. 7. (Color online) Nanolithography on silicon oxide showing Don Quixote and Sancho Panza.

Figure 7 illustrates the capabilities of WSXM lithography. The nanolithography is probably the smallest portrait of Don Quixote and Sancho Panza, the well-known characters of Cervantes' novel. It has been performed by applying a bias voltage of about 12 V between a silicon AFM tip and a silicon substrate. This voltage, together with the thin water layer that is commonly adsorbed on surfaces at air ambient conditions,[21] produces local oxidation[22] resulting on the ~30 nm wide lines observed in the figure. The voltage sequence and the parameters for the appropriate tip-sample motion have been set using the above described lithography features.

E. Perspectives

WSXM is not a finished software application; on the contrary it is under continuous evolution driven by the development team and the users around the world. In these final lines we will describe two important tendencies in our software.

Atomically resolved images do not necessarily reflect the position of the atoms as in a hard ball model. On the contrary, they are a complicated convolution of the tip and sample density of states. *Ab initio* calculations are currently applied with great success to understand the physics involved in SPM images.[23] WSXM has been adapted to display the results of a well-recognized *ab initio* program: SIESTA.[24] As new results emerge the capabilities of WSXM to analyze theoretical results will be more important.

The very last developments in WSXM are related with noncontact AFM. By combining new dynamic mode with the lithography capability WSXM allows us to move atoms at room temperature in an ultrahigh vacuum environment.[25]

WSXM is now used by thousands of persons related with science and technology. In the forthcoming years we will keep working on improving the capabilities of the software helping people to get the best possible results in their works.

ACKNOWLEDGMENTS

The authors are certainly indebted to all the WSXM developers that have worked in this project since 1998 and the many WSXM users around the world; in particular, the authors would like to thank the New Microscopy Laboratory students and Nanotec testers, their patience and enthusiasm

03 October 2023 07:38:31

have been of the greatest importance for the test and development of WSXM.

[1] G. Binnig, H. Rohrer, Ch. Gerber, and E. Weibel, Phys. Rev. Lett. **49**, 57 (1982).
[2] G. Binnig, H. Rohrer, Ch. Gerber, and E. Weibel, Phys. Rev. Lett. **50**, 120 (1983).
[3] D. M. Eigler and E. K. Schweizer, Nature (London) **344**, 524 (1990).
[4] R. Piner and R. Reifenberger, Rev. Sci. Instrum. **60**, 3123 (1989).
[5] P. Zahl, M. Bierkandt, S. Schröder, and A. Klust, Rev. Sci. Instrum. **74**, 1222 (2003).
[6] WSXM free software at www.nanotec.es
[7] The current hardware implementation incorporates a M67 board with a Servo16 module from Innovative Integration. See www.innovative-dsp.com for details.
[8] J. Hammers, R. M. Tromp, and J. E. Demuth, Phys. Rev. Lett. **56**, 1972 (1986).
[9] C. Gómez-Navarro *et al.*, Nanotechnology **13**, 314 (2002).
[10] P. J. de Pablo, J. Colchero, J. Gómez-Herrero, and A. M. Baró, Appl. Phys. Lett. **73**, 3300 (1998).
[11] E. Ganz, S. K. Thies, I. S. Hwang, and J. Golovchenko, Phys. Rev. Lett. **68**, 1567 (1992).
[12] J. M. Gómez-Rodríguez, J. J. Sáenz, A. M. Baró, J.-Y. Veuillen, and R. C. Cinti, Phys. Rev. Lett. **76**, 799 (1996).
[13] I. Brihuega, O. Custance, R. Pérez, and J. M. Gómez-Rodríguez, Phys. Rev. Lett. **94**, 046101 (2005).
[14] W. H. Press and W. Thomas, *Numerical Recipes in C. The Art of Scientific Computing* (Cambridge University Press, Cambridge, 1998).
[15] J. M. Gómez-Rodríguez, A. M. Baró, L. Vázquez, R. C. Salvarezza, J. M. Vara, and A. J. Arvia, J. Phys. Chem. **96**, 347 (1992).
[16] R. Wiesendanger, *Scanning Probe Microscopy and Spectroscopy: Methods and Apllications* (Cambridge University Press, Cambridge, 1994).
[17] At present the timing constant is set to $\Delta t = 70\ \mu$s with the hardware implementation shown in Fig. 2.
[18] See, e. g., D. R. Baselt, S. M. Clark, M. G. Youngquist, C. F. Spence, and J. D. Baldeschwieler, Rev. Sci. Instrum. **64**, 1874 (1993).
[19] A. Rosa, E. Weilandt, S. Hild, and O. Marti, Meas. Sci. Technol. **8**, 1 (1997).
[20] F. Moreno-Herrero, P. J. de Pablo, M. Álvarez, J. Colchero, J. Gómez-Herrero, and A. M. Baró, Appl. Surf. Sci. **210**, 22 (2003).
[21] A. Gil, J. Colchero, M. Luna, J. Gómez-Herrero, and A. M. Baró, Langmuir **6**, 5086 (2000).
[22] R. García, M. Calleja, and F. Pérez-Murano, Appl. Phys. Lett. **72**, 2295 (1998).
[23] O. Paz, I. Brihuega, J. M. Gómez-Rodríguez, and J. M. Soler, Phys. Rev. Lett. **94**, 056103 (2005).
[24] J. M. Soler, E. Artacho, J. D. Gale, A. García, J. Junquera, P. Ordejón, and D. Sánchez-Portal, J. Phys.: Condens. Matter **14**, 2745 (2002).
[25] Y. Sugimoto, M. Abe, S. Hirayama, N. Oyabu, O. Custance, and S. Morita, Nat. Mater. **4**, 156 (2005).

03 October 2023 07:38:31