



# Ionscan: scanning and control software for proton beam writing

A.A. Bettiol \*, C.N.B. Udalagama, J.A. van Kan, F. Watt

*Centre for Ion Beam Applications, Department of Physics, National University of Singapore, Singapore 117542, Singapore*

Available online 25 March 2005

---

## Abstract

The proton beam writing technique relies on a precise beam scanning and control system that offers a simple yet flexible interface for the fabrication and design of microstructures. At the Centre for Ion Beam Applications, National University of Singapore, we have developed a suite of programs, collectively known as *Ionscan*, that cater for the specific needs of proton beam writing. The new version of *Ionscan* is developed using the *Microsoft Visual C++*. *NET* development environment in conjunction with a *National Instruments* analog output card and *NI-DAQ* drivers. With the benefit of the experience gained in proton beam writing over the years, numerous enhancements and new features have been added to the scanning software since the first version of the program that was developed using *LabVIEW* [A.A. Bettiol, J.A. van Khan, T.C. Sum, F. Watt, Nucl. Instr. and Meth. B 181 (2001) 49]. These include the ability to perform combined stage and magnetic (or electrostatic) scanning, which is particularly useful for the fabrication of long waveguides and microfluidic channels over lengths of up to 2.5 cm. Other enhancements include the addition of the *Ionutils* program which gives the user the ability to design basic structures using an ASCII file format that was developed. This format contains basic information on the shape to be irradiated including the way in which it is scanned.

© 2005 Elsevier B.V. All rights reserved.

*PACS:* 07.05.Hd; 7.10.Cm; 7.78.+s

*Keywords:* Proton beam writing; Beam scanning; Software

---

## 1. Introduction

Proton beam writing is a direct write sub-micron lithographic technique that utilizes a focused beam of protons to pattern mainly polymeric materials such as polymethylmethacrylate

---

\* Corresponding author. Tel.: +65 6874 4138; fax: +65 6777 6126.

*E-mail address:* [phybaa@nus.edu.sg](mailto:phybaa@nus.edu.sg) (A.A. Bettiol).

(PMMA) and SU-8. For any direct write lithographic technique to function, a scanning system comprised of computer software and hardware is required in order to take care of stage movement, beam deflection and fluence normalization. As a lithographic technique, proton beam writing is analogous to electron beam writing in the sense that both techniques utilize a beam of highly focused charged particles for resist patterning. However, proton beam writing has more stringent requirements from a scanning system that cannot be obtained from commercial electron beam writing packages. These unique requirements arise from the characteristics of particle accelerators, the way in which fluence needs to be normalized and the magnetic rigidity of MeV ion beams.

At the Centre for Ion Beam Applications (CIBA), National University of Singapore, we have been developing a scanning and control system that is optimized for proton beam writing. Early versions of the software were developed using the National Instruments *LabVIEW* programming environment and a National Instruments analog output card [1]. This combination was initially chosen due to the ease with which scanning and control algorithms could be implemented, and to reduce development time. As the software developed in complexity it was decided to move over to the *Microsoft Visual C++* development environment and to use the National Instruments *NI-DAQ* drivers. The use of the *NI-DAQ* drivers allows us to easily support any of the National Instruments analog output cards under the *Microsoft Windows* operating system without requiring any major revisions to the code.

## 2. Software design

The *Ionscan* software suite is comprised of three individual programs to control beam scanning, stage movement and file conversion. The division of the scanning and control software into these three main tasks is a consequence of the way in which proton beam writing experiments are usually performed. For example, it is common for a user to spend some time designing and converting scan pattern files offline before the actual irradiation experiment.

Secondly, stage movement and sample positioning is often performed at the start of a run during which beam focusing and sample calibration experiments are performed. Many of the features that have been incorporated into the software are the result of user feedback and the extensive experience gained in proton beam writing over the years. The different components of the *Ionscan* software are now discussed.

### 2.1. *Ionscan*

The main function of the beam scanning program, known as *Ionscan*, is to scan the focused ion beam in a vector style pattern with controlled blanking and timing defined either by a normalization detector or at a uniform rate. The details of the scan pattern are contained in a binary file with an *epI* extension. This file is read by the *Ionscan* software and the data used to generate the scan pattern which is in the form of a sequential train of three simultaneously clocked unipolar or bipolar voltages corresponding to an *X* position, a *Y* position and a blank level. The *Ionscan* interface is shown in Fig. 1.

In order for the *Ionscan* software to be able to deliver the correct particle fluence, the sample must be first calibrated. The method that is used for calibration depends on the sample to be irradiated, the size of the structure to be irradiated and the desired fluence. The most commonly used method for fluence normalization and calibration is Rutherford backscattering spectrometry (RBS). To use this calibration method, a particle detector with a known surface area and in a known geometry is required. This detector is usually a Si surface barrier detector; in our case we use an annular surface barrier detector with a solid angle of 62 msr. At the start of an experiment, an RBS spectrum is first collected from the sample with sufficient statistics for the data to be fitted using an analysis software package like SIMNRA [2]. From the fit, the number of ions per steradian in the spectrum, and the total number of normalization counts are extracted and entered into the appropriate *Ionscan* dialog window. Fluence normalization can be done with any detector that gives a yield proportional to the number of inci-

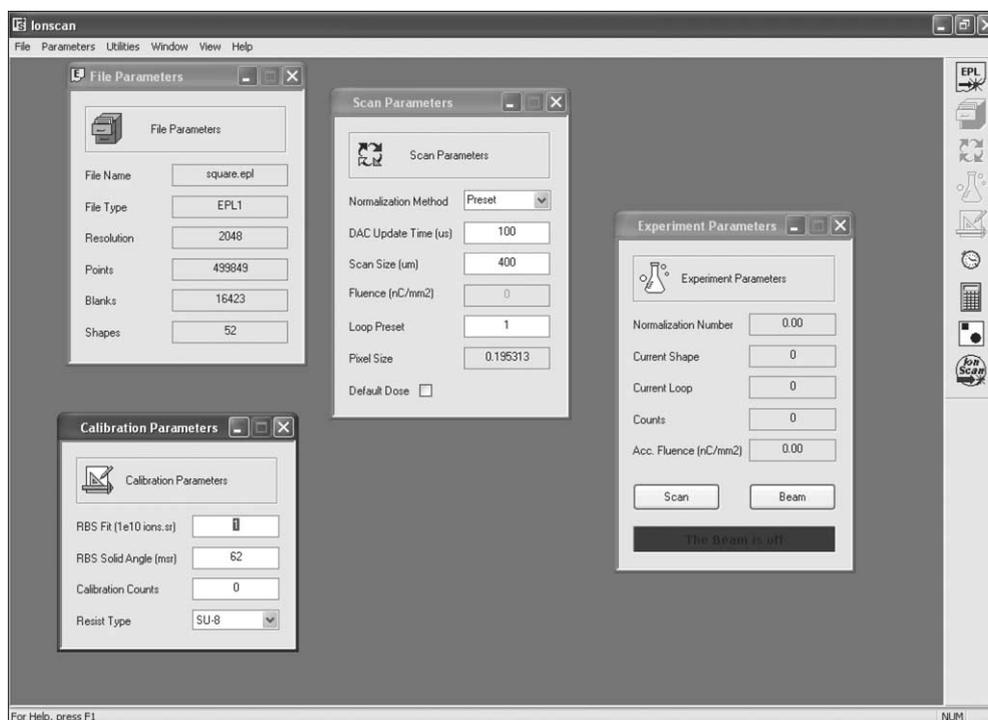


Fig. 1. The *Ionscan* software graphical user interface running under windows XP. This is one of the three programs that make up the *Ionscan* proton beam writing scanning and control software suite of programs. The software is developed using the *Microsoft Visual C++*. *NET 2003* development environment.

dent ions. The *Ionscan* software uses this data, and other information like the scan size and the desired fluence to calculate a number called the *normalization number*. This is the number of counts required in order to expose the sample with the desired fluence measured by the detector used for normalization.

Once the scan file has been read, and the sample calibration details entered into *Ionscan*, one of four different normalization modes can be used to deliver the desired fluence to the sample. These normalization modes are named preset, pixel, shape and figure. If preset normalization is chosen, the user can enter the number of times that the scan pattern is repeatedly scanned. If this normalization mode is chosen, the user needs to calculate the total time required for the exposure based on the beam current (detector count rate) and the update time between the irradiated pixels. A dose calculator dialog is included in *Ionscan* to assist the

user in making this calculation. If pixel normalization is chosen, the beam will remain at each pixel until the required fluence (counts) is measured by the normalization detector before moving to the next pixel. In between shapes, the beam is clocked internally at an update time specified by the user. The scan rate in between shapes is in effect the same as it would be if any of the other normalization methods were chosen. Pixel normalization is potentially the most accurate normalization mode: However if the number of counts required for normalization per pixel is low then this method is prone to substantial errors. Typically if RBS is used as the normalization signal for low current proton beam writing at sub micron dimensions, then the number of counts per pixel may be less than 1: In this case one of the other normalization methods is preferable. If shape normalization is chosen, the beam is repeatedly scanned over a shape until the desired fluence is reached before

moving to the next shape. A shape is defined as a group of adjacent pixels that are separated from the next group of pixels by a path where the beam is blanked. This normalization method decreases the number of times the beam is scanned over regions of the sample where the beam is blanked. The fourth normalization method is called figure scanning. This method scans the whole pattern, including regions where the beam is blanked, until the desired fluence is reached. This method is essentially an automated version of preset scanning.

Recently the ability to perform multiple exposures has been included in the *Ionscan* software. This feature allows the user to list in an ASCII file all the exposures that are required for a run, including the file name, the desired dose and the  $x, y, z$  co-ordinate of the stage. The *Ionscan* software will then sequentially expose each pattern at the corresponding  $x, y, z$  co-ordinate and record the measured fluence for each exposure. This enables the user to automate the whole experiment for one load of samples.

## 2.2. Ionutils

The *Ionutils* software package currently supports file conversion to and from monochromatic *bitmap*, *ascii* and *epl*, the native file format used by *Ionscan*. The *epl* file format is a binary file that contains a header and an ordered list of coordinates,  $(x, y, b)$ . These coordinates correspond to the  $x, y$  position of the beam in units of pixels, and the beam blanking level  $b$ . In addition to the irradiated points, the *epl* file also contains the points in between irradiated regions. The *epl* file therefore fully describes the path taken by the beam during an irradiation, including the path taken in between the irradiated structures while the beam is in fact blanked. Since the *epl* file fully specifies the scan pattern, the *Ionscan* program does not need to perform any calculations with regard to the scan path and as a consequence is simpler to program.

The user can specify the irradiated points in ASCII format by listing in two columns the  $x$  and  $y$  coordinates that are irradiated. During the conversion, the user is asked to specify the file resolution

(i.e. 128, 256, 512, ..., 32,768) and the minimum shape distance. The minimum shape distance is the distance in unit of pixels below which neighboring pixels are considered as being part of the same shape. This parameter is required in order to calculate the regions of the scan pattern where the beam is blanked, and the path which is taken between shapes. In order to reduce magnetic scan hysteresis, the paths in between shapes are included in the pattern vector. They are scanned with the same update rate as the rest of the pixels in the pattern.

*Ionutils* also supports a new file format which is called an *emc* file. The *emc* file is an ASCII file with an *emc* extension that has been developed specifically for proton beam writing. The file format was developed because it was observed that most of the useful patterns that are irradiated using proton beam writing are made up of simple shapes like circles and squares. The file format allows the user to define several basic shapes that can be easily converted to the *epl* format for proton beam writing. The *emc* shapes, called entities, contain information of their design (e.g. for a circle the radius and centre co-ordinates) and more importantly the order in which the points within the shape are scanned. This feature is unique to the *emc* format. Other file formats like AutoCAD *dxf* contain similar information on the parameters that define basic entities but they do not have any information on the point order. The point order is an important parameter that can considerably reduce the number of blank points in the figures. Since the blank points are also updated due to hysteresis problems, a proper ordering can make the overall write time much shorter.

The *emc* file consists of a header and any number of entity keywords with their corresponding parameters. The header consists of two parts: (i) the version keyword (!EMC1) and (ii) a resolution keyword followed by the file resolution (!RES). Valid file resolutions include numbers corresponding to  $2^n$  where  $n = 7, 8, \dots, 16$  (i.e.: 128, 256, 512, 1024, ..., 65,536). These numbers are the native resolutions of the 12 bit or the 16 bit DAC's on the scanning hardware. An example of the *emc* file format is shown in Fig. 2(a), and the corresponding figure generated by this file is shown in Fig. 2(b).

(a)

!EMC1	.....	Header for emc version 1.0
!RES	.....	Resolution keyword
1024	.....	Resolution value (128,256,512,1024....65536)
!SSQU	.....	Spiral square keyword
340	.....	centre x-coord
340	.....	centre y-coord
101	.....	Side width
!SCIR	.....	Spiral circle keyword
680	.....	centre x-coord
340	.....	centre y-coord
50	.....	Inner radius
120	.....	Outer radius
!RREC	.....	Raster rectangle keyword
0	.....	Scan direction
620	.....	First x-coord
620	.....	First y-coord
740	.....	Last x-coord
840	.....	Last y-coord
!SREC	.....	Raster rectangle keyword
1	.....	Scan direction
220	.....	First x-coord
740	.....	First y-coord
400	.....	Last x-coord
800	.....	Last y-coord

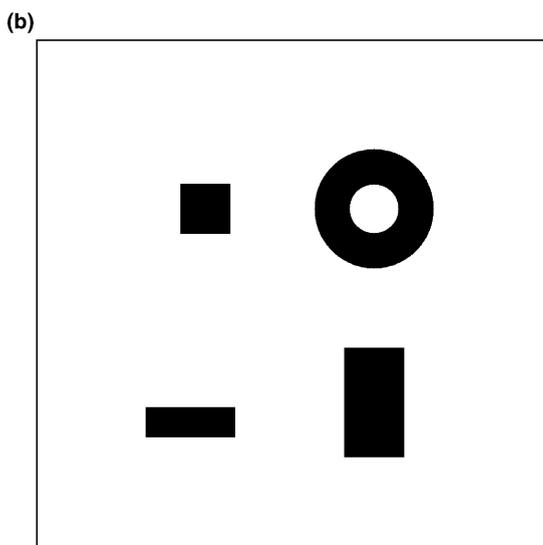


Fig. 2. An example of the new *emc* file format for generating *epl* scan pattern files for proton beam writing. The source 2(a) and the resulting Fig. 2(b) are shown.

Fig. 3 shows the order in which points are scanned for the four basic shapes that are currently supported by the *emc* file format. Rectangles can be raster scanned with or without a silhouette. The silhouette helps to smooth out edge roughness that can occur due to magnetic scanning hysteresis by scanning the beam always along the edge of the shape. Spiral scanning can also improve the edge roughness for circles and squares. In the spiral square case, the beam scanning always begins and ends in the centre of the shape. This scan

method was found to improve the edge smoothness of squares since any unwanted resist exposure that can result when the beam is switched on or off occurs in a region that is already exposed.

### 2.3. Stage control

The third software included in the *Ionscan* suite of programs is called *StageControl*. *StageControl* allows the user to perform several tasks related to stage movement. The software sends commands to the stage controller via a GPIB card allowing the user to move the stage in either a free run mode or to a specified position. The position of the stage is continually monitored and displayed for the user. Other functions included in the software are the ability to move the stage a predefined step (in units of micron), the ability to independently set the speed for all three stages, and a recording function that allows the user to record the co-ordinates of important positions so that the user can easily return to the desired co-ordinate at a later time. The ability to perform some basic line scans using the stage has also been included. This function allows the user to scan the stage in either the *x* or *y* direction, and simultaneously scan the beam magnetically (or electrostatically) in the perpendicular direction, while monitoring the normalization signal. This is an important feature of the *StageControl* software as it allows the user to make linear structures over the whole length of the stage movement which is 2.5 cm. The line scan feature has been successfully used to make long microfluidic channels and linear waveguides [3,4].

## 3. Hardware

A schematic of the hardware setup for proton beam writing is shown in Fig. 4. The *Ionscan* software is currently compatible with either the PCI 6731 (16 bit) or the PCI6711 (12 bit) National Instruments cards. These cards can provide up to four analog output signals, two of which are used for the scanning signal and one for beam blanking. The minimum update time possible with either of these cards is 1  $\mu$ s per pixel. This update time is faster than is usually required, the main limitation

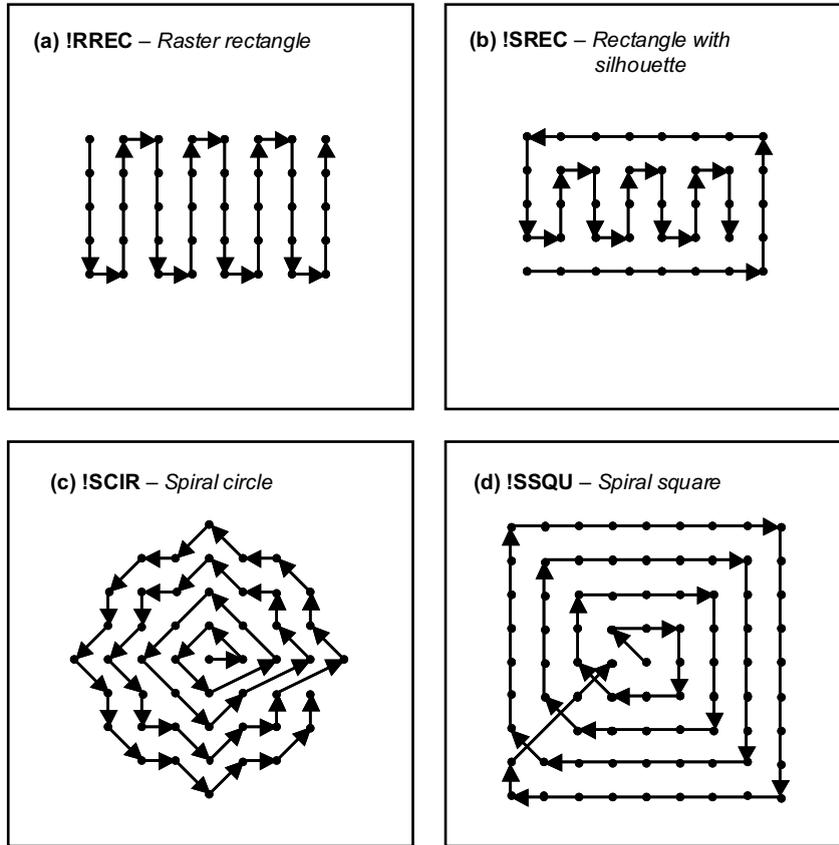


Fig. 3. The point scan order for (a) a raster rectangle, (b) a raster rectangle with a silhouette, (c) a spiral circle and (d) a spiral square is shown for the four *emc* file entities supported by the current version of the *Ionutils* software.

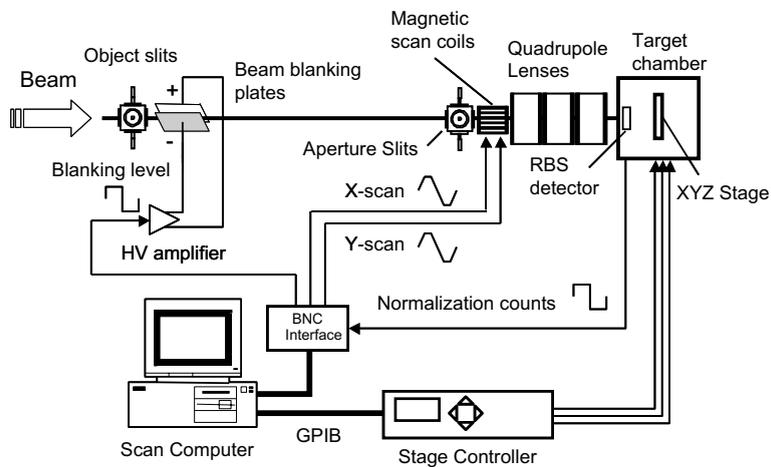


Fig. 4. A schematic diagram of the scanning and control hardware setup at the Centre for Ion Beam Applications, National University of Singapore proton beam writer.

Table 1

A summary of the hardware that is used in the proton beam writing set up at the Centre for Ion Beam Applications, National University of Singapore

Hardware	Purpose and main features
National Instruments Analog output card PCI 6731	Used to provide 3 digital to analog converters and 1 counter timer for scanning, blanking and normalization <ul style="list-style-type: none"> <li>• Resolution = 16 bit (<math>\pm 32,768</math>)</li> <li>• Analog output voltage = <math>\pm 10</math> V</li> <li>• Minimum update time = 1 <math>\mu</math>s/pixel</li> </ul>
National Instruments GPIB card	Used to control the XYZ stage via the 8200 controller
National Instruments BNC2110 interface block	Used to interface the scanning system software to the scan amplifier, the blanking high voltage supply and the normalization signal
EXFO Burleigh Inchworm <sup>®</sup> TSE-820-V XYZ stage and an 8200 series controller	Used to provide mechanical stage movement and for scanning long lines in samples <ul style="list-style-type: none"> <li>• Minimum step size (Encoder resolution) = 20 nm</li> <li>• Maximum speed = 1.5 m/s</li> <li>• Maximum travel = 2.5 cm</li> </ul>

being the hysteresis in the magnetic scanning. A single 24 bit counter timer on the card is used also to monitor a normalization signal that is in the form of a TTL pulse. Stage control is achieved by using a National Instruments GPIB card that is connected to an 8200 series EXFO Burleigh Inchworm<sup>®</sup> stage controller. The Inchworm<sup>®</sup> stage can provide a total of 2.5 cm of movement in the *X*, *Y* and *Z* directions. A summary of the hardware used in the proton beam writing scanning system is given in Table 1.

#### 4. Conclusion

At the Centre for Ion Beam Applications, National University of Singapore we have developed a scanning and control system optimized for proton beam writing. The system has been successfully used to fabricate structures in various samples including polymers and silicon [5,6]. The experience that has been gained in the fabrication of such structures has dictated the direction in which the software and hardware has been developed. The current system supports new features like combined stage and magnetic scanning, and the introduction of a new file format for efficient design of pattern files. The *Ionscan* suite of programs will continue to be developed and new features will be included in future versions of the

software. Future versions of the software will include support for other industry standard pattern file formats (like dxf and GDSII) and the ability to perform arbitrary stage and magnetic scanning.

#### Acknowledgments

This work was supported by a Defence Innovative Research Program grant from the Defence Science and Technology Agency, Singapore, and a grant from the Agency for Science Technology and Research, Singapore.

#### References

- [1] A.A. Bettiol, J.A. van Kan, T.C. Sum, F. Watt, Nucl. Instr. and Meth. B 181 (2001) 49.
- [2] M. Mayer, SIMNRA User's guide, Technical Report IPP 9/113, Max-Planck-Institut für Plasmaphysik, Garching, Germany, 1997.
- [3] K. Ansari, J.A. van Kan, A.A. Bettiol, F. Watt, Appl. Phys. Lett. 85 (2004) 476.
- [4] T.C. Sum, A.A. Bettiol, J.A. van Kan, F. Watt, E.Y.B. Pun, K.K. Tung, Appl. Phys. Lett. 83 (2003) 1707.
- [5] J.A. van Kan, A.A. Bettiol, F. Watt, Appl. Phys. Lett. 83 (2003) 1629.
- [6] E.J. Teo, M.B.H. Breese, E.P. Tavernier, A.A. Bettiol, F. Watt, M.H. Liu, D.J. Blackwood, Appl. Phys. Lett. 84 (2004) 3202.