

COMPUTER MANUAL SERIES No. 17

Data Acquisition for X Ray Microprobe

User's Manual

INTERNATIONAL ATOMIC ENERGY AGENCY, 2002

The originating Section of this publication in the IAEA was:

Instrumentation Unit
Seibersdorf Laboratories
International Atomic Energy Agency
Wagramer Strasse 5
P.O. Box 100
A-1400 Vienna, Austria

DATA ACQUISITION FOR X RAY MICROPROBE
IAEA, VIENNA, 2002
IAEA/CMS/17

© IAEA, 2002

Printed by the IAEA in Austria
September 2002

FOREWORD

A modified data acquisition software for X ray microprobe was developed by the Physics Group, Instrumentation Unit, IAEA Laboratories at Seibersdorf, with assistance from M. Bogovac, Croatia.

The software consists of data acquisition (scanning and calibration), automatic positioning and micro-movement of sample, data reduction and evaluation. The acquisition software was designed in order to support different measurement set-ups which are applied in low-energy nuclear physics.

The modification was done in 1999–2000 under the projects Nuclear Spectrometry and Utilization of Particle Accelerators. The manual supersedes the first version entitled Microanalysis Data Acquisition and Control Program published under Computer Manual Series, No. 9 in 1996.

The software described in this manual is freely available from the IAEA upon request. For comments, suggestions, installation and/or user help please contact:

IAEA Laboratories Seibersdorf
Instrumentation Unit, XRF Group
A-2444 Seibersdorf, Austria
Fax: +43-1-2600-28222
Tel.: +43-1-2600-28244
e-mail: S.A.Bamford@iaea.org

EDITORIAL NOTE

The use of particular designations of countries or territories does not imply any judgement by the publisher, the IAEA, as to the legal status of such countries or territories, of their authorities and institutions or of the delimitation of their boundaries.

The mention of names of specific companies or products (whether or not indicated as registered) does not imply any intention to infringe proprietary rights, nor should it be construed as an endorsement or recommendation on the part of the IAEA.

CONTENTS

1. X ray microprobe.....	1
1.1. Introduction	1
1.2. Setup	1
1.2.1. General.....	1
1.2.2. Specification	2
1.3. X ray beam focusing	4
1.3.1. Multilayers	5
1.3.2. Refractive lenses	6
1.3.3. Capillaries and polycapillaries	7
1.3.4. X ray total external reflection	8
1.4. Micropositioning	10
1.4.1. Motion controller	11
1.4.2. Amplifier.....	12
1.4.3. Actuator.....	12
1.4.4. Feedback (Encoder)	14
1.4.5. Specification	15
1.5. Data acquisition	16
1.5.1. Getting started.....	16
1.5.2. Scanning.....	27
1.5.3. Examples.....	37
1.5.4. Tips	45
2. MNG.....	48
2.1. Data menu	48
2.1.1. Load	48
2.1.2. Save as	49
2.1.3. Save.....	49
2.1.4. Add.....	49
2.1.5. Subtract	49
2.1.6. Compose	49
2.1.7. Replay	50
2.1.8. View.....	51
2.2. System menu	51
2.2.1. Select MCA.....	52
2.2.2. MCA	52
2.2.3. Host.....	57
2.2.4. Set as default	60
2.3. Acquire menu	60
2.3.1. MCA	60
2.3.2. Host.....	68
2.3.3. Report.....	69
2.3.4. Save.....	69
2.3.5. Save as	69
2.3.6. Load	69
2.3.7. Set as default	69
2.4. Desktop	70

2.4.1. Customize	70
2.4.2. Action controls.....	72
3. MDI	75
3.1. Display menu.....	75
3.1.1. Open.....	75
3.1.2. Open all.....	76
3.1.3. Open all on start.....	76
3.1.4. Read.....	76
3.1.5. Write.....	76
3.1.6. Create single.....	76
3.1.7. Create map.....	77
3.1.8. Print.....	77
3.1.9. Setup printer.....	79
3.1.10. Exit.....	79
3.2. Window menu	79
3.2.1. Tile.....	79
3.2.2. Cascade.....	79
3.2.3. Arrange icons.....	79
3.2.4. Close all.....	79
3.2.5. Arrange spector.....	79
3.2.6. Save arrangement.....	79
3.2.7. Window list.....	79
3.3. Region menu.....	80
3.3.1. Zoom.....	82
3.3.2. Back.....	82
3.3.3. Home.....	83
3.3.4. Sum.....	83
3.3.5. Gauss fit.....	83
3.4. ROI menu	84
3.4.1. Create.....	84
3.4.2. Delete.....	84
3.4.3. Edit.....	84
3.4.4. Project.....	86
3.4.5. Keep always.....	87
3.4.6. Update sum.....	87
3.4.7. Shape.....	87
3.5. View menu	87
3.5.1. Single.....	87
3.5.2. Map.....	91
3.5.3. Isometric.....	94
3.5.4. Colors.....	96
3.5.5. Fonts.....	99
3.5.6. Desktop.....	100
3.5.7. Directories.....	103
3.5.8. Tool bar.....	104
3.5.9. Menu bar.....	104
3.5.10. Status bar.....	106
3.5.11. Updating.....	107

3.5.12. Show restart	107
3.5.13. Show exception	107
3.5.14. Save	107
3.5.15. Save as	107
3.5.16. Load	108
3.6. Tools menu	108
3.6.1. Calibrate	108
3.6.2. Overlap	109
3.6.3. Compose	109
3.6.4. Copy as bitmap	109
3.7. Action menu	109
3.8. Scan menu	109
3.8.1. Scan properties	110
3.8.2. Export	114
3.8.3. Extend	117
3.8.4. View	117
3.8.5. History	117
3.8.6. Motor	119
3.8.7. Go to	119
3.9. Video menu	119
3.9.1. Select	120
3.9.2. Properties – Matrox Meteor	120
3.9.3. Image	122
4. Motor	125
4.1. Coordinate system	125
4.2. Basic controls	126
4.2.1. Movement	126
4.2.2. Speed	126
4.2.3. Positions	127
4.2.4. Position units	127
4.3. Select board	127
4.3.1. Board properties	127
4.3.2. Refresh	133
4.3.3. Go there	133
4.3.4. Go to zero	133
4.3.5. Reset board	133
4.3.6. Reset motors	133
4.3.7. Recover origin	133
4.3.8. Application properties	134
APPENDIX	135

1. X RAY MICROPROBE

1.1. Introduction

X ray fluorescence (XRF) is a widely used non-destructive technique for chemical analysis. In the conventional XRF spectrometer, the sample is uniformly irradiated and fluorescence spectrum averaged over the entire irradiated area is obtained. This leads to the determination of the elements and their average concentration in the whole bulk sample.

In the imaging XRF spectrometer different points of sample (pixels) are irradiated by a narrow (typically 10-30 μ m) primary beam. At each point, X ray fluorescence spectrum is collected and the concentration of a particular element versus position may be obtained (concentration maps). However, the real concentration maps require advanced (and time-consuming) data evaluation techniques. The simplest type of maps shows only the number of X rays detected per pixel at the energy that corresponds to the characteristic X rays of a given element of interest.

If X rays are used as a primary beam the sample is mounted on micropositioning system, and the technique is called *micro-beam X ray fluorescence* or simply *X ray microprobe*. If charged particles are used as a primary beam, instead of X rays, the method is called scanning particle microprobe (specially electron scanning microprobe and proton induced X ray emission microprobe which is also known as PIXE microprobe).

Despite their great chemical and crystallographic sensitivities, X ray microprobe techniques have been limited by inefficient X ray focusing optics as well as weak X ray sources. Interest in X ray microprobe was revived by the development of ultra-high intensity synchrotron X ray sources but still limited due to very high cost of such facilities. The most recent development in the X ray focusing optics has enabled using of a conventional X ray tube as source of primary beam and first commercially available systems has been released.

1.2. Setup

1.2.1. General

There are no specified components and standard arrangements for a XRF microprobe acquisition scanning and control system. Therefore systems differ from laboratory to laboratory. However, a typical X ray microprobe setup features the following basic elements:

- X ray source and X ray optics
- micropositioning system
- high-resolution energy dispersive detector with simple front-end electronics
- data acquisition software.

The schematic block-diagram of XRF microprobe in IAEA Laboratories at Seibersdorf is shown on the Fig. 1.2.1 and the real system on the Fig. 1.2.2.

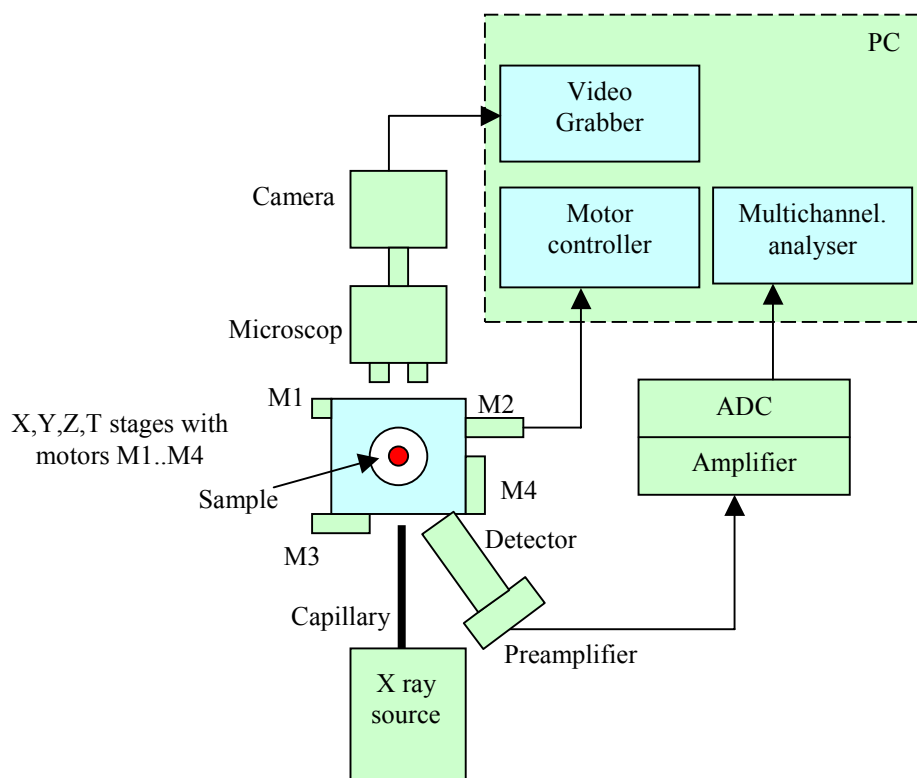


Figure 1.2.1. Block scheme of X ray microprobe at IAEA Laboratories Seibersdorf.

The sample can be moved in front of capillary by four motors (M1..4) in three orthogonal spatial directions and rotated as well. The motors are connected to the board plugged into the data acquisition PC computer. Two of four motors define the sample position in the plane perpendicular to the capillary, that is, x and y coordinates. For each pair (x,y) and preset time, energy signals from detector are processed by standard front-end electronics (preamplifier, amplifier and Analog to Digital Converter (ADC)), and converted signals are stored in multi-channel analyser operating in pulse height analysis (PHA) mode. After the preset time elapses, collected spectrum is analysed or optionally stored onto the disk, and sample is moved to the next position. Such an acquisition is mainly defined by the fact that the moving of the motor is slow compared to the pulse processing and moving of the beam not possible. Therefore an event-by event acquisition is not suitable.

To scan an appropriate part of the sample, a magnified image of sample can be captured and calibrated in such a way that units on this picture are correlated with actual sample size and position. This is supported by the system consisting of the microscope, video camera and video grabber.

1.2.2. Specification

The specification for the experimental setup shown in the Fig 1.2.2 are as follows:

- **X ray source**
Philips; 3kW, Long Fine Focus, Mo anode

1. X RAY MICROPROBE

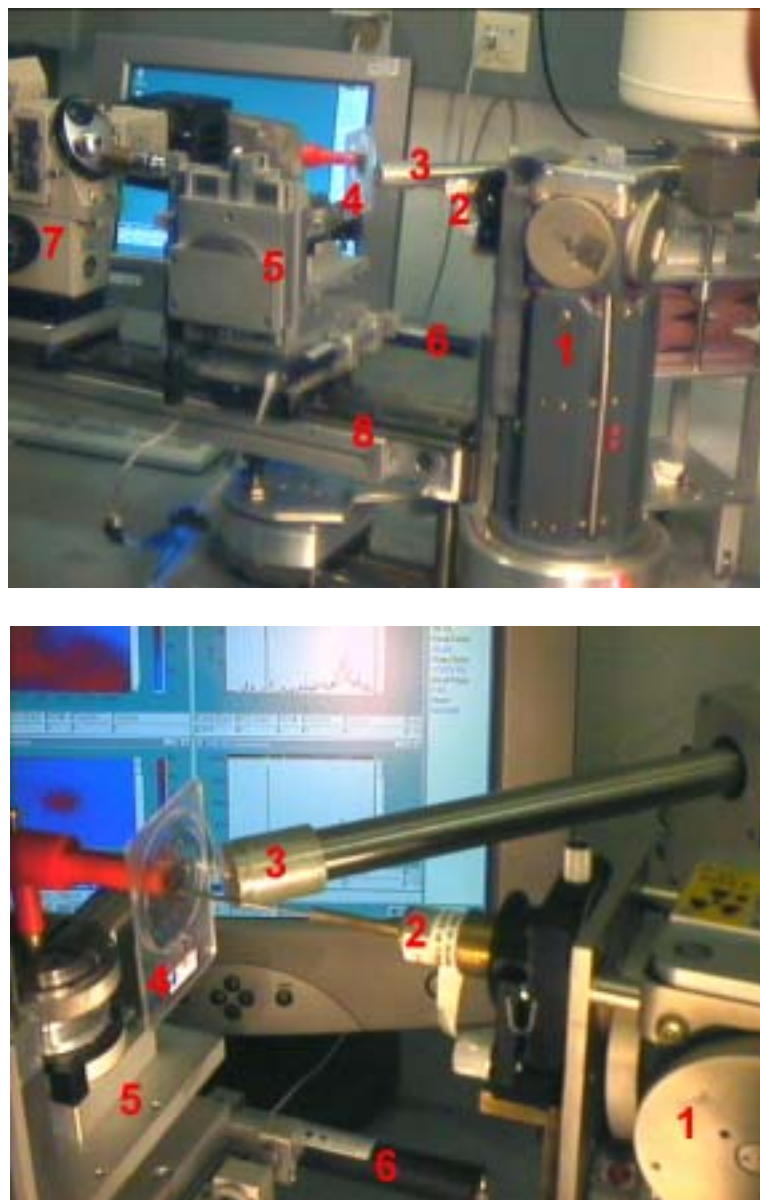


Figure 1.2.2. X ray fluorescence microprobe facility at the IAEA Laboratories in Seibersdorf, features the following components: X ray tube (1), capillary for X ray beam focusing (2), high resolution X ray detector (3), sample (4), sample holder (5) with micrometer step motors (6), microscope with CCD camera (7), trail (8) and data acquisition system.

□ **X ray optics**

Capillaries: several with exit diameter 12, 30, ... μm

Polycapillaries: X ray optics lens (spot size 35 μm , focus length 6.2 mm)

□ **Video image**

Microscope: OLYMPUS BH2-UMA with two lenses X40, X10

Camera: Panasonic GP-KR222 (analog)

Frame grabber: Matrox Meteor

Table 1.2.1 Video image specification.

	Video image					
	size [pixels]		projected area [μm^2]		resolution [$\mu\text{m}/\text{pixel}$]	
Lens	horizontal	vertical	horizontal	vertical	horizontal	vertical
X40	768	576	414	313	0.539	0.544
X10	768	576	1648	1244	2.146	2.160

□ **Micropositioning system**

Motor controller: Physik Instrumente C812

Actuators: Physik Instrumente M-224.20:

-Faulhaber DC-micromotors 1331012S (3W, 12V, 11300rpm)

-Faulhaber Magnetic encoder (15 x 4 counts per revolution)

-Faulhaber Spur gearheads series 15/2: reduction ratio $(31/9)^4$

-Drive screw pitch 0.5mm/rev

-Travel: 25mm

-Maximal velocity 0.67mm/sec

-Maximal push/pull force: 40N

-Backlash: 5 μm

-Theoretical resolution:

translation: 16891.1 encoder counts/mm

rotation: 19444.4 encoder counts /deg

Stages: Physik Instrumente XYZR (magnetic with ball bearings)

□ **Data acquisition**

Detector: Oxford Pentafet model 6454

-Resolution 165eV

-Area 80mm²

-Preamplifier:

-Amplifier: Canberra spectroscopy amplifier model 2020

-ADC: Canberra model 8075

-Multi Channel Analyser: Canberra S100

-Software: Custom written (Spector)

1.3. X ray beam focusing

All X ray optics makes use of the wave nature of X rays. Therefore the refraction, reflection or diffraction and interference of X rays are fundamental for X ray optics. Although these properties are well known from visible light and widely used there, much shorter wavelength (typically around 0.1nm for 10keV X ray comparing with 600nm of visible light, $\lambda[\text{nm}] = 1.2398/E[\text{keV}]$) and different interaction with matter makes the focusing of X rays much difficult.

X ray beam size and intensity are critical parameters for X ray microprobe and similar techniques. Therefore a lot of effort has been made in past few decades to construct an efficient focusing system. The techniques can be summarised in three groups.

The first is based on diffraction and interference. Such are Bragg and multilayer mirrors, zone plates and phase gratings. Their common characteristic is spectral selectivity. Therefore they do not allow one to handle beams with a wide energy spectrum.

1. X RAY MICROPROBE

The refraction techniques have been considered impracticable for long time due to very small index of refraction and considerable absorption. But recently, it has been demonstrated that with the using of the so-called compound refractive lens (stacking between 20 and 150 small lenses with strong curvature of 0.2mm) a considerable focusing effects on relative short distance (1-2 meters) can be achieved.

The reflection techniques are most widely used in X ray focusing. They are based on the total external reflection. Typical examples are total reflection bent mirrors, capillaries and polycapillaries. These techniques are especially useful for focusing broad bandpass X rays. The capillaries and polycapillaries are widely used in X ray microprobe with conventional X ray sources.

1.3.1. Multilayers

A multilayer is a stack of (usually two) alternating thin layers (A,B) deposited N times on each other. Layer A usually consists of a strongly absorbing material (metal). Layer B is a spacer made of a low-density material. The resulting structure has a repetition period $\Lambda = t_A + t_B$ where $t_{A,B}$ are the single layer thicknesses. Due to different optical indices n_i of the two layers, the whole stack acts like a one-dimensional artificial Bragg reflector when exposed to an X ray beam. The properties of multilayers can be described in analogy to those of single crystals. Neglecting absorption and refraction, the wavelength λ , periodicity Λ , and Bragg angle θ are approximately related by the well known Bragg equation

$$2\Lambda \sin \theta_n \approx n\lambda$$

Since the multilayer period Λ ranges typically between 2.0 nm and 10.0 nm, the first order Bragg angle θ for hard x rays ($E = 5 \dots 100$ keV) is rather small ($0.2^\circ \dots 2.0^\circ$). The scattering power increases with increasing optical contrast between the layer materials. The reflectivity is strongly affected by interface roughness between adjacent layers. The penetration depth is limited due to strong absorption and extinction effects leading to a relatively low spectral resolution of about 10^{-2} . On the other hand, this offers a relatively high photon flux as compared to single crystals. By variation of the thickness ratio $\Gamma = t_A / \Lambda$ (the equivalent of the structure factor of single crystals) one can suppress undesired harmonics in the reflected spectrum.

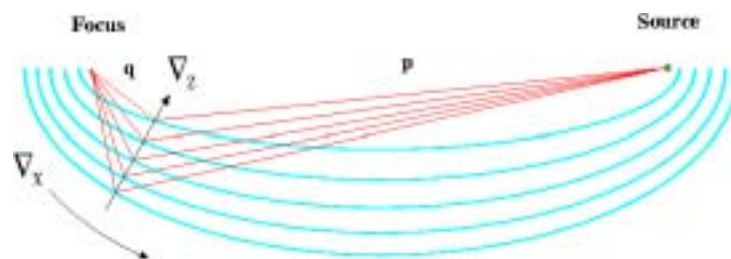


Figure 1.3.1.1. Focusing geometry using a multilayer mirror.

1.3.2. Refractive lenses

It is well known from visible light optics that a convex lens with equal radii of curvature R on both sides has focal length F

$$F = R/2(n - 1).$$

This means that for visible light a glass lens with index of refraction $n \approx 1.5$, radius of curvature of 10cm gives focal length of 10cm. On the other hand the index of refraction for X rays in matter can be written as $n(\lambda) = 1 - \delta(\lambda) - i\beta(\lambda)$ where $\beta(\lambda)$ is absorption index, $\delta(\lambda)$ is the refractive index decrement and λ is X ray wavelength (see 1.3.4 X ray total external reflection). The $\delta(\lambda)$ is typically between 10^{-5} and 10^{-7} .

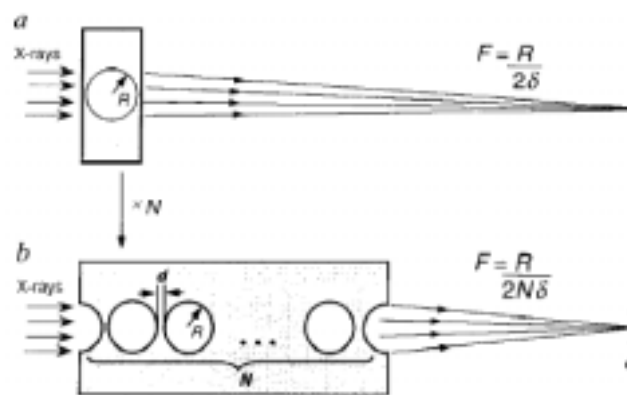


Figure 1.3.2.1. Principles of X ray focusing by a compound refractive lens.

A few things can be concluded from above:

- real part of n is smaller than one and focusing lens for X rays must have concave form rather than convex.
- $1 - \delta(\lambda)$ is very close to one and such a lens would have very small radii in order to get reasonable focus length. For example, for aluminium $\delta(\lambda) = 2.8 \cdot 10^{-6}$ and $F = 1m$ requires $R = 5.6\mu m$.
- although $\beta(\lambda)$ is small compared to one, X ray absorption in the lens material is not negligible.

All this has led to the widespread view that there is no chance of manufacturing refractive X ray lenses. However, the problems can be overcome by stacking many individual lenses behind one another with spacing d as smaller as possible (e.g. $20\mu m$), by making radius of curvature reasonable small (e.g. $200\mu m$) and choosing low-Z lens (see fig 1.3.2.1).

In 1996, the first refractive lens was built and feasibility of refractive optics for hard X rays was demonstrated. The lens was built in aluminium by drilling cylindrical holes. Such a lens focuses only in one direction. Focusing in both directions, resulting in a point focus, can be achieved in two ways. The simplest is to use two cylindrical arrays in crossed geometry. More complex approach uses

spherical hole shape. The lenses of second generation have parabolic shape and focal length of 1-2 m. Until now they are used only in experiments with synchrotron radiation sources.

1.3.3. Capillaries and polycapillaries

The reflectivity (ratio of intensity between incident and reflected beam) of X rays is normally extremely small and reasonable reflectivity can be achieved only below the critical angle for total reflection (see 1.3.4 X ray total external reflection). For all practical materials it is in the range of milliradians. In case the reflecting surface represents a certain channel limited in space, such as hollow capillary, X ray beam can be passed down the capillary tube, being reflected from the internal surface with very little loss. The loss depends on the shape of capillary and roughness of its surface. The most common shapes are straight, tapered, conical and ellipsoidal. Beam divergence is typically from 5 to 10 mrad for Cu $K\alpha$ X rays. The intensity gain versus an aperture collimator can be up to two orders of magnitude greater, depending on the system geometry and the source spectral characteristics.

The spot size depends on the exit diameter. On the Fig 1.3.3.1 are shown horizontal and vertical beam profiles for a capillary with declared exit diameter of 13 μm measured at XRF laboratory at IEAE Laboratories Seibersdorf.

By using many capillaries one can produce much intensive beam. This, so called polycapillary lens or Kumakhov lens after its Russian inventor, is basically an array of hollow optical fibers. It is possible to make arrays of hollow fibers, each containing hundreds or thousands of individual channels. There are two types of polycapillary optics which are differentiated by the way in which they focus or control x rays: Multifiber and monolithic.

Multifiber lenses are especially well suited for operation with line X ray sources in point mode and for large area parallel beam applications. The fibers are individually threaded through a series of thin screens bearing hole patterns precisely controlled to a variance of micrometers. A frame assembly supports the screens and maintains alignment. This optics contain between several hundred and several thousand fibers, each of which contain well over a thousand channels. In multifiber lenses, the channel diameter remains constant and uniform throughout the length of the fiber.

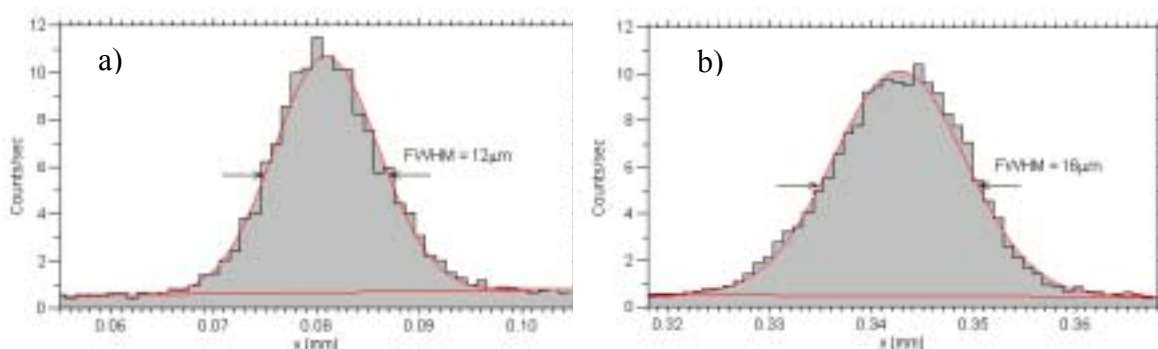


Figure 1.3.3.1. Horizontal and vertical profile of an X ray beam focused by a glass monicapillary (similar to one shown on the Fig 1.2.2) with exit diameter around 10 μm at distance of cca 2mm from outer exit. The profile was measured using Mo X ray tube, with 4 μm thick cross using steps of 1 μm .

1. X RAY MICROPROBE

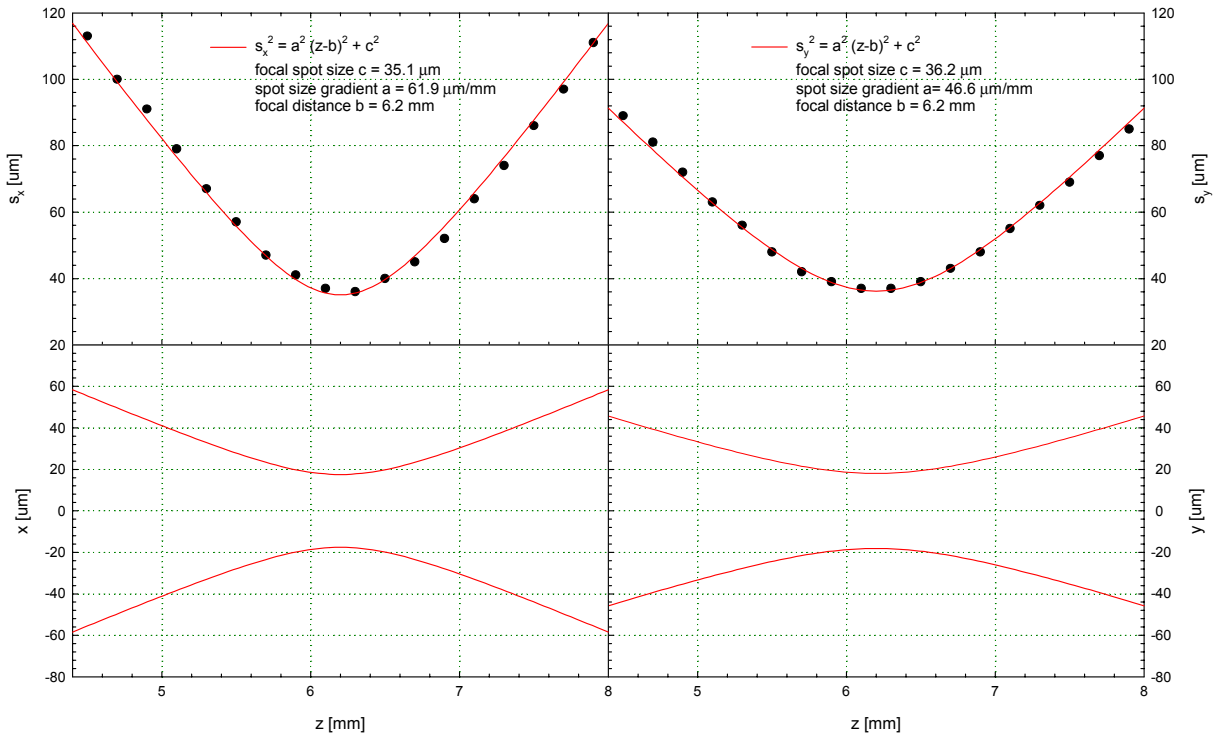


Figure 1.3.3.2. X ray beam spot size s_x and s_y in the horizontal and vertical direction respectively as a function of distance z from a monolithic polycapillary lens (791-01, X ray Optical Systems). The corresponding beam profiles x and y are shown on the bottom graphs. Measurements were performed with Mo tube.

Channel diameter, fiber diameter, capillary curvature, optic cross section, and even capillary composition are tailored to the specific application, providing peak performance. The multifiber X ray collimators produce parallel beam footprints ranging from few cm^2 up to few tens cm^2 . Focusing/converging multifiber optics can yield spot diameters as small as $400 \mu\text{m}$, depending on the energy level.

Monolithics, the second type of polycapillary optic, are smaller and more compact than the multifiber polycapillary lenses. Monolithic capillary of fibers that have been fused together and tapered so that no external frame assembly is required for alignment. Each channel is orientated toward the X ray source or sample. The most significant difference between a monolithic optic and a multifiber is that monolithics form significantly smaller-diameter parallel beams for collimating applications (0.5 to 5 mm) and smaller spot sizes for focusing applications (20-200 μm), depending on energy level.

In the Fig 1.3.3.2 is shown beam spot size, measured at various distances from exit of a monolithic polycapillary lens (X ray Optical Systems).

1.3.4. X ray total external reflection

X ray reflection and refraction is usually described in the approximation of the classical dispersion theory. In the case of mono-energetic X rays of wavelength $\lambda = hc/E$ it gives index of refraction $n(\lambda)$

$$(1) \quad n(\lambda) = 1 - \frac{N}{2\pi} r_0 \lambda^2 \sum_j \frac{\rho_j}{A_j} f_j$$

where N is Avogadro's number, $r_0 = e^2/m_e c^2$ classical electron radius, the sum is over elements in material, the ρ_j , A_j and f_j are the density, atomic mass and form factors of element j . In general $f_j = f_{0j} + f_j' + f_j''$, where f_{0j} is the Thompson scattering, f_j' and f_j'' are real and imaginary anomalous dispersion corrections. The index of refraction is usually written in the form

$$(2) \quad n(\lambda) = 1 - \delta(\lambda) - i\beta(\lambda)$$

where $\beta(\lambda)$ describes the attenuation of X rays in matter with linear absorption coefficient μ

$$(3) \quad \beta(\lambda) = \frac{N}{2\pi} r_0 \lambda^2 \sum_j \frac{\rho_j}{A_j} f_j'' = \frac{\mu}{4\pi} \lambda$$

and $\delta(\lambda)$, so called real part decrement of the index of refraction, in near forward scattering ($f_{0j} \approx Z_j$, Z is atomic number)

$$(4) \quad \delta(\lambda) = \frac{N}{2\pi} r_0 \lambda^2 \sum_j \frac{\rho_j}{A_j} (f_j' + Z_j)$$

On the other side, it is well known from visible light optics that, when light goes from one dens medium with index n' to one less dens medium with index n ($n' > n$), there is a certain critical incident angle θ_c^{light} above which all incident beams are totally reflected. In light optics the θ_c^{light} is measured from normal to the surface and it given by relation

$$(5) \quad \sin \theta_c^{light} = \frac{n}{n'}$$

This feature of visible light is known as *total internal reflection*. For instance θ_c^{light} goes from 42° to 30° as n'/n goes from $2/3$ to $1/2$. The relation (5) directly follows from the Snell's law (W. Snell, 1621). In the case of X rays, assuming the attenuation $\beta(\lambda)$ is zero and X rays incidence from vacuum ($n'=1$), the critical angle θ_c measured from surface (grazing angle) of material with index of refraction (2) is equal

$$(6) \quad \cos \theta_c = 1 - \delta(\lambda)$$

For X rays range of energies $\delta(\lambda)$ is very small for all know elements (see table 1.3.4.1 for Al and Be). Therefore, in absence of any attenuation ($\beta(\lambda) = 0$), index of refraction is very close to one and equation (6) can be developed into Taylor series $\cos \theta_c = 1 - \theta_c^2 / 2 + \dots$. Keeping only the first order term, one can get for the critical angle

$$(7) \quad \theta_c = \sqrt{2\delta(\lambda)}$$

Table 1.3.4.1. Attenuation $\beta(\lambda)$ and real part decrement $\delta(\lambda)$ of index of refraction for Al and Be at various energies

E[keV]	Be		Al	
	$\delta[10^{-6}]$	$\beta[10^{-9}]$	$\delta[10^{-6}]$	$\beta[10^{-9}]$
8	5.334	2.419	8.579	158.20
10	3.412	1.095	5.468	66.10
15	1.515	0.341	2.414	13.50
20	0.852	0.195	1.355	4.40
30	0.379	0.113	0.601	0.95

Substituting (4) into (6) and assuming material with only one type of atoms and $f' \approx 0$ one gets

$$(8) \quad \theta_c [\text{mrad}] = \frac{28.8}{E[\text{keV}]} \sqrt{\frac{\rho[\text{g/cm}^3]}{A[\text{a.m.u}]} Z}$$

As one can see from the table 1.3.4.2, the critical angles for various materials are in the range of milliradians.

Table 1.3.4.2. Critical angles θ_c for various materials at two energies

Material	$\rho[\text{g/cm}^3]$	$\theta_c (8.04 \text{ keV}) [\text{mrad}]$	$\theta_c (17.5 \text{ keV}) [\text{mrad}]$
Plexiglass	1.20	2.88	1.34
Glassy carbon	1.55	3.14	1.45
Quartz	2.20	3.75	1.72
Silicon	2.32	3.84	1.77
Aluminium	2.69	4.07	1.86
Titanium	4.54	5.17	2.38
Germanium	5.31	5.50	2.53
Steel	7.80	6.84	3.14
Tantalum	16.60	9.27	4.25
Platinum	21.41	10.47	4.83

Another important parameter for the beam focusing techniques utilizing reflection is ratio of the intensity between incident and reflected beam, so called reflectivity $R(\theta)$. The reflectivity can be calculated by means of Fresnel's equation which gives

$$(9) \quad R(\theta) = \frac{\left| \theta - \sqrt{\theta^2 - \theta_c^2 - 2i\beta} \right|^2}{\left| \theta + \sqrt{\theta^2 - \theta_c^2 - 2i\beta} \right|^2}$$

where θ is angle of the incident beam measured from surface. Fig 1.3.4.1 shows reflectivity of SiO_2 calculated by formula (9). The shape of the curves indicates that such a reflector can be used as low pass energy filter. By fine tuning angle of reflector, one can remove high-energy components from incident beam.

1.4. Micropositioning

The micropositioning system is used to move sample under the manual or automatic (software programmable) control with positioning accuracy and resolution in the micron or sub-micron range. The system features motion controller, amplifier, actuator and translation stages.

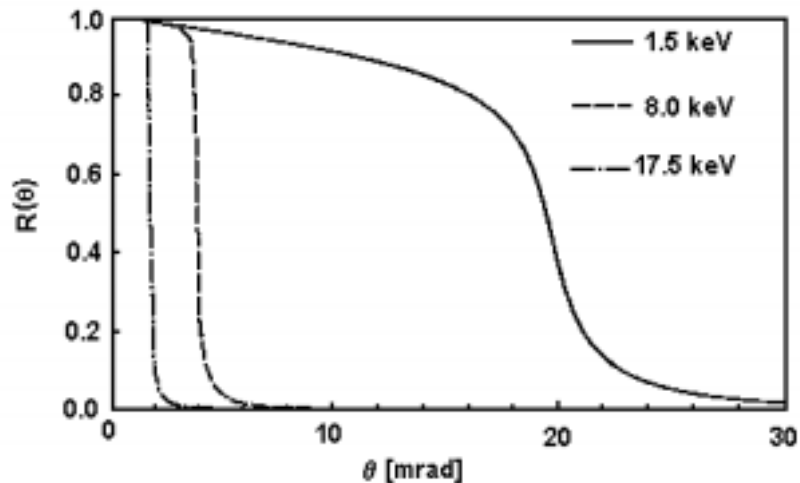


Figure 1.3.4.1. Reflectivity $R(\theta)$ of SiO_2 as a function of incident angle θ for various energies of X rays.

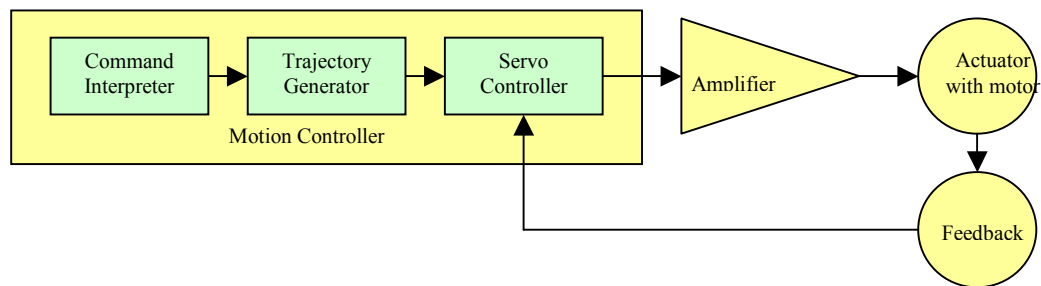


Figure 1.4.1. Functional diagram of a motion control system.

1.4.1. Motion controller

A motion controller is a device in a motion system that generates electronic signals instructing the motion mechanics to move or stop. If a feedback system exists, then signals measuring the actual motion output are returned to the controller. The controller compares the actual to desired output and generates an error signal upon which corrective action is taken. A controller can have various features such as data communications, input/output lines, memory for storing motion programs, and encoder feedback processing for closed-loop positioning. Although a displacement can take several seconds, the controller has to work at higher speeds to make calculations and output updated motion commands.

Common controllers use three types of control methods. They are *position* control, *velocity* control and *torque* control. The position type of control moves the load from one known fixed position to another known fixed position. Feedback, or closed-loop positioning, is important for precise positioning. Torque control measures the current applied to a motor with a known torque coefficient in order to develop a known constant torque.

Motion controllers are typically available as bus-based cards or in stand-alone configurations. Designed to be incorporated within a host computer, bus-based cards are available in most popular formats, including ISA, PCI, compact PCI, STD, PC-104, and VME. By residing on the computer's internal expansion bus, the cards can provide communication speed and flexibility. Unlike bus-based cards, stand-alone controllers operate without requiring installation in a computer. They have their own power supplies and enclosures. Communications take place via RS-232 serial links. The PI C-812 controller is implemented as ISA board and supports RS232 communications as well.

1.4.2. Amplifier

A motor amplifier, also referred to as motor driver, receives input signals from a controller and converts them to power to drive a motor. Basic step motor drives are fairly simple devices consisting of several power transistors that sequentially energise the motor phases according to the number of digital step pulses received from the controller. More advanced microstepping versions, which are common, control the phase currents in a staircase sinusoidal fashion, permitting a motor to operate with a much greater resolution and reducing a step motor's inherent resonance problem. Brush motor drives typically accept a ± 10 -V analog signal from the controller. This signal corresponds to a current or voltage command. Four power transistors arranged in an H-bridge control both the direction and magnitude of current in the motor windings. Like a brush drive, a brushless motor drive typically accepts a ± 10 -V analog signal from the controller. However, just as the typical brushless motor has a three-phase winding (delta or wye), brushless drives typically contain six power transistors arranged in three half-H-bridges as well as additional logic to control their timing as a function of rotor-position feedback. This logic is often expanded upon to employ advanced commutation schemes, such as sinusoidal current control and phase advance, to extract even more performance. The PI C-812 has onboard amplifier for four motors up to 2W.

1.4.3. Actuator

Actuators provide a method of converting a motor's rotary motion into linear motion. Options include leadscrews, ball screws, rack and pinion, or belt/cable/chain drives. A very popular technique for moving loads is to use the axial translation of a nut riding along a rotating screw. Leadscrews use sliding contact, so their wear rate is directly proportional to usage. The advantages of leadscrews include self-locking capability, low initial costs, ease of manufacture and a wide choice of materials. Usually the actuators host encoder, motor and gearhead. The whole assembly sometimes is referred to as actuator (Fig. 1.4.3.1).

The most popular of the many types of motors available are step, brush-type DC motors and brushless DC motors. Step motors are often selected simply because they can be run open loop; that is, without any feedback sensor. The position is simply controlled by the number of steps send to the motor. However, to support this, motor must keep continuous current in windings, which then causes motor heating. Brush motors are constructed with a wire-wound armature rotating within a magnetic field produced by a stationary permanent magnet assembly. As the motor rotates, current is sequentially applied to the appropriate armature windings by a mechanical commutator consisting of multiple brushes sliding on a ring of copper segments. Brush motor technology is quite mature and can provide very high performance.

However, the mechanical commutator often becomes a limiting factor. Brushless motors incorporate electronics to eliminate the need for mechanical commutation. Essentially an inside-out brush motor, a brushless motor consists of a permanent magnet rotor with windings distributed about the outer stationary housing. The mechanical commutator is replaced by either extra transistors and logic in the motor drive and noncontact Hall effect sensors in the motor; a commutating encoder; or, in some cases, extra software in the controller or drive. Compared to brush motors, brushless motors typically exhibit lower rotor inertia, lower winding thermal resistance, and no sliding mechanical contacts. These motors can offer higher top speeds, higher continuous torque, and faster acceleration without the brush dust, commutator wear, contamination, arcing, and electromagnetic interference (EMI) or radio-frequency interference (RFI) of a brush assembly. Recent advances in semiconductors, specifically in power devices and microprocessors, have significantly reduced the cost disadvantage of brushless motor systems. The majority of new designs employ brushless technology. Comparing with step motors, the DC motors have lesser motor heating, produce very little vibration and move smoothly. On the other side they need more complex feedback.

Gearboxes (see Fig. 1.4.3.1) are used to transmit rotational motion and power. They are frequently used in reduction mode to produce increased resolution that may be difficult or impossible to achieve with standard motors. A 10:1 gearbox, for example produces one turn of its output shaft for every 10 turns of its input shaft. A 200-step motor with this gear combination would be viewed as a device, which has an effective resolution of 2000 points per revolution of the output shaft. In practice, non-integer gear ratios are used for the following reason: In the process of cutting gears, there is a small but inevitable error in tooth profile. The profile error usually manifests itself in the last tooth. If the gear ratio would be an exact integer than this tooth would repeatedly mesh with same tooth pairs on the counterpart gear. In practice 31:9 ratio is often used. The gear ratio may be changed adding or removing one or more 31:9 passes to the input end of the gearhead. This gives for total gear ratio $(31/9)^n$ where n is number of passes. The Seibersdorf's system gear box also uses the same ratio but

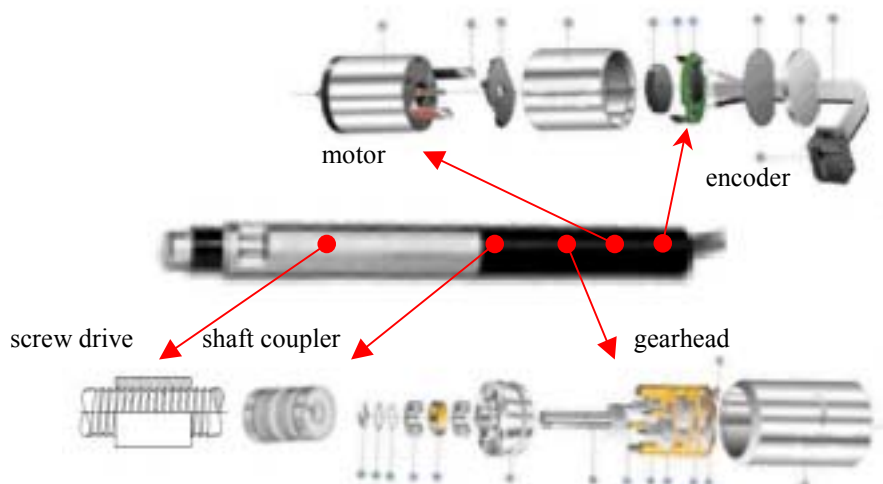


Figure 1.4.3.1. Physik Instrumente PI M-224.20 High Resolution DC Mike Actuator features Faulhaber's 3W brushless DC micromotor series 1331012S, corresponding magnetic encoder with 60 counts/rev, gearhead with ratio $(31/9)^4$ and drive with 0.5.

has $n=4$. This implies that for about every 141 turns made at the input the output is reduced to only one turn, therefore increasing resolution.

Couplings (see Fig. 1.4.3.1) are used in a drivetrain to transmit power and motion between two independent shafts, which may not be perfectly aligned. Flexible couplings generally allow for some parallel and angular misalignment. Depending on their design, they may accommodate more misalignment, have higher torsional stiffness and load capacity, or be capable of higher speeds. For example, helical couplings feature no backlash and constant velocity under misalignment and they are adaptable to high-speed applications as well.

Usually samples can't be directly mounted on the actuator, because of their very small transverse load capacity. Therefore they are mounted on platforms called mechanical stages. The stages are two mechanical surfaces separated by suitable bearings. The most common bearings are ball and crossed-roller. The stages can either translate (linear stages) or rotate (rotary stages). The translation and rotation are constrained to have only one degree of freedom. By combining two or more stages one can get motion in two or more axes. For example XY or XYZ combination are common. The most critical parameter for stages is load capacity.

1.4.4. Feedback (Encoder)

A feedback device transforms a physical parameter like position into an electrical signal for use by a motion controller. They are most often mounted integrally with the motor, but sometimes on output gear shaft. Common feedback devices include encoders for position feedback, tachometers for velocity feedback, and accelerometers for acceleration feedback. The position parameter is ultimate for micro-positioning. If encoder output consists of electronic signals corresponding to an increment of linear or rotational movement only relative position/angle is possible. Such encoders are called incremental and are widely used in micropositioning. Quadrature encoders are a particular kind of incremental encoder with of at least two output signals, commonly called channel A and channel B. The channel B is offset 90 degrees from channel A. The addition of a second channel provides direction information in the feedback signal. The ability to detect direction is critical if encoder rotation stops on a pulse edge. Without the ability to decode direction, the counter may count each transition. A rotary encoder can differentiate a number of discrete positions per revolution. This number is called *points or counts per revolution* and is analogous to the steps per revolution of a stepper motor. Linear and rotary stages and actuators incorporating indirect metrology use rotary encoders measuring the motor shaft or leadscrew angle to report position. For example encoder with 60 counts per revolution attached to motor that drives 4 passes 31/9 gearhead, coupled to drive screw with 0.5 mm pitch gives $60 \times (31/9)^4 \div 0.5 = 16891.1$ counts/mm. The most common encoders are optical and magnetic. The first features a disc or a plate containing opaque and transparent segments passes between a light source (such an LED) and detector to interrupt a light beam. The electronic signals that are generated are then fed into the controller where position and velocity information is calculated based upon the signals received (the speed is determined mathematically by differentiating pulses in real time).

1.4.5. Specification

In the micropositioning a few specifications are used in order to feature the system capabilities. They are summarized on the Fig. 1.4.5.1 and explained as follows.

Resolution is the smallest position change that a motion system can detect, which is not the same as the minimum incremental motion. Also referred to as display or encoder resolution, it is usually determined by the encoder output, but due to inefficiencies in the drivetrain—like hysteresis, backlash and windup—most systems cannot make a minimum incremental move equal to the resolution unless the encoder is directly measuring the delivered motion.

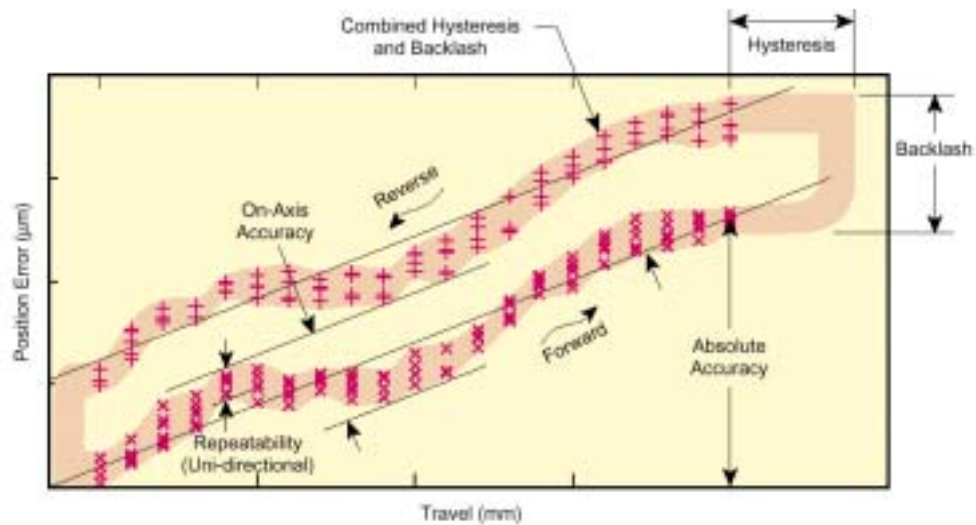


Figure 1.4.5.1. Error vs. position for a stage translated four times in each direction.

Backlash is lost motion upon reversing direction of motion. Backlash especially arises from gear meshes but also from the compressibility of materials. When you are driving a stage and there is backlash in the system, and then you reverse the actuator, initially nothing happens; the stage doesn't start reversing until the system backlash is compensated. Often, this means not until the gear teeth make contact in the reverse direction. Some micropositioning systems exhibit a highly reproducible and consistent backlash of about few μm . It can be compensate for reproducible backlash when moving from one known position to a target position when using software driven devices.

Hysteresis describes the difference in behavior of an instrument operated firstly in one direction and then in the reverse. Usually the backlash dominates hysteresis.

Accuracy is maximum expected difference between the actual and the ideal position for a given input. Accuracy of a motion device is highly dependent on how the actual position is measured.

Precision or Repeatability is the range of deviations in output position that will occur for 95% of the motion excursions from the same error-free input. Manufacturers often specify *uni-directional repeatability*, which is the ability to repeat a motion increment in one direction only. This specification side-steps issues of backlash and hysteresis, and therefore is less meaningful for many real-world

1. X RAY MICROPROBE

applications where reversal of motion direction is common. A more significant specification is *bi-directional repeatability*, or the ability to achieve a commanded position over many attempts regardless of the direction from which the position is approached.

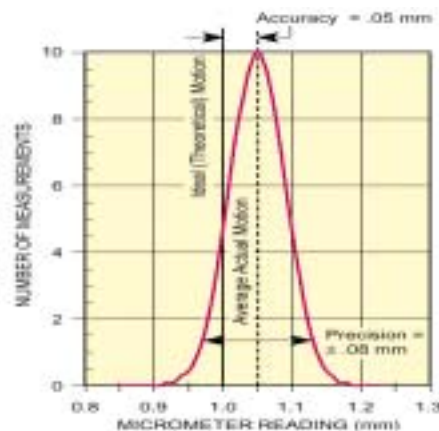


Figure 1.4.5.2. The difference between accuracy and precision in micropositioning systems.

1.5. Data acquisition

1.5.1. Getting started

1.5.1.1. Program structure

The X ray microprobe acquisition is fully controlled by a program package called Spector (*Spectrum collector*). It is a general-purpose multi-parameter data acquisition software originally developed for the FAST Com-Tec MPA/PC multi-parameter hardware (PC card and external ADC bus-box) running on the Win95/NT operating system. The system is in use in a number of laboratories for a variety of applications in a low energy nuclear physics (few particle nuclear reactions, PIXE microprobe, small-angle X-ray scattering (SAXS)...). The modular program structure allows adjusting to various types of hardware.

The software operates in two basic modes: list and PHA. In the PHA mode the data coming from ADC are sorted by dedicated hardware and stored there in associated memory. The PHA requires a simple pulse-height sorting. The software periodically copies an external memory and shows the corresponding spectra on the computer monitor. In the list mode data are collected event-by-event and stored in a buffer, that is, a successive memory locations of an external memory. The software reads the buffer, makes data sorting (on-line data analysis), shows the sorted data in graphical form, and optionally, stores event-by-event data to the storage device for additional off-line analysis. In the XRF microprobe application the list mode is utilised in such a way that each event is defined by the sample x and y position (two parameters) and energy spectrum collected at this position (third 'parameter').

The software consists of several applications and related drivers/dynamic link libraries. The XRF microprobe uses three main applications: MNG, MDI, and MOTOR.

1. X RAY MICROPROBE

The Spector's manager, MNG.EXE, is the main acquisition program and performs high-level hardware control, data input/output and data sorting to one or two-dimensional histograms called displays. It uses several dynamic link libraries to perform different tasks. To enable customized data analysis a user written parameters, that is, any function of ADCs channels are possible to define by creating dynamic link library with generic name prmlib.dll. The off-line analysis is supported by another dynamic library called replay.dll. A motor controller specific library that is loaded on start and released on stop of acquisition supports motor driving.

The graphical presentation is dedicated to the executable called Multi Display Interface, MDI.EXE. It also features analysis tools: display manipulation like arithmetic operation between displays, projection, and graphical comparison, ROIs etc. In addition it enables to define scan area graphically or numerically, display and grab video image by using appropriated dynamic link library.

The MOTOR.EXE enables to drive up to four motors by using graphical user interface or keyboard. It is motor board independent and uses the same library as mdi.exe.

The various program parts communicate via Windows message queues and memory sharing. The whole software is controlled by user-friendly window controls (dialog boxes, list boxes and other common controls, toolbars, menus...). To support full control from a command line or a batch file mode an interpreter called Parser has been developed. In addition, a control from a remote computer via RS232 or network is possible.

1. X RAY MICROPROBE

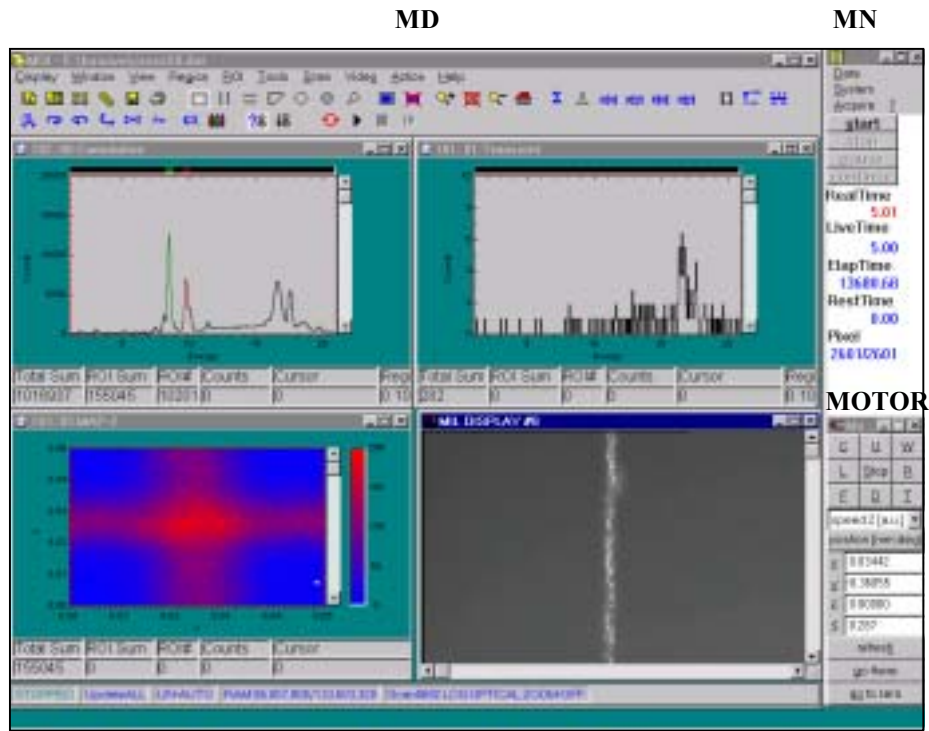
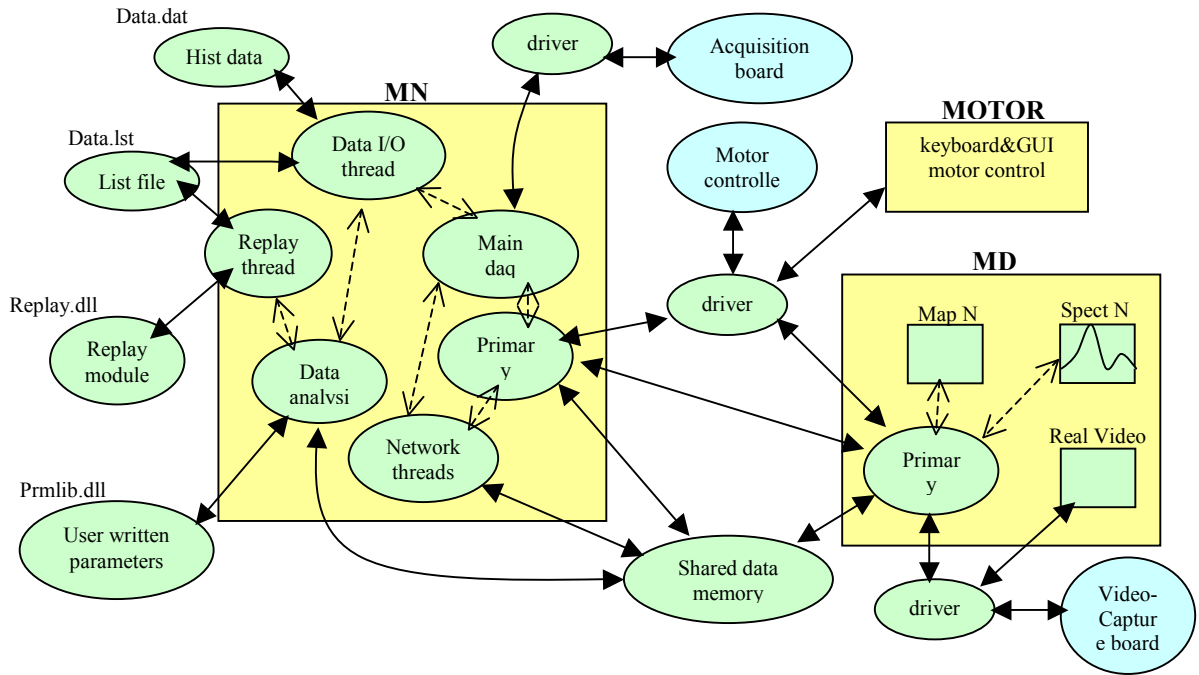


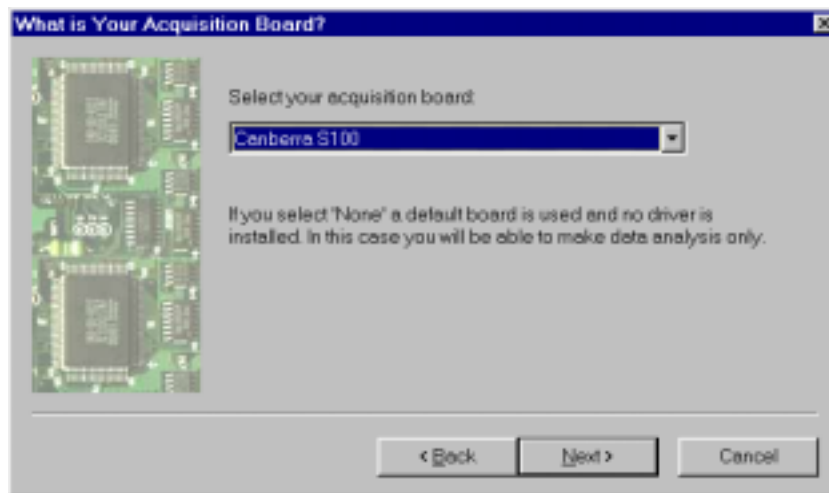
Figure 1.5.1.1. Spector's basic components: schematic diagram (up) and spectrum graphical user interface (down).

1.5.1.2. Program installation

The program installation is a simple wizard that guides you through the process of installation. The wizard consists of the following basic steps:

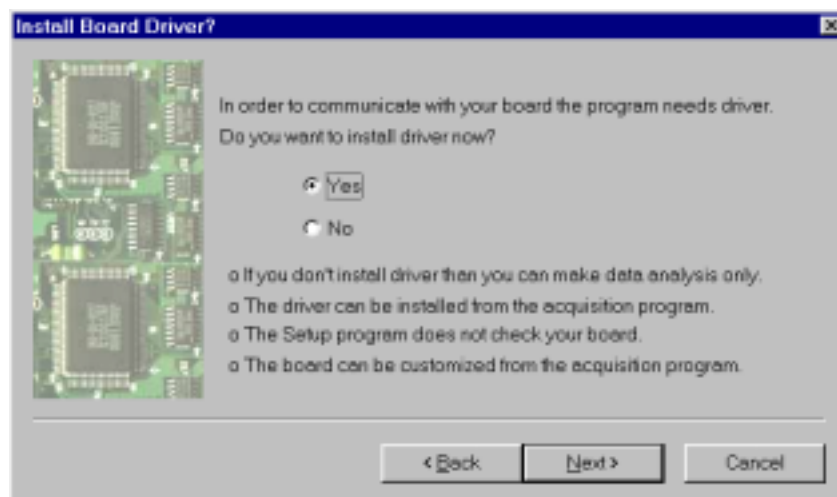
1. *What is your data acquisition board?*

If you select none the program installs Canberra S100 but without any driver, therefore you will not be able to measure. In addition the wizard skips to step 4. Note that you can select the board from already installed program.



2. *Install board driver?*

Select 'No' if you plan to make data analysis only. Otherwise select 'Yes'. If you select 'No' the wizard does not need any board information and skips to step 4. If you



select 'Yes' the wizard will install corresponding driver (s100.sys for Canberra S100, mpapc.sys for FastComTec MPA/PC, dsprot31.sys for DiconLab DSProt31) but only under the Windows NT/2000.

1. X RAY MICROPROBE

3. *Acquisition board information*

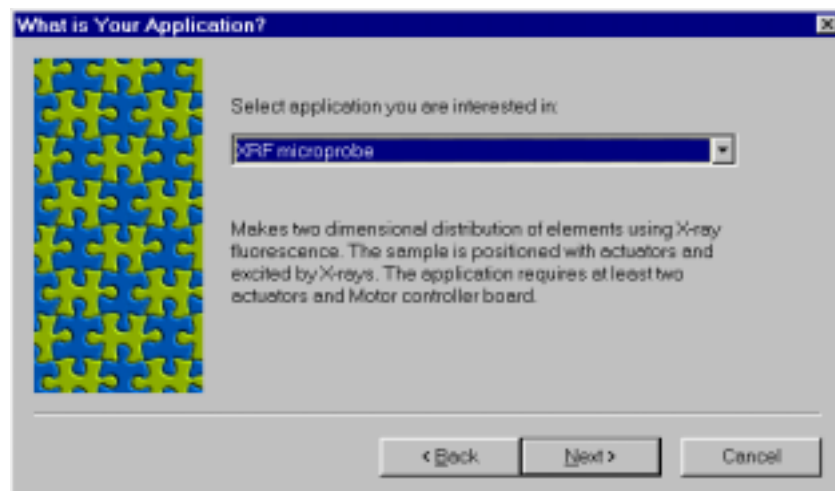
The information depends on the board. Usually it is only base I/O address. All other properties that may be application dependent or not necessary to access the board are set to default. Latter, you can adjust them from already installed program. The address you select here must match jumper selection on your board. For correct



jumper settings see board manual or 2.2.2 MNG, MCA. The wizard always offers default information.

4. *What is your application?*

Select here XRFM microprobe.



5. *What is your motor controller board?*

This step appears only for the XRFM and TXRF application. In these cases you need motor controller board installed on the same computer and at least two actuators. If you don't have any board installed then select none and wizard will skip to the step 7. Note that current version supports only PI C812 motor controller board.

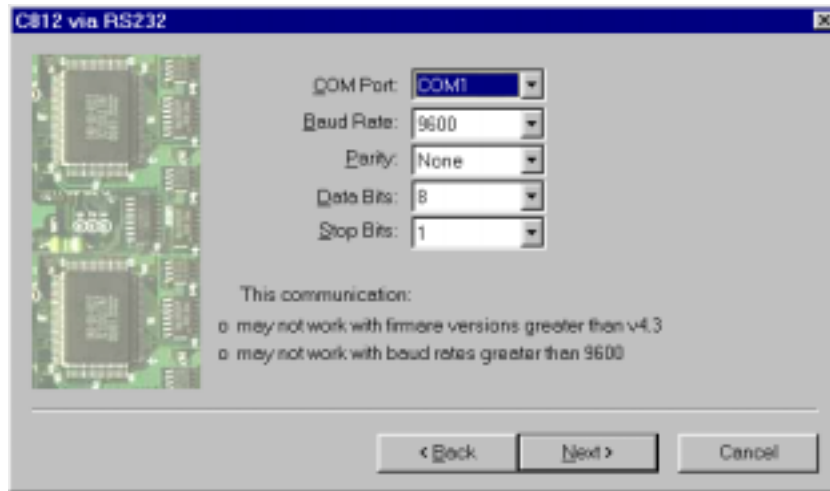
6. *Motor controller information.*

The PI C812 has two channels of communication: via serial port (RS232) or ISA bus. In both case the program asks only for the information necessary to access the board.

1. X RAY MICROPROBE

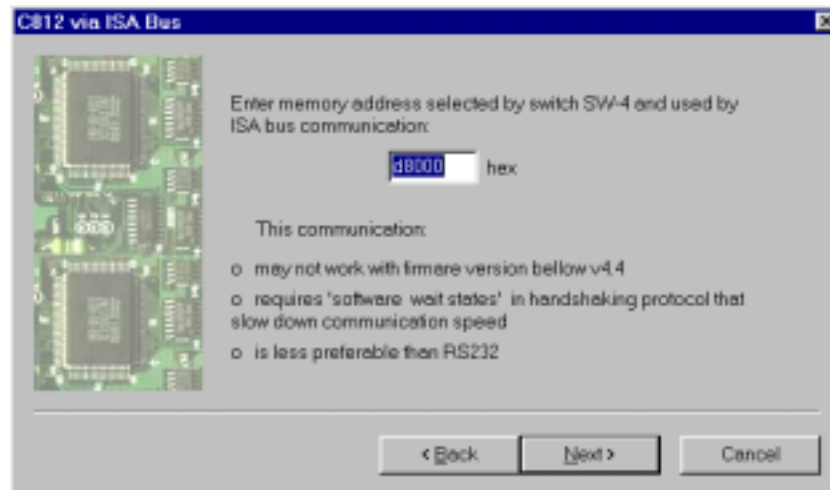
All other properties (gain, torque...) are set to default and you can later change them with using Motor application. They are stored in Motor.ini file.

The RS232 channel is preferred due to hardware handshaking problems in ISA mode. If you select RS232 the wizard asks for the host's serial port settings. You may



need to change Port ID. In addition you need a serial communication cable (see 4.3.1.1.2 Motor section).

For ISA bus you should enter base memory address. Use the default value if not in conflict with other hardware. When running in ISA mode program uses several software wait-states that are set to default and may not be proper for your computer. After installation, it is highly recommended to run Motor application, Board properties sheet – Communication dialog and make the following: (1) 'loop-speed test', (2) disable 'auto loop-speed test', (3) test communication.



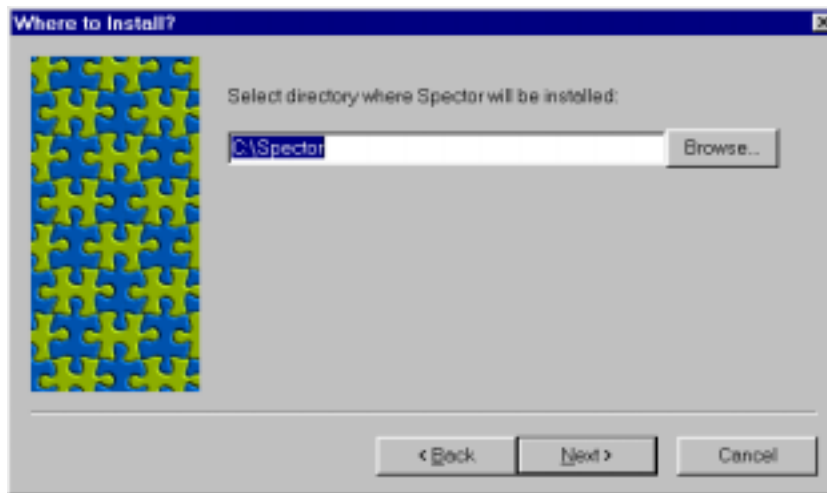
For both channels the default values correspond to default positions of board's jumpers and they are highly recommended for selection. Note that program supports only firmware versions that are listed in the table 4.3.1.1.1 of the Motor section.

You can install/update board from already installed software with using Motor application. Therefore you can proceed.

If you don't get controller working from the installation program or you realise some problem in moving motors then read 4.3.1.1 Motor, Board properties, C812 and run Motor application ... 'Board properties' command.

7. *Where to install?*

You should enter programs root directory. Normally it is 'c:\spector'. The program creates several sub-directories. The most important are 'bin' that contains binary files, 'data' with data files and 'scan' with scan related information.



8. *Complete Install*

Starts installation.

Note: In order to scan you can use video image, but it is not necessary. If you want to use video image then the program requires Matrox Meteor capture-board. The board must be already installed together with Intellicam software. The Intellicam is not required by the Spector, but it installs all necessary Matrox's libraries (mil.dll, milhost.dll, multiff.dll, milvga.dll and milvhook.dll) that Spector's installation CD does not contain. The files will be installed in Window's system32 subdirectory. Other Matrox's board are also possible but they are not tested.

1.5.1.3. Running program for the first time

After installation you can run the program from the 'Start... Programs ... Spector ... Mng'. It starts the main acquisition program MNG and spectra graphical interface MDI.

The MDI shows three empty spectra in cascade: two single and one map. You can load an example supplied by the installation CD with using of the MNG 'Data Load' command. It opens system's dialog box that shows the following three examples:

- cross.dat – 50µm x 50µm XRF image of 4µm thick tungsten cross
- hash1.dat – 6mm x 6mm XRF image of a piece of iron
- hash2.dat – 2mm x 2mm XRF image of a piece of iron

❖ Load hash1.dat. You can do it with using of the MNG 'Data Load command'. It opens standard Window's Open dialog. There you can see above three data files. Select hash1.dat. and program will opens four spectra in cascade. You can tile them by the MDI 'Window... Tile' command or by pressing second bitmap in the first row

1. X RAY MICROPROBE

of the MDI's toolbar (see Fig 1.5.1.3.1). It gives one map that corresponds to Zn-K_α and other to Fe-K_α characteristic X ray line. One single spectrum corresponds to the total Energy spectrum collected over the full scan area and other is collected at last scan pixel. You can do the following:

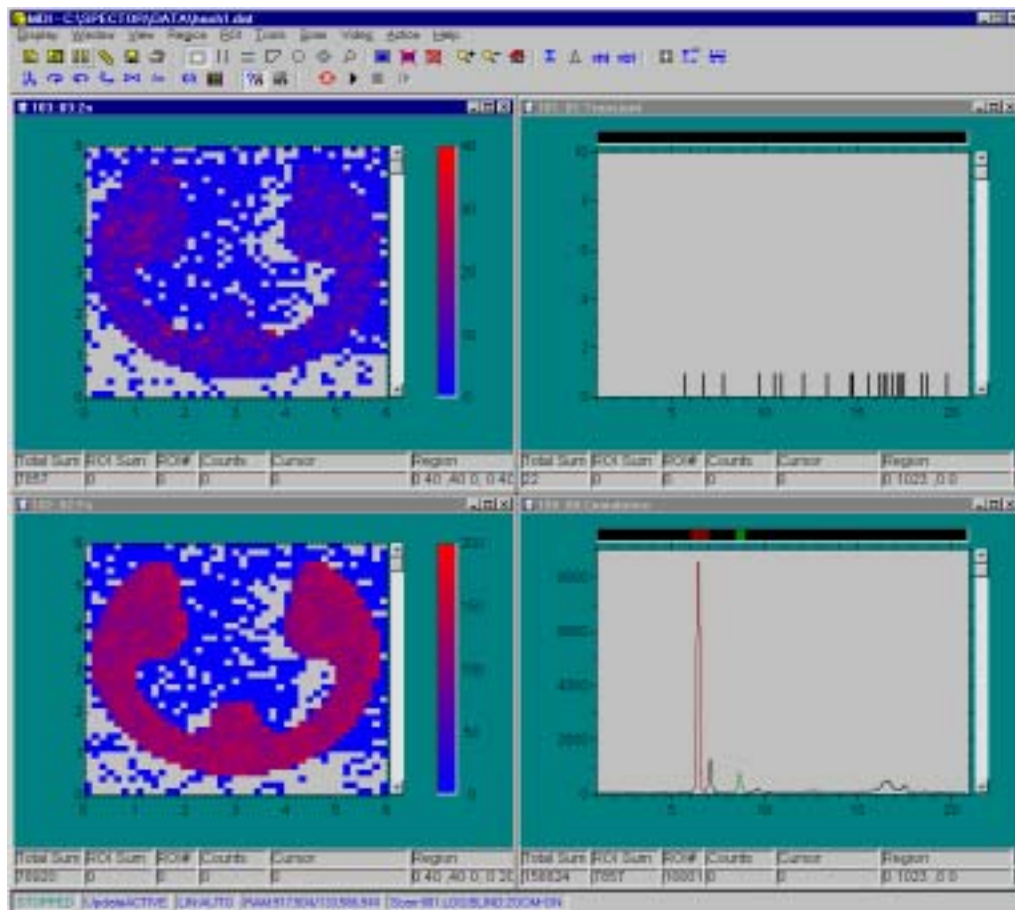


Figure 1.5.1.3.1 Program view after loading 'hash1.dat' and tiling spectra windows

- ✓ Click by left button on the title bar of one of map's window to activate this window.
- ✓ Click on the left mouse button inside the map to get cursor and read coordinates and number of events at this coordinate at one of controls bellow the map.
- ✓ Move cursor quickly by pressing left button and dragging mouse inside the map. Drag it to the left-bottom corner of the map. For this position you can read (0,0) coordinate in channels and (0,0) in mm. If you drag mouse to the right-top corner you can read (40,40) channels and (6,6) in mm. This means that scan area is 6mm x 6mm.
- ✓ Move cursor in fine steps with using of arrow keys. Note that each time you click an arrow the coordinate in channels change for one channel and 150 μ m. This means that scan resolution, that is, distance between two successive motor positions, was 150 μ m in both directions.
- ✓ Press right mouse button somewhere inside the map, drag it and finally release. This creates a rectangle. If you apply 'Region zoom' command you will get this rectangle enlarged. To go back use the 'Region Home' command.

1. X RAY MICROPROBE

✓ Repeat above but don't zoom. Instead of that use 'ROI... Create' command. It creates a ROI and shows it in inverted colour. You can delete it with using of 'ROI... Delete' command.

✓ Activate single spectrum with name 'Cumulative' on the same way as you did for map. Above the spectrum image there is a bar with two colours. Each of them corresponds to one ROI, that is, X ray line of Tungsten and to one of maps as well.

✓ Click left button over one of colours on the above bar. This activates corresponding ROI and you can read total number of events in this ROI. The number is equal to the total sum in corresponding map.

✓ Double-click left button on the same place as above. Its zooms the ROI. To go back use the 'Region Home' command.

✓ If you want to zoom an arbitrary part of the spectrum then use the same method as described for the map.

✓ If you want to create a new ROI use same method as described for map.

❖ Load hash2.dat. The example shows also two maps. One map corresponds to Zn-K_α and other to Fe-K_α characteristic X ray line.

✓ Verify that scan area was 2mm x 2mm and resolution 50 μm.

✓ Use 'Display Read' command. It opens system dialog box. Select there for 'File of types' SPECTOR (*.dat). This shows has1.dat and hash2.dat. Select the first and press OK. Now you can see a list of spectra from hash1.dat sample. Select third in the list (Fe) and press OK. It opens Fe map. Now you can compare two maps and check which part of image from has1.dat was scanned in the has2.dat.

❖ The data you have loaded ('dat' extension) are already sorted in XRF maps and only Cumulative spectrum is available. In order to make off-line analysis or to see spectra in each pixel you should read list files ('lst' extension). The installation CD contains two list files that correspond to above two data files. In order to read list file do the following:



Figure 1.5.1.3.2 The Start dialog box in replay mode.

✓ Open MNG Data menu and check 'Replay' item.

✓ Press MNG 'Start' button. It brings up 'Start replay' dialog. Press there 'Browse' button. It brings up a standard Window's 'Open' dialog with a list of 'lst' files. Select 'hash2.lst' and press Open button. To exit the 'Start' dialog press OK. Now, the program is reading list file. You can watch how the maps are updating. Wait the process terminates.

- ✓ Tile all windows. You can see exactly the same data that you saw by the hash2.dat .
- ✓ Activate one of maps, place cursor at arbitrary position and apply ‘Tools... Show Cumulative...’ command. It opens a new window that contains spectrum collected at sample position defined by the cursor.
- ✓ Create a ROI on one of two XRF maps as explained above and apply ‘ROI... Show Cumulative’ command. Again the standard Window’s ‘Open’ dialog is brought up. Select ‘hash2.lst’ and press Open button. It opens a new window that contains energy spectrum collected at all sample points that belongs to selected ROI.
- ✓ Close all windows.
- ✓ Switch off ‘Replay’ mode.

- ❖ Load cross.dat. The example shows also two maps. One map corresponds to $W-L_{\alpha}$ and other to $W-L_{\beta}$ characteristic X ray line. One single spectrum corresponds to the total Energy spectrum collected over the full scan area and other is collected at last
- ✓ Verify that scan area was $50\mu\text{m} \times 50\mu\text{m}$ and resolution $1 \mu\text{m}$.
- ✓ Repeat all steps from the first example

1.5.1.4. Measuring for the first time

It is assumed that you have run the program for the first time. If not then see 1.5.1.3 ‘Running program for the first time’.

- ✓ Find a small piece of material that contains Fe or Zn (sample) and mount it on the holder.
- ✓ Run the Motor application. It is accessible through the ‘MDI ... Scan ... Motor’ command or from the Window’s ‘Start’ menu. Move the ‘sample’ with using of the Motor control. If motors do not move then refer to 4.3 Motor, Board properties. If you want to see the border on XRF image than move the ‘sample’ on such a way that watching from exit of capillary the border is a little bit up and right (for example 2-3 mm).
- ✓ Load hash1.dat examples from 1.5.1.3 ‘Running program for the first time’. It defines two XRF maps: one for $Fe-K_{\alpha}$ and one for $Zn-K_{\alpha}$, each of them with dimension 40pixels x 40pixels. In order to change it you can use MNG ‘Acquire... Displays... Edit... Map resolution’ command. The preset time inherited from example is 0.5sec per pixel. You can change it by the MNG ‘Acquire... MCA... Presets’ dialog box.
- ✓ Apply the MDI ‘Scan... Export... Rectangular Area’ command. It brings up a property sheet. There, select for Position ‘*in the front of capillary...*’ and then press ‘*read motor*’ button (see Fig. 1.5.1.4.1). After that set for Size 6mm x 6mm or similar. Finally, press OK to exit from the property sheet. Note that scan area size in millimetres together with above map size in pixels defines scan step size in millimetres, that is, resolution of your target XRF image in the same units.
- ✓ Turn on the beam and press MNG ‘Start’ button. It brings up ‘Start Acquisition’ dialog shown on the Fig. 1.5.4.1.1. Check ‘Display file’ box and write bellow a ‘dat’ file name. The program will automatically save all spectra and maps in the file immediately after measurement stops. Check ‘list file’ and enter bellow a ‘lst’ file name. This will save spectra for each scan pixel during measurement and close file immediately after measurement stops. If you don’t check a box the corresponding type of data will not be stored. Still you can save maps and cumulative spectrum manually after measurement stops but not spectra per pixel.

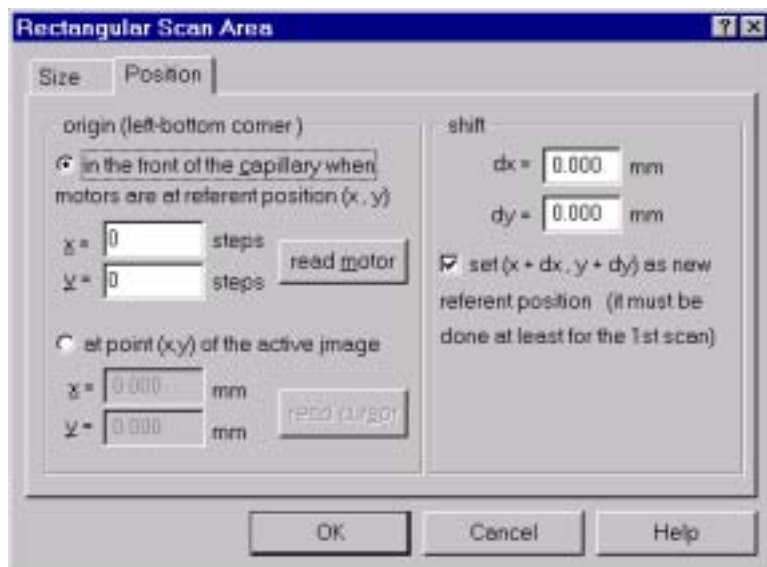


Figure 1.5.1.4.1. The 'Position' dialog box of the 'Rect, Scan Area' property sheet.

That's all! You are already measuring with the program for the first time.

Have a look in the Cumulative spectrum. If the peak of Fe-K_α and Zn-K_α X ray lines does not fit into ROI this means that you don't have the same amplification as it was in the example. Nothing strange. Stop the measurement. Activate ROI for Fe by pressing on its bar, drag button around Fe-K_α line and use the 'Region... Replace ROI' command. This moves ROI to the proper place. Do the same with Zn's ROI. If you still wonder about wrong calibration let fix it. Use the 'Tools... Calibrate'



Figure 1.5.4.1.1 The 'Start' dialog box in the acquisition mode.

command. Now, you can repeat measurement.

Once you have finished measurement you will get XRF image of your sample. From there you can define a new scan area and run again. To do this: create a ROI on the XRF image, apply Scan .. Export... Image ROI command and start measurement again. The result will be new XRF images of sample area defined by previously created ROI.

By reading the next paragraph you can better understand how the scanning works, use more program's features and eventually video image in order to define scan area.

In the next step you can do the examples shown in 1.5.3 Examples and if necessary, read related topics in section 2 MNG, 3 MDI and 4 Motor that explains particular program's commands.

1.5.2. Scanning

The scanning is performed in such a way that sample is moved by the micro-positioning system in the front of a fixed beam. In order to scan a sample you should select a part of its area and define steps in which micro-positioning system will move the sample in the front of the beam, that is, capillary or lens.

The scan-area can be selected from a video image or you can define it in reference to the sample point that is currently in the front of the capillary. The first case refers to the *optical* scan and the last to the *blind* scan mode. In both cases the result will be an XRF image (target image) with rectangular frame that you can use to redefine new scan area by inheriting scan type.

In all cases the area can be defined with using of the 'Scan' menu item in the MDI application, particularly the 'Scan... Export... Rectangle' for rectangular shape and the 'Scan... Export... Image ROI' command for arbitrary shape. In both cases the program creates a new logbook file (scan#.log) and overwrite existing scan data file (scan.dat) that is used by the MNG to drive motors.

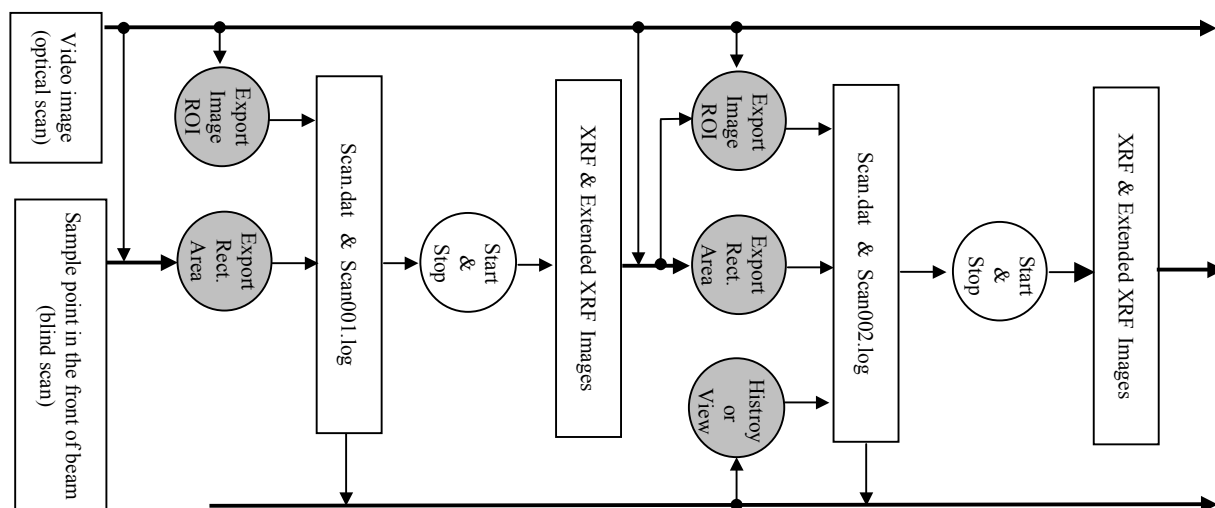


Figure 1.5.2.1 All possible ways to define scan-area in a series of measurements. The corresponding MDI commands are indicated by the gray circles.

In any time you can activate an old scan area, that is, scan logbook file, by the 'Scan... History' command and repeat measurement. In addition you can view and modify current active scan area by the 'Scan View' command. The scan area can have various shapes for example: a simple rectangle, polygon, group of polygons, circle, circle pie, ...

The step size is defined indirectly by choosing size of the target image in pixels and taking into account size in millimeters of the minimal rectangle that surrounds above defined scanning area. The pixel size a target image can be defined by the ‘Acquire Displays’ command in MNG application.

The program supports different direction of scan, which can be defined by the ‘Scan Properties’ dialog in the MDI application. Regardless of the scan direction, it always starts to scan from left-top corner of selected rectangular scan area.

1.5.2.1. Geometry

By default, it is assumed that axis of horizontal (x-axis) and vertical (y-axis) actuators are perpendicular to the beam (z-axis) and all together form a three-dimensional left-handed coordinate system (see Fig. 1.5.2.1). Normally, sample is in the x-y plane and rotation motor shows zero degree. You can read this angle by the ‘Motor’ application.

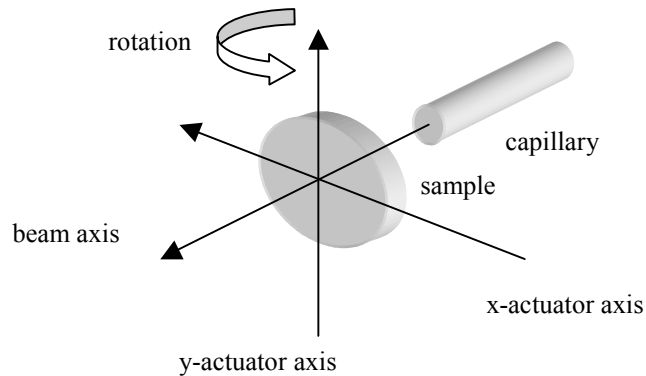


Figure 1.5.2.1. The scan geometry. For each actuator the encoder counts increase in the direction of the corresponding arrow.

If the Motor application does not show zero for rotation angle, the program automatically assumes that sample is not perpendicular to the beam. Therefore it moves sample in x direction on such a way that distance from irradiated point to the exit of capillary is always equal. In addition it takes care of capillary making proper order of z and x translations. This means that rotation angle should be always zero, if the sample is perpendicular to the beam. Generally, it is advantageous if the x-actuator axis is parallel to the plane of sample.

For each actuator (or simply motor) the program uses a constant that defines number of encoder counts per millimetre. This constant A is intrinsic for each actuator and defined as a function of encoder counts per revolution N_e , gearhead ratio r_g and drive screw pitch size s_d with the following relation,

$$(1.5.2.1.1) \quad A = \frac{N_e r_g}{s_d}$$

The constant a may be defined by the Motor application. For its value please refer to the Section 1.2 and the specification of the experimental setup, particularly

micropositioning system. The values are stored in ‘Motor.ini’ file. For default values refer to 1.2.2 Setup, Specification.

1.5.2.2. Blind scan

In Blind scan the area is defined with reference to (a) a sample point that is in front of capillary or (b) part of an XRF scanned image obtained from (a).

The blind scan is very simple because all you need is to select an area on your sample surface and export it. In addition you have to define XRF image resolution.

1.5.2.2.1. Coordinate system

Blind scan assumes the x and y motors are moved in the scanning coordinate system which (0,0) coordinate in millimetres (origin) is set at point O positioned in

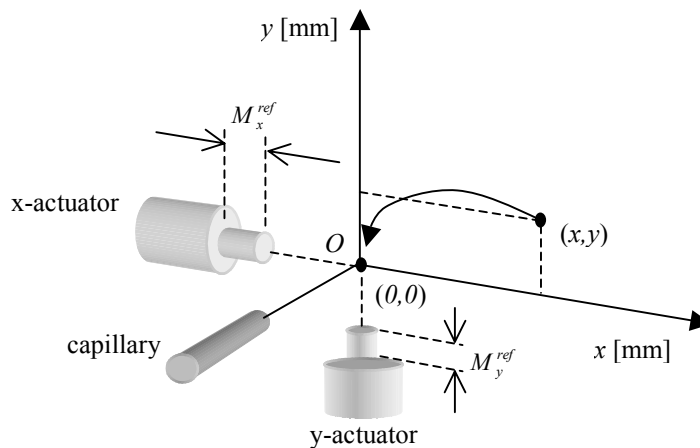


Figure 1.5.2.2.1.1. The Scanning coordinate system for a blind scan.

the front of capillary when motors are at position (M_x^{ref}, M_y^{ref}) steps (see Fig 1.5.2.2.1.1). This motor position is called motor referent position. Therefore, you should define this referent position with using of the ‘Scan... Export... Rectangle Area’ command. You must define this position at least one time. Usually you don’t need to define it more than one time in a series of measurement. This is because new scan area can be easy defined from a collected XRF image keeping the same referent position. You are free to move motors with Motor application and it has no effect on the referent motor position.

During scan, x and y motors are moved on such a way that all scan points are placed, one by one, in the front of capillary, that is, to the coordinate (0,0) millimetres in the scanning coordinate system. In order to scan an arbitrary point on the sample with coordinates (x,y) millimetres one should move x and y motor to the position (M_x, M_y) steps

$$\begin{aligned}
 M_x &= A_x x + M_x^{ref} \\
 M_y &= A_y y + M_y^{ref}
 \end{aligned}
 \tag{1.5.2.2.1.1.)}$$

where A_x and A_y are number of steps (encoder counts) per millimeters for x and y motor respectively (see expression 1.5.2.1.1),

The XRF image obtained by the scan is always shown in the Scanning coordinate system. Therefore to move a point shown on the XRF image in the front of capillary, one should apply the same formula 1.5.2.2.1.1. For this purpose you can use MDI ‘Scan... Goto’ command.

1.5.2.3. Optical scan

In Optical scan the area is defined with reference to (a) an optical image or (b) an XRF scanned image obtained from (a). This scan needs optical image which can be obtained by camera coupled to microscope and with using of the ‘Video’ menu item in the MDI application, specifically the ‘Video... Image... Grab’ command. In addition the optical image must be calibrated and related to the beam position, which is explained in the following chapter.

The optical scan is more complicated than blind, because it requires optical calibration.

1.5.2.3.1. Optical image

The program supports two positions of camera relative to the beam, that is, to the capillary: 180^0 - when camera see the back side of sample during scan, and 0^0 - when camera see the front side of sample. In both cases it is assumed that camera plane is parallel to the axis of X and Y actuators.

In order to verify that camera X and Y axes are parallel with actuator’s axes you can use a micrometers cross and lens with lower magnification. Move the center of the cross on such a way that it appears first on the left most part and then to the right most part of the video image. If camera is not rotated you should see the center at the same y coordinate. Therefore rotate camera and repeat above procedure as long as the two coordinates are different. Still the camera plane may be tilted, that is, not perpendicular to the Z actuator axis, which is not so easy to correct.

Before using an optical image to define a scan area, the image must be calibrated. This means that position (x_p, y_p) in pixels on the image vs position (x, y) in millimeters in a plane perpendicular to the beam/parallel to camera, must be established. The (x, y) position refers to scanning coordinate system (see 1.5.2.3.2 and 1.5.2.3.3). In addition the position of capillary on the optical image must be known. The transformation curves for x and y direction are defined by straight lines with offsets b_x and b_y , and slopes (also called resolution) a_x and a_y respectively, as follows

$$\begin{aligned} x &= a_x x_p + b_x \\ y &= a_y y_p + b_y \end{aligned} \tag{1.5.2.3.1.1}$$

The slopes are defined by the microscope/camera system and do not change as long as you keep the same all of the following: video image format, camera type and microscope magnification. The slopes can be easy found by the ‘Video... Image... Calibrate’ procedure in the MDI application. See section 1.2 and table 1.2.1 for particular values of the resolution a_x and a_y . Note that table shows resolution in $\mu\text{m}/\text{pixel}$ while the program uses mm/pixel .

The offsets are arbitrary numbers that define $(0,0)$ position in millimeters on an optical image. As we can grab one image for each motor position, generally we can have as many offsets as there are motor positions. But if we want to show all images in the same coordinate system, than only one offset and one motor position are necessary and sufficient. This coordinate system is called ‘scanning or referent coordinate system’ and x and y motor position is called ‘referent motor position’. The image obtained at referent motor position is called ‘referent image’. This means that:

- calibration offset you define is valid only for referent motor position,
- each time you move motors and grab an image, the program recalculates calibration offset on such a way that any object has equal coordinates on all images
- any fixed object (does not move with motors) that you can see on a grabbed image has valid coordinates only if x and y motors are at referent position.

Therefore, when you define the calibration offset and motor referent position then the position of capillary is unique and each sample point can be positioned in the front of capillary.

The scanning coordinate system and position of capillary (and the axis of rotation if camera is at 180^0) are usually established during a procedure called ‘optical calibration’, that is explained in 1.5.2.3.3.

1.5.2.3.2. Coordinate system for camera at 180^0

The scanning coordinate system (see Fig. 1.5.2.3.2.1.) is established when x and y motors are at fixed, arbitrary position (M_x^{ref}, M_y^{ref}) steps (encoder counts), that is called motor referent position on such a way that $(0,0)$ coordinate in millimeters (origin) is set at arbitrary selected point O on the image/sample. In addition, the image’s resolution in pixels per millimeter should be known.

Before scanning, the sample is first rotated around the axis of rotation x_A millimeters and then x and y motors are moved on such a way that all scan points are positioned, one by one, in the front of capillary, that is, to the coordinate (x_c, y_c) millimeters. In order to scan an arbitrary point with coordinates (x, y) millimeters on the video image one should move x and y motor to the position (M_x, M_y) steps (encoder counts).

$$M_x = +A_x x + [M_x^{ref} + A_x (x_c - 2x_A)]$$

$$(1.5.2.3.2.1) \quad M_y = -A_y y + [M_y^{ref} + A_y y_c]$$

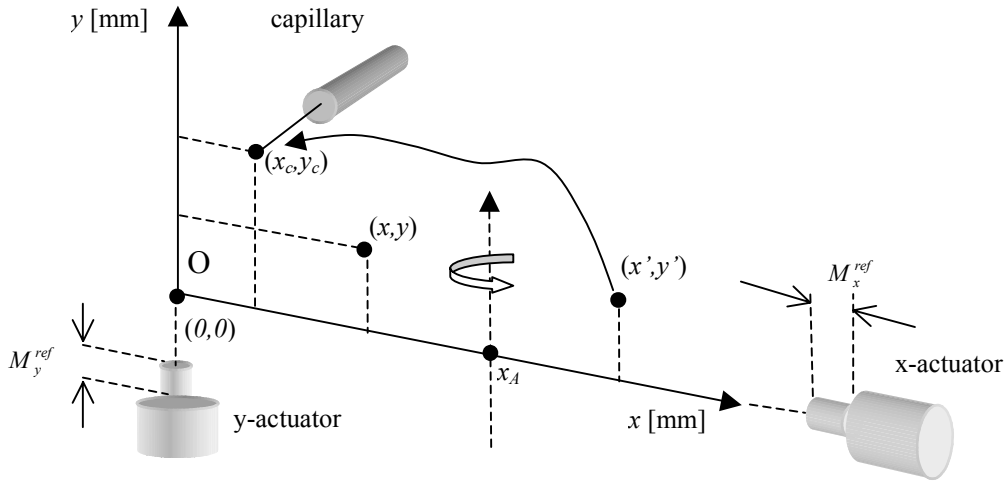


Figure 1.5.2.3.2.1. The Scanning coordinate system for optical scan and camera at 180° .

where A_x and A_y are number of steps (encoder counts) per millimeter for x and y motor respectively (see formula 1.5.2.1.1).

The XRF image obtained by the scan and video image as well are always shown in the Scanning coordinate system. Therefore to move a point shown on these images in the front of capillary, one should apply the same formula 1.5.2.3.2.1. For this purpose you can use MDI ‘Scan... Go to’ command.

1.5.2.3.3. Coordinate system for camera at 0°

The scanning coordinate system is established (see Fig. 1.5.2.3.3.1.) when x and y motors are at fixed, arbitrary position (M_x^{ref}, M_y^{ref}) steps (encoder counts), that is called motor referent position, on such a way that $(0,0)$ coordinate in millimeters (origin) is set at arbitrary selected point O on the video image. In addition, the image’s resolution in pixels per millimeter should be known.

During a scan, x and y motors are moved on such a way that all scan points are placed, one by one, in the front of capillary, that is, to the coordinate (x_c, y_c) millimeters. In order to scan an arbitrary point (x, y) millimeters on the video image one should move x and y motors to the position (M_x, M_y) steps (encoder counts)

$$\begin{aligned}
 M_x &= -A_x x + [M_x^{ref} + A_x x_c] \\
 (1.5.2.3.3.1) \quad M_y &= -A_y y + [M_y^{ref} + A_y y_c]
 \end{aligned}$$

where A_x and A_y are number of steps (encoder counts) per millimeter for x and y motor respectively (see formula 1.5.2.1.1).

The XRF image obtained by the scan and video image as well are always shown in the Scanning coordinate system. Therefore to move a point shown on these images in the front of capillary, one should apply the same formula 1.5.2.3.3.1. For this purpose you can use MDI ‘Scan... Got o’ command.

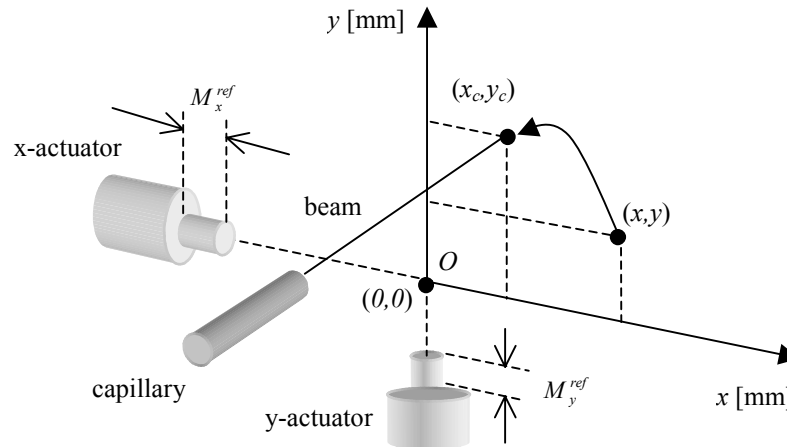


Figure 1.5.2.3.3.1. The Scanning coordinate system for optical scan and camera at 0^0 .

1.5.2.3.4. Calibration

The main purpose of the optical calibration procedure is to establish scanning coordinate system and find there position of capillary and the axis of rotation (if camera is at 180^0).

There are two types of procedures. One of them requires that you can see exit of the capillary, that is, beam position, with using of the microscope and camera. In addition you need a micrometer thick cross. This cross can be mounted on a separate or on the same holder together with sample. In the last case you don't need to remove cross after calibration, therefore any errors related to the holder handling are removed. However, in the case of camera at 180^0 the cross must be visible by camera after 180^0 rotation. For example, you can glue a small cross on a piece of Mylar and then Mylar on a Plexiglas holder together with sample. It is convenient to put cross at left bottom corner of holder. In any case, take care that part of sample you are interested in and cross as well can be moved in the front of capillary that is, all they should be inside the range of motors.

1.5.2.3.4.1. Procedure I

This procedure you can use for both positions of camera, 180^0 and 0^0 as well. It does not require that capillary exit or beam position is visible by the microscope and camera.

You need a micrometer thick cross, which can be mounted on the holder together with a sample or on a separate holder.

□ Step 1: Referent image calibration

Here you need to define offset and slope of the linear transformation between position in millimeters in the scanning coordinate system and pixels on the screen. The slope is defined by the microscope magnification and camera resolution. The offset is arbitrary.

If you didn't change magnification or camera resolution, you can leave existing values and go to the next step. Eventually you can set $(0,0)$ for offset.

If you changed magnification or camera resolution or you want to check the slope then perform the 'Image Calibration' that is described in the 3.8 'MDI...Video... Image... Calibrate'. This procedure is primary used to calculate slope for both x and y-axes. Therefore you should enter new slopes in the Optical dialog. As offset is arbitrary, you can overwrite it with $(0,0)$ values or define it later when you mount the sample. For the moment we don't need the dialog or its property sheet, therefore you can press Ok to exit.

Note that default slope values are defined in 1.2.2 Setup Specification.

□ Step 2: *motors referent position*

Actually, this step is performed together with step three. For the moment just read bellow.

If you would move the center of cross and grab it, you would see that it has the same coordinate on all images. This means that program shows images in coordinates of a system that is always at rest. This is so called scanning or referent coordinate system. Therefore, there is only one position of motors for which grabbed image has valid coordinates for both, objects that are not mounted on the holder (like capillary) and those that are mounted on the holder. In other words, fixed object has valid coordinates on a grabbed image only if x and y motors are at unique position. This is called motors referent position. It is usually $(0,0)$ steps, although it can be any value. In this case, the referent motors position will be defined when cross is just in the front of capillary exit. This is done in the next step; therefore you don't need to enter any new value now.

□ Step 3: *capillary position*

In order to find the capillary exit, that is, beam position we are going to move centre of the cross directly in the front of capillary exit.

To do this we have to find an XRF image of the cross's centre. This problem is already solved in the Example 2 (see 1.5.3.2) without using microscope. In other words, it is not necessary to see capillary by microscope. Therefore you should follow all steps from Example 2 at least to the step seven, and then return here. Note that if your cross is not parallel with x and y axis of actuators than you should perform also step eight to eleven from above example.

In order to move the cross in front of the capillary, you have to use coordinates in millimetres of the cross's centre you have just found and transform them into motors coordinate. This can be done with using of the 'Scan... GoTo' command. It brings up a dialog box that calculates corresponding coordinates of x and y motors in steps (encoder counts). Use the Motor application to move the motors there. After that the centre of the cross will be exactly in the front of the capillary exit

When the cross's centre is in the front of capillary it is recommended to reset x and y motors and then enter $(0,0)$ for the 'referent motor position' in the Optical dialog of the 'Scan... Properties' sheet. If you don't want to reset motors you should read current motor position and enter these values instead of $(0,0)$.

So far we found coordinates of the cross in reference to a coordinate system(s) established in blind scan(s). In order to find position of the capillary in the optical scanning coordinate system, move the motors on such a way that you can see cross's centre on the screen. Then grab image with using of the 'Video... Image... Grab' and read coordinate of the centre in millimetres. Enter these values as the 'capillary position' in the same dialog. Now you can close the dialog and property sheet by pressing Ok button.

1. X RAY MICROPROBE

□ Step 4: *camera position*

In this step select position of camera in the Optical dialog. If it is at 180^0 , you should define axis of rotation, otherwise skip to the step two. In both cases press OK and exit from the dialog.

□ Step 5: *axis of rotation*

This step is required only if the camera is at 180^0 .

In order to define axis of rotation you should remember current coordinate of the cross's centre (actually the position of capillary), rotate it from 0^0 (where it is now) to 180^0 and then grab its image. Note that after rotation you may not see the cross, therefore move it in the view of microscope, grab an image and read position of cross.

The axis of rotation is at half distance between the cross's centre position at 180^0 and 0^0 . Therefore enter half of the sum of above two positions in the Optical dialog of the 'Scan... Properties' sheet.

1.5.2.3.4.2. Procedure II

This procedure is used if you have camera at 180^0 . It requires that you can see the exit of capillary with microscope and camera, that is, to estimate the position of the beam on the video image.

In addition, you need a micrometer thick cross, mounted on a transparent holder or on the same (also transparent) holder together with sample, in order to find axis of rotation and make correction of the beam position as well.

□ Step 1: *microscope positioning*

Microscope must be positioned on such a way that you can see capillary exit using the microscope (for example around the center of video image). Now, take a picture of the capillary by the 'Video... Image... Grab' command and find there its exit (as precisely as possible) in channels (not millimeters). Remember this value.

□ Step 2: *referent image calibration*

If you didn't change microscope magnification and camera format you can skip this step. Eventually if you don't like position of $(0,0)$ point, you can change only offset in the 'Scan Properties... Optical' dialog. For example to put $(0,0)$ coordinate at channel (x_c, y_c) , set offsets to $(x_o, y_o) = (-R_x/x_c, -R_y/y_c)$, where R is resolution of image. The resolution should not be changed.

The calibration can be done by using the 'Video... Image... Calibrate' procedure. The procedure defines origin (offset) and slope (resolution) of the x and y axis calibration lines. The offset defined there you can change if you want. This procedure can be done only once for a particular microscope magnification and camera format.

□ Step 3: *motor referent position*

If you performed calibration in the step two, than x and y motors were reset at custom position. Take this position as referent by entering $(0,0)$ for the '*motor referent position*' in the 'Scan Properties... Optical' dialog box. In addition the cross is at 180^0 .

In the most cases you skip step two, therefore mount the cross on holder or sample together with cross and rotate it for 180^0 to view cross by microscope. Move

motor to a convenient position, for example with cross in the center of display. Now, reset x and y motors (to zero) by using Motor application's system menu command. Then enter (0,0) for the 'motor referent position' in the 'Scan Properties... Optical' dialog box. Note that it was not necessary to reset motors. In that case, you should enter current x and y motor coordinates as the referent motors position.

□ **Step 4: *capillary position***

Grab an image with using of the 'Video... Image ...Grab' command. Note that motors must be at position defined in the step three. Place cursor on the position from the step one, read coordinates in millimeters and enter it as the '*capillary position*' in the 'Scan Properties... Optical' dialog box. Note if your cross is not mounted on the holder together with sample than you can get picture of capillary directly.

□ **Step 5: *axis of rotation***

Grab an image of the cross's center with using of the 'Video... Image... Grab' command and then read position of the center in millimeters. It is recommended to save this picture by 'Video... Image... SaveAs' command. Now, move motors to zero, rotate them for 180^0 (to 0^0), grab image and read the position in millimeters as you did before. The average value of this two readings enter as '*axis of rotation*' in the 'Scan Properties... Optical' dialog box.

After this step you have all necessary information. However, the position of capillary exit or beam may not be very precise established. Making an XRF image of the cross's center can check and correct this. Also, you can make similar correction on the XRF image of the sample. However, it is recommended to proceed to the step six, although not necessary.

□ **Step 6: *correction of the capillary position and axis of rotation***

This is described in six steps that follow bellow. Note, after the step four you have already mounted cross and its image at 180^0 . In addition, the cross is at 0^0 . Therefore you can do there the step three and skip to the step five.

Checking and correcting optical calibration (camera at 180^0)

The error in position of the axis of rotation and capillary as well are the main reasons for missing selected scan area. In the formula which program uses to move the sample in the front of the capillary, the positions **A** and **C** of the axis and capillary respectively, enter always as **2A-C**. The precision of the calibration depends on the overall value and not on the particular terms. To adjust the whole expression we can alter, for example only position of the capillary. For this purpose, we should take XRF image of a suitable object. This can be done by a few- micrometer thickness cross as follows:

□ **Step 1**

Mount the cross and move motors on such a way that cross appears somewhere around the center of the video image

□ **Step 2**

Grab video image with motors at 180^0

1. X RAY MICROPROBE

- Step 3
Select a ROI (for example 300 μm x300 μm) around the cross-center and export it
- Step 4
Rotate the cross for 180⁰
- Step 5
Make a horizontal line scan. It requires defining a map with dimension in y-axis equal to one channel and resolution in x direction comparable to the beam size (for example 1/2 of the beam size). This gives for beam size of 30 μm and 300 μm scan, around 60 channels in x directions. The time preset of 1 second should be enough. The map must correspond to the element of which the cross is built (how to create map refer to the 'MNG... Acquisition... Displays'). After run is finished save data with using the 'MNG... Data SaveAs'.
- Step 6
Make a vertical scan. This is similar to step 5 but with exchanged number of channels in the x and y direction. Save data with using of the 'MNG... Data SaveAs'.

After each of steps 5 and 6, you will get XRF image of the cross as a peak in one-dimensional map. To see the peak, create a full ROI on the map and project it on to the corresponding axis using the 'ROI ...Project' command. If you can't see the peak, this means that you have selected too small ROI comparing to the error in the capillary/axis of rotation position or too small resolution. Enlarge the ROI and number of channels in the map keeping the same or increasing the resolution. Then repeat step 5 and 6 until you can see the cross on the XRF images.

To correct the position of the capillary defined in the 'Scan Properties Optical' dialog box, you should add $x_{\text{xrf}}-x_{\text{vid}}$ to the already defined x position and subtract $y_{\text{xrf}}-y_{\text{vid}}$ from y position. The 'xrf' indicates position of the cross on the XRF-image and 'vid' on the video-image.

The new established position of the capillary corrects the 2A-C term on such a way that no correction of the position of the axis of rotation is necessary.

1.5.3. Examples

Before performing examples listed in this paragraph it is highly recommended to read and repeat 1.5.1.3 *Running program for the first time* and 1.5.1.4 *Measuring for the first time*. Here are shown three examples starting from the scratch: two without optical calibration (so called blind) and one with optical calibration.

The first example shows a typical 'blind' measurement. It is performed in four-step procedure: (1) scan area definition (2) XRF map definitions (3) preset time settings (4) start measurement.

The second example is not typical measurement but it is useful exercise for optical calibration procedure.

The third example shows typical measurement with using optical image to define scan area.

1.5.3.1. XRFM image of an IC chip without optical calibration (blind scan)

Assumption:

- Spector is running in XRFM mode

You can check it in the 'MNG... System... MCA' property sheet by opening 'Application' dialog box. There should be active XRF microprobe. If not, then select it and press OK. If yes, it means that you already have some XRF maps defined and you need to redefine them in the step two.

- detector is calibrated (not necessary)

It is calibrated if single spectra in MDI show both energy and channels. To see spectra apply 'MNG... Data... View' and then 'MDI... Display... OpenAll' command. If detector is not calibrated you can do it as described in 2.3.1.2.1 MNG, Edit XRFM, Detector Calibration.

- IC is mounted on the holder and rotated 90^0 respect to the beam and rotation motor shows 0^0 .

□ Step 1: *Rectangular scan area definition*

Here we will define two-dimensional rectangular scan area. Later, after obtaining the first XRF image, you will be able to define an arbitrary scan area shape that refers to this image.

Move the sample on such a way that left bottom corner of area you are interested in is in the front of capillary. In other words, the area should be in the first quadrant like on the Fig. 1.5.3.2.1 for the cross.

Then open the 'Scan... Properties' dialog box. There, select for the scan path '*vertical with jump*' or '*horizontal with jump*' option. These are two preferred ways of scanning because they take into account backlash. The backlash is important if scan step-size is small, that is, typically less than $50\mu\text{m}$. The '*scan speed*' can be set to maximal (10 units). The '*zoom*' should be checked. Note that gain, torque and similar motor related parameters are defined by the Motor program. Normally, you don't need to change default values unless there are some problems with moving motors. Upon pressing Ok button, the main scan properties are set.

After setting the scan properties, apply the 'Scan... Export... Rectangle' command. It brings up a property sheet with two dialog boxes: Size and Position.

In the Position dialog, select the '*origin in the front of capillary when motors are at position*' button and press the '*read motor*' button. The program will read current motor coordinates and show it in the corresponding edit controls. Then set $(0,0)$ for (dx,dy) shift, and check the '*Set $(x+dx,y+dy)$ as a new referent position*' button.

In the 'Size' dialog box enter size of the scan rectangle area you want to scan. Finally press Ok button. Now you have well defined rectangular scan area of size $x \times y$ that includes sample area you are interested in. Actually, the program creates scan.dat file in the `..\scan` subdirectory of the Spector's root directory. In addition, it creates scanxxx.log file which you can later edit with the 'Scan View' or 'Scan... History' and eventually modify.

□ Step 2: *XRF map definition*

In this step you should define maps that the program collects data into. Each map is defined by its size (in pixels) and additional information like ROI used by

analysis routine. Generally, each map corresponds to one element you want to analyse, that is, its characteristic X ray line. The analysis used in this example assumes that you have defined one ROI for each element. The ROI should be defined around the position of the corresponding characteristic X ray line.

To define ROI you should first open existing cumulative spectrum with using 'MDI... Display... OpenAll' command. To run MDI use 'MNG... Data... View' command. In the cumulative spectrum you can find already defined ROIs. You can delete all of them. Then create ROIs for all lines you are interested in. In this case it can be only Ni-L_α line.

In order to define map(s), you should open the 'Acquire... Display' property sheet. There, you will find previous maps. Some of them may be without any ROIs for example if you have created less ROIs than it was originally. Delete all maps and create one new for Ni- L_α.

Note that you could also first open the 'Acquire Display' property sheet, press the 'set default' button, open cumulative spectrum in MDI, redefine existing ROI/add new ROIs, and then go back to the same property sheet in order to define new maps.

The third approach is to define everything, including ROIs, from the 'Acquire Display' property sheet.

□ Step 3: *Preset time setting*

To define preset time use the 'MNG... Acquire... MCA' property sheet.

□ Step 4: *Starting scan*

Run measurement by pressing the 'Start' button in MNG client area.

After the run is finished you will get XRF map similar to the Fig. 1.5.3.3.1. From this map you can define scan area of any shape simple by creating a ROI there, and then applying the 'Scan... Export... Image ROI' command. If you want to define area that is partially or totally out of the collected XRF image then use Expanded XRF map. Finally, to define a rectangle area use 'Scan... Export Rectangle Area'.

1.5.3.2. XRFM Image of 4μm Tungsten cross without optical calibration (blind scan)

Assumption:

- Spector is running in the XRFM mode

You can check it in the 'MNG... System... MCA' property sheet by opening 'Application' dialog box. There should be active XRF microprobe. If not, then select it and press OK. If yes, it means that you already have some XRF maps defined and you need to redefine them in the step two.

- detector is calibrated

It is calibrated if single spectra in MDI show both energy and channels. To see spectra apply 'MNG... Data... View' and then 'MDI... Display... OpenAll' command. If detector is not calibrated you can do it as described in 2.3.1.2.1 MNG, Edit XRFM, Detector Calibration.

- cross is mounted on the holder and rotated 90⁰ respect to the beam and rotation motor shows 0⁰.

□ Step 1: *Scan area definition for horizontal and vertical scan*

Here we will define two-dimensional rectangular area in order to make both vertical and horizontal scan. Later, we will make it one-dimensional by setting map dimension to one pixel in direction we don't need to scan. In this case the program ignores scan size defined in the direction for which map dimension is one pixel.

First, in order to define the scan area, you should move the cross on such a way that its center is somewhere in the first quadrant of an imaginary coordinate system which center is in the exit of capillary, as it is shown on the Fig. 1.5.3.2.1. The exact position is not important and distance from the center of the cross to the capillary can be any. It is recommended to put the center as close to capillary as you can, but still not moving it outside of the first quadrant.

Then open the 'Scan... Properties' dialog box. There, select for the scan path 'vertical with jump' or 'horizontal with jump' option. These are two preferred ways of scanning because they take into account backlash. The backlash is important if scan step-size is small, that is typically less than $50\mu\text{m}$. The 'scan speed' can be set to maximal (10 units). The 'zoom' should be checked. Note that gain, torque and similar motor related parameters are defined by the Motor program. Normally, you don't need to change default values unless there are some problems with moving motors. Upon pressing Ok button, the main scan properties are set.

After setting the scan properties, apply the 'Scan... Export... Rectangle' command. It brings up a property sheet with two dialog boxes: Size and Position.

In the Position dialog box check 'in the front of capillary when motors are at position' button and press 'read motor' button. The program will read current motor

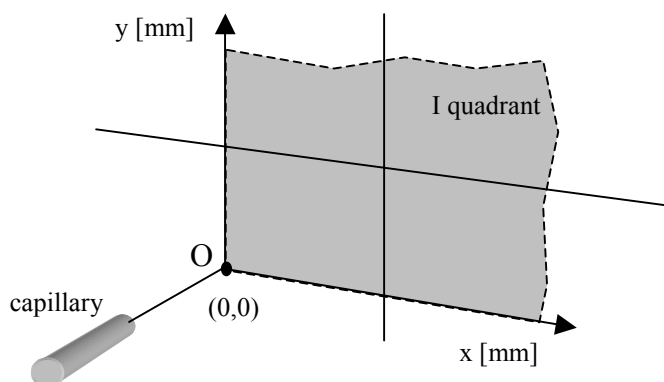


Figure 1.5.3.2.1. Properly positioned cross for vertical and horizontal scanning.

coordinates, showing them in the corresponding edit controls. Then set $(0,0)$ for (dx,dy) shift and check 'Set $(x+dx,y+dx)$ as a new referent position' button.

In the Size dialog box enter size of the scan rectangle area. The x and y size should be a little bit larger than x and y distance from center of the cross to the exit of capillary. Finally press Ok button.

Now, you have well defined rectangular scan area of size $x \times y$ that includes the center of your cross. Actually, the program creates scan.dat file in the $\backslash\text{scan}$ subdirectory of the Spector's root directory. In addition it creates scanxxx.log file, which you can later edit with 'Scan View' or 'Scan... History' and eventually modify.

1. X RAY MICROPROBE

□ Step 2: *XRF map definition for horizontal scan*

In the third step you should define map the program will collect data into. Each map is defined by its size (in pixels) and additional information used by analysis routine, like ROI. Generally, each map corresponds to one element you want to analyses, that is, its characteristic X ray line. The analysis used in this example assumes that you have defined one ROI for each element. Each ROI should be defined around the position of the corresponding characteristic X ray line.

To do this, apply the 'Acquire... Display' command. It opens the 'XRFM Displays Properties' dialog box. There, press the '*set all default*' button. It creates a map that uses a simple ROI based elemental analysis. In addition, two single spectra are created: one that shows cumulative and other that shows 'per pixel' (transient) spectrum. To set resolution of your map in pixels, press the '*map resolution*' button. It brings up dialog that accepts x and y resolution. In order to find vertical wire, enter for the x direction a number equal to the x-size of the scan rectangle from the step one, divided by FWHM of your beam. For the y resolution set one pixel.

The default ROI covers full range of spectra. In order to define a proper ROI, first apply the 'MNG... Data... View' command which starts/activates MDI, if it is not already started, and then 'MDI... Display... Open All' command. It brings up two single displays and one map. It was assumed that detector was already calibrated. Therefore, you can modify ROI created by the program when you applied '*set all default*' command. To do this, create region by dragging right mouse button around the position of one of W characteristic X ray lines and then apply the 'ROI replace' command. Note that if your detector is not calibrated you can leave ROI as it is, make calibration during a scan and then redefine ROI.

□ Step 3: *Preset time setting for horizontal scan*

To define preset time use the 'MNG... Acquire... MCA' property sheet. For example one-second Live Time preset should be enough to observe the wire.

□ Step 4: *Starting horizontal scan*

Run measurement by pressing the 'Start' button in the MNG client area.

During acquisition you can monitor map by making projection onto the X-axis. To do this, select full region on the map, create ROI and then apply the "MDI... ROI... Project X' command. It creates one-dimensional display, which changes from the left to the right part of the spectrum.

If the scan size was chosen properly, you should get a peak in the projection spectrum, which comes from the wire. If not, it means either the scan size is too small or the wire does not cross first quadrant. Correct it and run again. Remember position of the peak. Assuming the vertical wire is parallel to the y-axis, the peak shows x-coordinate of the cross's center.

□ Step 5: *XRF map definition for vertical scan*

In this step you should adapt already defined map for making scan in the vertical direction in order to find position of the horizontal wire of the cross.

1. X RAY MICROPROBE

To do this you should define new x and y map resolution similar to the step two. This time, enter resolution in the x direction equal one pixel. For resolution in the y direction make calculation like in the step two for the x direction.

- Step 6: *Preset time setting for horizontal scan*

You can keep the same preset time used in the step three.

- Step 7: *Starting vertical scan*

This is similar like the step four. Run measurement by pressing the 'Start' button in MNG client area.

During acquisition you can monitor map by making a projection onto the y-axis. To do this, select full region on the map, create ROI and apply "MDI... ROI... Project Y' command. It creates one-dimensional display, which changes from right to left part of the spectrum.

If scan size was chosen properly, you should get peak at position of cross in the projection spectrum. Remember this position. Assuming the horizontal wire is parallel to the x-axis, the peak shows y-coordinate of the cross's center.

- Step 8: *Scan area definition for two-dimensional scan around the center of the cross*

In this step you will make a rectangular scan to obtain a two-dimensional XRF image of the cross around its center. Therefore you should define new rectangular scan area with proper position and size. The size is up to you. The position should be related to the values you found in the step four and seven. You can define the position on few different ways.

First way to define a scan area, is to use the 'Scan... Export... Rectangle' dialog box on the same way as above. Before that you should move motors on such a way that center of the cross is in the front of capillary exit/beam. To do this, apply 'Scan... Goto' command. It brings up corresponding dialog box. There, enter position of the cross's center in millimeters that you found in the steps four and seven. Then press the 'Calculate' button. It gives coordinate of motors in steps when cross is in the front of capillary. Now use the motor.exe to move motors there. Once the motors are at proper position, you can apply the 'Scan... Export... Rectangle' command. In the 'Position dialog' box check '*in the front of capillary when motors are at position*' button and press the 'Read motors' button. For the two required offsets, set half of the values you are going to select for the size, and both with negative sign. If you want to set zero in the left bottom corner of the map, then check the '*Set (x+dx,y+dx) as a new referent position*' button. In the 'Size' dialog box of the same property sheet set size (for example 100 μ m x 100 μ m). Apply Ok to export the scan area you've just defined.

1. X RAY MICROPROBE

Second way to define scan area is by using of the Extended XRF map. You can access it by the 'Scan Extend' map. Before that you should define its properties by the 'Extend Map' dialog in the 'MDI Scan Properties' property sheet. There, choose the size at least equal to the size of the current scan area. Then apply the 'Scan... Extend' command which brings map with picture of your last scan. In this case, it is the vertical line obtained after the step seven. On the map you can find coordinate of the cross. Select a rectangle region around it and apply the 'Scan... Export... Image... ROI' command. You can also put cursor at left bottom corner of the region and use the 'Scan... Export... Rectangle' dialog box to define rectangle relative to this point.

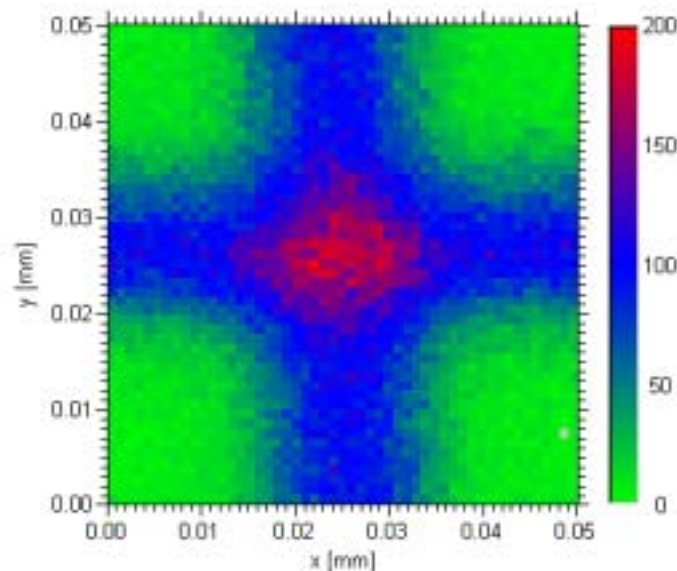


Figure 1.5.3.1.2. XRF image of 4 μm thick Wolfram cross: capillary exit diameter cca 12 μm , step size 1 μm , preset 5 sec per pixel.

In this case you should check in the 'Position' dialog box a button labeled 'origin at point (x,y) of active image' and press 'read cursor'. Set (0,0) for the (dx,dy) shift. The button 'set (x+dx, y+dy) as new referent position' has no effect in this case. In the size dialog box define the size of rectangle. Comparing to the graphical definition of ROI, this approach has advantage that it allows you to define exact size of the scan area in millimeters. On the other size, by the graphical method you can define an arbitrary shape.

□ Step 9: XRF map definition for two-dimensional scan

This can be done like in the step two or step five. Actually, the number of pixels in the map defines the step size of the scan in millimeters. Therefore, you should again use the 'Map Resolution' dialog box in the 'MNG... Acquire... Displays' property sheet. For example if you want step size of 5 μm along the full scan range of 100 μm in x or y direction, than you should set map resolution to 21 pixels in the same direction.

□ Step 10: Preset time setting for two-dimensional scan

In this step you can redefine preset time selected in the step six. The preset time is always compromise between collecting spectrum that contains a good statistics, and measuring for a short time.

□ Step 11: *Starting two-dimensional scan*

Run measurement by pressing the 'Start' button in the MNG client area. The result is shown on the Fig. 1.5.3.1.2.

1.5.3.3. XRFM image of an IC chip with optical calibration

Assumption:

- Spector is running in XRFM mode

You can check it in the 'MNG... System... MCA' property sheet by opening Application dialog box. There should be active XRF microprobe. If not, then select it and press OK. If yes, it means that you already have some XRF maps defined and you will need to redefine them.

- detector is calibrated

It is calibrated if single spectra in MDI show both energy and channels. To see spectra apply 'MNG... Data... View' and then 'MDI... Display... OpenAll' command. If detector is not calibrated than you can do it as described in the 2.3.1.2.1 MNG, Edit XRFM, Detector Calibration.

□ Step 1: *Optical calibration*

In this step you will establish the scanning coordinate system and find there position of capillary, and axis of rotation (important only if your camera is at 180⁰) All you need to define is contained in the Optical dialog box of the 'Scan... Properties' property sheet. This step is described in the Optical scan section of this paragraph. For example the Procedure I is the most general and does not depend on the camera position.

□ Step 2: *scan area definition*

Now, move all motors to zero position including rotation, mount the sample, and rotate it by 180⁰ and then translate on such a way that area you are interested in is in the view of microscope, grab its image by 'Video... Image... Grab' and save it. Note that you can grab more images, one for each part of sample and then later you don't need to rotate motors again.

Then open the 'Scan... Properties' dialog box. There, select for scan path '*vertical with jump*' or '*horizontal with jump*' option. These are preferred ways of scanning because they take into account backlash. The backlash is important if scan step-size is small, that is typically less than 50µm. The '*scan speed*' can be set to maximal (10 units). The '*zoom*' should be checked. Note that gain, torque and similar motor related parameters are defined by the Motor program. Normally, you don't need to change default values unless there are some problems with moving motors. Upon pressing Ok button, the main scan properties are set.

To define a scan area mark a region on the grabbed image, create ROI and then use 'Scan... Export... Image ROI'.

□ Step 3: *XRF map definition*

See Example I step two.

1. X RAY MICROPROBE

- Step 4: *Preset time setting*

See Example I step three.

- Step 5: *Starting scan*

See Example I step four.

The result is shown on the Fig. 1.5.3.3.1. You can eventually observed a shift of the XRF image comparing to the video image. For example an object that is shown on the video image at coordinate (x_{vid}, y_{vid}) can be on the XRF image at coordinate (x_{xrf}, y_{xrf}) . To correct it you should change position of the capillary defined in the 'Scan Properties Optical' dialog box as follows: add $x_{xrf} - x_{vid}$ to the x and subtract $y_{xrf} - y_{vid}$ from y position of capillary.

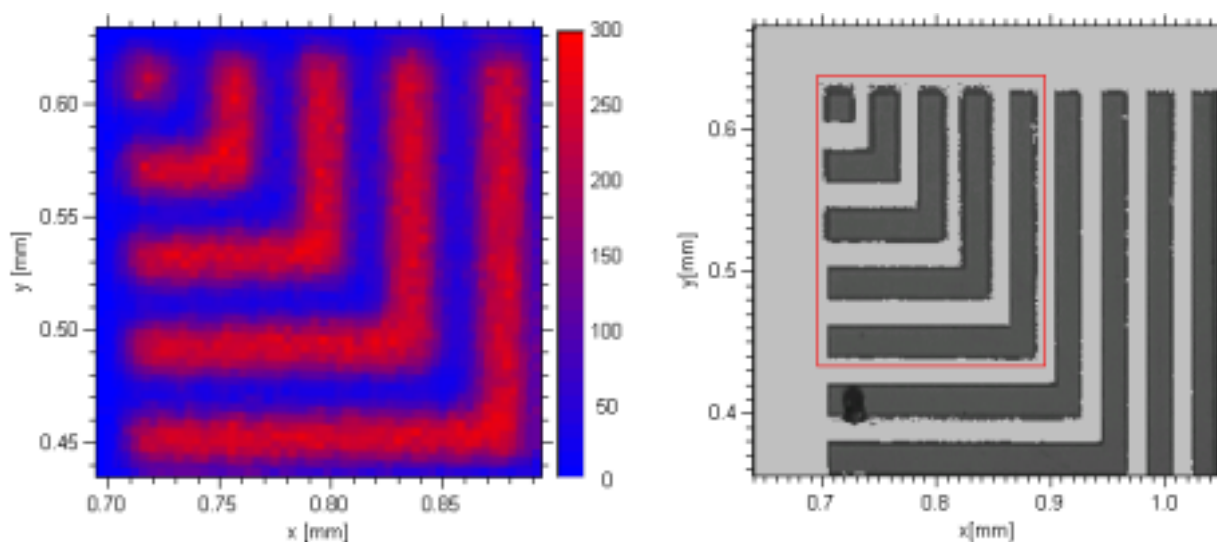


Figure 1.5.3.3.1. XRF image (left) of an IC chip shown on the video image (right) region indicated by rectangle: size $200\mu\text{m} \times 200\mu\text{m}$, step size $4\mu\text{m}$, capillary exit cca $12\mu\text{m}$, preset time 5 sec.

1.5.4. Tips

Scan with using optical image

- **Camera position:** be sure that (x,y) axes of camera are parallel with (x,y) actuators axes. To do this grab two images of the cross's center by using lens with lower magnification: first at left most and then at right most visible position of cross. In both cases the center must be at same vertical position. The tilt (z axis of the camera) can't be adjusted.
- **Beam direction:** if beam is not parallel with trail an additional error in position may be introduced (for example to rotate sample you need to move the whole stages from capillary and after taking image to bring it back – all this procedure may change

distance from capillary to sample). To check the direction very precisely you can use micrometers cross and find position of its center as a function of distance from capillary. To do this you can make ‘blind’ scan around the center of the cross for each distance from capillary.

- ❑ **Sample positioning:** in order to grab image one should rotate your sample (camera at 180^0) and therefore move the whole stages on ‘secure’ distance from capillary. When moving stages back one should reproduce the stages position. To do this it is convenient to fix a ‘liminator’ on the trail between stages and capillary on such a way that stages can’t be moved from measurement position in the direction to the capillary. This also saves capillary from damaging if stages were not fixed by corresponding screw.
- ❑ **Sample rotation:** always move motors to (x,y) referent position (which is usually 0,0) before rotating sample for 180^0 . It minimizes errors in position when scanning an area far from microscope when motors are in referent position.
- ❑ **Looking for capillary position with using cross:** check that cross wires are parallel with camera and actuator axes. If not you should make 2-dim scan rather than a horizontal and vertical scan only (in order to get capillary position). The checking may be done on the same way as verifying camera position, but this time the whole horizontal wire should not change its vertical position.
- ❑ **Grabbing image:** grab images on such a way that microscope and stages are always approximately at the same position on the trail. The trail surface is rough and differences as large as $50\ \mu\text{m}$ in y axis may be observed at different places on the trail
- ❑ **Grabbing image:** once you have sample in the front of camera it is convenient to grab several images of interesting areas on the sample. This may save time especially if camera is at 180^0 (requires sample rotation) or far from capillary.
- ❑ **Microscope focusing:** if turned too much, the screw for rough focusing may move microscope up or down relative to the sample. The trail and microscope platform may not be parallel due to roughness of trail. The errors as much as $50\ \mu\text{m}$ were observed.
- ❑ **Correction of the optical calibration:** if an XRF image, collected with using of a scan area defined on a video image, is shifted comparing to the video image you can correct current position of capillary (x_c, y_c) without additional measurement. Look for a point that you can recognize on both images (for example a corner or center of an object). If the point has (x_{XRF}, y_{XRF}) coordinates on the XRF image and (x_{Vid}, y_{Vid}) on the video image than corrected capillary position (x_c^{new}, y_c^{new}) is given by the following relation

$$x_c^{new} = x_c + (x_{XRF} - x_{Vid})$$

$$y_c^{new} = y_c - (y_{XRF} - y_{Vid})$$

Enter corrected values in the Scan... Properties... Optical dialog box instead of already existing values.

Scan general

- ❑ **Backlash:** if precision is critical then use vertical or horizontal scan path with ‘jump’. It minimizes backlash. In addition enable ‘go there with backlash correction’ feature of the Motor application (see application’s system menu and ‘Application Properties’ item). Then apply ‘go there’ command after each manual positioning done by the ‘move’ buttons.

1. X RAY MICROPROBE

- ❑ **Motor limits:** never go with motors in the extreme position (bellow 0 mm marker and above 25mm marker). The motors may have no limit switches and in the worst case it may block motors (only solution is to dismount motor). In the 'better' case, your optical calibration may be lost due to inelastic forces in the stages.
- ❑ **Line scan:** in order to make line scan use Polygonal marker, create 'line' ROI and export it with 'Scan Export Image ROI'. If the line is horizontal then define map size in vertical direction 1pixel. Similar for vertical line use 1pixel map size in horizontal direction. You can also define line scan from the 'Scan Export Rectangle Area' dialog box. In this case it must be either horizontal or vertical. If the line is horizontal then vertical map size must be 1 pixel and vertical scan area size in millimeters is ignored. Similar for horizontal line.
- ❑ **Point scan:** in order to perform scan at particular point of an XRF image do the following: transform the point coordinate shown on the image to motor coordinates with using 'Scan... Goto' and move motors to the calculated coordinates (in encoder counts) by the Motor application. This causes that selected point is moved in the front of capillary. Save your data with 'Data Save' command and remove all maps by the 'Acquire ... Display... Edit... Delete All Maps' command. Finally start measurement. Note that you have to stop measurement manually. Another approach which does not need to delete maps is to create a small ROI around the point, export it and define map size 1pixel x 1pixel.
- ❑ **Multi pattern scan area:** you can define several independent scan area patterns, but they can be only polygonal (including rectangle). To do this use Polygonal marker when creating scan ROI: for each corner of the first pattern click on the right button. When the last corner is selected move cursor on the first corner of the next pattern and press 'Tab' key. In order to finish (at last corner of the last pattern) press 'End' or 'Esc' key.

Measurement general

- ❑ **Changing preset while measurement is on:** Sorry, it is not possible.

2. MNG

The MNG application is the main acquisition program. It prepares the system for running, starts the measurement, controls the acquisition and finally stops it and saves collected data.

However, the MNG is not intended for a visual data presentation. Moreover, it does not feature any graphics capability. Therefore, you should use the MDI application to visualize on-line data.

To perform the measurement, you should first select the data acquisition board, also called MCA. Usually, it is done during the program installation. However, you can select a new MCA without running the installation program again. To do this, use the 'Select MCA' command in the 'System' menu.

After selecting a MCA, you should prepare it for measurement. Actually, the installation procedure performs the basic initialization. If necessary, you can modify it by the 'MCA' command in the 'System' menu. For example, there you can change the resolution of your ADCs.

There are MCA properties that are often modified like the time pre-sets, coincidence-time window etc. To change it, use the 'MCA' command in the 'Acquire' menu.

If you run the program in the list mode, then you should define the on-line data sorting, that is histograms. To do this, use the 'Display' command in the 'Acquire' menu.

To start the measurement, press the 'Start' button in the application's client window.

Usually, the collected data are automatically saved on the end of the measurement. To customize this and similar data output features use the 'Data' item of the 'Acquire' menu.

All the basic MNG commands are grouped in the following three menus: Data, System and Acquire. Each of them is explained separately as follows.

2.1. Data menu

The Data Menu contains commands that enable reading or collecting the data. In addition, you can combine the data from two or more measurements. The program uses two types of data files: histogram and list files. The both of them reproduce histograms that are originally measured. In addition, you can create new histograms from the list files. A histogram file contains measured spectra, that is, already sorted pulse heights. Usually, it is a binary file with extension 'dat', but also may be text file. The histogram file can be directly loaded into the memory using the 'Data Load' command.

A list file contains raw pulse heights in the same order as they were acquired by the system. By default, it is a binary file with extension 'lst', but you can convert them into the text file as well. One can't read list files with using of the 'Data Load' command. The reading of the list files, also called replay, is similar to the live acquisition. First you should switch the program into the replay mode by the 'Data Replay' and then apply the 'Start' command. The replay reproduces the original histograms or builds a new one.

2.1.1. Load

The command reads all histograms from a histogram file and overwrites already loaded histograms.

First, the Windows 'Open' dialog box is brought up. It contains a list of all files located in the default data directory. Only files with extension and format selected by the '*Files of type*' are shown.

To change the default directory use the 'Directories' dialog box. It is available by the 'Host' command in the 'System menu.

To view the data use the 'View' command in the 'Data' menu. It brings up the MDI application. However, you can start the application as is done for normal applications.

To read a list file, first check the 'Replay' item in the 'Data' menu and then apply the 'Start' button.

2.1.2. Save as

The command saves all histograms into a histogram file with name, location and format specified by the command.

First, the Windows 'Save As' dialog box is brought up. There you should enter the file name and select location and data format. If the file name does not contain extension, the program automatically adds it according to the selected format.

2.1.3. Save

The command saves all histograms into an existing histogram file with already defined file name, location and format.

The file attributes are the same that was used to load the data or start acquisition. The file name is shown on the MDI title bar. If this name is labelled as the 'noname' the command brings up the Windows 'Save As' dialog-box.

2.1.4. Add

The command reads all histograms from a file and adds it to the existing histograms.

It is assumed that both sets of histograms are compatible. It means equal number of histograms and the same resolution of the corresponding histograms.

See the 'Load...' command for an input file selection.

2.1.5. Subtract

The command reads all histograms from a histogram file and subtracts it from existing histograms.

It is assumed that both sets of the histograms are compatible. It means equal number of histograms and the same resolution of corresponding histograms. The negative results are set to zero.

See the 'Load...' command for an input file selection.

2.1.6. Compose

The command performs custom arithmetic operation between two histogram files called operands.

First, the 'Compose' dialog-box is brought up. It lets you select the operation type and operands.

The operands are defined by the fields labeled with the '*Data sources*'. If any of the two operands is located on the disk, enter its file name or apply the '*Browse...*' button to look for it. If the operand is already loaded into the memory press the '*Memory*' button or enter '*NULL*' in the corresponding data source field.

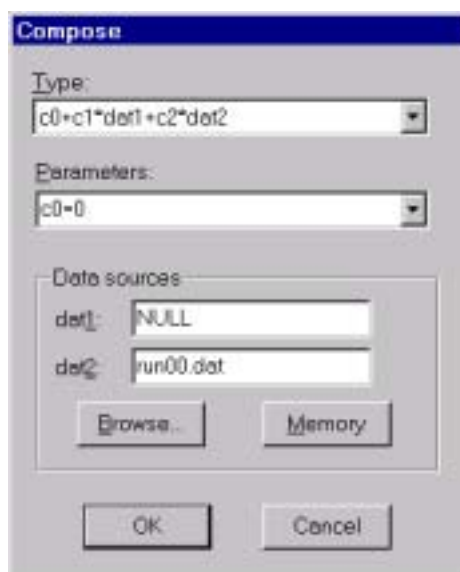


Figure 2.1.6.1 The 'Compose' dialog box.

The type of the operation is selected from the 'Type' list and its parameters by the 'Parameter' list.

The results of the operation are always rounded to the integers and negative values are set to zero.

2.1.7. Replay

The command enables to read a list file.

To start the reading you should apply the 'Start' command. It brings up the 'Start Replay' dialog box that enables you to select a file name and replaying attributes.

Note that replay is possible only if the program is running in the list mode and the 'Replay' item in the 'Data' menu is selected. Otherwise, the 'Start' command brings up a usual start dialog box.

To select the file name use the 'File name' field or apply the 'Browse' button.

Any list file contains event-by-event data. Therefore you can use it to build new histograms, different from the originals. To define new displays use the 'Displays' command in the 'Acquire' menu. After that start the replay and check the 'Modified' box. Otherwise, the program makes replay based on the original configuration stored at the beginning of each list file.

The new replayed data overwrites existing data in the memory unless the 'Append' box is checked. In this case, you can start a new replay for each file and accumulate overall histograms.

The replay starts from the first available event, which is located at the beginning of the data section. It proceeds in a successive order and terminates with the last event at the end of the file. To change the default start and end positions, edit the 'Event from' and the 'Event to' entries, respectively. For the XRFM application, event is related to a point on the sample and each event contains a full spectrum collected at this point.



Figure 2.1.7.1. The 'Start Replay' dialog box.

The input list file is assumed binary unless you check the 'Ascii input' box. By definition, the 'Ascii' list file is n-column text file where column n contains channel of ADCn. It contains no display information, therefore it is always replayed using the current defined histograms.

If you want to transform your original binary list file into the n-column text file, then check the 'Ascii output' box.

The 'Scan' option is available only for the XRFM or SPM application. If it is checked, the program skips events that are not contained in the scan region. To define scan region, use 'MDI Utility Scan' for the SPM or 'MDI Scan Export' command for the XRFM application.

The 'Stabilize' option is used to make correction due to the ADC drift during data acquisition. It is not available in the XRFM and SPM application.

2.1.8. View

The command starts MDI application (binary mdi.exe).

The MDI application is used for graphical presentation of your data. In addition, the MDI features data analysis. For more about the application see chapter three, 'MDI'.

2.2. System menu

The System menu lets you change the system properties that are not frequently modified.

The both MCA and host are properly initialized during the program installation. However, there are some properties, which you may need to change from time to time like the resolution of the ADC or MCA application. In addition, you can use your MCA in different modes of operations. All these properties are listed in the common property sheet called 'MCA'.

Similar to the MCA, you can define the program properties related to the Host, like process and thread priorities, default program directories etc. For this purpose the 'Host' property sheet is used.

In addition, the system menu features the 'Select MCA' dialog box used to select a new MCA.

2.2.1. Select MCA

The command installs selected MCA.

Usually, the MCA is selected and installed during the program installation. Using this command, you can do it later without running the installation wizard again. However, the command uses default parameters whenever it is possible. The parameters are taken from the corresponding '.inf' file in the '..\bin' subdirectory of the application's root directory. Therefore, to change any default value, use the 'MCA' command in the 'System' menu, after selecting the MCA.

2.2.2. MCA

The MCA command lets you modify the basic properties of the MCA you are using or you already used to collect the data.

The command brings a property sheet that shows MCA properties in the three pages that are common for all MCAs: MCA, ADC and Application.

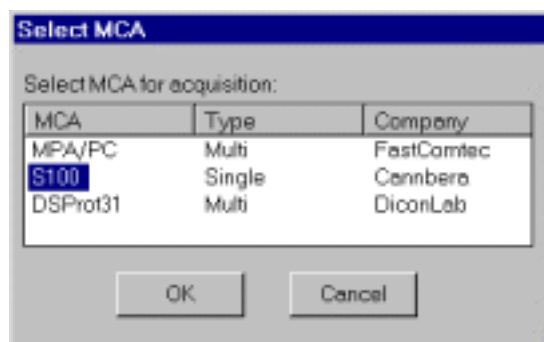


Fig 2.2.1.1. The 'Select MCA' dialog box.

Usually the MCA you have selected and installed matches the loaded data. This means that data were collected on the same type of the MCA that you installed. However, you can load a new data that were not collected on this MCA. In this case, the properties of the MCA that is compatible to the data are shown and acquisition is disabled. The properties are taken from a corresponding '.inf' file.

2.2.2.1. MCA

The MCA page lists basic properties of the MCA hardware, not including the ADCs.

A MCA may support one or more ADCs. A single-parameter MCA refers to one ADC and multi-parameter to a group of ADCs. This feature is indicated by the 'Connected ADCs' item.

The 'Data storage' mode features the MCA capability to process data. It may be the PHA or List mode.

In the simple PHA mode, for each ADC a simple histogram is built, that is, the frequency of digitized amplitude versus the digitized amplitude. If the mode is not

listed, it is assumed the MCA operates in the simple PHA mode. Otherwise, the PHA mode may be more versatile and specified by additional properties.

In the *List* mode, the board stores converted data to an internal buffer in a successive order. The buffer is transferred to the host in the real-time.

The capability to handle coincidence between ADCs is indicated by the entry '*Coincidence*'. It can be turned on or off. Usually, this property is specified with the additional items listed by the 'MCA' command in the 'Acquire' menu.

2.2.2.1.1. S100

The Canberra System S100 MCA is a PC-board that multiplexes up to four ADCs. The current program version supports only the first channel, that is, one ADC in the full resolution.

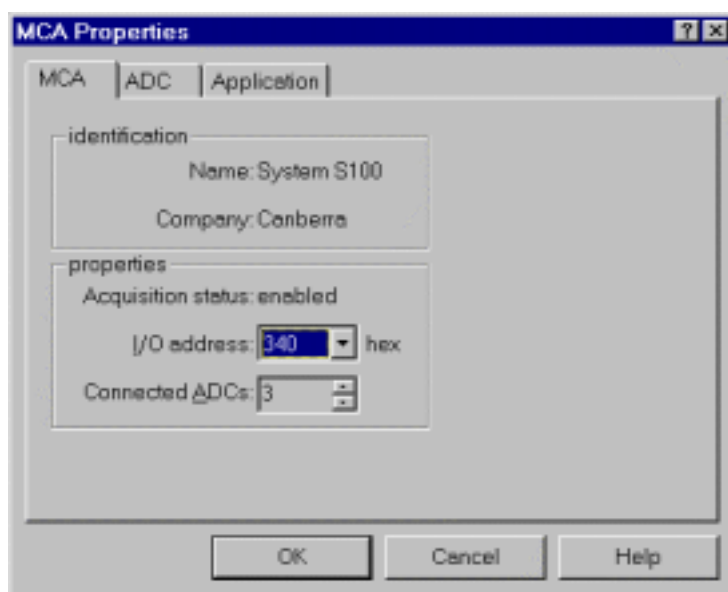


Figure 2.2.2.1.1.1. The 'MCA' dialog box for the 'S100' MCA in the XRFM application.

Therefore, the value for the '*Connected ADCs*' item should be set to one, except for the XRFM application when it should be three. In the last case the first two ADCs are treated as a virtual and they are assigned to the x and y position of the sample.

The MCA uses I/O ports to communicate with the host. Therefore, the '*I/O address*' item should correspond to the on-board jumper settings as indicated by the table 2.2.2.1.1.1. Note that the slave boards are not supported, that is, the board must be Model 4610 Master board.

The MCA has no on-board ADC. If you use Canberra 870x ADC, then set the Gain and Range equal to the '*Resolution*' and offset to zero.

Table 2.2.2.1.1.1. The I/O port base addresses for the Canberra System S100

MCA	Jumpers			Base I/O Address(hex)
	W6	W8	W9	
#1	2-3	2-3	2-3/1-2	340/300
#2	2-3	1-2	2-3/1-2	348/308
#3	1-2	2-3	2-3/1-2	350/310
#4	1-2	2-3	2-3	358

2.2.2.1.2. MPA/PC

The Fast ComTec MPA/PC MCA is a multi-parameter PC-board that features the list mode and the coincidence.

It supports up to eight ADCs in a LBB package or three ADCs in a SBB package. In the first case, the ‘*Connected ADCs*’ item must be set to the number of interfaces in the bus box. In addition, the jumpers in the ADC interfaces must be properly set.

To measure ADC dead time, the MCA uses internal clocks, two in the PHA and one in the list mode. Using the ‘*DtClk*’ column you assign whether the system measures dead time for particular ADC or not and which clock to use. A clock counts an overall dead time for all ADCs, if more than two ADCs are assigned to it.

If the coincidence feature is enabled, you should select coincidence time GAL-code ‘*Gal-CT*’. This must correspond to the GAL chip. The chips are placed on the first card in the bus box, which connects ADC interfaces with your MPA/PC card. Once you have selected the GAL-code, you can set the coincidence time window. To do this, use the ‘Coincidence’ dialog.

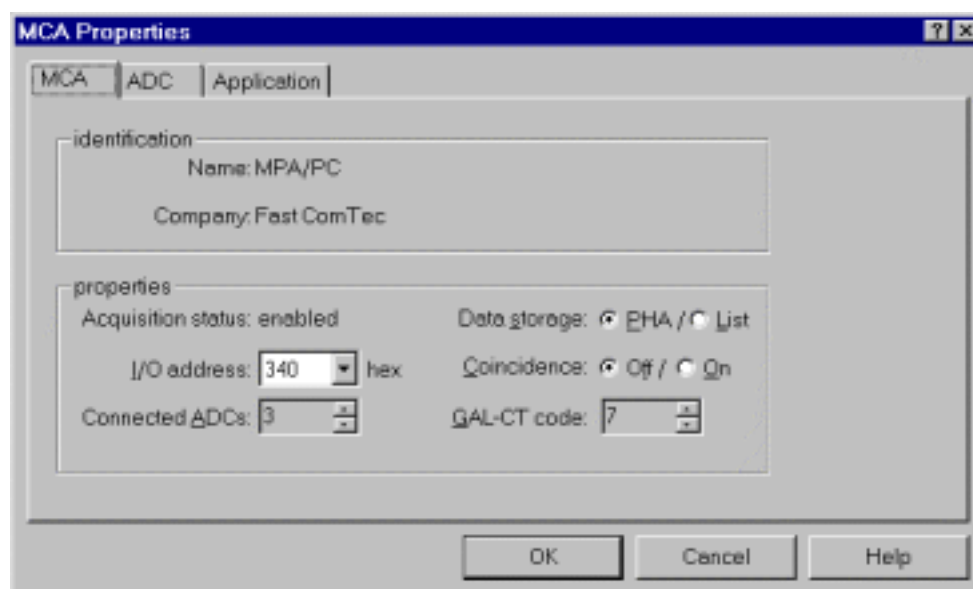


Figure 2.2.2.1.2.1. The ‘MCA’ dialog box for the ‘MPA/PC’ MCA.

2.2.2.1.3. DSProt31

The DiconLab DSProt31 is a multi-parameter board that features list mode and coincidence.

The MCA uses I/O ports for communication with the host. The ‘*I/O address*’ item should correspond to the on-board jumper labeled with the ‘*Address select*’.

The ‘*Timer#0 period*’ and the ‘*Timer#1 period*’ items sets period for the TMS320C31 32-bit timers to generate TINT0 and TINT1 timer interrupts respectively. The first timer, Timer0 is used to measure the Real time and Live time for each ADC. The second timer, Timer1 measures coincidence window. The ‘*Timer#0 period*’ item defines the resolution of the Real time. The value of order of milliseconds or greater is considered as acceptable from the point of the on-board CPU time consumption. The ‘*Timer#1 period*’ item setting depends on the resolution you want to measure coincidence window.

The '*Coff file*' entry selects a low-level data acquisition code, that is, TMS320C31 executable code. Each acquisition mode has a default file, but it may be overwritten by a user written code.

After the low-level acquisition program has been loaded, the execution is redirected to the address specified by the '*Go address*' item. It is usually the starting address of the program's '.prog' section.

The MCA supports up to 2 or up to 32 NIM ADCs, depending on the ADC interface. You can adjust the MCA to any of the NIM ADCs, on such a way to specify the polarities of the ADC handshaking signals in the ADC page of the MCA property sheet.

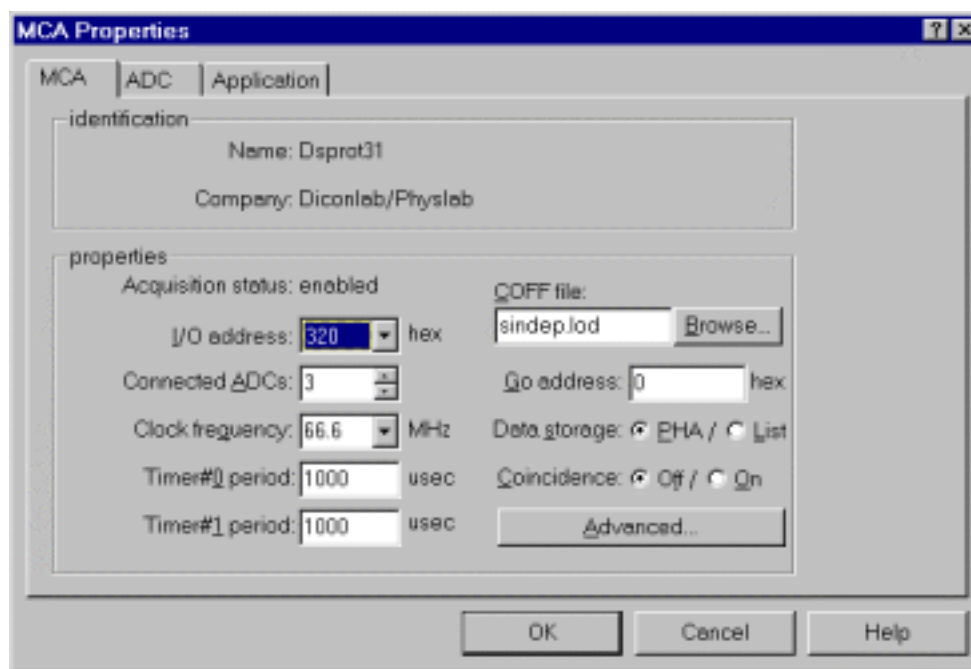


Figure 2.2.2.1.3.1. The 'MCA' dialog box for the 'DSProt31' MCA.

2.2.2.2. ADC

The ADC page refers to the analog to digital converter part of the MCA. It may be either on-board or external NIM type ADC. For each type, the program supports a corresponding page.

These pages contain one row for each ADC. The properties are listed in columns. To edit a property, click on it by the mouse. To exit, press the 'Enter' or the 'Esc' key.

Some MCA may support more than one ADC. Therefore, to enable or disable a particular ADC you should check or uncheck the corresponding box in the first column labeled 'ADC'.

The NIM (Nuclear Instrumentation Module) type of ADC features standard handshaking signals that are summarized in the table 2.2.2.2.1. For some type of MCA, it is possible to adjust their internal circuits to the polarity of the ADC handshaking signals. In this case, the program supports a list of the polarities and you need to enter their current settings for each of ADC connected to the MCA. Otherwise, you have to adjust the polarities on each ADC to the value specified by the MCA.

The resolution of the ADC defines the maximum number of the ADC channels used to convert the input pulse height. This parameter corresponds to the ‘*Resolution*’ item in the ADC dialog box. This resolution may be additionally reduced by the MCA itself or by the program. It is defined by the ‘*Input Resolution*’ item in the same dialog box and it corresponds to the amount of the memory allocated to the corresponding spectrum. Any MCA, which uses NIM ADC, can’t adjust automatically the resolution of your ADC. Therefore, you should do it manually.

Table 2.2.2.2.1. The NIM ADC handshaking signals

Signal	Description
<i>ENC</i>	Enable converter
<i>DRY</i>	Data ready
<i>DEN</i>	Enable data
<i>DACC</i>	Data accepted
<i>DT</i>	Dead time
<i>INV</i>	Invalid conversion

In the case of the XRFM application, the settings for the first two ADCs are not important and the third ADC always refers to the ADC connected to the MCA.

2.2.2.3. Application

The application page refers to the on-line data processing. It can be supported by the on-board hardware and software, the host or both of them.

Whether an application appears in the list box or not depends on which MCA you use and which data storage mode was selected in the ‘MCA’ dialog box.

Default

The program builds normal one or two-dimensional histograms also called displays. The displays may be defined by the ‘Displays’ command in the ‘Acquire’ menu.

In the PHA storage mode, the histograms are allocated on the on-board memory and field by the MCA.

In the list mode, the data are transferred event-by-event and the histograms are built off-board in the host memory.

TXRF

In the Total reflection X ray florescence (‘*TXRF*’), data storage capabilities are the same to the ‘*Default*’ application, with exception that only one-dimensional displays are possible.

In addition, the program drives up to four motors to adjust position and angle of the spectrum modifier and the sample holder.

The X ray beam may be monitored by a video image obtained with a CCD camera and a video capture card.

XRFM

In the X ray florescence micro-probe (‘*XRFM*’) mode, for each sample position the program collects one X ray energy spectrum.

In addition, the program builds two-dimensional XRF-images, for example, on such a way that for each selected ROI in the X ray energy spectrum, it stores number of counts as a function of the sample position.

The position of the sample is controlled by two micrometer-motors. The motors are driven by the program according to the selected area on the sample. The area may be selected from a video image, XRF-image or relative to the sample point that is in the front of the beam.

The energy spectrum is built on-board. Therefore, the MCA does not need to support the list mode. However, the program runs in the list mode and each event represents measurement at a particular sample position. This gives three parameters: x coordinate, y coordinate and the energy spectrum at (x, y) coordinate. The data are optionally stored on the disk in a successive order.

The program flow is described as follow:

1. move sample on such a way its coordinate (x,y) is in the front of the beam
2. collect energy spectrum at the coordinate (x,y)
3. analyze energy spectrum from the step two and update X ray maps
4. select a next point from a scan pattern and go to the first step

SPM

The Scanning particle micro-probe ('*SPM*') is a similar application to the 'XRFM'. It is supported only by the MCA with list mode and coincidence.

In difference from the XRFM, the sample is fixed while either on-board hardware or an external scanner drives the beam.

Like in the XRFM application, for each position of the beam the system collects three parameters: x coordinate, y coordinate and energy. If it is supported by the MCA, additional detectors may be added.

Custom

The data processing is performed by a user written DLL (dynamic link libraries). It is supported by the list mode only.

Another way to use the custom data processing is to define parameters by the 'prmlib.dll'. For more information see 'Displays' command in the 'Acquire' menu.

2.2.3. Host

The Host command lets you modify the basic properties of the program and host resources that are controlled by the program.

The command brings up a property sheet containing several pages described as follows.

2.2.3.1. Control

Sets the program control mode.

You can control the acquisition from the same computer you used to install the MCA, also called server, or from a remote computer also called client. This can be done on a several ways. They are listed on the 'Control' page and explained as follows:

None

All commands are applied from the program menus and dialog boxes on the Server. This is the default mode.

Local user written program

The commands are sent from a user written program in the form of Windows 'SendMessage' instruction. The program should run on the server.

Local Spector command interpreter

The commands are sent from a command interpreter using 'Control' application (binaries `ctl.exe` and `parser.dll`). This requires to write a script in a simple language and then run it either line-by-line or in a batch mode. The 'Control' application should run on the server.

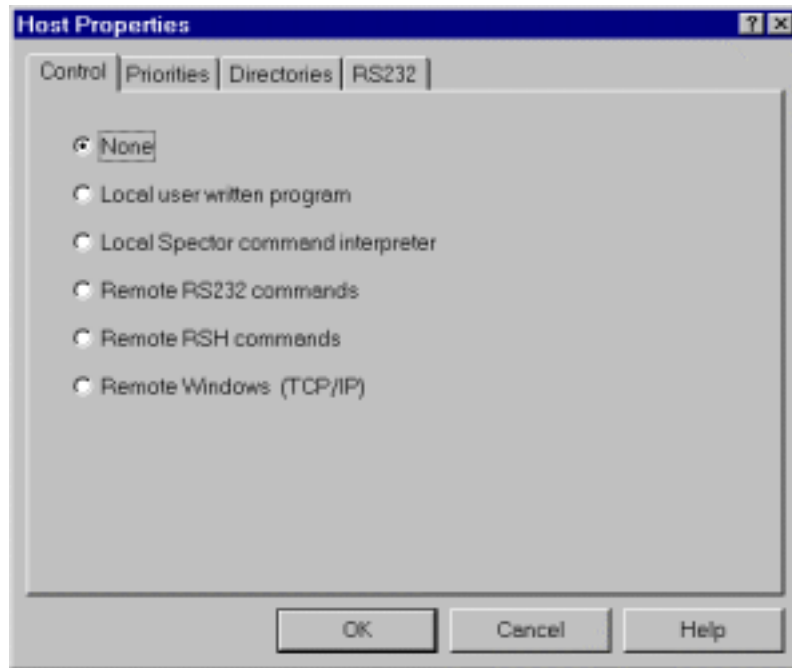


Figure 2.2.3.1.1. The 'Control' dialog box.

Remote RS232 commands

The commands are sent from a remote host via RS232 serial port. The remote host may run 'Control' application or a user written program.

Remote RSH commands

The commands are sent from a remote host using the RSH protocol.

Remote Windows

The commands are sent from the 'Spector', which runs on a remote host using the TCP/IP protocol. It allows you full access to the Spector-Server including graphics.

2.2.3.2. Priorities

The 'Priorities' page enables you to set base priority levels for all Spector's threads and processes. For more processes and threads see Appendix 2.

If both MDI and MNG run smoothly you don't need to change default values that are shown on the Fig. 2.2.3.2.1.

The Spector has two processes, MNG and MDI. By default both of them has normal priority class. Additionally, the MNG has a several threads: a default thread called '*mng*' is created when the program is loaded into the memory and has normal priority.

It receives any mouse or keyboard input and command message as well; '*daq*' thread is a data acquisition supervisor. It receives and synchronizes all data

acquisition commands. The 'daq I/O' thread performs I/O task during the data acquisition. The priority value of the 'daq' and 'daq I/O' threads are set to 'below normal' due to the dynamic priority boost, which increases their priority level. The MDI runs only primary thread which priority is boost by the program in order to draw spectra smoothly.

Apart from above listed threads the Spector has several threads for network support. They are started only if you select 'Remote Windows (TCP/IP)' in the 'System... Host... Control Remote' dialog box. In order to change their priorities you should edit 'spector.ini' file.

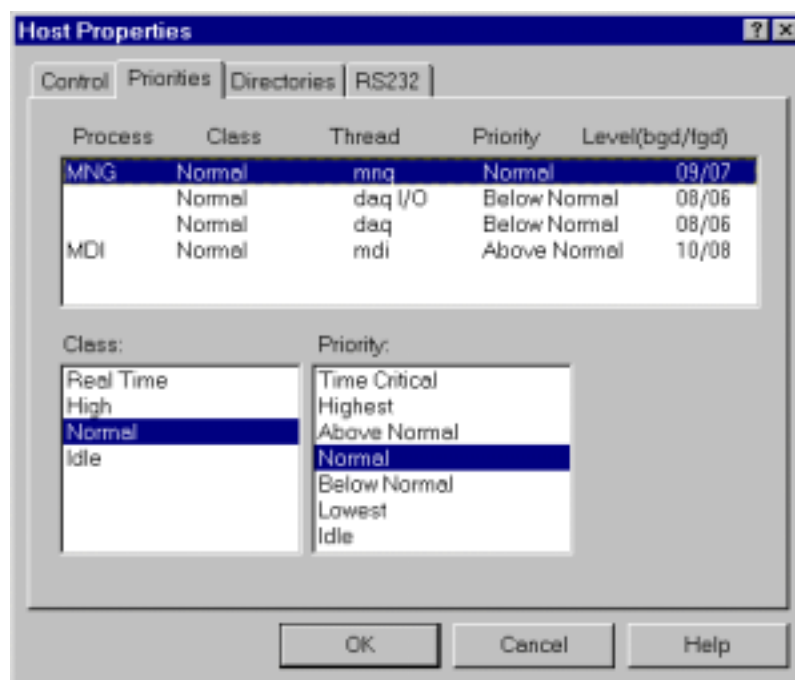


Figure 2.2.3.2.1. The 'Priorities' dialog box.

2.2.3.3. Directories

The Spector software comprises a various type of files that are placed in corresponding directories as follows:

- *Calibration:* histogram calibration files (cal).
- *Control:* command script files (ctl)
- *Custom:* DLL libraries used by the Custom application (dll).
- *Data:* acquisition data files (dat, lst, sp\$, as\$).
- *Fitting:* files related to fitting (fit).
- *ROI:* files that contain Region of interests (roi).
- *Stab:* files related to the ADC drift stabilization (stb).
- *Print:* print-out files.
- *Video:* video image file (bmp).
- *Scan:* files used by the XRFM and SPM application for scanning.

2.2.3.4. RS232

Sets serial communication ports (COM1 to COM4) properties. However, the properties are not applied until 'mpacom.dll' library is loaded. It happens when the program control is turned into 'Remote RS232 commands'.

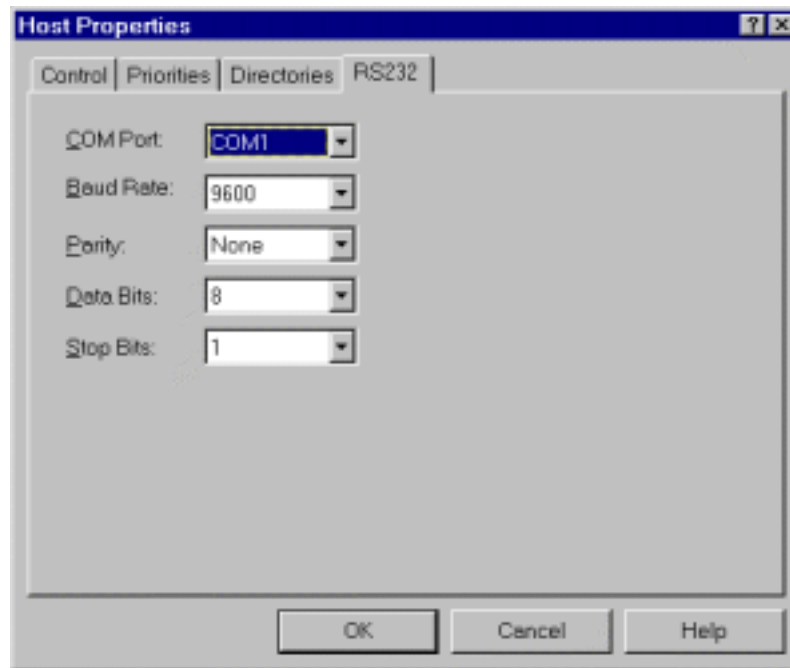


Figure 2.2.3.4.1. The 'RS232' dialog box.

2.2.4. Set as default

See the same command in the 'Acquire' menu.

2.3. Acquire menu

The Acquire menu lets you change the system properties that are more frequently modified than those listed under the 'System' menu. The properties are grouped in the three property sheets: 'MCA', 'Host' and Displays. They are explained as follows.

2.3.1. MCA

The MCA property sheet lists frequently used properties of the MCA hardware. The properties are grouped into a few pages described as follows.

2.3.1.1. Presets

The 'Presets' dialog box defines data acquisition presets. The presets cause the data acquisition to terminate when specified condition is satisfied. The condition depends on application and MCA. The Real time and Live time presets are common in the both cases. The Real time stops acquisition when specified time elapses since the start of measurement.

The Live time takes into account the time that ADC's circuits are busy processing an analog input (dead time). This time is measure on such a way that corresponding timer is stopped when the ADC is busy or 'dead'. Therefore, the Live time counter always counts slower then the Real time counter or in the best-case equal if there is no input-signal.

The difference between the Real and Live times gives an overall ADC busy or Dead time.

Assuming the ADC Dead time per each event is fixed and smaller than the average time between two successive counts, then the true number of events N is related to the measured number M as follows

$$N = M \frac{RealTime}{LiveTime}$$

If the ADC average 'Dead time' per event is not fixed then the above formula can't be applied until the former is much longer than average time between two successive events. For more discussion see: Knoll, G. F., 'Radiation Detection and Measurement', 1979 John Wiley & Sons Inc p.99-103.

2.3.1.2. Displays

The Displays command lets you define one or two-dimensional histograms called displays.

In the PHA mode, the histograms are usually collected by the MCA as one-dimensional spectra. Some MCA supports two-dimensional spectra also. In the list mode, the histograms are usually built by the software running on the host.

The number of displays, their dimension and purpose depends on the data MCA and application. Please, refer to the 'MCA' command in the 'System' menu for more information about the supported MCA and Applications.

The Display command brings up a property sheet that contains several pages. The main page is called 'Edit' and its view depends on the Application. Therefore, the dialog features are discussed for each application separately.

In addition, there are few more pages that set properties that are not defined by the Edit page.

2.3.1.2.1. Edit XRFM

The dialog contains a list of displays used by the XRFM application and dialog controls that enable you create, change or delete the displays.

A brief description of the XRF data processing is described in the 'System MCA Application'. Here is given a procedure how to create displays and described each of the property pages as well.

By default, the XRF application creates two single displays. They always appear as the first two displays in the list. Therefore, your list always contains at least two displays, which you can't remove.

The first is called 'Cumulative' and it contains spectrum accumulated at all scanning points of the sample. The second is called 'Transient' and it contains spectrum at the most recently measured sample point. Therefore, each time the system skips to the next sample point, it resets the 'Transient' and updates the 'Cumulative' spectrum. You can change their names or titles using the '*edit...*' control.

The resolution of the energy spectra is defined by the resolution of the corresponding ADC and it can be set by the 'ADC' page of the 'MCA' property sheet in the 'System' menu.

Usually, you define one or more regions of interest called ROI on the 'Cumulative' spectrum. Usually they are related to the characteristic X ray lines of the elements you are interested in. In the simple mode of analysis the program sums all events in a ROI to build corresponding elemental map. To recognize these lines more easy, it is recommended first to calibrate the detector.

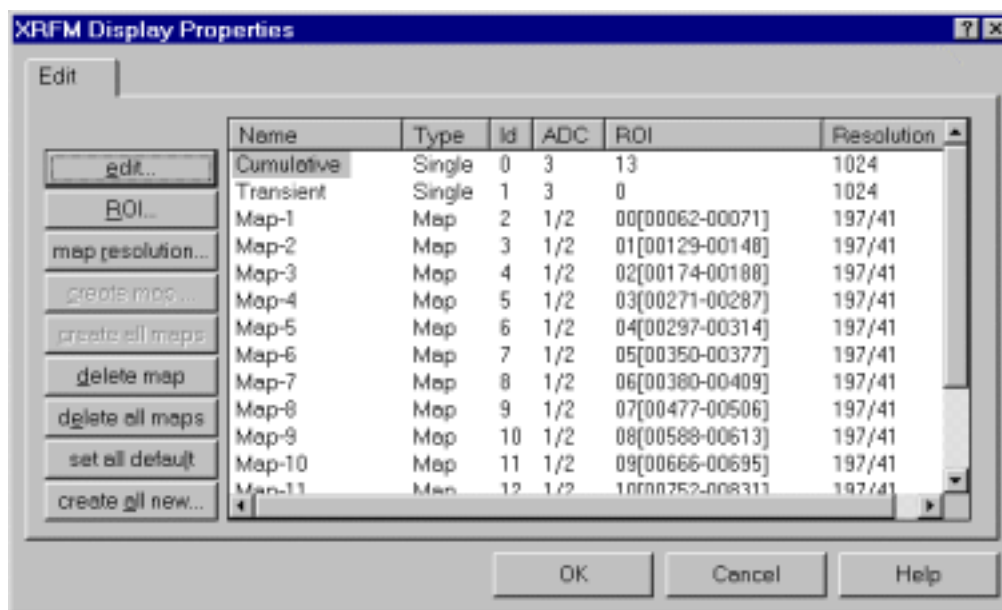
Detector Calibration

Figure 2.3.1.2.1.1. The 'Edit' dialog box for the XRFM application.

In order to calibrate detector using the XRFM application, first set all to the default values by the 'set all default' button. It creates one full scale ROI in the 'Cumulative' spectrum and one associated two-dimensional map. After that, you should export a small rectangular scan area using the MDI 'Export Rectangular Area' in the Scan menu of the MDI application.

An alternative way to 'set all default' approach is to delete all maps, that is, to define no maps. In this case the program does not move motors and you don't need to export any scan area.

In both cases the second step is to start acquisition and collect one calibration spectrum with a sample that features a few characteristic X ray lines. You can also change a few samples in the same measurement, one for each line. After collecting the spectrum, use the 'Calibrate' procedure in the 'Tools' menu of the MDI application. You can save the calibration using the same procedure.

The second way to calibrate detector is to switch to the 'Default' application. In this case, you don't need to export any scan region. Just collect the spectrum, calibrate it and save the calibration using the same 'Calibrate' dialog box in the MDI application. Afterwards you can switch again to the XRF application and load the calibration file using the 'Calibration' dialog box.

The fourth and the simplest way to make the calibration, is loading a XRFM data with already calibrated detector. In this case, it is assumed the calibration has not been changed. In addition, you can delete all ROIs and maps.

Now you can select the ROIs referring to the calibration. It can be done directly on the 'Cumulative' spectrum shown in the corresponding MDI window or by the 'ROI' command in the 'Display' dialog.

For each selected ROI, you should create one two-dimensional map. You can do it for all map at once by the 'create all maps' button or one by one map, using the 'create map' button.

The dimension of the map may be defined by the *'map resolution...'* command. All maps have the same dimension, which you can change even after creating the maps.

A selected map may be deleted by the *'delete map'* command. Note that each time you delete one ROI in the 'Cumulative' spectrum, the program removes its corresponding map.

Procedure

The display setup is summarized in the following procedure. Note that steps 1-3 are not necessary if you have calibrated detector and you loaded a valid data for X ray lines identification:

1. apply the 'Display' command in the 'Acquire' menu
2. press *'set all default'* button
3. press *'Ok'* button
4. export a Rectangular scan area using the 'MDI Scan Export'
5. make a short measurement by pressing the 'start' button
6. delete all ROIs and then create new ROIs on the Cumulative display shown by the MDI
7. apply the 'Displays' command in the 'Acquire' menu
8. press the *'map resolution...'* to define map resolution
9. press *'create all maps'*
10. press *'Ok'* button

Note that you can also create the ROIs from the 'Edit' page in the 'Display' property sheet by using the 'ROI...' button. This should be done after the step 7. In this case, you can skip the step 6.

Edit...

The command lets you to modify properties of the current selected display.

A 'Single display' or 'XRF Map Properties' dialog is brought up depending on which one is currently selected. It shows properties of the selected display.

If a map is currently selected, the 'Edit' dialog brings up a dialog that depends on the type of analysis that are used to build the map. a list of all ROIs defined for the 'Cumulative' spectrum. The dialog shows map's name and axis titles can be modified with using of corresponding controls.

If the 'Cumulative' or 'Transient' display is currently selected, the dialog shows only properties that can be changed: axis titles and name. In order to change resolution you should use 'ADC' page of the 'MCA' property sheet in the 'System' menu.

Roi...

The command creates ROIs for the currently selected display.

It is primary used to create ROI for 'Cumulative' spectrum which than enables you to define XRF map. Although not common, you can use it to define ROIs for each of map.

You can define up to 100 ROIs in the 'Cumulative' spectrum.



Figure 2.3.1.2.1.2 The 'XRF Map resolution' dialog.

Map resolution

The command sets XRF map x and y dimension in the channel units, that is, pixels on the sample.

All maps have the same dimension. You can change the default value before or after creating the maps. In any case, it is applied to all maps. The actual resolution in millimeters depends on the scan size that is defined in MDI with using of the Scan menu item (for example Scan... Export... Rectangular area) or directly from already existing XRF map or video image.

Create map

The command creates two-dimensional display called XRF map.

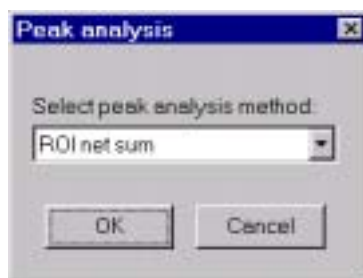


Figure 2.3.1.2.1.3 The 'Peak analysis'

The maps are built on such a way that for each pixel on the selected scan area the program makes analysis of a corresponding spectrum. Therefore you should first select mode of analysis from a dialog box that is brought immediately after applying the command. The current version supports only two modes: ROI total sum and ROI net sum.

ROI sum

In the ROI sum mode the program sums all events in a selected ROI and stores result at corresponding location in the map.

Normally the ROI corresponds to energy of X ray characteristic line for a given element. Therefore it is also referred to as peak ROI. The peak ROI must be selected on the Cumulative spectrum before applying 'Create' command. However the sum is performed on the Transient spectrum. The storage location is defined by the beam position on the sample.

The command brings up dialog box in which you can select ROI from a list of existing ROIs defined on the Cumulative spectrum. Therefore it is assumed that ROI is already created.

In addition you can define x and y-axis titles and name of the map as well.

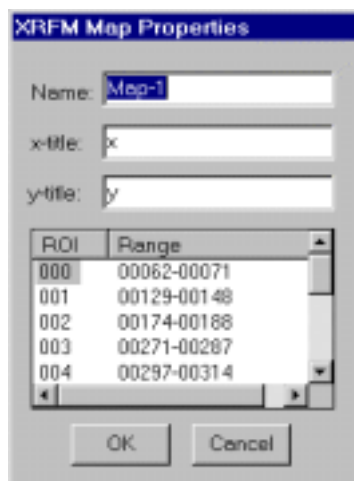


Figure 2.3.1.2.1.4 The 'XRFM Map Properties' dialog box for the 'ROI sum' analysis mode.

ROI net sum

This method is similar to the ROI sum but includes linear background correction. It is suitable for isolated peaks with good statistics.

The net sum is calculated inside the peak ROI that should be defined as in the case of the ROI sum method. The linear background is calculated on the following way: The program first finds average background in two so called background regions, one on the left side and the other on the right side of the peak ROI. The 'left' and 'right' backgrounds are assigned to the channels that are in the middle of the two background regions. The background in the peak ROI channels are then calculated by the linear interpolation. In addition a constant is added to the calculated background. All negative net sums are suppressed to zero.

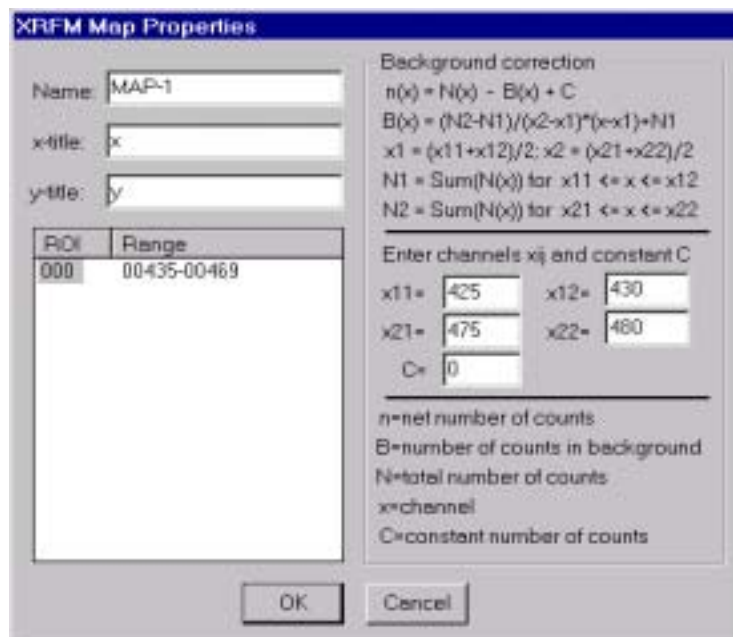


Figure 2.3.1.2.1.5 The 'XRFM Map Properties' dialog box for the 'ROI net sum' analysis mode.

The command brings up corresponding dialog that enables you to define boundaries of the two background regions, constant and peak ROI as well. In addition you can define x and y-axis titles and name of the map as well.

Create all maps

The command creates as many maps as there are ROIs in the ‘Cumulative’ spectrum. All maps use ‘ROI sum’ mode of analysis.

If there is no ROI in the ‘Cumulative’ spectrum, the control is disabled (grayed button).

Create all new...

Assuming the detector is calibrated, you can create both, ROIs and maps simultaneously by applying ‘create all new...’ command.

The command brings up the ‘Periodic table of Elements’ dialog box. It contains one button for each element in the periodic system table and a list of their X ray K and L lines.

An element is selected if the corresponding button is in the ‘checked’ state and one of its X ray is chosen.

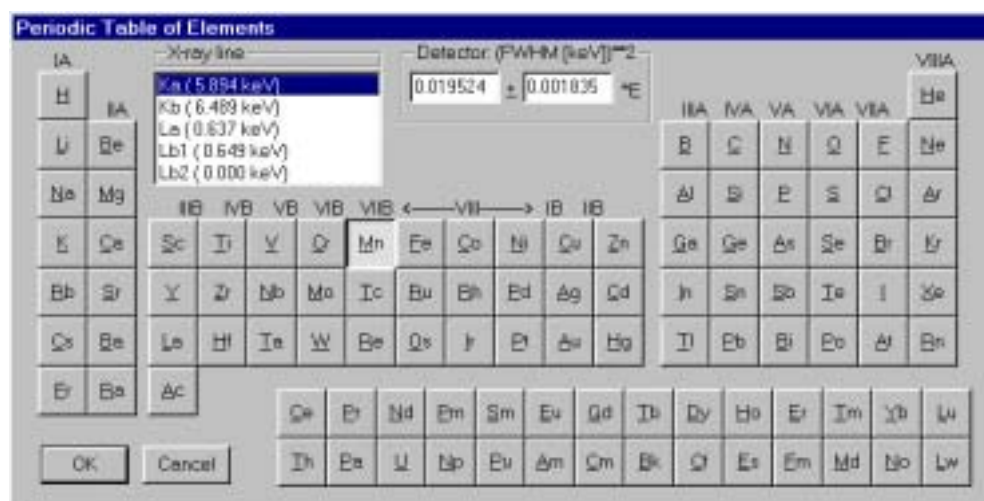


Figure 2.3.1.2.1.6 The ‘Periodic Table of Elements’ dialog.

The program automatically creates a ROI and a map for each selected element. The ROI is centered on the X ray line and has a width FWHM in keV calculated from the formula

$$FWHM[keV] = \sqrt{a + bE[keV]}$$

The coefficients a and b are defined by the controls labeled ‘Detector’. They are related to the detector’s resolution parameters as follows:

$$a = 2.355^2 \epsilon^2 N^2$$

$$b = 2.355^2 \epsilon FE$$

where ε is energy needed to produce electron-hole pair, F dimensionless factor which is a measure of the departure from the assumption of Poisson statistics in the generation of the electron-hole pairs (Fano factor), N is noise in number of electrons that is produced in charge collection and pulse processing, and E is energy of X ray in. For Si(Li) detector at 77K, $\varepsilon=3.76 \text{ eV}$, $F=0.122$ (values from 0.1 and 0.13 may be found in literature) and N depends on the detector and electronics and for good quality systems $N=7.89$ which gives total resolution 141 eV at 5.894 keV of Mn- K_{α} characteristic line. In this case

$$a = 0.0048810[\text{keV}^2]$$

$$b = 0.0004587[\text{keV}]$$

Note that constants a and b calculated above, give ROI which width is equal to the FWHM of the characteristic X ray line. In the case of low counting rate and poorer statistics you can use larger values in order to decrease total measurement time. In this case you should multiply both of them with a same factor that is larger than one (see Fig. 2.3.1.2.1.6 and note that factor 4 is used).

Delete map

The command deletes currently selected map. The corresponding ROI is not deleted.

Delete all maps

The command deletes all maps. The corresponding ROIs are not deleted.

2.3.1.3. Cleanup

The 'Cleanup' page enables you to select the displays (histograms) that content are set to zero on the start time.

The page lists all defined displays. Each display is represented by a row including its name and associated ADC. If a name is checked then the content of the corresponding display is set to zero. It is done when the acquisition is about to start. If you don't check a name, the system keeps the corresponding histogram on such a way that new histogram is superimposed on the previous. In the PHA-storage mode it means that MCA is not cleared and each spectrum bin contents starts from the existing value and not from zero.

2.3.1.4. Flush

The Flush page box enables/disables display transfer from data acquisition board to the host in the PHA data storage mode.

In the PHA mode, displays are built on-board while the program periodically transfers their content onto the memory allocated on the host. On this way, the displays are updated on the host.

The page lists all defined displays and enables you to turn on/off the data transfer by checking a corresponding box. Similar to the 'Cleanup' page, each display is represented by a row including its name and associated ADCs. If a display's name is checked then its contents is updated during the measurement.

You can enable or disable flushing in any time during the measurement. To do this, use the MDI application. You should click on the right mouse button when the mouse cursor points on the window of the corresponding display.

Note that the flushing has no effect on the MCA. That is, the MCA continues data collecting and processing regardless of the 'Flush' flag.

2.3.2. Host

2.3.2.1. Output

The 'Output' dialog box' customizes the data storage onto the disk.

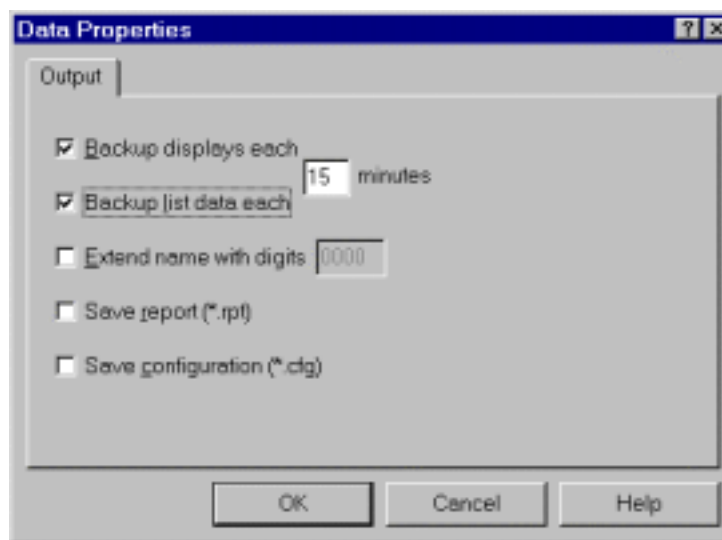


Figure 2.3.2.1.1. The 'Output' dialog box in the list mode.

Backup displays

The option protects all histograms from being lost due to the system crash during a measurement.

The displays may be optionally stored in the output data file that is defined in the 'Start' dialog. By default, the file is created after the measurement stops. However, if happens that the system crashes during a measurement then the displays will not be stored onto the disk, To prevent this you can force the program to make backup periodically during the measurement.

This option is enabled by checking the 'Backup displays'. The backup file is called 'backup.dat' and it is created in the default data directory. The backup occurs at the time intervals selected by the corresponding edit control.

Backup list data

The option protects list data file from being lost due to the system failure during a measurement.

The event-by-event data are optionally stored in the list-output data file that is defined in the 'Start' dialog. If that option is selected, the corresponding file is created on start but it is not closed until the acquisition terminates. However, if it happens that the system crashes before the end of run, then the list data output file is lost. To prevent this you can force the program to close and again open the file periodically. The reopening occurs at the time intervals selected by the corresponding edit control.

Extend name with digits

The option builds a ‘serial like’ output file name.

The program assign an integer to the output files (dat, lst, cfg, rpt) on such a way that the four last letters in the name takes a digit form. Therefore, if the name (not including the extension) contains more then four letters, those started from the fourth is overwritten. The appended digit is incremented each time you start the measurement. You can define its current value by the edit control that follows the command check-box.

Save report

The option saves report file each time the data (‘dat’ or ‘lst’) are saved.

The report file has ‘rpt’ extension and contains some basic text information about the program configuration. It is for a personal use only.

Save configuration

Saves configuration file each time the data (‘dat’ or ‘lst’) are saved.

The configuration file has extension ‘cfg’. The same information is built into the corresponding data file. Therefore, it is not necessary to save it. However, this enables loading the configuration explicitly without having any data.

2.3.3. Report

The command generates a short overview of the current selected properties.

It brings up the ‘Notepad’ application that displays the report.

2.3.4. Save

The command saves all acquisition properties (configuration) into the file with extension ‘cfg’.

The existing file is overwritten without any notice. It takes the same name as the most recent loaded configuration file. You can look for the name in the Report (see ‘Acquisition Report’ command). If you want to store it in a new file then use the ‘SaveAs...’ command. If you want to set it as a default configuration, then use the ‘Set As Default’ command.

2.3.5. Save as

The command saves all acquisition properties (configuration) into the file with a user-defined name.

The command brings up the ‘Save As’ dialog box that enables you to enter the name. The file takes default extension ‘cfg’ if no extension is specified. If you want to set it as a default configuration, then use the ‘Set As Default’ command.

2.3.6. Load

The command loads previously saved acquisition properties (configuration) from a file.

2.3.7. Set as default

The command sets the current acquisition properties (configuration) as default.

Each time you start the program it automatically loads the default configuration. The default configuration file name is stored in the program initialization file (spector.ini). If the program can’t find the default configuration file, it takes information that is built-in the code.

2.4. Desktop

The application's client window is called 'Desktop'. It contains acquisition controls that enable you to control flow of a measurement and monitor various parameters during acquisition. The current version supports three types of controls: 'Action', 'Counter' and 'Title'. The 'Action' controls are used to control the acquisition flow, for example to start measurement. The 'Counters' usually show various timers/counters and they purpose is described by 'Title' controls. In addition, the 'Title' may be used to display a user-defined text or information.

The desktop may be stored on, or retrieved from the disk. For each



Figure 2.4.1 The MNG Desktop view in the stop status (left) and during customisation (right)

application, the program uses one default file that you can overwrite with a custom defined desktop. The current program version does not store desktop information into the data file. Therefore, each time you load new data, the system keeps desktop or takes the default (if new data and old application are incompatible).

2.4.1. Customize

One can define position and view of the desktop controls. To do this the desktop must be first 'opened' for editing. To do this, click on the right mouse button over the desktop area.

Generally, the view of the desktop depends on the application. On load time, the program reads desktop information from a desktop-file. If the file can't be found there, the program loads it from executable. For each application, there is one desktop-file. The table 2.4.1.1 shows a list of all desktop-files and associated applications. In the future, the desktop may be associated with data as well.

The customization allows you to modify desktop file. You can change position and view of the controls, but with the current program version, you can't add new controls.

Table 2.4.1.1 The MNG Desktop data-base files

Application	Desktop file
XRFM	Xrfm.dsk
SPM	Spm.dsk
Custom	Custom.dsk
default-list	List.dsk
default-PHA	Pha.dsk

Although you can do it by editing the desktop file, it is not recommended. The ‘*Open*’ item in the ‘*Desktop pop-up menu*’ let you access to the desktop controls. The menu is invoked by clicking on the right mouse button with mouse pointer anywhere inside the desktop (except over a control).

After gaining the access to the desktop, you should first select a control. To do this, move the mouse pointer over the control and click on the left button. In that case, the control gets selection boundary. Each time you select next control, the previous loses selection. To select a group of controls you should draw a selection rectangle around the group. To draw the rectangle, drag pointer from one to the other (opposite) corner. The selected control(s) may be displaced using arrow keys or dragging the mouse pointer.

2.4.1.1. Open/close

The ‘*Open*’ item disables desktop control features, enables desktop customization and turns itself into the ‘*Close*’ state. The ‘*Close*’ terminates the customization returning the controls in the normal, acquisition mode.

2.4.1.2. Edit

Sets desktop-control properties. A property dialog box is invoked. Another way to start the same dialog is to double-click the left mouse button when its pointer is placed on the control. The dialog let you set the basic desktop-control properties. The properties, hence the dialog view, depends on the type of control.

2.4.1.3. Title properties

Size&Pos

Sets the size and position of the desktop-control in the screen pixel units. The position is taken relative to the left top corner of the desktop client area

Style

‘*Left*’, ‘*Right*’ and ‘*Center*’ display text flush left, right, or center respectively, in a simple rectangle. Words that would past the end of a line are automatically wrapped to the beginning of the next line. The ‘*Visible*’ style makes control visible. If this style is not set, the control will be initially invisible. The ‘*Disable*’ style produces ‘grayed’ text. The ‘*Border*’ style draws a rectangle around the control.

Color

Sets color of the desktop-control. The control has its window's color and text's background/foreground colors. For each of them, you can select one of the colors from the default system palette.

Text

Sets text of the desktop-control.

2.4.1.4. Action properties

2.4.1.5. Counter properties

Style

'*Left*', '*Right*' and '*Center*' display text flush left, right, or center respectively, in a simple rectangle. Words that would past the end of a line are automatically wrapped to the beginning of the next line. The '*Visible*' style makes control visible. If this style is not set, the control will be initially invisible. The '*Border*' style draws a rectangle around the control. If '*Read only*' style is checked, it prevents editing the content of the control. The content of the counter changes automatically, by the program. Therefore, you don't need to edit it. If you don't check the box, you will be able to edit it. The '*Disable*' style produces 'grayed' control's text.

Color

Sets color of the desktop-control. The control has its window's color and text's background/foreground colors. For each of them, you can select one of the colors from the default system palette.

2.4.1.6. Save

The command saves desktop into the default desktop file. The default desktop file depends on the application. The file gets the same name as application and extension 'dsk' ('txrf.dsk', 'spm.xrf', 'xrfm.dsk'). In the case of a default application, the name becomes 'list.dsk' and 'pha.dsk' for list and PHA data storage respectively.

2.4.1.7. Save as

The command saves desktop into a user-selected file. The 'Save As' dialog brings up that let you select the file name. This file is not used by the system, until you save them with the default name (refer to 'Desktop Save' for default file names). Usually, the command is used to make desktop backup.

2.4.1.8. Load

The command loads desktop from a file. The 'Open' dialog brings up that let you select the file name. The loaded desktop is not used as default until you apply 'Desktop Save' command. The program always starts using the default desktop file.

2.4.2. Action controls

There are four basic '*Action*' desktop-controls. They enable you to control the flow of measurement: to start, pause, continue and stop the acquisition.

2.4.2.1. Start

Starts acquisition. The control invokes the ‘Start’ dialog box to set the output file name. In addition, you can enter a short (128 characters in size) information, called header that is stored together with your data.

In the PHA data storage mode you can store histograms and all associated information (calibration,...), which together form displays. By default, the displays (histograms) are stored into data file in a binary format. Such file has default ‘dat’ extension and it is placed in the default data directory. If you don’t specify the extension or path, the default values are used. If you specify ‘sp\$’ or ‘as\$’ extension then the file is saved in the corresponding ASCII format. In any other case, the file has default binary format. To save histograms you should check the box labeled ‘Display file’.



Figure 2.4.2.1.1. The ‘Start’ dialog box in the PHA mode.

In the list-data storage mode, you can also store data event-by-event. The corresponding list file has ‘lst’ extension and it is placed in the default directory. Only the default binary format is supported. If you want to transform list data into ASCII



Figure 2.4.2.1.2. The ‘Start’ dialog box in the list mode.

format use ‘Data Replay’ command. The header information is also stored together with the list data file. The list file enables you to ‘reconstruct’ your original displays (histograms). Therefore, if you save ‘lst’ file then it is not necessary to save ‘dat’ file. To enable list data dumping on the disk you should check the box labeled ‘List file’.

The histograms are stored **after** the acquisition is terminated. In the same time, the list file is closed. Therefore, it is useful for a longer measurement to enable file backup-ing (see Acquisition Properties Output). It causes that displays are periodically stored on the disk during the measurement. In the same time, the list file is reopened (closed and immediately opened).

To make repeatedly runs use Spector's Control program (binary ctl.exe). There you can write a script.

2.4.2.2. Pause

Pauses measurement.

The timers and any on/off-board data processing are suspended until the 'Continue' command is applied. The current version does not support to change preset time during pause.

2.4.2.3. Continue

Returns the acquisition in the same status as it was immediately before the 'Pause' command was applied.

2.4.2.4. Stop

Stops acquisition.

The data are automatically saved if it was specified on the start time.

3. MDI

The MDI application features graphical presentation and spectra manipulation. The spectra also refer to histograms or displays.

Spectra collecting during acquisition are defined with using of the MNG application while the MDI periodically reads them and shows on the screen. Apart from this you can use MDI to create an auxiliary one and two-dimensional spectra. Usually, they are not updated during measurement unless they are one-dimensional projection.

Each spectrum is shown in a separate window. By closing a window or MDI itself you don't destroy associated spectrum only if it was defined by the MNG application.

Apart from acquisition spectra the MDI supports two-dimensional video images.

3.1. Display menu

The Display menu contains commands that let you load, create, close and print one and two-dimensional spectra (displays).

The basic set of displays that are used during acquisition should be created by the MNG application. You can *open* them for viewing, but you can't delete them. Therefore, if you close such display or even the MDI application, it has no effect on the displays and they remain in the memory. However, using the MDI you can create some temporary displays during measurement that lasts until you close them.

3.1.1. Open

The command opens a new display.

It brings up '*Open Display*' dialog that contains a list of one and two-dimensional displays created by the MNG (see '*MNG Acquisition Displays*'). A two-dimensional display can be opened either as a map ('*map view*') or in isometric projection ('*isometric view*'). To choose the view select appropriate button.

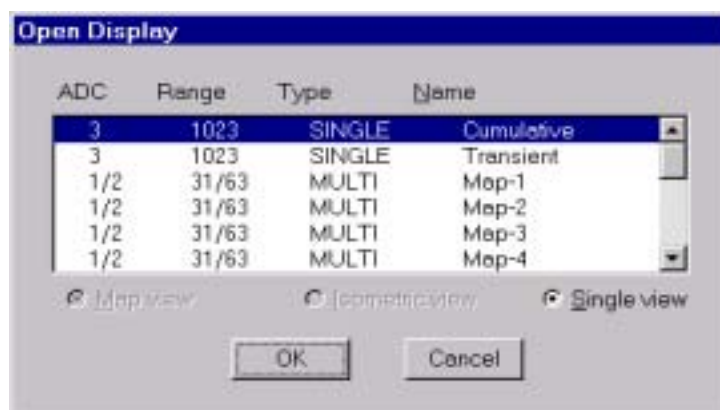


Figure 3.1.1.1 The '*Open Display*' dialog box.

To open a display click on the corresponding item in the list and then press '*OK*' button, or double-click on the item. A new display window will be created and shown as the active window. You can close and open it again without losing the data. Moreover, you are allowed to close the MDI application without affecting the data. In addition, you can open more instances of the same spectrum. This means that a spectrum is kept in the memory until you delete it from MNG. This happens when

you load new data or use ‘Delete’ command in the ‘MNG Acquisition Displays’ dialog.

3.1.2. Open all

The command opens all available displays.

The window of the last opened display becomes active. The windows should be created by ‘MNG Acquisition Displays’ dialog.

3.1.3. Open all on start

The command enables or disables automatic opening of all displays each time the acquisition is started or a new data are loaded.

3.1.4. Read

The command reads display from a user-selected file and shows it in a separate window.

This is an inverse of the ‘Write’ command. The file name and format should be entered in the dialog box that is automatically opened. Usually, the command is used to read previously stored display that had been created by the MDI application. See also ‘Display Read’ command.

3.1.5. Write

The command writes data of the active display in ASCII format (extension .ASC).

For each count in the spectra one line is written that contains x(y) channel number and channel content, separated by a blank character. In addition, the GANAAS format with extension .SPE is supported. It is a special ASCII format with a short header, which can be used to evaluate spectra with the GANAAS software from the I.A.E.A.

The command starts ‘Save As’ dialog that let you select file name and format. To save all displays you should use ‘MNG Data Save’ command. Note this does not save displays that were created by ‘Display Create’ command. Therefore, the only way to save such displays is to apply ‘Write’ command.

3.1.6. Create single

The command creates one-dimensional auxiliary spectrum.

The spectrum can’t be used to collect data into. Moreover, it lasts until the MDI is active and can’t be recovered after that. One should use the ‘Display ... Write’ command to save it.

The data are stored as 32-bit unsigned integers and initially all channels contain zero counts. To store data into the display you can use ‘Tools Compose’ command or do it from the Control application.

The command brings up the ‘Create Single’ dialog. There you can define the basic display feature.

The ‘Name’ defines display name; the ‘Title’ sets title on its axes; the ‘Resolution’ is an integer that determines the number of channels. The ‘Minimum’ and ‘Maximum’ are real numbers that are used to calibrate x-axis (linear transformation). They correspond to the first (ch=0) and last (ch=Resolution-1) channel, respectively. Therefore, if you want to display channels on the axis then set *Minimum*=0 and *Maximum*=*Resolution*-1.

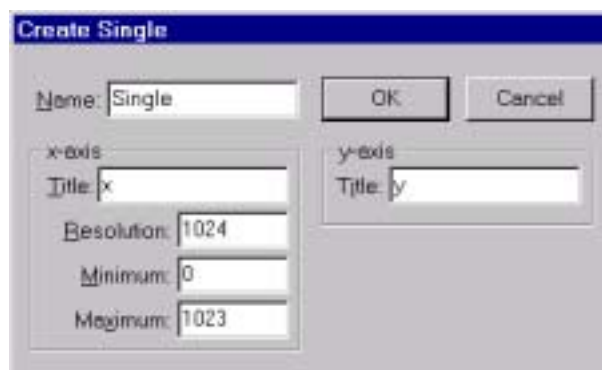


Figure 3.1.6.1 The 'Create Single Display' dialog box.

3.1.7. Create map

The command creates two-dimensional auxiliary spectrum.

The spectrum can't be used to collect data. Moreover, it lasts until the MDI is active and can't be recovered after that. One should use the 'Display ... Write' command to save it.

The data are stored as 32-bit unsigned integers and initially all channels contain zero counts. To store data into the display you can use 'Tools Compose' command or do it from the Control application.



Figure 3.1.7.1 The 'Create Map' dialog box.

The command brings up the 'Create Map' dialog. There you can define the basic display feature. The 'Name' defines display name. For each axis, you can define the following items: the 'Title' sets title on its axes; the 'Resolution' determines the number of channels. The 'Minimum' and 'Maximum' are real numbers that are used to calibrate x/y-axis (linear transformation). They correspond to the first ($ch=0$) and last ($ch=Resolution-1$) channel, respectively. Therefore, if you want to display channels on particular axis then set $Minimum=0$ and $Maximum=Resolution-1$.

3.1.8. Print

The command prints one or more histograms to the printer.

It is not intended to print final work. Rather, use it for draft, multi-histogram printouts.

The command brings up 'Print' dialog that let you customize printout. The dialog features an automatic histogram placement. To enable this you should check the 'Paper zones-Enable' box. In that case, the paper is virtually divided into two

dimensional, equal-sized areas called ‘zones’. The number of zones in the horizontal and vertical direction can be selected by ‘*Paper zones-Vert.*’ and ‘*Paper zones-Horiz.*’ lists, respectively.

To place a histogram in a zone you should select it from the upper list of histograms and then apply ‘*Add*’ command. In that case, the selected histogram appears on the bottom list and it is indicated by the shadow area on the page layout. To select all histograms apply ‘*Add All*’. To deselect, that is remove from printout, one or all histograms use ‘*Remove*’ and ‘*Remove All*’ respectively. The zones are filled by the histograms automatically and in the successive order.

The position of a histogram inside a zone is the same for each zone. This is set by the ‘*Left*’, ‘*Right*’, ‘*Top*’ and ‘*Bottom*’ zone margins that are defined in the units of 1/100th of the vertical and horizontal zone size.

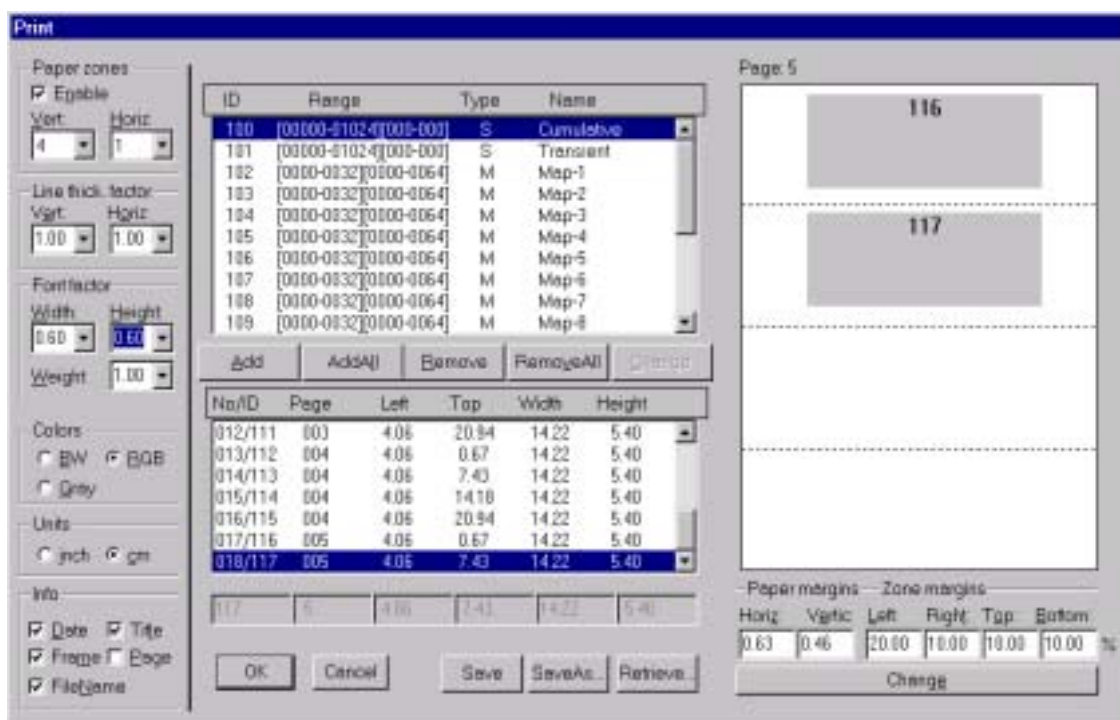


Figure 3.1.8.1. The ‘Print’ dialog box.

One can define the histogram position manually. To do this the ‘*Paper zones - Enable*’ box should be unchecked. In that case, you can use the four edit-controls below the bottom list. Starting from the leftmost, they define page, left and top corner position, height and width, respectively.

The group of checkbox called ‘*Info*’ let you to print a few additional information: the ‘*Date*’- date of printing; the ‘*FileName*’- name of file that contains printed histograms; the ‘*Title*’- title for each histogram; ‘*PageNum*’ - number of each page; ‘*Frame*’- rectangular border.

The size of drawing object may be adjusted by ‘*Thickness factor-Vert*’ and ‘*Thicknes facot-Horiz.*’. You can also use ‘*Options Axis View*’ and ‘*Options Single View*’ commands to adjust thickness of lines and size of symbols. To adjust font size use ‘*Font factor – Width*’, ‘*Font factor – Height*’ and ‘*Font factor – Weight*’.

3. MNG

3.1.9. Setup printer

The command setups your printer.

A standard Windows dialog box is opened that let you set properties of you printer.

3.1.10. Exit

The command closes the MDI application and frees its resources from memory.

The acquisition data are not lost, because they are allocated by the MNG application. Therefore, you can retrieve them by starting again the MDI application.

Any resources that are not allocated by MNG are lost. Such are: any display produced from a display defined by MNG (projection) and any display created by 'Display Create' command.

3.2. Window menu

The Window menu let you manipulate with MDI windows. Each window contains one display.

3.2.1. Tile

The command arranges all opened windows over the full MDI client area trying to make the same size for all windows.

3.2.2. Cascade

The command arranges all opened windows in a cascade.

3.2.3. Arrange icons

The command arranges all minimized windows in a series at the bottom of the MDI client area. Each minimized window is represented by an icon.

3.2.4. Close all

The command closes all opened windows. The same as 'Display CloseAll' command'

3.2.5. Arrange spector

The command arranges the main windows of the MNG and MDI application according to the default values. That is, sets positions and dimensions of the former, as they were when 'Window Save Arrangement' command was applied.

3.2.6. Save arrangement

The command saves the current positions and dimensions of the MNG and MDI main windows as default. Therefore, you should adjust the size and position of the windows before applying the command. Next time you start the application, they will take these positions.

3.2.7. Window list

At the end of the Window pop-up menu, all created MDI windows are listed with their names. In addition, the current active window is selected. If you change the selection, a corresponding display becomes active and it is displayed in front of all others.

3.3. Region menu

The Region menu contains commands related to the Region. The Region is a portion of one or two-dimensional display (histogram) that contains a selected channels. Each display may have only one Region.

Each time you define a new region, the previous is lost. They are mainly used to enlarge a part of spectrum. However, if you are especially interested in a Region, you can declare it to be ROI (Region of Interest).

In the case of two-dimensional displays, the channels may be bounded by a various geometrical shapes like rectangle, polygon, circle and a pie (sector). There are two special cases of rectangle: one with maximal width called y-slice and the second with maximal height called x-slice. For each shape, one marker corresponds that you can use to select the channels. The marker can be chosen by 'Region Shape' command.

The one-dimensional displays have Region that contains successive channels. The corresponding marker has x-slice shape with vertical lines related to the edge channels. Moreover, you can define rectangular Region for one-dimensional displays.

Table 3.3.1 List of possible Region shapes

type	description
Rectangle	channels form rectangular that sides are parallel with x and y axis; defined by four corners; supported by one and two-dimensional displays
x-Slice	special case of Rectangle that has maximal size in the y direction; defined by four corners type; supported by one and two-dimensional displays
y-Slice	special case of Rectangle that has maximal size in the x direction; defined by four corners type; supported by one and two dimensional displays
Polygon	channels form one or more polygons; each polygon may have arbitrary number of corners; supported by two-dimensional displays
Circle	channels form a circle; defined by center and a point on the circle
Pie	channels form a circle; defined by center and a point on the circle; supported by two-dimensional displays

To construct a region you can use mouse or keyboard. In addition, you can construct a ROI using a dialog box. Here is described how to select a Region by mouse and keyboard.

Moving the display cursor over the display area

To read a content of a particular channel you should move the display cursor at the screen pixel that corresponds to this channel. You can do it by mouse or keyboard. If you use mouse, then *move the mouse pointer and then click the left-button*. The other way is to *press the left-button and drag the mouse*. The position of display cursor is shown bellow the histogram. If the spectrum contains more channels than there are pixels available for drawing, it may happen that more channels share the same pixel. In that case, the program shows channel containing the highest number of counts. Therefore, you may not be able to read the content of a particular channel or select it as a region boundary point. To overcome the problem, you can *use the arrow keys*. In that case, the cursor is moved on such a way that it never skips any channel.

Using mouse to select rectangle region

1. select rectangular shape using 'Region Shape' command or a corresponding toolbar bitmap
2. move the display cursor at one of corners
3. press the right mouse-button
4. drag the mouse to the opposite corner
5. release the mouse

Notes:

While dragging mouse is in progress, the cursor pointer follows screen pixels. Therefore some channels may be skipped. To overcome this problem use keyboard. After the 3rd step the display cursor turns into the up-arrow shape. It goes back into the original shape after the 5th step.

Using keyboard to select a rectangle region

1. select rectangular shape using 'Region Shape' command or a corresponding toolbar bitmap
2. move the display cursor at one of corners
3. press 'Shift' key
4. move the cursor by one of arrow keys, keeping the 'Shift' key pressed
5. release all keys at the final corner
6. press 'End' or 'Esc' key (this terminates the procedure and turns the cursor shape back into the original shape)

Notes:

After the 4th step, you can release and then press the keys again to continue. Before pressing the 'End' key, you are allowed to move the cursor with mouse. That let you first move the cursor very fast but roughly in channels and then adjust the channels using keyboard. In the 3rd step, the display cursor turns into the up-arrow shape. It goes back into the original shape after the 5th step.

Using mouse to select a polygonal region

1. select polygonal shape using 'Region Shape' command or a corresponding toolbar bitmap
2. move the cursor to next corner and click on the right-button
3. press 'End' or 'Esc' key after selecting the last corner

Notes:

Each time you apply 2nd step, a new corner is defined and you are free to move the mouse to the next corner. To check the position you can click on the left mouse button. In any time you can use keyboard to select a new corner (Shift + Arrow key, see bellow). You can create **multi-polygonal** ROI. To do this, use 'Tab' key if you want to close one polygon and skip to the next. However, place the cursor at the first corner of the next polygon before applying 'Tab',

Using keyboard to select polygonal region

1. select polygonal shape using 'Region Shape' command or a corresponding toolbar bitmap
2. move the cursor to a corner using 'Shift + Arrow' key and press 'Enter' key
3. press 'End' or 'Esc' key after selecting the last corner

Notes:

Each time you apply 2nd step, a new corner is defined. In any time, you can release the keys and use mouse to approach the corner faster. You can create more polygons that belong to the same ROI. To do this, use 'Tab' key if you want to close one polygon and skip to the next. However, place the cursor at the first corner of the next polygon before applying 'Tab'.

You can define only one region for each display. Therefore, each time you define new region, the program deletes previous. However, if you assign it as a Region of interest (ROI), than the system keeps it in the memory and creates additional resources that enables you to use it for data analysis. To create a ROI use '*Region Create ROI*' command.

Using mouse to select a circle region

1. select circular shape using 'Region Shape' command or a corresponding toolbar bitmap
2. move the cursor to the center
3. press right-button
4. drag the mouse to any point on the circle
5. release the button

Notes:

In the 3rd step the display cursor turns into the up-arrow shape. It goes back into the original shape after the 5th step. To check the position you can click on the left mouse button.

Using mouse to select a pie region

1. select circular shape using 'Region Shape' command or a corresponding toolbar bitmap
2. move the cursor to the center
3. click on the right-button
4. move cursor at the end of the first segment line
5. click on the right-button
6. move cursor at the end of the second segment line
7. press 'End' key

Notes:

In the 3rd step, the display cursor turns into the up-arrow shape. It goes back into the original shape after the 5th step. To check the position you can click on the left mouse button.

3.3.1. Zoom

The command enlarges current selected region to fit the whole area available for the spectrum drawing.

The size of the area depends on the window size. For one-dimensional displays and rectangular Region, the y-boundaries are ignored unless you check 'Auto-scale' in the 'Options Scaling' dialog box.

3.3.2. Back

The command restores previous selected region. Successive 'Back' commands have no effect.

3.3.3. Home

The command restores a display to the basic configuration on such a way that full spectrum is shown. In addition, the vertical scale is restored to fit maximal number of counts.

3.3.4. Sum

The command calculates number of counts contained in the current selected region. The edge-points are taken into account.

3.3.5. Gauss fit

The command fits spectrum contained in a selected region to the Gaussian function. The fit procedure assumes linear background. The fitted curve is displayed and a dialog box shows the results:



Fig 3.3.5.1. The 'Logfile Options' dialog box

The full-width at half-maximum FWHM and Position of the Gaussian can be changed and a 'New Fit' can be performed. In addition, they can be fixed to the

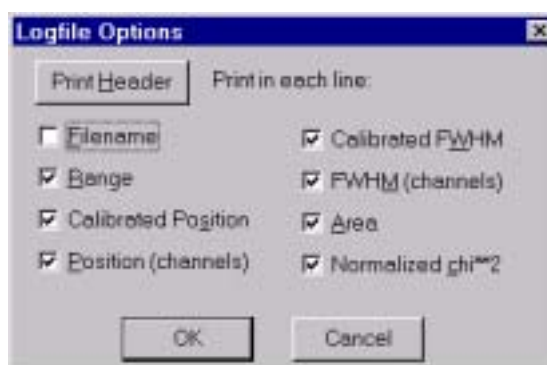


Figure 3.3.5.2. The 'Single Gaussian Peak Fit' dialog box

entered value by marking the respective checkbox. If the spectrum is calibrated, the 'Position' and 'FWHM' may be displayed in calibrated units. The 'Area' of the Gaussian is also shown. For all values also the standard deviations are given. The value of 'Q' is the normalized χ^2 . To take into account the systematic error of the line-shape, you may multiply the errors with the square-root of 'Q'. Click on 'Save' to

append a line containing the results to a 'Logfile' with the specified name. The 'OK' closes the dialog and lets the fitted function in the display also if it is refreshed, whereas after the 'Cancel' the curve no longer will be shown in a refreshed display. The 'Options...' opens a new dialog box to define the information in the 'Logfile'. A several quantities are written in standard text format with 'Tabs' as separators and a 'Newline' character at the end of each line. On this way, the file can be read with standard calculation programs like EXCEL. Click on the 'Print Header' to write a header line.

3.4. ROI menu

A ROI is a group of channels that let you manage associated data on an easy way. For example: project and sum. You can define up to 99 ROI per each display.

In difference from ROI any Region lasts until you select new one or close its display (see Region Menu). However, if you are especially interested in a Region, you can declare it as a ROI (Region of Interest). This is done by the 'ROI Create' or the 'Region Create ROI' command. In that case, the program copies Region resources and keeps them in the memory until you destroy them. In addition, you can create a ROI directly from a dialog box. In both cases, the program assigns an identification or referent number.

In a one-dimensional spectrum, each ROI is indicated by a different color on the ROI bar that is located bellow the spectrum. In addition, the spectrum channels are drawn in the same color.

In a two-dimensional spectrum, ROI channels are shown either by inverted colors or boundary lines are drawn. This can be customized by 'Options Map View' dialog.

3.4.1. Create

The command creates a ROI from current selected Region. It does the same as 'Region Create ROI' command.

3.4.2. Delete

The command deletes active ROI and activates previously defined ROI. This command has no effect on the ROIs defined in the MPA/PC Manager until the 'ROI Keep always' is checked.

3.4.3. Edit

The command creates new ROI(s) or deletes/modifies ROIs that were previously created.

You can save all ROIs into a file and load them from a file. The command is applied on activate display. It brings up a dialog box that let you manipulate with ROIs. There are two different dialogs, one for one-dimensional (single) and one for two-dimensional (map) displays.

3.4.3.1. One-dimensional ROI

The one-dimensional ROI (single) is defined by a starting (left edge) and a terminating (right edge) channel. The edge channels are included into the ROI.

The dialog includes a list of all ROIs for currently active display and a few controls to create new and delete/change selected ROI. To define the left and the right



Figure 3.4.3.1.1. The 'Create one-dimensional ROI' dialog box.

edge channels use 'From' and 'To' edit control, respectively. By 'Add' you can add a new ROI in the list.

If you want to change any ROI from the list, you should first select it. This causes that its channels appear in the edit controls and you can modify it. After that,

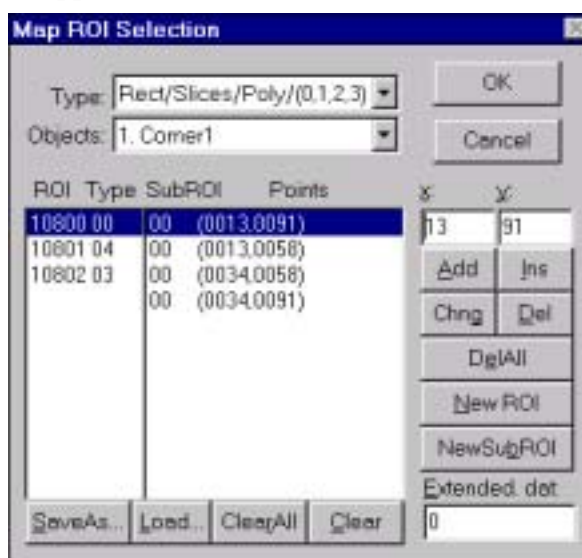


Figure 3.4.3.2.1. The 'Create two-dimensional ROI' dialog box.

press 'Change' button.

The 'Delete' command removes selected ROI from the list. The 'SaveAs..' command saves all ROIs from the dialog list in a user selected file. The inverse command is 'Load...'. Note that you can't save or load ROIs for all displays simultaneously.

3.4.3.2. Two-dimensional ROI

Each time you want to create (add) a new ROI, first you should select the type. It can be done from the 'Type' list. The following shapes are supported by the program:

Rectangle: channels form rectangular that have sides parallel with x and y axis; defined by four corners; type identification number 0;

Slice: special case of Rectangle that has maximal size in the x or y direction; defined by four corners type; identification number 1 for x-slice and 2 for y-slice;

Polygon: channels form one or more polygons; type identification number 3; defined by number of polygons and each polygon by arbitrary number of corners

Circle: channels form a circle; defined by center and a point on the circle; type identification number 4;

Pie: channels form a sector; defined by center and two points on the edges of straight lines; type identification number 4

After selecting the 'Type' you should apply the 'New ROI' command. It displays ROI identification number and its type in the columns 'ROI' and 'Type' respectively. This allows you to 'Add' the 'x' and 'y' coordinates.

For the 'Polygonal' shape ROI you can define more sub-polygons. For each new sub-polygon, enter (x,y) coordinates of any of its corner and press 'NewSubROI' button. The program automatically closes the polygon. Therefore, you don't need to repeat the first corner at the bottom of the list.

You can use 'Insert' to insert a point above the selected one in the list, 'Delete' to delete any point in the list, or 'DeleteAll' to remove all points. The 'Clear' command deletes selected ROIs, while 'ClearAll' deletes all ROIs from the list

The 'SaveAs' command saves all listed ROIs into a user-selected file. The command invokes a 'Save As' dialog that let you define file name. If the file extension is not specify explicitly it takes name 'roi'. In addition, the file is stored in a default directory. To change the default directory use 'Options Directories'. To load a new set of ROIs from a file use 'Load' command name. To change the default directory use 'Options Directories'.

3.4.4. Project

Any ROI defined on a two-dimensional display may be projected onto x, y or a custom defined axis. The result is always stored in one-dimensional spectrum that is automatically created and activated. The command is always applied on the current active display and its active ROI. The projection is automatically updated during the measurement.



Figure 3.4.4.3 The 'X/Y Net Projection' dialog box.

3.4.4.1. Project-X

The resultant one-dimensional display contains at channel x_i number of events that is equal to the sum of all channels in the ROI that have the x-axis coordinate $x=x_i$ (projection onto the x-axis).

3. MDI

3.4.4.2. Project-Y

The resultant one-dimensional display contains at channel x_i number of events that is equal to the sum of all channels in the ROI that have the y-axis coordinate $y=x_i$ (projection onto the y-axis).

3.4.4.3. Project-X/Y

The same as 'ROI Project-X' and 'ROI Project-Y' commands, including a correction for background. The command invokes a dialog box that let you customize the projection. The '*projection axis*' defines axis on which the projection is done. If you select 'Y' for the axis of projection, then '*left background*' and '*right background*' margins define two regions in the x-axis that are used to calculate the background. Similar, if you select 'X' for the axis of projection, then '*upper background*' and '*lower background*' margins define two regions in the y-axis that is used to calculate the background. The command may be applied on the rectangular ROI only.

3.4.5. Keep always

If this option is active (checked) then each ROI, created by MDI, is kept in the memory after closing corresponding display. This means that ROI lasts until you do anything of bellow:

- delete ROI by the 'ROI Delete' command or dialog box,
- delete corresponding display from MNG 'Acquisition Display' dialog.

However, if the display was created by the 'Display Create' command, then corresponding ROI lasts until you delete the display.

3.4.6. Update sum

The command enables automatic updating (refreshing) of an active ROI sum during measurement.

3.4.7. Shape

The command selects Region shape. Each time you define a new Region the currently selected shape is used. In addition, the shape is indicated by the corresponding toolbar bitmap state.

3.5. View menu

The 'View Menu' contains commands that let you customize various program components like the view of display, menu bar, status bar, tool bar.

The majority of properties is defined by the default configuration file ('mdc' extension). Normally, the file is read on the program start time. Therefore, you have to update it if you want to keep the changes. You can do it by the 'Save As' or 'Save' command from the 'View Menu'.

The 'View Menu' commands bring up a corresponding dialog or a property sheet. They are described as follows

3.5.1. Single

The command lets you set properties of currently active one-dimensional histogram. (in order to set colors, metrics and fonts please, refer to the 'Color', 'Desktop' and 'Fonts' commands respectively).

The command opens the ‘*Single Display Properties*’ property sheet that contains a set of dialog boxes, one for each group of common properties. Almost all of the dialog boxes feature the same two buttons: ‘*Set All To Default*’ and ‘*Set All As Default*’. The former causes that any new opened display has the same properties as those selected by the dialog box. In addition, to use the same properties in the next MDI session apply the ‘*View Save*’ or the ‘*View SaveAs*’ command. The last button sets all properties listed in the dialog box to their default values.

Scale

The ‘Scale’ dialog box sets the vertical (count) axis scale.

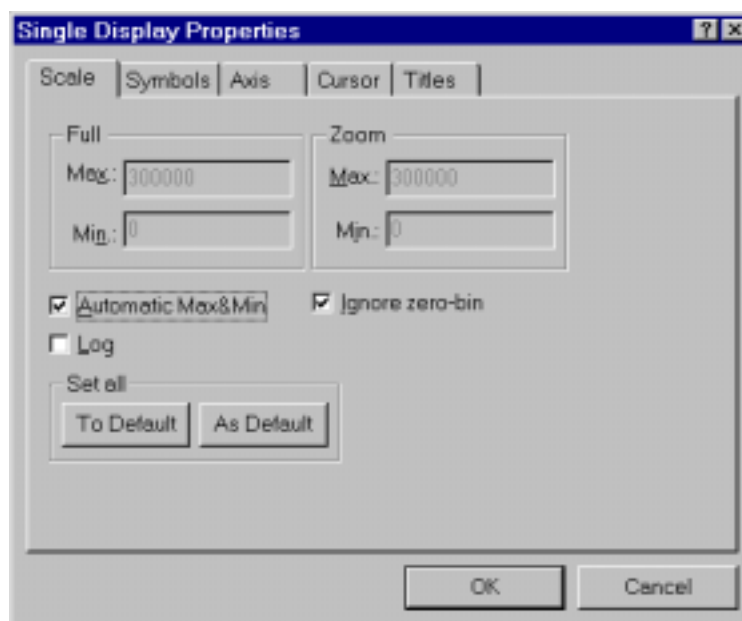


Figure 3.5.1.1 The Single ‘Scale’ dialog box.

Each one-dimensional histogram features two scales. The ‘*Full*’ scale is always associated with all channels in the histogram. The ‘*Zoom*’ scale refers to the channels that are currently shown by the display. If the ‘*Automatic Max&Min*’ box is checked then the program calculates both scale ranges automatically. For linear scale, this is done on such a way that maximum is a little bit greater than the highest number of counts and minimum is set to zero. For the logarithmic scale, the maximum is rounded to the closest power of ten and minimum set to one. To define custom ranges you should unchecked above box. Still, the program adjusts logarithmic scale if the selected minimum and maximum are not power of ten. If the ‘*Log*’ box is checked then the scale is logarithmic, otherwise it is linear.

Some acquisition boards use the first channel, here referred as the ‘zero-bin’, for special purpose. If the ‘*Ignore zero bin*’; is checked this channel is not shown, therefore, the scale is not adjusted using its content.

You can also adjust the ‘*Zoom*’ scale using the thumb of the vertical scroll bar that is placed on the right side of the histogram. The top and bottom of the bar corresponds to the maximum and minimum of the full scale respectively. The position of the thumb corresponds to the maximum of the zoom-scale. Therefore, by moving the thumb, you can adjust the maximum of the visible part to the maximum of the entire histogram.

For the ‘*Set All To Default*’ and the ‘*Set All As Default*’ buttons refer to this property sheet introduction.

Symbols

All counts in the spectra are drawn by the same type of symbol. Using this dialog box, you can select the type They are listed in the ‘*Type*’ combo box. The symbol may be a line that connects counts or a pattern that represents counts.

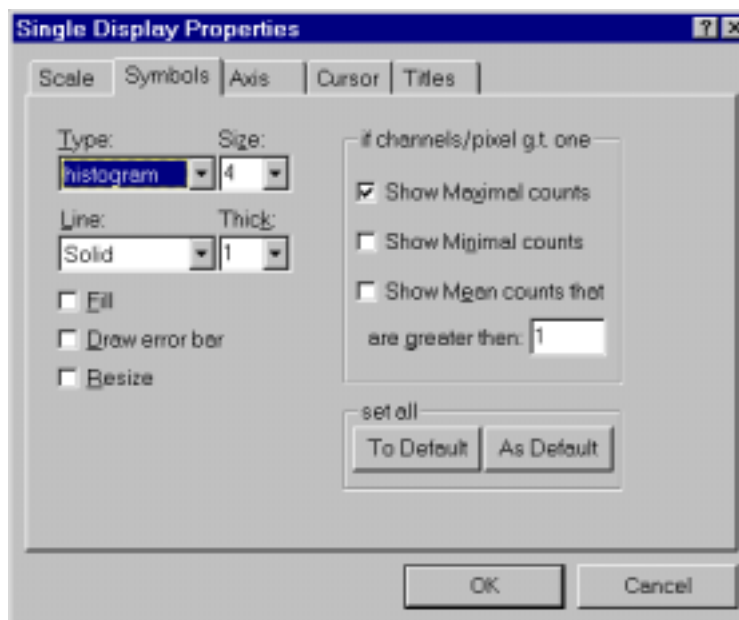


Figure 3.5.1.2 The Single ‘*Symbols*’ dialog box.

For each line-type symbol, you can define its thick using the ‘*Thick*’ combo box. The following line-types are possible:

- *histogram* connects two successive counts by one horizontal and one vertical line;
- *spline I* connects two successive counts directly with a straight line;
- *line* produces vertical lines that go from the x-axis to each count

For a pattern-type symbol, you can define its size using the ‘*Size*’ combo box and pen thick by the ‘*Thick*’ combo box. In addition, the pattern-symbols can be filled. To enable it, check the ‘*Fill*’ box. The following pattern-symbols are possible:

- *rectangle*
- *circle*
- *triangle up*
- *triangle down*
- *cross*
- *snow-flake*

If you check the ‘*Error bar*’ box a vertical line is drawn over the symbol. The line length is equal to the square root of the number of events.

If the visible spectrum contains more channels than the corresponding pixels on the screen, it may happen that two or more channels share the same pixel. In that case, the program may draw maximal, mean value of the counts or minimal counts. This is defined by *'Max Counts'*, *'Mean Counts'* or *'Min Counts'* respectively. The channels which content is below this *'Threshold'* value is not taken into account for the mean value calculation.

For the *'Set All To Default'* and the *'Set All As Default'* buttons refer to this property sheet introduction.

Axis

The *'Axis'* dialog box sets the graphical properties of the y, also called vertical or counts, and x, also called horizontal or channels, axis. The axis contains the following graphical objects: frame, grid, ticks, tick numbers and titles.

The axis *'Frame'* is a rectangle drawn around the histogram. You can disable drawing of one or more its sides using the *'Frame Style'* dialog box. The *'Frame'* is drawn by a pen which properties are defined by the *'Frame Line'* and *'Frame Tick'* controls.

The axis *'Grid'* is a set of equidistant, horizontal and/or vertical lines that are perpendicular to the axes: x-grid to the x-axis and y-grid to the y-axis. The *'Grid'* is drawn by a pen, which line style and thick you can set with *'Grid Line'* and the *'Grid Tick'* controls respectively. This is done only if the *'Draw'* box is checked.

The *'Ticks'* are short lines at the axes that mark the distance in axis units. They can be drawn on each of the four frame borders separately, by checking the *'Top'*, *'Bottom'*, *'Left'* and *'Right'* boxes. Their thickness and size are defined by the *'Tick Thick'* and *'Thick Size'* controls, respectively. The *'Ticks'* may be drawn inside or outside of the frame, which is defined by the corresponding combo controls. The *'Ticks'* are labeled by numbers, which are shown only if the *'Ticks'* are enabled.

The axis *'Titles'* are defined by the *'Titles'* dialog box of the same property sheet. The colors are defined by the *'Color'* dialog box and font by the *'Fonts'* dialog box. You can use the *'Desktop'* command to customize axis size and position.

For the *'Set All To Default'* and the *'Set All As Default'* buttons refer to this property sheet introduction.

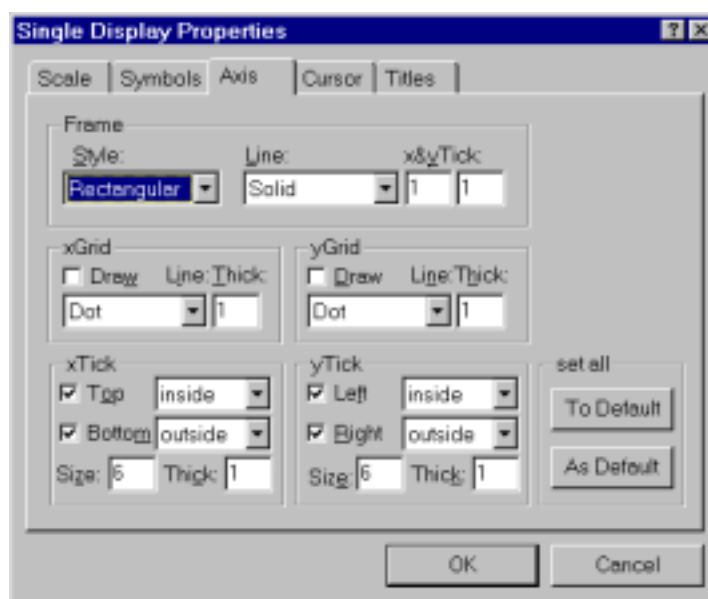


Figure 3.5.1.3 The Single *'Axis'* dialog box.

Cursor

The ‘Cursor’ dialog box sets the properties of the histogram cursor. It is used to read number of counts at a particular channel. You can move it by the mouse or

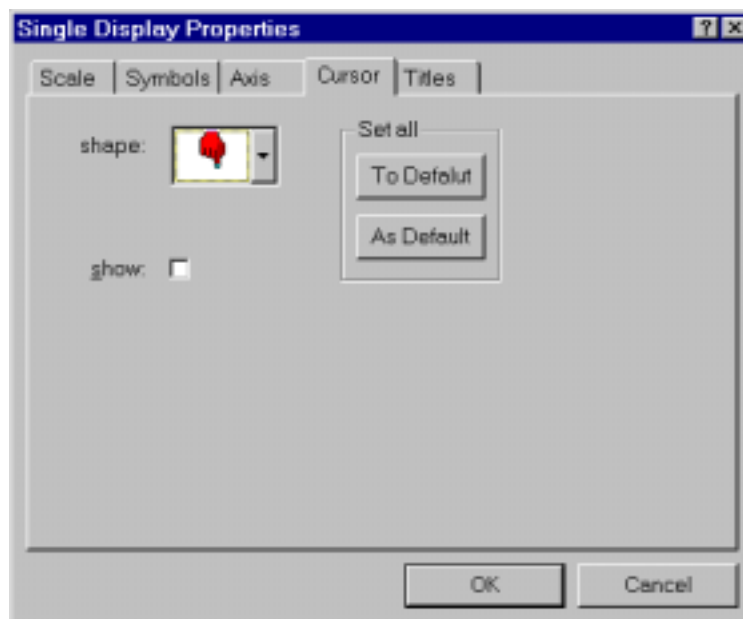


Figure 3.5.1.4 The Single ‘Cursor’ dialog box.

keyboard. Each time you move the mouse cursor and click the right button, the histogram cursor appears at the corresponding channel. If there are more channels than pixels, the program automatically selects a corresponding channel, which may not be the required channel. In that case, you should use keyboard to move the histogram cursor.

To select the cursor shape, use the ‘*shape*’ combo box. To remove the cursor, uncheck the ‘*show*’ box. Note that each time you click by the mouse on the spectrum and the count is inside the visible scale, the cursor appears again.

For the ‘*Set All To Default*’ and the ‘*Set All As Default*’ buttons refer to this property sheet introduction.

Titles

The ‘Title’ dialog box sets histogram title and its axes labels. Note that the new titles have no effect on the original displays created by the MNG.

3.5.2. Map

The command lets you set properties of currently active one-dimensional histogram. (for setting colors, metrics and fonts please, refer to the ‘Color’, ‘Desktop’ and ‘Fonts’ commands respectively).

The command opens the ‘Single Display Properties’ property sheet that contains a set of dialog boxes, one for each group of common properties.

Almost all of its dialog boxes feature the same two buttons: ‘*Set All To Default*’ and ‘*Set All As Default*’. The former causes that any new opened display has the same properties as those selected by the dialog box. In addition, to use the same properties in the next MDI session apply the ‘*View Save*’ or the ‘*View SaveAs*’

command. The last button sets all properties listed in the dialog box to their default values.

Scale

The 'Scale' dialog box sets the vertical, also called z or count, axis scale. The scale is represented with colors that can change either in the logarithmic or linear grade. This is defined by the 'Log Color' box.

Each two-dimensional histogram features two scales. The 'Full' scale, that is always associated with all channels in the histogram and the 'Zoom' scale that refers to the channels that are currently shown by the display. In the both cases, if the 'Automatic Max&Min' box is checked then program calculates scale ranges automatically. For linear scale, this is done on such a way that maximum is a little bit greater than the highest number of counts and minimum is set to zero. For the logarithmic scale, the maximum is rounded to the closest power of ten and minimum set to one. To define custom ranges you should uncheck above box. Still, the program adjusts logarithmic scale if the selected minimum and maximum are not power of ten.

Normally, the scale range is mapped into the color indexes on such a way that first and last color index corresponds to the minimum and maximum of the 'Full' scale respectively. If you want to use the 'Zoom' scale as the referent scale for the color mapping, then you should check the 'Adapt color to Zoom'. To adjust the color indexes use the 'View Colors' command. The referent scale, together with corresponding colors are drawn on the right side of the map.

You can also adjust the zoom scale using the thumb of the vertical scroll bar that is placed on the right side of the histogram. The top and bottom of the bar corresponds to the maximum and minimum of the full scale respectively. The position of the thumb corresponds to the maximum of the zoom-scale. Therefore, by moving the thumb, you can adjust the maximum of the visible part to the maximum of the entire histogram. Note that only counts in the range of the zoom scale are shown.

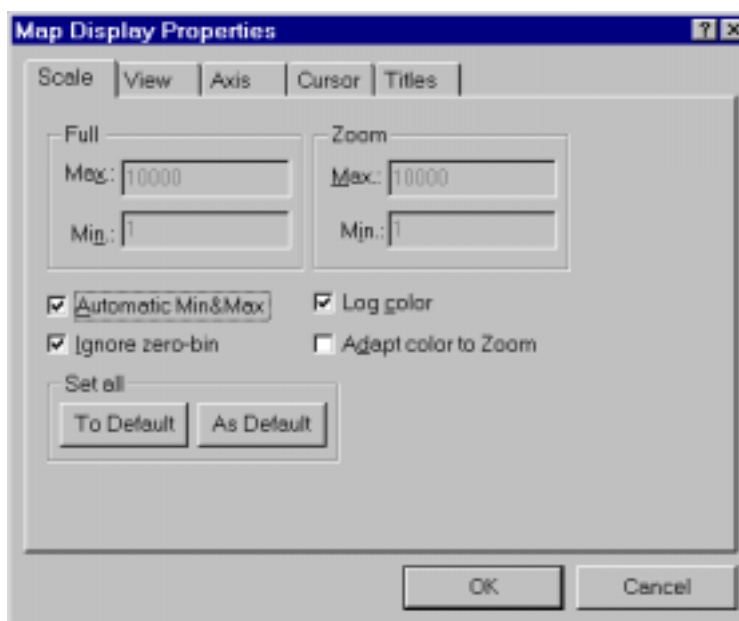


Figure 3.5.2.2 The Map 'View' dialog box.

For the ‘*Set All To Default*’ and the ‘*Set All As Default*’ buttons refer to the introduction into this property sheet.

Views

Maps are either drawn in a ‘*bitmap*’ or ‘*vector*’ graphical mode, which is defined by the ‘*Graphical mode*’ combo box. In each of modes, counts are shown by the same symbol but in different colors. The symbol can be selected by the ‘*Symbol type*’ combo box.

In the ‘*bitmap*’ mode, maps are drawn on the surfaces with a fixed dimension, which is usually different than the surfaces used for their displaying. This dimension is defined by the ‘*Bitmap resolution*’. The only exceptions are maps drawn by the ‘*dot*’ symbol. In this case, the horizontal and vertical dimensions are equal to the number of channels in the x and y direction respectively. When displaying map drawn in the ‘*bitmap*’ mode, the program maps drawing surface (source) onto the displaying surface (destination). This means that pixels on the source surface are stretched into rectangles/lines or removed from the bitmap, depending on the sizes of the drawing and displaying surfaces. In any case, program does not need to redraw source bitmap each time the displaying surface changes its size.

In the ‘*vector*’ mode, the drawing surface is equal to the displaying surface. Therefore, each time the display window changes its size, the program resizes and redraws the whole map. After the full map is drawn, the program copies it onto the displaying surface.

The ‘*dot*’ symbol type combined with ‘*bitmap*’ mode is recommended as the standard, because such maps are drawn fastest and it is easy to manipulate with them. The checkbox ‘*Invert ROI pixel*’ gives a choice how the ROIs are displayed in a map: if checked then the complete ROI is displayed in inverted colors, otherwise only the boundary lines are drawn.

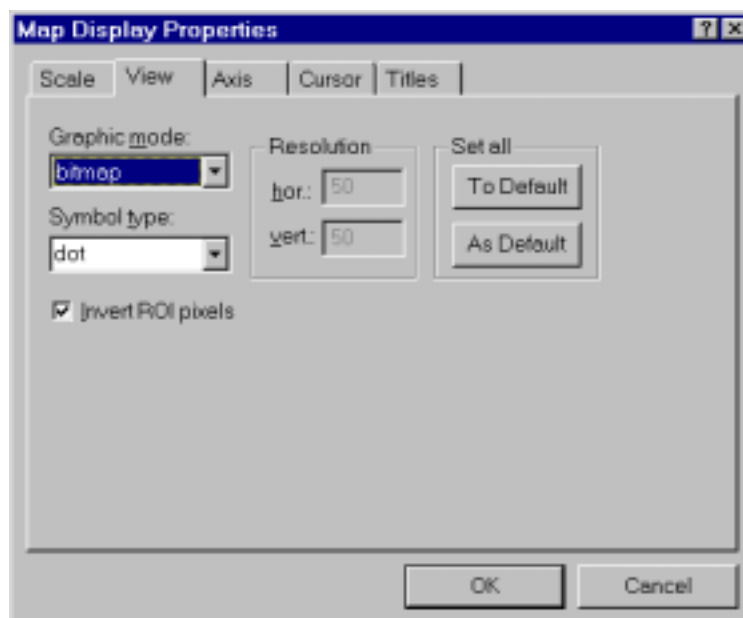


Figure 3.5.2.2 The Map ‘View’ dialog box.

For the '*Set All To Default*' and the '*Set All As Default*' buttons refer to the introduction into this property sheet.

Cursor

See the '*View...Single...Cursor*'

Titles

See the '*View...Single...Titles*'

3.5.3. Isometric

The command lets you set properties of currently active two-dimensional histogram drawn in the isometric projection. The term 'isometric' refers here to several single spectra drawn behind each other. For setting colors, metrics and fonts please, refer to the '*Color*', '*Desktop*' and '*Fonts*' commands respectively.

The command opens the '*Isometric Display Properties*' property sheet that contains a set of dialog boxes, one for each group of common properties.

Almost all of its dialog boxes feature the same two buttons: '*Set All To Default*' and '*Set All As Default*'. The former causes that any new opened display has the same properties as those selected by the dialog box. In addition, to use the same properties in the next MDI session apply the '*View Save*' or the '*View SaveAs*' command. The last button sets all properties listed in the dialog box to their default values.

Scale

The '*Scale*' dialog box sets the vertical (also called z or count) axis scale.

Each two-dimensional histogram in the isometric projection features two scales. The '*Full*' scale, that is always associated with all channels in the histogram and the '*Zoom*' scale that refers to the channels that are currently shown by the display.

If the '*Log*' box is checked, than the scale counts are drawn at vertical positions that change in the logarithmic grade, otherwise in linear. In addition, the symbols may be drawn in colors. The symbol's color scale grade is specified by the '*Log color*' box on the same way as the former. Normally, the color indexes are calculated on such a way that first and last color index corresponds to the minimum and maximum of the '*Full*' scale respectively. If you want to use the zoom scale as the referent scale for the color mapping, then you should check the '*Adapt color to Zoom*'. To adjust the color indexes use the '*View Colors*' command.

If the '*Automatic Max&Min*' box is checked, the program calculates scale ranges automatically. For linear scale, this is done on such a way that maximum is a little bit greater than the highest number of counts and minimum is set to zero. For the logarithmic scale, the maximum is rounded to the closest power of ten and minimum set to one. To define custom ranges you should unchecked above box. Still, the program adjusts logarithmic scale if the selected minimum and maximum are not power of ten.

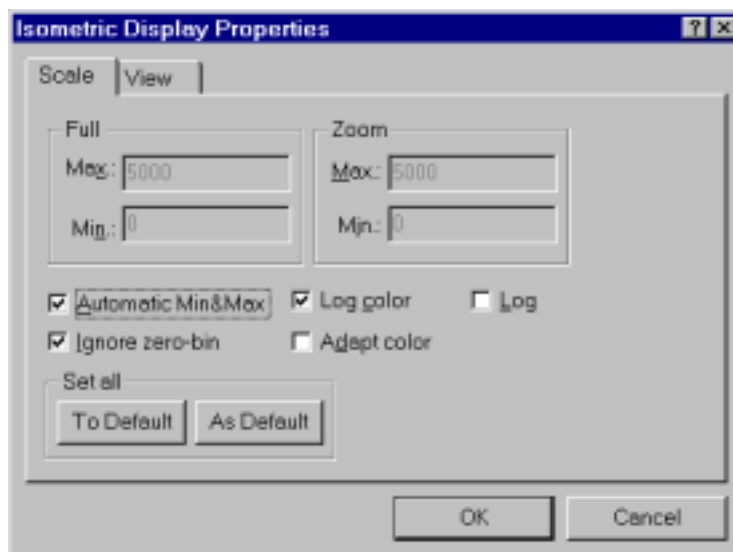


Figure 3.5.3.1 The Isometric 'Scale' dialog box.

View

The 'View' dialog box lets you set symbols used to draw the display and viewer perspective as well.

The '*precession*' defines angle around the vertical axis can be chosen in multiples of 90 degrees.

The '*tilt*' defines angle between the x and y-axis. It can be chosen in the range from 15 to 89 degrees.

The '*height*' specifies the percentage of the z-axis length respective to the whole drawing, it can be entered between 0 and 99.

With '*show hidden*' it can be specified whether the hidden parts are drawn. If '*b/w*' is checked, the spectra are drawn black and white

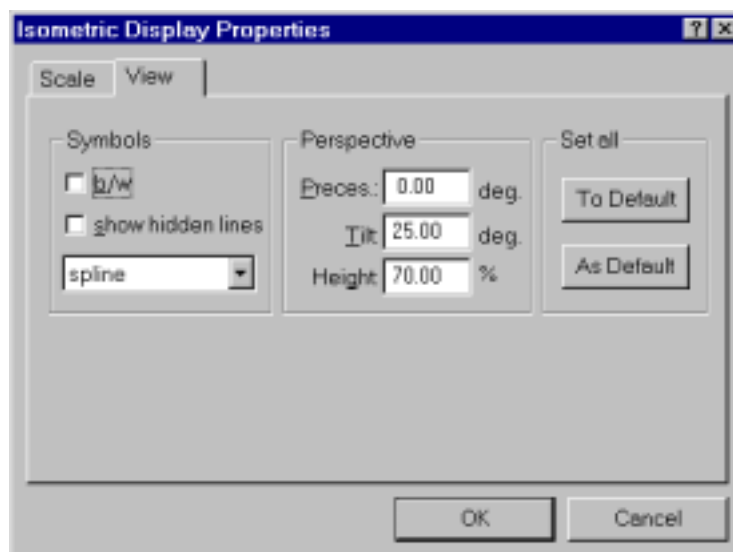


Figure 3.5.3.2 The Isometric 'View' dialog box.

3.5.4. Colors

The command adjusts all colors used by the program.

It brings up the ‘Colors’ dialog box than enable you to define colors. All displays are drawn with the same color scheme.

To handle colors, the program uses palette with N items (in the present version $N \leq 256$). The number of items may be defined by using the ‘*Palette*’ list box. Twenty items are static and can’t be changed. They are divided into two parts: the first half are placed at the bottom while the rest are at the top of the palette. Usually the static colors are used to draw all graphic objects except the map color scale.

The central part of color palette, that is, from item 10 up to the $N-10$ is normally used to draw gradual color scale in two-dimensional displays. Therefore, this part of palette is called map colors.

The ‘*Save*’ command in the Colors dialog box saves the color setup into the default color file. The default file is loaded automatically each time you start the program. The file becomes default if you load it by the ‘*Load*’ command and then apply the ‘*UseAsDef*’ command. To save the color setup in a user defined file, use the ‘*SaveAs...*’ command.. All color files has default extension ‘col’ and they are stored in the ..\bin subdirectory.

Map colors

The number of colors used to draw map is defined by the ‘*Map levels*’ item and it can be up to $N - 20$, where N is the total number of palette entries define by the ‘*Palette*’ list. By default, the first level corresponds to the smallest number of counts and to the first color in the color scale, that is, 11th in the palette. You can reorder it and assign any color to any level. To do this use the MAP entries in the ‘*Color Select for*’ list box. For example, MAP001 corresponds to the first level and usually it is assign to the first color of the colors scale (11th in the palette).

In the case of the fine grade color scale, it may be problem to distinguish visually zero counts from others. For this purpose, the program uses the ‘*Map background*’ for the first-level color. It is listed in the ‘*Color select for*’ list in the ‘Colors’ dialog box. Usually this color is assigned to one of the static colors.

All map color-scale colors, that is, palette entries from 10 to $N-10$, can be defined on one of the two following ways:

- by a list of RGB triplets: for each color three integers in the interval [0..255]
- by three functions that show color component relative intensity versus color component relative index (the components are R, G and B).

In the both cases the color information are stored in the color scale files that has name ‘scale\$.rgb’ where \$ is indexed from 00 to 99. These files are listed in the ‘*Map scale*’ list box. The file that contains data currently used by the program is selected when the dialog is just brought up. Actually, the program does not use the color scale files directly. Instead of that, it copies all information in the color file (*.col) in appropriate format. In addition, it keeps the color scale file index as a reference to the color scale information. Index 99 indicates that none of the existing color scale files is used to build the color scale. This means that previously selected file is invalid or does not exist and you should select a new valid file.

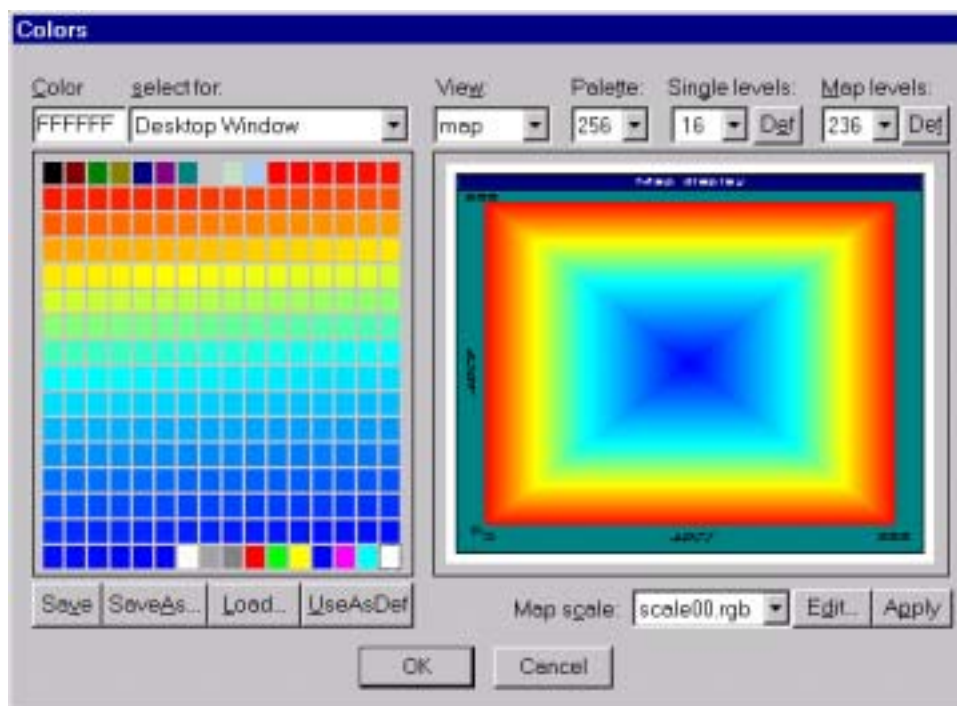


Figure 3.5.4.1. The 'Color' dialog box.

To select new full map color scale you should select one of the color scale files listed in the 'Map scale' list box. You can also edit currently selected file by the 'Edit' button and then modify it.

RGB triplet map color-scale format

The RGB triplet colors are stored in text files with the following structure:

The number in the first line should be always one (integer), then follows lines with intensity of R, G and B components, each of them in the range from 0 to 255 (integers) and separated by blank. The bottom of the scale corresponds to the second line (first RGB triplet) while the top of the scale corresponds to the last line (last RGB triplet). The number colors is defined by the number of lines in the file.

In order to edit this file the program automatically starts default Window's editor (notepad). After making changes you should save the file, close the editor and finally press the 'Apply' button.

Spline I map color-sale format

In the spline format, the colors are also represented by the R, G and B components. In addition, the intensities (I) of each component are set of linear functions of the color index (X). The intensity and index are defined in the range [0.1] and the functions are constructed from a finite set of points (up to eleven) in the (X,I) space using linear spline method. The data are stored in corresponding file in the following format:

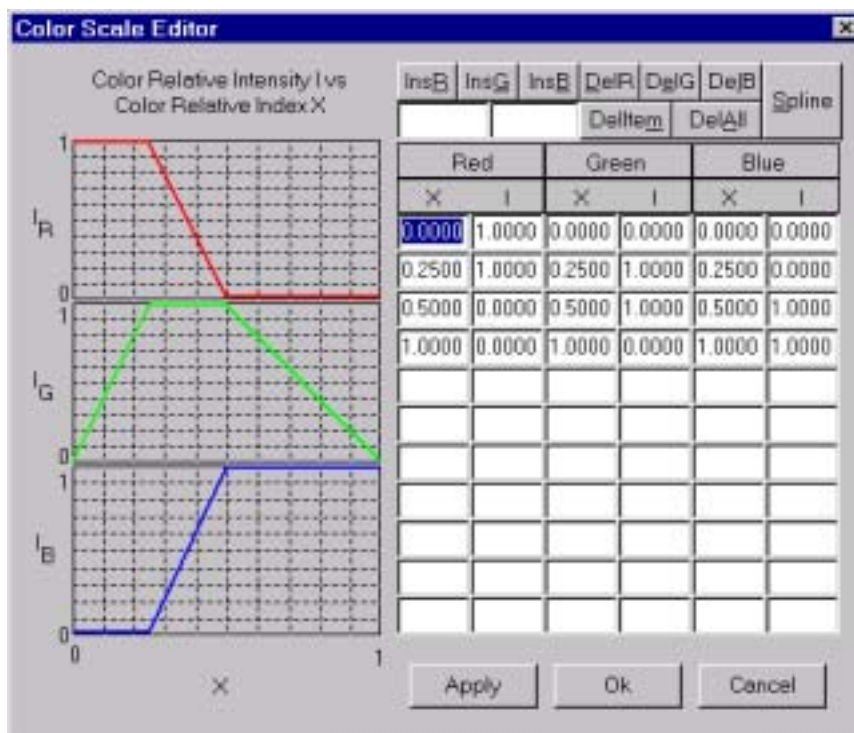


Figure 3.5.4.2. The 'Color Scale Editor' dialog box.

The number in the first line must be zero (integer). The second line contains number of points for color component (integer). Then follow lines that contain declared points, that is, X and I values (two real numbers per line separated by blank). First must be listed data for R components. After the last pair of R, follows the number of points for G and then its point. At the end are similar information for B color component.

You can edit corresponding file with using a standard editor or program's Color scale editor.

The Color scale editor enables you to define appropriate spline shapes on an easy way and show them graphically. It is automatically started upon pressing the 'Edit' button and if color scale is in the spline format.

The dialog features a table of (X,I) coordinates. The X coordinate represents relative color index and must be in increasing order. For each X, a corresponding relative intensity I must be defined. You should enter at least two (X,I) pairs. The first must have the relative index X=0 and the last X=1. The points may be entered directly in the table without taking care of their order.

You can also use two edit controls on the top of the table to define new pair of X and I and then apply one of the following buttons: 'InsR', 'InsG' or 'InsB'. To delete an entry, first click on it by the mouse and then apply the 'DeleteItem' button. To delete all items of the R, G or B color-component, use the 'DelR', 'DelG' or 'DelB' respectively. The 'DelAll' buttons deletes all entries in the table.

After applying the 'Spline' button the dialog sorts all items and draws all curves.

Single display colors

The program supports up to 100 ROIs per each one-dimensional histogram (display). The ROIs have zero-based index and they are indicated on the spectrum by

colors. For example, all pixels in the spectrum that belong to the same ROI are drawn in the same color.

To define the number of the colors use the ‘*Single levels*’ list. This number may be up to $N-4$ and corresponding colors may be any from the palette. By default, the number is equal to sixteen and colors are selected from the twenty static colors. The one of the rest static color is used to draw single display background, and the three other to draw foreground histograms: maximum, mean and minimum. For example, if the drawing surface has less pixels then there are channels in the visible part of the histogram, the program may display channels that contains the largest number of events (maximum), the smallest (minimum) or mean value (mean).

For each ROI there is one item in the ‘*Color select for*’ list box. They are labeled with ‘*ROI*’ and have two digits index. The ROIs take the color with period equal the number selected for the ‘*Single levels*’ To define color for ROI i you should select the ROI_i from the list and then click on the color you want to choose. It is convenient to use only static colors for the ROIs otherwise, any change in the map colors affects the single histograms view.

Colors for other drawing objects

Apart from the ROIs and map color scale, display may have several other graphical objects for which colors can be adjusted. They are listed in the ‘*Color Select for*’ dialog box. It is common to assign static color for these items.

3.5.5. Fonts

The command sets font that is used for drawing texts related to the histograms. It brings the ‘Font’ dialog box that lets you select font properties

The same type of font is used for all histograms. The size you select here corresponds to the referent histogram’s window that has 701 pixels in width and 393 pixels in height. These dimensions are corrected for the margins that can be selected in the ‘View ... Desktop’ and then normalized to the actual window size.

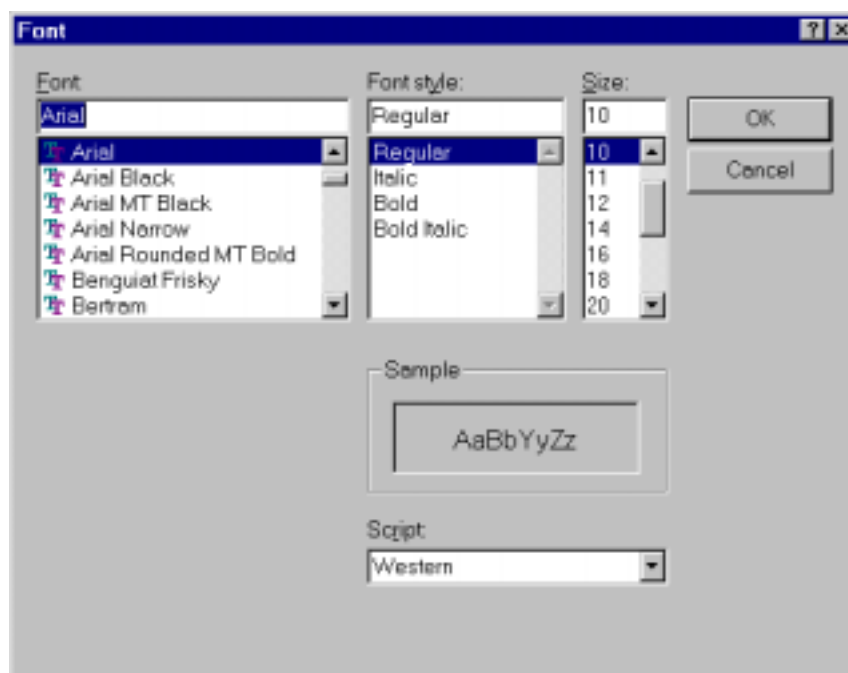


Figure 3.5.5.1 The ‘Font’ dialog box.

3.5.6. Desktop

The *'Desktop'* command customizes size and position of display objects.

The *'Desktop'* refers to the entire client area of the window associated to any histogram. This window is called parent window. It owns the following objects that are called bars:



Figure 3.5.6.1 The *'Desktop'* popup menu.

- *histogram* bar: contains axis frame, grid, ticks and count symbols
- *axis scale* bar: contains tick numbers
- *axis title* bar: contains axes title
- *color scale* bar: shows vertical scale in corresponding colors
- *counter* bar: contains histogram related information (counts, sum...)
- *text* bar: contains static text that usually refer to one of the counter bars

The *'Desktop'* objects have various properties. The size and position respects to the parent window are common properties to all of them. In addition, they can have 3D-look style or simply you can hide each of them. Before applying the *'Desktop'* command, it is recommended to cascade all windows by the *'Window...Cascade'* command. This gives the default dimension to all windows.

Upon applying the *'Desktop'* command, the toolbar is removed and menu bar contains *'Exit'* command only. Now you have to select display that you want to customize. Then, move the mouse cursor inside the window at a position where nothing is drawn and click on the right button. It opens a popup menu. Select there the *'Open'* item. Now, if you click by mouse one of bars, than it is selected. Any selected bar can be moved or resized as ordinary window. To select a group of bars, you should draw a rectangle around all of them (or at least to intersects each of them with this rectangle). You can do it by the left mouse button. It is possible to move a group of bars together, but not resize more then one bar at time.

To edit bar properties, first select the bar and then apply *'Edit'* command from the *'Desktop'* popup menu. An equivalent approach is to double-click on the bar. This brings up a property sheet. If no bars are selected and you applied the *'Edit'* command from the popup menu, or you double clicked inside the desktop area but not on any bar, then the *'Desktop'* properties sheet brings up.

After setting all properties, you can save them as default by applying the *'Save'* command. This means that on the run time, the program takes the last saved information to build the corresponding type of display. This is done on such a way that all bars are drawn in the proportional size

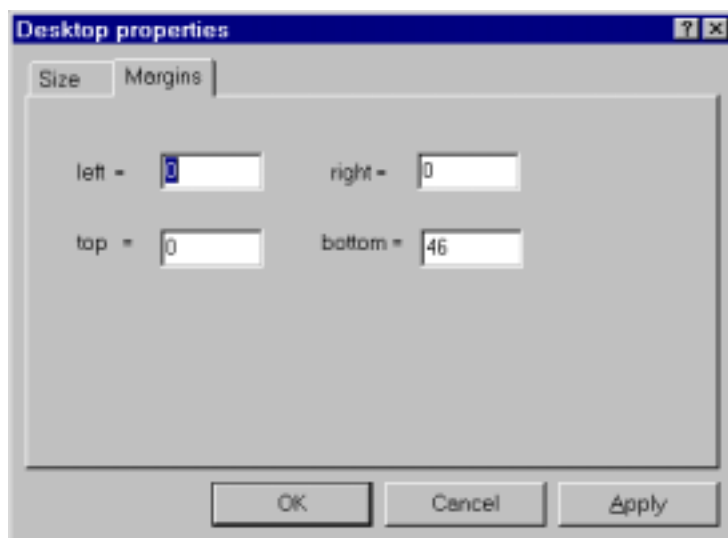


Figure 3.5.6.2 The 'Desktop Margins' dialog box.

. The 'single.dks', 'map.dks' and 'iso.dsk' files are default for single, map and 3d histograms, respectively. If the program can't find the default files it uses built-in properties. Note that initially all displays of the same type have bars with equal properties. You can also store properties in a different file and load it later. To do this, use the 'Load' and 'SaveAs' command.

The properties of each bar are described as follows.

Desktop Properties

The 'Desktop' properties are grouped into two dialog boxes that are described as follows.

The 'Size' dialog defines dimension of the window you have opened. The values listed here are updated automatically, each time you resize window using the mouse. In addition, each time you change these values, the window is automatically resized.

The 'Margin' dialog defines borders outside of which no bars are supposed to be drawn except counters and texts. For example, if you want to put counters and texts to the right side of the display, then set the right margin equal to the width of the widest counter or text. It is assumed that other bars are arranged on such a way that they do not overlap margins.

Counter bar properties

The 'Counter' bar properties are grouped into few dialog boxes that are described as follows.

The 'Size&Position' dialog box define dimensions in pixels and position relative to the left top corner of the parent window (desktop). Each time you resize or move a bar using the mouse, these values are updated automatically.

The 'Style' dialog box defines how the bar is drawn. The bar is not drawn if the 'Visible' box is unchecked. The border is defined by checking one or more of the following boxes: 'Thin Border', 'Sunken' and the 'Static edge'. In the first case a thin border is drawn around the bar, while two other options produces a 'three-dimensional' border. If the 'Resize' box is checked, then the size of the bar and its text are adjusted to the size of the parent window. In addition to the resizing, you can set the moving properties. They are listed in the corresponding list box. If 'do not move' item is selected, then the position of control is fixed. This option is normally used to keep the bar at the top of the parent window or on its left edge. If you want to keep the bar on the bottom of the window then use the 'move vertical' option. In that case, the bar is shifted up or down each time the parent window decreases or increases its vertical dimension. The shift is equal to the increment of the window size. In any case, the horizontal bar position is not changed. The meaning of the 'move horizontal' is the same as above but applying it to the horizontal direction. This is normally used if you want to keep the bar at the right edge of the parent window. The above three moving options are usually applied on the bar that has fixed size, that is, the 'Resize' box is unchecked. In this case, it is useful to define desktop margins and set position of the bar inside the margins. For example, you can place a bar on the bottom of the

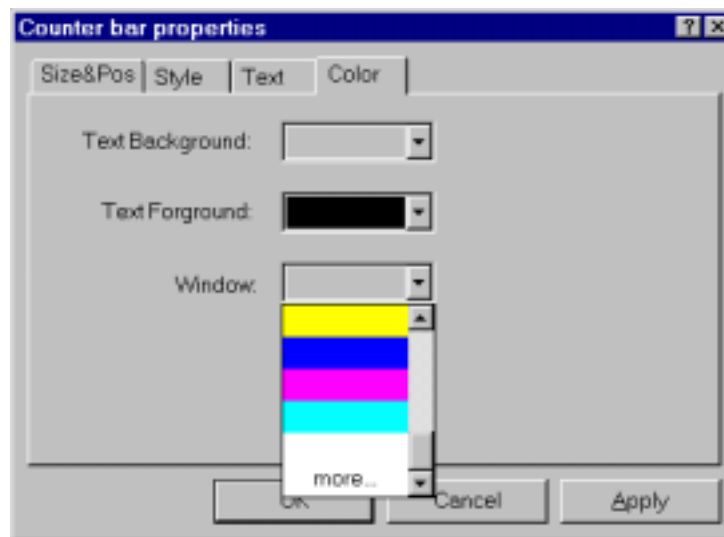


Figure 3.5.6.3 The 'Counter Bar Color' dialog box.

desktop, set bottom margin to the bar's height and use the 'move vertical' style. The 'move proportional' moves bar on such a way that it has always the same position relative to the parent window size. This is normally used if the bar resizing is enabled, that is, the 'Resize' box is checked. The text shown by the bar can be formatted using the styles listed in the 'Text' group.

The 'Text' dialog is normally not used, because the counter shows always a predefined value.

The 'Color' dialog box lets you to define colors of the text, its background and the window associated with bar. The dialog lists static controls that are normally used by bars. If you select the last item in the list, a new dialog is brought up. It contains colors that are used for drawing maps. Therefore each time you change the map palette this will affect the bar color.

Text bar properties

The 'Text bar' properties are similar to the counter bar. Therefore, you can refer to the 'Counter bar properties' for majority of properties. Here are discussed only additional properties.

The text shown by this bar can be multi or single line. If you check the 'Single-line' box than bar shows the whole text in one line. In that case, you can align text in vertical and horizontal direction. Otherwise, the text is shown in more lines and you can't align it in the vertical direction. The 'Break words' box is used with multi-line texts. If the box is checked then lines are automatically broken between words if a word would extend past the edge of the bar.

The text of the 'Text bar' is defined by the 'Text' dialog box. It features multi-line editor control, which enables you to enter up to fifty characters.

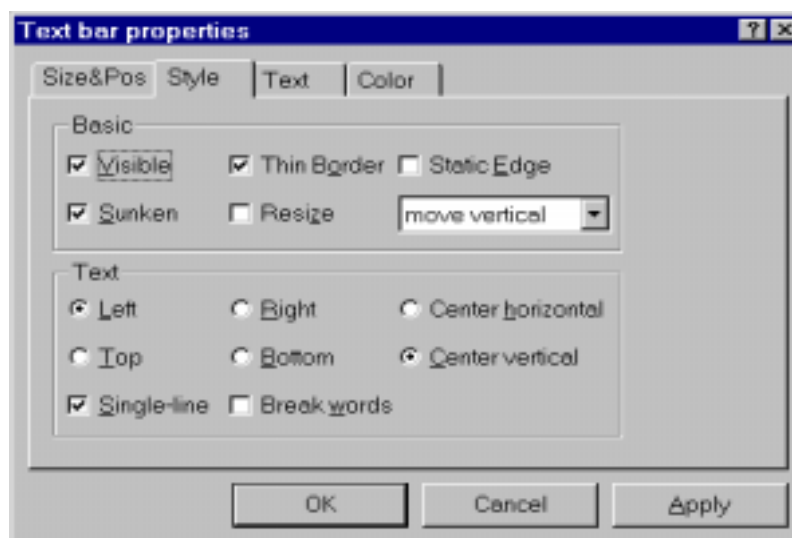


Figure 3.5.6.4 The 'Text bar Style' dialog box.

Color bar properties

See the 'Counter bar properties' for more information about the 'Size&Pos' and 'Style' dialog boxes. To change colors in the 'Color bar' see the 'View Colors' command. See also the 'View Map' command.

Scale bar properties

See the 'Counter bar properties' for more information about the 'Size&Pos' and 'Style' dialog boxes. To change the numbers in the scale use 'View Single' or 'View Map' command. To change the colors use the 'View Colors' command

Display bar properties

See the 'Counter bar properties' for more information about the 'Size&Pos' and 'Style' dialog boxes.. To change the colors use the 'View Colors' command

3.5.7. Directories

For more information see 'MNG System Properties Directories'.

3.5.8. Tool bar

The command Sets the 'Toolbar' view.

It opens 'Tool Bar' dialog that let you customizes your 'Tool bar'. The 'Tool bar' is a widow placed below the main window title and 'Menu bar' that contains a set of bitmaps. The bitmaps allow you to apply 'Menu' command directly by clicking on the corresponding bitmap. Not all commands have the corresponding bitmap.

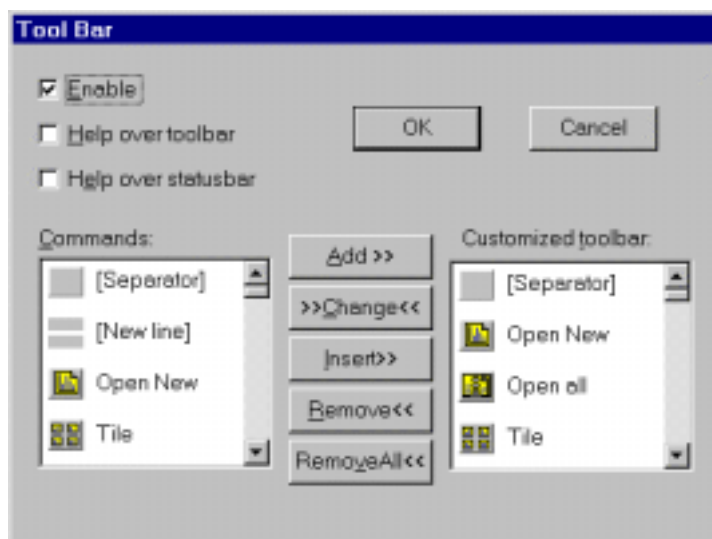


Figure 3.5.8.1 The 'Tool Bar' dialog box.

To enable the 'Tool bar' you should check the 'Enable' box. If you want the program displays a single line of descriptive text each time you put cursor over it, you should check the 'Help over toolbar' button. A little more information is possible to have on the 'Status bar'. To enable that feature you should check the 'Help over statusbar' button. The 'Status bar' is a narrow window placed on the bottom of the program window.

The list of commands supported by the toolbar is shown by the 'Customized toolbar' list. All available commands that are possible to use in the toolbar are listed in the 'Command' list. The toolbar bitmaps appear in the toolbar window in the same order as they are listed in the 'Customized toolbar list'. If you want to put a place between two bitmaps, insert one or more 'Separator' bitmaps. To place a bitmap in the next row, insert the 'New Line' bitmap.

3.5.9. Menu bar

The command sets Menu bar view.

The menu bar is a window positioned bellow the program title that contains commands. The Menu Bar command runs 'Menu Bar Editor' application that let you customize Menu bar.

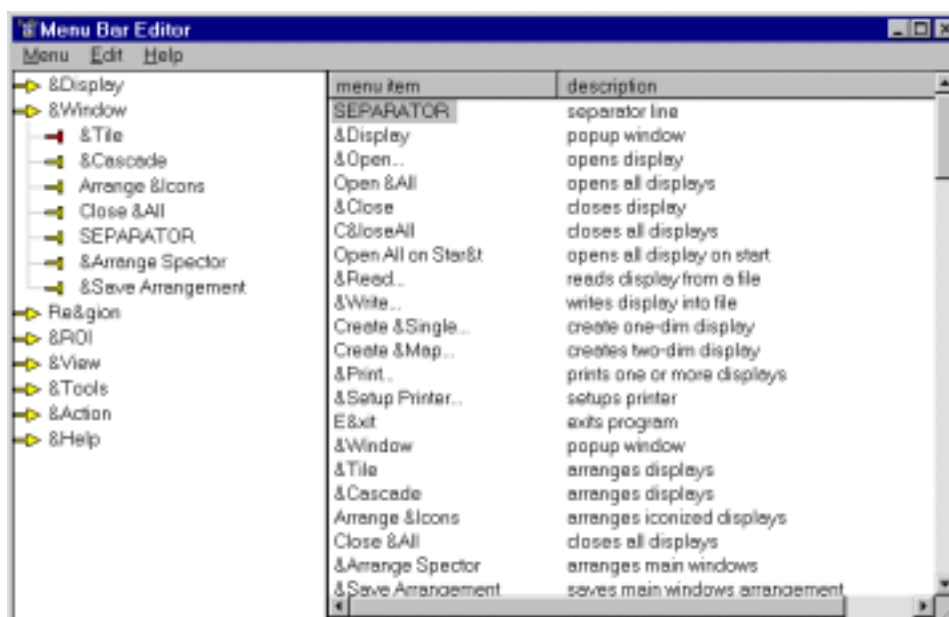


Figure 3.5.9.1. The Menu bar editor.

The 'Menu Bar Editor' features two windows: a tree view window on the left side that contains defined menu items and a list view window on the right side that contains a list of all available commands.

On the start, the tree view shows currently used menu that is stored in the text file 'mdi.mnu'. You can modify it using the commands listed in the 'Edit' popup menu item. For example, if you want to add one item in the menu use the following procedure. First, select item in the tree view after which you want to make insertion. Then select item in the list view, which you want to insert. Finally, apply the 'Edit Insert' command

If you want to apply modification, use the 'File...Apply' command. It automatically updates 'mdi.mnu' file and sends signal to MDI to update its menu bar according to the new settings.

The Menu bar is built from tree basic types of items that is defined by name and position:

- *command menu item* refers to one of MDI commands. In the tree view, it is pointed by an inverted 'T' icon. In the list view, a short command description is associated with it.
- *separator menu item* appears in MDI as a line between two command items. It has name SEPARATOR in both lists.
- *popup menu item* appears in MDI as a popup window that contains one or more command items, separator items or additional popup menu items. In the list view, it is pointed by an 'arrow' icon. When double clicked it shrinks showing sub-items. In the list view the 'window popup' description is associated the item.

Menu Open

The command reads menu from a file. It brings up the 'Open' dialog box, that lets you choose a file with 'mnu' extension and associated text format. If the file is successfully opened and read, its name appears on the main window title bar.

3. MDI

Menu Save

The command saves menu into a file which name is written on the main window title bar. If there is no file name, the program invokes the 'SaveAs' command

Menu SaveAs

The command saves menu into a user-selected file. It brings up the 'Save As' dialog box that lets you define file name.

Menu Apply

The command saves menu item shown by the three view into the default file named 'mdi.mnu'. In addition, it updates MDI menu.

Menu Exit

The command exits program.

Edit Delete

The command deletes currently selected item in the tree view window. To invoke command you can also press the 'Del' key.

Edit Delete All

The command deletes all items in the tree view. You can also press 'Shift' and 'Del' keys simultaneously to run the command.

Insert

The command adds item that is selected in the list view window, immediately after the item that is selected in the tree view. If the last is popup menu, the former becomes its sub-item, that is one level higher. To run the command you can also use the 'Ins' key.

Add

The command adds item that is selected in the list view window, immediately after the item that is selected in the tree view. The inserted item has always the same level as the reference. To run the command you can also use the 'Shift' and 'Ins' keys simultaneously.

Restore

The command restores menu to the original form, that is taken immediately after the 'Open' or 'Default' command. The associated accelerator key is 'Esc'.

Default

The command builds default menu. The default menu contains all available commands. It is good starting point to build a new menu. The commands can be run by the 'Home' key.

3.5.10. Status bar

The command enables or disables the 'Status bar'.

The 'Status bar' is a narrow window placed on the bottom of the program window. It contains some useful information like acquisition status, available memory, and scale properties. In addition, it displays a short help message what the toolbar icon does.

3.5.11. Updating

The command sets mode and period of the displays refreshing during the measurement. It brings up 'Updating Properties' dialog box.



Figure 3.5.11.1 The 'Updating Properties' dialog box

The MDI uses only one thread to draw all displays. The displays are drawn in successive order: after the last display in the queue was drawn, the system again starts to draw the first one. You can select the period between successive drawing of the first display in the queue. This is done by the 'Update Rate'. The value may be increased by the program if drawing task takes more time. In the case of the XRFM application, the updating is done usually after each motor step and can be adjusted by the corresponding edit control. By the 'Updating Mode' option, you can put into the drawing queue either active display or all displays.

3.5.12. Show restart

The command enable/disables 'Action Restart' icon in the toolbar. This icon or the corresponding menu item in the action menu is useful during the setup for a fast clear and restart, but may be dangerous in a longer measurement. Therefore, it can be disabled here.

3.5.13. Show exception

The command enable/disables displaying of the exceptions, for example, access violation.

An exception is an event that occurs during the execution of a program, and that requires the execution of software outside the normal flow of control. If this option is enabled and the program goes out of control a message is displayed that may help programmer to resolve the problem. Normally the flag is set off because the message may be generated a loop blocking the program.

3.5.14. Save

The command saves all properties defined in the 'View' menu to the default configuration file. By default, the configuration file has extension 'mdc' and it is placed in the '..\Bin' directory.

3.5.15. Save as

The command saves all properties defined in the 'View' menu to the user-defined configuration file. The command opens 'SaveAs' dialog that lets you select

3. MDI

the file name. All configuration files have default extension 'mdc' and they are stored in the '..\Bin' subdirectory.

3.5.16. Load

The command loads configuration from a file and sets it as default. It opens 'Open' dialog that lets you select the file name. The selected file will be loaded automatically each time you start the program.

3.6. Tools menu

3.6.1. Calibrate

The command transforms channels into arbitrary units using polynomial transformation.

It starts 'Calibration' dialog box that let you enter the calibration input-data. The 'order' selects the order of the polynomial function used for the transformation.

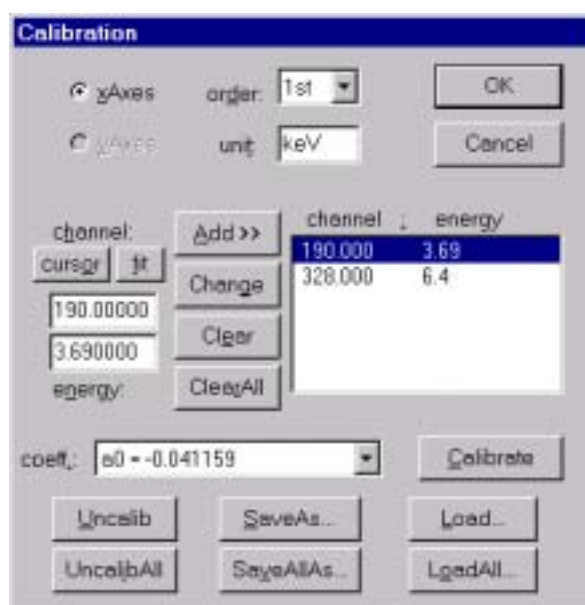


Figure 3.6.1.1 The 'Calibration' dialog box.

For each calibration point you should define two values: channel in the 'channel' field and transformed value into 'energy' field. By applying the 'Add' command, you transfer the values into the list of calibration points. The cursor position can be entered quickly by pressing 'cursor', and the result of the last Gaussian fit by clicking 'fit'. To correct a value in the list, you should first select it there, then change its value and finally apply 'Change' command.

To remove a value select it and press 'Clear'. Use 'ClearAll' to remove all points. After setting all points in the list, you should press 'Calibrate'. If you like to use the channel units again, press the 'Uncalib' button. To do the same for all displays use 'UncalibAll' command. The 'SaveAs...' and 'SaveAllAs...' stores calibration information for the current active spectrum and all spectra, respectively. To load this data, use 'Load' and 'LoadAll'.

3.6.2. Overlap

The command draws two or more one-dimensional (single) spectra in the same window (overlapping mode) with different colors and same vertical scale.

The command lets you select a group of one-dimensional spectra and draw all of them over the current active display. It brings up the 'Overlap' dialog that lets you select displays. The dialog contains two lists. The 'Display list' shows all one-dimensional spectra that you can draw over the current active display. The 'Overlap with' lists all displays selected for overlapping. Using 'Add' or 'Add All' you can select one or all of available displays. To enable the compare feature you should check the box that shows the name of the active display (followed by the 'Overlap with').

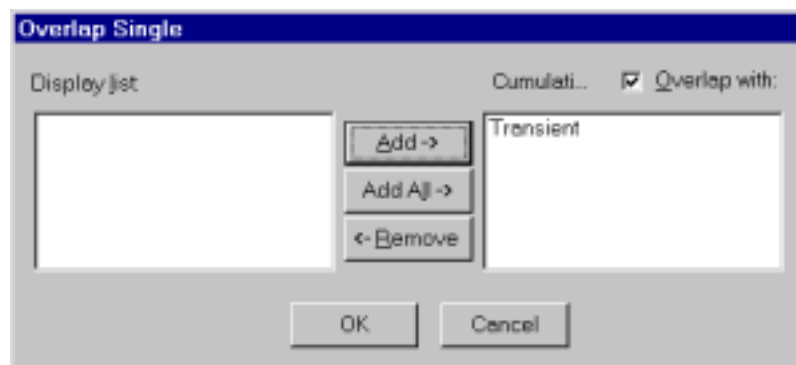


Figure 3.6.2.1 The 'Overlap' dialog box.

3.6.3. Compose

See the 'MNGI Data Compose'.

3.6.4. Copy as bitmap

The command copies display to the clipboard in the screen resolution as a screen dependent bitmap.

From the clipboard, the display can be pasted to another application, which supports bitmap format. Only the visible part of the display spectra will be copied.

3.7. Action menu

The Action Menu contains the items 'Start', 'Stop', 'Pause' and 'Continue' that perform the corresponding actions with the MPA/PC hardware. These controls are also available from the MNG. The Restart function can be used for a fast restart (stop and start) without calling the 'Start' dialog. You should enable it must in the View Menu.

3.8. Scan menu

The Scan Menu items let you select a scan region used for the micro-beam X ray fluorescence (XRFM) application in order to collect two-dimensional maps (XRF-images).

The scan region is a selected area on a sample that is going to scan with a stationary micro-XRF beam on such a way that sample is moving in the front of the capillary. The shape size and position of the scan area can be defined using this menu.

To select a scan area, you can use calibrated video image or already collected XRF-image and mark there a corresponding region of interest (ROI). In addition, you

can define a rectangular area which left-bottom corner is always in the front of the capillary, regardless of the sample position.

If you plan to use a video image of the sample to define a scan area, you should first make an optical calibration described in the 3.8.3. Once it is done, you can ‘grab’ a frame from a video image of your sample by using the ‘video Image Grab’ command, select a ROI and finally apply the ‘Export Image ROI’ command.

If you don’t use a video-image, you should move sample on such a way to bring the left-bottom corner of the sample area you are interested in into the front of capillary. After that, apply the ‘Export Rectangular Area’ command. Such mode of scanning is called ‘blind scan’.

In any case, you can define a scan area on the XRF image that is produced from already defined scan area, either blind or video. To do this you should select an area on the XRF-image and apply the ‘Export Image ROI’ command. Note you need already defined scan area. Therefore, the first valid scan area should be either ‘blind’ or defined from a video image.

If you want to select a scan area that does not fit into the frame of an existing XRF-image, you can open so called, ‘extended XRF-image’. This is done by the ‘Extend’ command. It brings up a new two-dimensional map with an XRF image in its center. The size and resolution of the map can be defined by the ‘extended Map’ dialog in the ‘Properties’ sheet. After selecting an area on the extended XRF-image, you should apply the ‘Export Image ROI’ command.

The currently defined scan area may be viewed by the ‘View’ command. If you want to actualize one of previously defined scan areas, then use the ‘History’ command. The both of commands enable you to modify and already defined scan properties.

If you want to position motors on such a way that selected point on the sample is brought in the front of capillary you should use ‘Goto’ command. It to calculate target motor coordinates and then move motor there using the ‘Motor’ application.

3.8.1. Scan properties

The command brings ‘Scan Properties’ sheet that lets you define basic scan features grouped in the following dialog boxes:

General

The program always starts scanning from the left-top corner of the scan frame and terminates either in the right-down or left-down corner depending on the scan area. The points that are inside the scan frame, but outside of the scan border are skipped (for more about the scan frame see Export Image ROI). The ‘Path’ item shows a list of figures that illustrate possible path from a starting to a terminating point. They are explained in the same order they appear in the list as follows:

- horizontal scan with jump at the right border to the left border
- vertical scan with jump at bottom border to the top border
- horizontal scan with no jump
- vertical scan with no jump

If the selected path includes jump then each sample point approach capillary from the same direction and eventual error in position that depends on the direction is minimal for all points (except for those at boundary when jump is terminated). On this way, any eventual shift between odd and even rows or columns are minimized. This

may be important for scan with very high resolution. On the other side, such path can increase the total measurement time.

The motors skip from one position to the other with equal speed selected by 'speed' control. The starting point is always approached at full speed. The speed grades are in arbitrary units. These units are related to encoder counts, which can be defined with using the Motor application.

If the 'zoom' item is enabled (checked) the collected XRF-image (target) has the same dimension in millimeters as the smallest rectangular that surrounds scan area selected on the existing XRF-image (source). Otherwise, the target and source XRF-image has the same size. For more about the effect of the 'zoom' feature on the resolution, see command 'Export Image ROI'. If the scan is selected on the video or extended XRF-image, then 'zoom' is enabled by default and any change of this item has no effect.

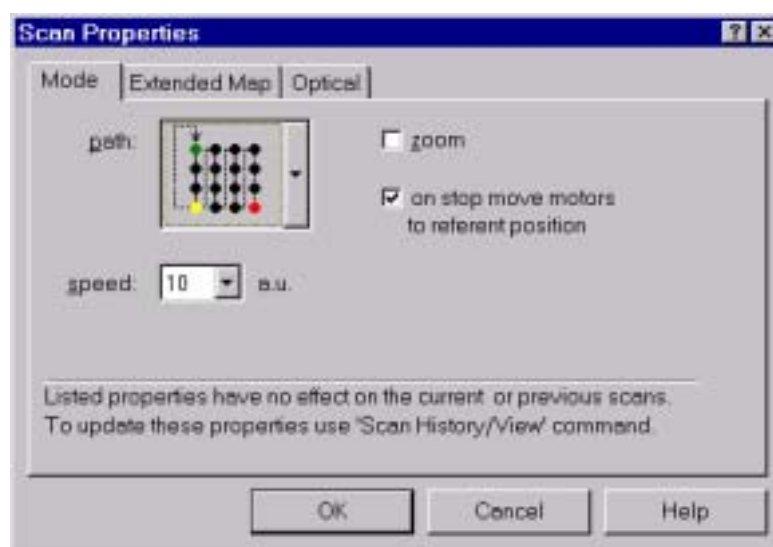


Figure 3.8.1.1. The 'General' dialog box of the 'Scan Properties' sheet.

After scanning is finished, the program moves motors to the referent position if 'on stop move motors to referent position' is checked. Otherwise, the motors keep last position. It is useful to keep motors in referent position if it is zero, but it can take some time. For example, if the computer must be restarted (switched off and then on) the motors are reset to zero position and scans you have defined are no more valid. In this case, you should recover motor referent position with using Motor application, which may sometimes failed. However, if this position was already zero, then you don't need to recover it.

The properties listed above have no effect on the current or previous scans. To update these properties you should use the 'Scan History' or the 'Scan View' dialog box.

Extended Map

The dialog lets you set size and resolution of the 'extended map' called 'extended XRF-image'. It is a two-dimensional map, which does not collect any data, but enables selecting a scan area that is outside of XRF images.

The map always shows one of the XRF images. You can select this image by the 'show XRF image' list box. The selected image is placed in the center of the 'extended map'.

To reserve enough space to fit the image and select a bigger scan area, adjust the ‘vertical’ and ‘horizontal’ size of the map. Once the size is set, choose the ‘vertical’ and ‘horizontal’ resolution in channels according to the resolution in

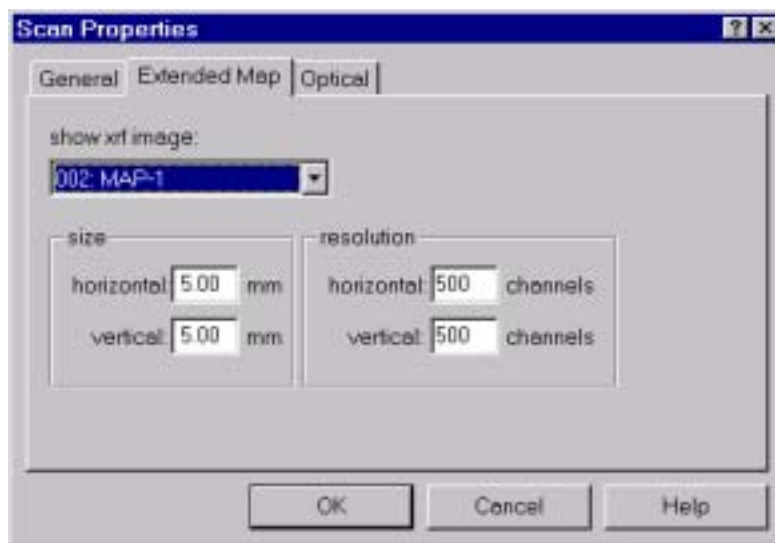


Figure 3.8.1.2. The ‘extended Map’ dialog box of the ‘Scan Properties’ sheet.

millimeters that you need. For example, with the size of 5 millimeters and required resolution of 10 μ m in the horizontal direction, you should set at least 100 channels for the same direction. This enables the program to calculate the scan border with sufficient precision.

Optical

The dialog box contains all information that let you define a scan region from a video-image.

Parameters defined in this dialog box are used by the program to move the sample in the front of capillary according to selected scan area. Refer also to the chapter 1.2 Setup and 1.5 Scanning for more information on setting dialog’s entries.

Optical-Referent image calibration:

The referent calibration is used by the program to transform pixel coordinate x_p and y_p on the video-image into millimeter coordinates x_{mm} and y_{mm} on a sample for horizontal (x) and vertical (y) direction respectively. For each direction, the calibration is defined by two coefficients, slope a and offset b .

$$x_{mm} = b_x + a_x x_p$$

$$y_{mm} = b_y + a_y y_p$$

The slope a is equal to the resolution, that is number of millimeters per each pixel of the image. It is not arbitrary and depends on the magnification of the microscope and size of the video-image (in pixels). The first depends on the lens you use. The last depends on frame grabber settings (see the ‘Video ... Properties’). For fixed lens magnification and video-image size, the resolution can be found experimentally by taking image of an object with well-known size. See 1.2 Setup for default values.

You can keep the existing values for the calibration coefficients if the microscope magnification and video-image size were not changed.

Both of the referent image calibration coefficients a_x and a_y can be easily found or checked by the procedure described at the 'Video ...Image ...Calibrate' command. The procedure requires a few-micrometer thickness cross that is mounted on the holder.

The offsets b_x and b_y are equal to the x and y coordinates in millimeter of a sample point that appears in the left-bottom corner of the video-image when the motors are at position defined by the 'Motor Referent Position' item. These coefficients are arbitrary and may be used to select the point on the sample that has (0,0) coordinate in millimeters. The default values are $b_x=b_y=0$.

When offset, resolution and the 'Motor Referent position' are defined, then the program establishes unique two-dimensional coordinate systems in a plane perpendicular to the capillary. It is called *scanning coordinate system* (SCS). This coordinate system is fixed (does not move with motors).

The program always shows video pictures in the SCS coordinates. This means that each time you move the sample and then take an image, the program automatically adjusts calibration of the new image. It is done on such a way that all points have the same coordinates like on an infinite image taken with motors in the referent position. Therefore, the same point always appears with the same coordinate in millimeters. If not, it means that either a wrong resolution is selected (it corresponds to a previous microscope magnification and video-image size) or motor

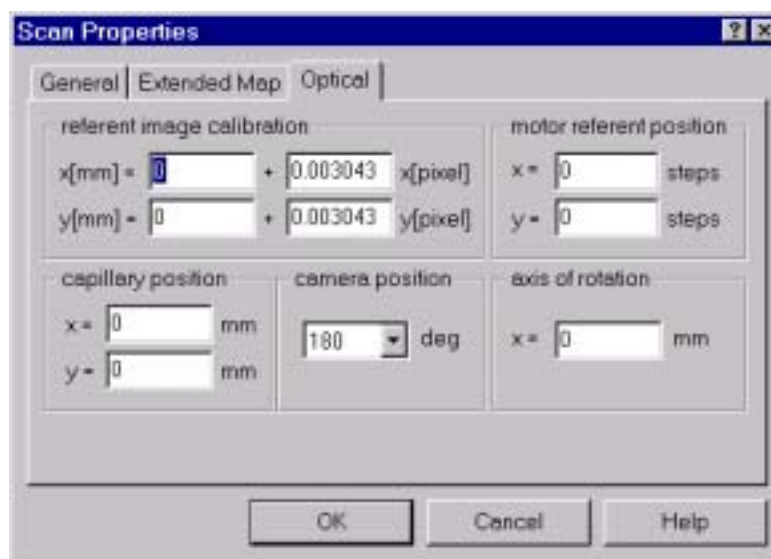


Figure 3.8.1.3. The 'Optical' dialog box of the 'Scan Properties' sheet.

calibration in millimeters has been changed since the last optical calibration. In the first case, recalibrate picture. In the last case use motor application (motor.exe binary) to correct motor calibration coefficients.

If you want to take coordinate of a fixed object, then motors must be in the 'Motor Referent position'. In other words, any fixed object (not mounted on the holder) that you can eventually see on a video image, has correct coordinate only if the image is taken with motors in the referent position.

Optical-Motor referent position

It is defined in encoder counts (steps) of x and y motors in the moment when the scanning coordinate system is defining (see also chapter 'X ray Microprobe... Scanning'). At which position the motors are with here selected coordinate in steps, it is up to you.

It is recommended to move motor on a preferred position and then reset x and y motor coordinates (set them to 0). It should be done before setting capillary position. Therefore (0,0) is default value.

The capillary position and position of axis of rotation are invalid if the motor referent position (and calibration coefficients as well) were changed in the meantime.

Optical-Camera position:

The program supports two positions of the camera:

- On the front side of the sample with optical axis at 0^0 respect to the capillary. In this case, select the first item in the list box (0^0)
- On the backside of the sample with optical axis at 180^0 respect to the capillary. In this case, select the second item in the list box (180^0)

Select the value that corresponds to your geometry.

Optical-Capillary position

It is a position of capillary (beam) defined in the scanning coordinate system. The capillary is fixed. Therefore, if you take its coordinate from a video image, then motors must be in the position defined by the '*motor referent position*' item when image is grabbing. Sometimes exit of capillary is covered by the aluminium and can't be seen by microscope. In this case you have to make a procedure that is described in the chapter 'X ray Microprobe... Scanning'.

Axis of rotation:

The axis of rotation is used only if the camera position is at 180^0 . Otherwise, it is ignored. To find it, you can mount a micrometer cross, take image and read position of the cross. Then rotate the cross and do the same. The position of the axis is then average value of this to readouts. Before each rotation it is recommended first to move motors to the referent motor position, which is normally (0,0).

3.8.2. Export

Export commands define scan area that is used later by the acquisition program to drive motors.

All scan parameters are stored into scan.dat file and copied to logbook file scan\$.log, where \$ is a unique integer. Both of files can be view and modified in any time.

3.8.2.1. Rectangular area

The command brings property sheet that lets you define a rectangular scan area with origin (left-bottom corner) that is related to one of the following:

- sample point that is in the front of capillary (or eventually shifted by a selected value) in the moment when you apply this property sheet (so called '*blind*' scan)

- sample point selected on a two-dimensional image (XRF, extended XRF or video, which was active in the moment when you opened this property sheet)

The 'blind' scan area does not use any video or XRF image. Therefore, if you don't have any referent image, you should use this property sheet to define valid starting scan area. Note that latter you can collect an XRF-image and use it to select new scan area, but blind scan area can be defined in any time.



Figure 3.8.2.1.1. The 'Position' dialog box of the 'Rect, Scan Area' property sheet.

In order to define blind rectangular scan area you should define its position and size. To define position you should move sample on such a way that origin (left-bottom corner) of the area you are interested in, is in the front of capillary. In the next step, you should enter current motor position in the 'Position' dialog. To do this check button '*origin in the front of capillary...*'. This allows you to enter the position either manually (as it is defined by motor application) or with using '*read motor*' command.

If you define scan for the first time after positioning motors then select no shift ($dx=0$, $dy=0$) and check '*set (x+dx, y+dy) as new referent position*'. This sets origin of scan area in the front of capillary, and calibrates image on such a way that (0,0) mm is in the origin (the slope of calibration curve is defined by motors itself (see the Motor application)).

Note that your scan area is well defined by the motor position (x,y) and can be easy reproduced. Later, if you want, you can scan area that is shifted by (dx,dy) mm from the current area. In this case, you can either keep current calibration offset or reset it. In the first case you should uncheck button '*set (x+dx, y+dy) as new referent position*'. This prevents program to move motor referent position. As the (0,0) coordinate in millimeters is always at point which is in the front of capillary when motors are in referent position your calibration does not change. Therefore, left top corner of the shifted scan area has no more (0,0) millimeters coordinate. In contrary, if you check the same button, then the program uses (x+dx,y+dy) as a new referent position and sets (0,0) coordinate in millimeters at origin of the shifted area. In any case, positive dx and dy shifts scan area to the right and up respectively, while negative dx and dy shifts it to the left and down side respectively.



Figure 3.8.1.2. The 'Size' dialog box of the 'Rect. Scan Area' property sheet.

If you want to define rectangular area relative to an image, then you don't need to move motor manually. The origin (left-bottom corner) of the new scan area is defined by a point on any of images (XRF, extended XRF or video). The coordinates of selected point you should enter in the 'Position' dialog. Note, the program requires that selected image is active before you open 'Scan Rectangular Area'. If you placed cursor at this point, then later you can enter coordinates by applying 'read cursor' command. If you want to place origin outside of the image select, then you can define corresponding (dx,dy) shift. The shift sign has the same convention as in the case of 'blind' scan.

Regardless of the origin the scan refers to, the size in millimeters of target XRF images (map collected using selected scan area) is equal to the size of the scan area and may be defined by the 'Size' dialog box.

The resolution of the target XRF-image, that is a distance in millimeters between two successive points, is equal to the size of the scan area divided by the number of channels in the target XRF-images. To change it, you should adjust the number of channels using the 'Acquisition ... Displays ... Map Resolution' command.

3.8.2.2. Image ROI

The command defines a scan region directly from an XRF, extended XRF or a video-image.

The images on which you define a scan area are called source images. The images that are collected from an exported scan area are called target XRF-images.

To define a scan region, you should first create a two-dimensional ROI on a source image (refer to ROI and Region menu for more information) and then apply the command. If you use a video image, the system must be optically calibrated (refer to 3.8.1 for more information).

The size in millimeters of the target XRF-images are the same and equal to the size of the scan frame. The scan frame is always rectangular, in difference from the scan border that may have any shape. If the source is a video image or extended XRF image, the frame is always defined as a smallest rectangle that surrounds selected scan area (ROI).

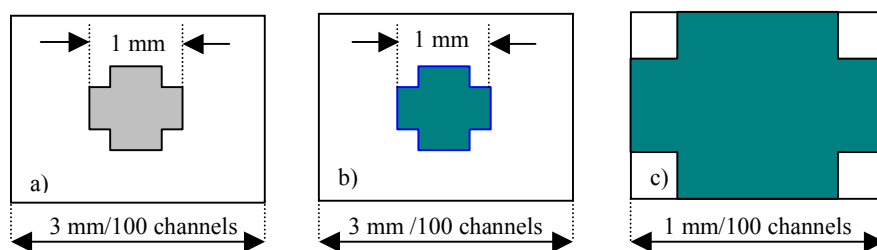


Figure 3.8.2.2.1 The effect of the 'zoom' feature. The scan area is defined on a source XRF image (a). The target XRF image (collected after the scanning) without zooming (a) has smaller resolution than the images (c) collected with zooming, assuming the both maps has the same number of channels.

If the source is an XRF-image, you can optionally select the full frame of the source image as a scan frame. In that case, the size of the source and target XRF-images are the same. This is enabled by disabling 'zoom' in the 'Scan Properties' sheet. If the zoom is enabled, the scan frame size, hence the target XRF-image, is equal to the smallest rectangle.

The resolution in millimeters of the target XRF-image, that is the distance in millimeters between two successive points, is calculated automatically by the program. To do this, the program divides the size of the scan frame with the number of channels in the target XRF-image. To define the number of channels use 'Acquisition... Display... Map Resolution' dialog box. Note that source XRF image does not need to have neither equal number of channels nor the same resolution in millimeters comparing to the target XRF-image. Fig 3.8.2.2.1 illustrates effect of zooming on the target XRF-image if the number of channels does not change.

3.8.3. Extend

The command creates a two-dimensional map called 'extended XRF-image' that lets you select a scan region partially or totally outside the frame of the existing XRF-images. Its properties can be defined in the 'Scan Properties' dialog box. After creating a ROI on the map, apply the 'Export image ROI' command. It defines scan area that corresponds to the selected ROI.

3.8.4. View

The command brings 'Scan View' dialog box that shows scan area used by the program to drive the step motors. The data are taken from 'scan.dat' file. For more information about the shown items, see the 'History' command. You can change some of these properties and then update the 'scan.dat' file with using the 'Update' command.

3.8.5. History

The command lets you actualize a scan area defined in the past. Each time a scan area is defined by one of the 'Export' commands, the program overwrites existing 'scan.dat' file and creates a corresponding logbook file. The logbook files have name 'scanx.log', where x goes from 0 to 999 and stays there until you clear all previous generated files.

This command creates a dialog box that lets you view scan area defined by any logbook file. To browse the files, use 'Previous' and 'Next' buttons. Each time you press on these buttons a new logbook file is loaded. Upon applying 'Export', the program automatically copies selected logbook file into the 'scan.dat' file.

3. MDI

On the right side of the dialog box there is an image of the scan area defined by the selected logbook file. The scan area is always drawn inside a white rectangle, which represents frame of a source image used to export the area.

The dialog box list the following scan properties (some of them you can modify):

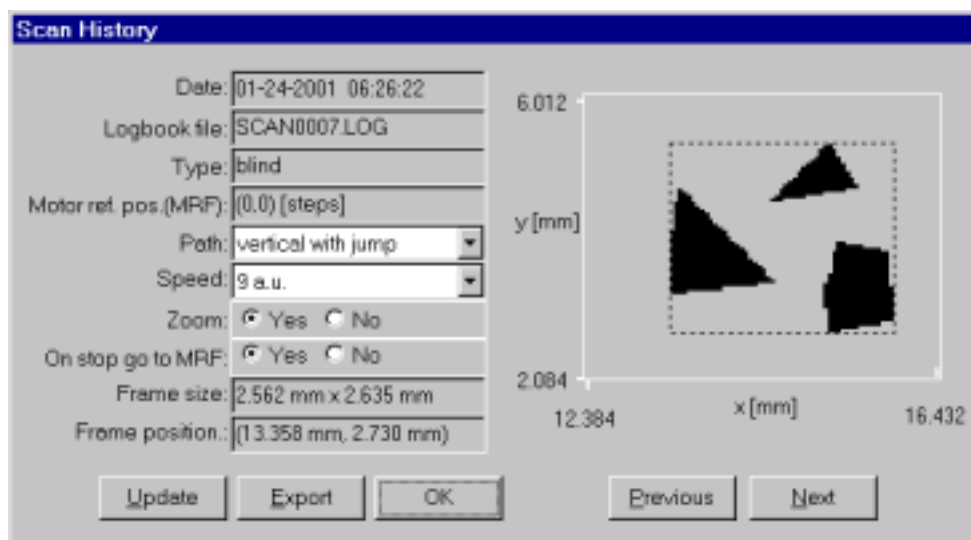


Figure 3.8.6.1. The 'Scan History' dialog box.

File

File name that contains shown scan area information

Date

Time stamp when scan was created (exported)

Type

Video if video images are used for scanning otherwise *Blind*

Motor referent position

Position of motors when referent image was calibrated

Scan path

Order in which points are scanned. The paths with 'jump' adjective take into account backlash compensation

Scan speed

Speed of all motors during scan. The speeds are in arbitrary units. In order to set these units in encoder counts/sampling time for each motor use Motor application.

On stop go to MRF

After scan is finished, motor goes to referent position if this item is checked.

Zoom

If zoom is enable (flag 'yes') than the target XRF image has dimension of a smallest rectangle that contains whole scan area. In this case, the picture of the scan

3. MDI

area illustrated by the dialog-box contains a rectangle drawn by dot lines. If zoom is not enabled (flag 'no'), then the target XRF image has the same dimension as the source image

Frame size:

Size in millimeters of the target XRF image that is defined by the scan area

Frame position:

Coordinate of the left-bottom corner of the above frame. The coordinates are in scanning coordinate system.

3.8.6. Motor

The command runs Motor application (binary motor.exe). The application is used to position motors manually. There, you can define basic motor properties. For more information see 4. Motor.

3.8.7. Go to

The command transform coordinate of a sample point on the video image into motor coordinates on such a way that point is in the front of capillary when motor are on calculated coordinates.

This command is useful if you want to bring particular sample point, identified on an image, in the front of capillary. In order to get spectrum at this point you can do any of the following:

- Move motors to the calculated position, switch to default application (see 'MNG... System... MCA ... Application') and start acquisition.
- Move motor to the calculated point, delete all maps and start acquisition
- Export a small rectangular with left-top corner at this point, set map resolution to 1x1 pixel and start measurement

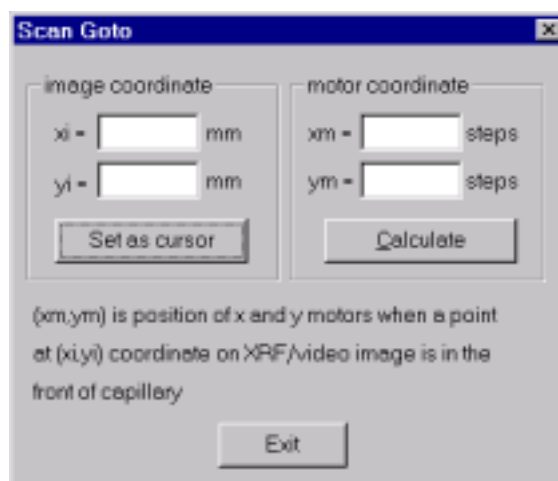


Figure 3.8.7.1. The 'Scan Goto' dialog box

3.9. Video menu

The Video Menu items let you display a video image from a color or black and white camera in a real or pseudo-real time, depending on the hardware capabilities. This requires a camera and a video capture board, which is also called frame grabber.

The software supports capture boards that are listed by the 'Select' command. You need to select your board only once time, usually after the program installation. Before that, you should install the board according to the instructions supplied by manufacturer. After selecting the board, you should configure it using the 'Properties' command.

For each of supported boards, the program uses a separate library that refers to one or more manufacturer libraries and drivers. It is loaded upon the 'Open' command and unloaded upon the 'Close' command. Therefore, each time you run the program or after you apply the 'Close', it is necessary to issue the 'Open' command.

The 'Open' command brings up a window called Video-display that shows a stream of image frames from camera, in a pseudo-real or real-time. You can grab one of the frames by the 'Image Grab' command. The grabbed image appears in a separate window and you can save it by the 'Image Save' command. Latter, you can retrieve it by the 'Image Load' command.

The micro-beam X ray fluorescence application (XRFM) requires that image is calibrated, that is to establish the following relation: image pixel versus position on the sample in millimeters. For this purpose, the 'Calibrate Video Image' command is used.

3.9.1. Select

The 'Select' command lets you specify the video capture board.

The command brings up the 'Select' dialog box that contains a list of supported boards. You need to select your board only once, usually after the program



Figure 3.9.1. The 'Select Video' dialog box.

installation. Before that, you should install the board using the instruction supplied by manufacturer. To specify camera format and grabber properties use the "Properties" command.

3.9.2. Properties - Matrox Meteor

The 'Properties' command configures the hardware and software that enables you to see the video image. In addition, you can use it to customize your Video - display (a window showing live video).

The listed options depend on the capture board you have selected. The current settings are stored in the 'video.ini' file in the program's binary directory. The file is updated each time you apply new settings.

The command brings up a property sheet that contains the following dialog boxes:

Input

The 'Input' dialog box lets you select the input device (camera) data format and the active input channel.

3. MDI

The data format is a specification that enables the software to configure on-board digitizer on such a way it matches video signal format from the input device. The dialog lists only data formats supported by the selected board. Here is a list of all available formats:

format	description	Camera
M_RS170	RS-170, 640x480, 8 bits, 12.5MHz, analog	Mono
M_CCIR	CCIR, 768x576, 8 bits, 14.8MHz, analog	“
M_RS170_VIA_RGB	RS-170 using RGB module	“
M_CCIR_VIA_RGB	CCIR using RGB module	“
M_NTSC	NTSC, 640x480, 3x8 bits, 12.5MHz, composite	Color
M_NTSC_RGB	RS-170 RGB, 640x480, 3x8 bits, 12.5MHz	“
M_NTSC_YC	RS-170 Y/C(SVHS), 640x480, 3x8 bits, 12.5MHz	“
M_PAL	PAL I, 768x576, 3x8 bits, 14.8MHz, composite	“
M_PAL_RGB	PAL I RGB, 768x576, 3x8 bits, 14.8MHz	“
M_PAL_YC	PAL Y/C, 768x576, 3x8 bits, 14.8 MHz	“
M_SECAM	SECAM, 768x576, 3x8 bits, 14.8MHz, composite	“
M_SECAM_RGB	SECAM, 768x576, 3x8 bits, 14.8MHz	“

You should select one from the dialog list that matches your camera. The M_RS170 is usual settings for monochrome cameras and M_NTSC for color cameras.

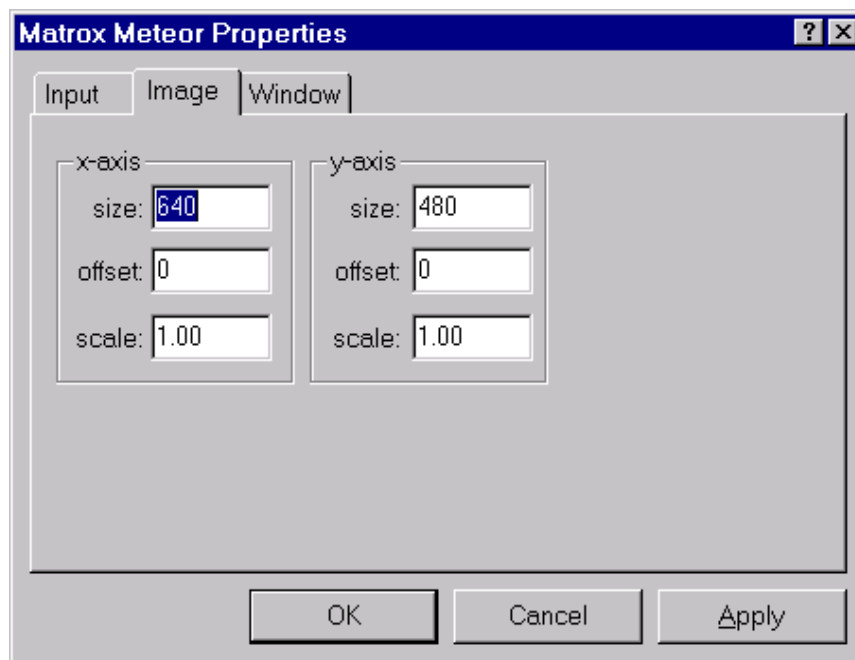


Figure 3.9.2.2. The 'Image' dialog box of the 'Properties' sheet.

For more information about the common formats see the 'Common video-signal formats and standards' in Appendix

The input channel parameter specifies the channel on which the digitizer is to input data (signal and sync). This parameter can be set to one of the values listed in the dialog boxes, depending on the number of channels available for the specified digitizer's data format. Usually it is channel 0.

Window

The 'Window' dialog box lets you set properties of the window that contains a video image.

The '*minimize*' item shows minimize-button control in the right-top corner of the window that may turn the last in the iconic state.

The '*maximize*' item shows maximize-button in the right-top corner of the window that may resize the last to the full screen.

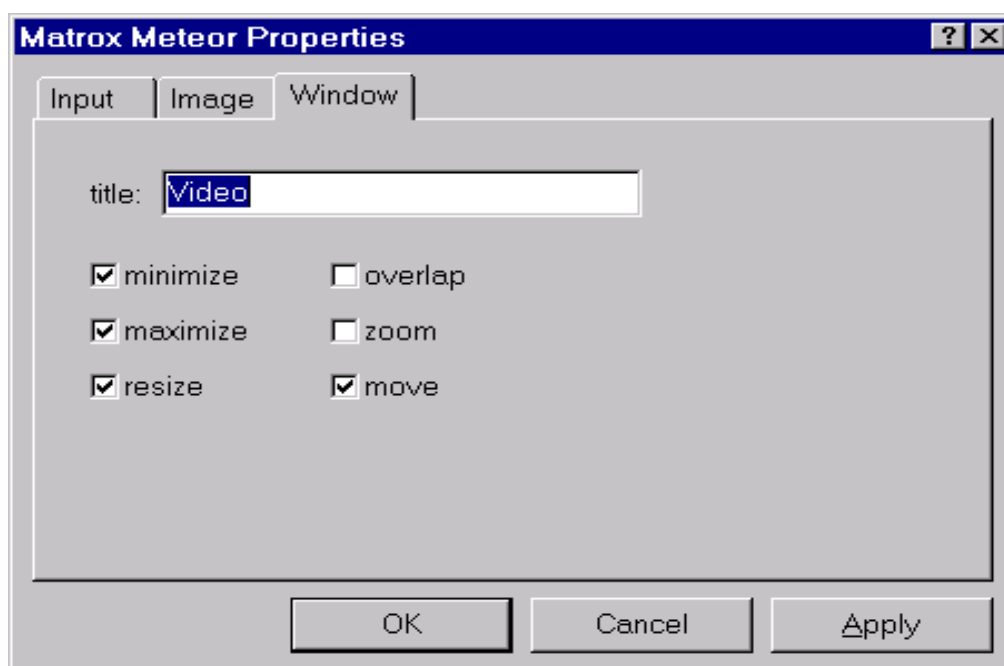


Figure 3.9.2.3. The 'Window' dialog box of the 'Properties' sheet.

The '*resize*' item enables to change the window size by dragging the cursor over the window border

The '*overlap*' item makes the window overlapped by other window. If it not checked, the window keeps on top.

The '*zoom*' item shows two scaling controls in the left-top corner of the window. Each time you apply the controls, display window size is reduced by factor two. Note the image is still grabbed in the full resolution.

The '*move*' item enables the window to change position.

3.9.3. Image

3.9.3.1. Grab

The command takes a single image (snapshot) from the Video-display and shows it in a new window. The image is grabbed always in the maximum size and resolution that is possible by the current camera format, regardless of the Video-display size.

3.9.3.2. Save

The command saves a Video-image in the bitmap format together with corresponding calibration.

3. MDI

First, you should grab image by the 'Image Grab' command. In addition, the corresponding image window must be active. The program automatically saves calibration in the file with the same name but with extension '.Cal'.

3.9.3.3. Load

The command reads bitmap and associated calibration file stored by the 'Image Save' command. In addition, it brings up a new window that contains loaded image.

3.9.3.4. Calibrate

The command brings up the 'Calibrate Video Image' dialog box that enables you establish the video-image pixels versus object millimeters transformation.

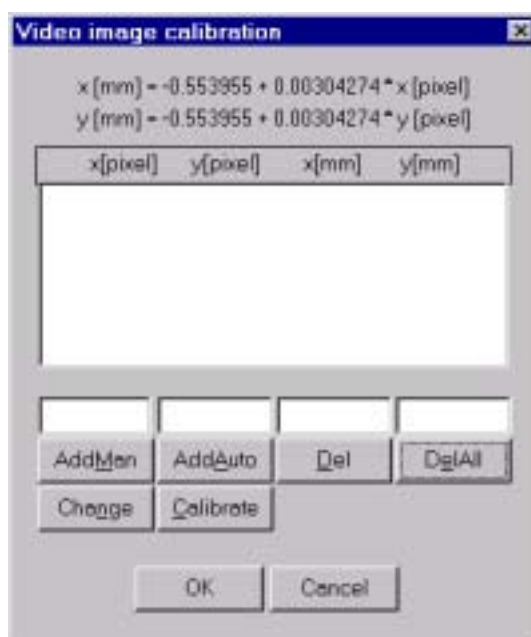


Figure 3.9.6.1. The 'Calibrate Video Image' dialog box

The video-image should be first captured by the 'Image grab' command. In addition, the corresponding image window must be active. The dialog box supports a list that is supposed to contain the calibration points.

For each calibration point you should enter both x and y coordinates in pixels and corresponding coordinates in millimeters. To do this, first enter the values in the edit controls below the list and then press the 'AddMan' button. The number of points is practically not limited. To correct already entered value; you should first select it in the list and then press the 'Change' button.

The calibration is performed upon pressing the 'Calibrate' button. Using all listed points; the program makes a least-square fit to the straight line and displays correlation coefficients at the top of the dialog.

The 'AddAuto' button is used for automatic calibration procedure described below. It assumes the system is supplied by the already calibrated step motors. This is the case of the micro-beam XRF fluorescence application. Therefore, the procedure is related to this application.

The XRF image referent calibration procedure

- Step 1
Mount a micrometer cross on the sample holder
- Step 2
Move the cross on such a way that its center appears on the Video-display at point you want to have (0,0) coordinate. For example, it can be in the middle of the Video-display
- Step 3
Reset x and y motors using the Motor application (motor.exe). The corresponding command is available in the application's system menu
- Step 4
Grab image by the 'Image Grab' command
- Step 5
Move cursor at the channel that corresponds to the center of the cross
- Step 6
Press the 'AddAuto' button. This gives the first calibration point
- Step 7
Repeat steps 2 to 6 for a few other positions of the cross, but skipping the step 3. For example, move the cross on such a way that it appears close to the corners of the video display. This will establish four additional calibration points.
- Step 8
Press the 'Calibrate' button.

This calculates so-called referent calibration coefficients and referent motor position as well. These values are also listed in the 'Optical' dialog of the 'Scan Properties' sheet. You can use this dialog box to change the calibration offset and referent motor position in any time, but you should not change the resolution (slope of the referent calibration lines).

Each time you grab a new image, either on the same or different motor position, the program automatically re-calibrate it taking into account the current motor position and the referent calibration coefficients.

4. Motor

The Motor application (binary motor.exe) lets you setup motor controller board and drive up to four motors by the computer keyboard or mouse.

The application supports different types of controller boards connected with your motors. To select the board you can use the ‘Select Board’ command in the program’s system menu. Note that for each board you need a corresponding library.

After selecting the board, you should set its properties by the ‘Board Properties’ dialog.

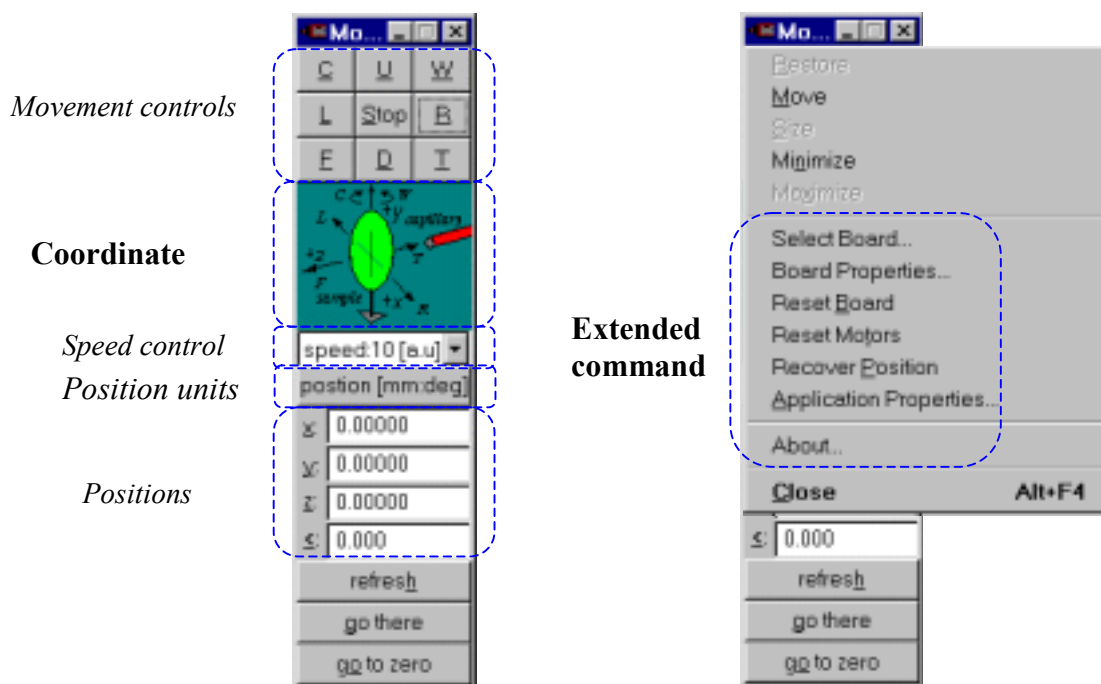


Figure 4.1. The Motor application main window (left) and its system menu box (right).

The application brings up a window (fig. 4.1) that contains a few basic controls for changing and monitoring motor’s position. In addition, the system menu box comprises a set of extended commands. The basic functionality of the controls and commands as well, are described separately as follows.

4.1. Coordinate system

The positions of motors are shown in metric units (mm, μm) or encoder counts (steps) in the orthogonal 3D coordinate system that is defined by the motor axes.

It is assumed that x and y motors has axes that are perpendicular onto the capillary, while the axis of z motor is parallel to the capillary (see Fig 4.1 left side).

The orientation of the axes is defined respect to the spectator that stays parallel to y-motor axis and watches into the exit of capillary. The x-axis in metric units goes from left to the right, z-axis from capillary to the spectator (beam direction), and y-axis from down to up. When measured in encoder counts x-axis has opposite direction than in metric units. The y and z-axis have the same direction in both cases.

The axis of rotation is along the y-axis. The clockwise rotation is in the direction of the left-hand fingers when the thumb is turned up (along the y-axis). The anti-clockwise is the same as above but using a right hand.

4. MOTOR

The zero of the coordinate system corresponds to an arbitrary position of the sample respect to the capillary or any other fixed object. To set the zero you should move the sample at position you want and then use the 'Reset Motors' command in the application's system menu.

Note that the transformation from encoder counts to millimeters/micrometer or degrees is linear with zero-offset and slope defined in the 'Board Properties' sheet, Motor dialog. Up to the signed that is defined by above convention, the slope is internal characteristic of the motor and encoder.

In order to connect coordinates shown by the Motor applications with those shown on a video or XRF image you should use 'Goto' command in the MDI application.

4.2. Basic controls

The basic controls enable you to move one or more motors simultaneously with a selected speed, while watching their instantaneous positions. In addition you can define target motor's position.

4.2.1. Movement

The movement controls move one or more motors infinitely in particular direction.

The controls are implemented by the push buttons shown on the fig 4.1. The buttons may be activated or deactivated. In the first case, they look like a 'pressed button', and in the second like a 'released button. To switch a button from one state to the other, you can use mouse or keyboard as it is indicated by the table 4.2.1.

If a button is activated the corresponding motor moves the sample infinitely in the direction indicated by the button. The sample moves in this direction until you deactivate the button, activate the button for opposite direction or press the stop button.

Table 4.2.1. The Movement controls description

button	keys	description
C	C or Alt+C or PgUp	<i>Rotate sample clockwise</i>
W	W or Alt+W or PgDn	<i>Rotate sample anti-clockwise</i>
L	L or Alt+L or Left arrow	<i>Move sample left</i>
R	R or Alt+R or Right arrow	<i>Move sample right</i>
U	U or Alt+U or Up arrow	<i>Move sample up</i>
D	D or Alt+D or Down arrow	<i>Move sample down</i>
F	F or Alt+F or Home	<i>Move sample from capillary</i>
T	T or Alt+T or Del	<i>Move sample toward capillary</i>
Stop	S or Alt+S or End	<i>Stop all motors</i>

More than one button may be active in the same time. Therefore, you can move the sample simultaneously in all directions.

4.2.2. Speed

The speed control causes the all motors run at selected speed. The speed is given in arbitrary units that are proportional to encoder counts/sec. If the torque load changes on the motors, the controller attempts to maintain the velocity by varying the motors current. In order to define units in encoder counts per sample time you should use Board properties dialog box.

4. MOTOR

4.2.3. Positions

The position of the sample is displayed by the four edit-controls labeled with the same name as the corresponding axis: x, y, z for translation, and < for rotation respectively.

If the motors are moving, the positions are automatically updated. When the motors are in the rest, you can edit the controls and then apply 'go there' command. To reset position to zero, use the 'Reset Motors' command in the application's system menu.

4.2.4. Position units

The position of sample may be shown in the metric units like millimeters (mm), micrometers (um) or degrees (deg) or internal motor coordinates called steps. The transformation from steps into the metric units is defined by the coefficients listed in the 'Motor dialog'. The dialog is accessible by the 'Board Properties' command in the application's system menu.

Each time you press the control, the units are switched and position controls are updated.

4.3. Select board

The command loads library for a selected controller board. The library contains executable code, which translates application commands into the board specific instructions and handshakes with the board as well.

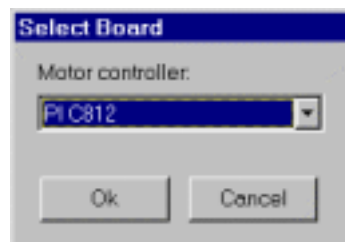


Figure 4.3.1 The 'Select Board' command dialog.

Usually, this command is applied when you change the board, or the program is running for the first time. Each time you start the program, it automatically loads the last selected library.

4.3.1. Board properties

The command sets basic motor controller board parameters. The parameters depend on the selected board and they are described as follows

4.3.1.1. C812

The PI C812 motor controller board of Physik Instrumente communicates with host through either ISA bus or serial port. In both cases the program uses c812.dll. In addition, the ISA bus channel requires mapmem.sys driver when program running on the Windows NT/2000. The driver is installed during installation of the Motor program or when you select the board with using of the 'Select board' command. If necessary the driver is updated each time you change board's properties.

The 'Board properties' command first brings up a dialog that enables you to select communication channel. Although the software enables you to change active communication channel without rebooting or restarting computer, the board itself requires setting several jumpers and switches. Therefore, each time you want to change the channel you must first turn off computer, make changes on the board, turn

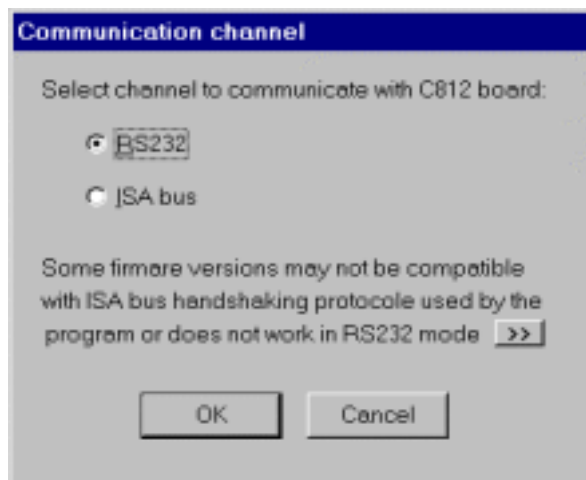


Figure 4.3.1.1 1 The Communication channel dialog box.

on the computer and come back to this dialog. Normally it is done only once - when board is installed for the first time. You should note that not all firmware versions are supported by the program. The list of compatibility are listed on the Table 4.3.1.1.1. For more about the controller, see the 'Information MS 26E C-804, C-812 DC-motor Controller. Before setting the hardware please read restrictions that refer to particular channel listed in the corresponding 'Communication' dialog box of the Board properties sheet..

Table 4.3.1.1.1 The firmware versions and software compatibility

Firmware	Communication channel	
	ISA	RS232
V5.5 + PAL U32F	Works	does not work
V4.3 + PAL U32F	does not work	works
V4.1 + PAL U32F	does not work	works

Normally you don't need to change current active channel, therefore you should press OK to proceed.

The 'Board properties' sheet contains a few dialog boxes which are equal for both channels except one that defines communication itself.

Communication-RS232

The serial (RS232) channel is preferred way of communication due to hardware problems in the ISA mode. In order to use it you should connect on-board connector J1 with serial port of your computer by connecting the pins as shown in the table 4.3.1.1.2.

Table 4.3.1.1.2 The cable connections for serial communication

C812 board connector J1	PC 25 pin Sub-D
1 (RXD)	2 (TXD)
3 (TXD)	3 (RXD)
7 (DTR)	6 (DSR)
9 (GND)	7 (GND)

Further you should set the following jumpers and switches:

- SW3:bit 8=ON
- SW2:bit 6=OFF; bit7=ON; bit8=OFF (see COM port settings bellow)
- jumper E3 must not be installed

The Communication dialog sets parameters related to the communication with board.

The serial port settings on the host are defined by the 'COM port settings' group of controls. The default values are:

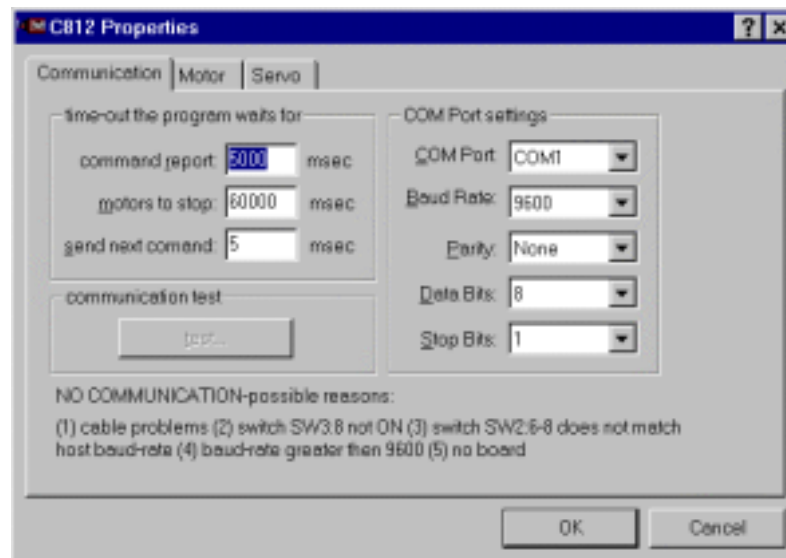


Figure 4.3.1.1.2 The 'Communication' dialog for RS232 channel (PI C812 board).

- Port ID: COM 1
- Baud rate: 9600 (must correspond to SW2:bits 6-8 settings)
- Parity none
- Data bits: 8
- Stop bits: 1

It is highly recommended not to change these values except Port ID. For example baud rates greater than 9600 may not work stable.

The software uses several time-outs when communicates with board. You can adjust three of them as follows.

The 'Send next command' defines minimum period the program waits before sending next command. It was noticed that in some cases C812 does not process a full command if the next command was sent to early, although no error in communication

was observed. A few milliseconds time-out is enough. The default is 5msec. You can test correctness of this time by using the 'test' button.

The '*Command report*' time-out is maximum time the program waits for C812 to send back report. A few seconds is enough. The default is 5sec. The values must be given in milliseconds.

The '*Motors to stop*' time-out is maximal time the program waits that motors move to target position. This value depends on the speed and distance. Therefore, an infinite value is ideal. On the other hand, if an error happens and motor does not move to the target position this time may cause a big delay in scanning or even stop it. The default value is 60 sec.

The '*Communication test*' sends several commands to the board and analyses reply. It is repeated as many times as defined by the corresponding dialog box. The tests is disabled if there is no communication with board.

Communication-ISA bus

The communication via ISA bus features some hardware handshaking problems that are fixed by introducing several software emulated wait-states. For this reason it may be slower than RS232. In order to use this channel several switches and jumpers must be adjusted as follows:

- SW3:bit 8=OFF
- SW4 bits 1,2,3,6=ON; bits 4,5,7,8=OFF (see 'Base memory' address below)
- Jumper E3 must be installed

The '*Base memory*' address defines referent memory address that program uses in order to communicate with board. It must match settings defined by switch SW4. The program automatically updates driver each time this value is changed.

The '*Command report*' and 'Motor to stop' time-outs have the same meaning for RS232 channel. Please see there for more information.

The '*Write*' wait-state is a fixed time the program waits before sending any character to PI C812 controller. This time is critical for correct communication. It may depend on computer speed and usually it is in the range of 2000-10000 μ sec (default value is 5000). The larger value minimizes handshaking problems but increases the command processing time. In order to test it, use the 'Test communication' button. Closely related to the write-time is the loop speed test, which must be properly performed in order to measure time correctly.

The '*Read*' time-out is a fixed time the program waits before reading any character from PI C812 controller. This time is less critical. It is suggested to use a value in the range 10-100 μ sec. The default value is 10.

The '*Wait-state loop speed test*' is critical for correct communication. The wait-state is implemented through simple loops. The execution time per one loop is defined on the basis of the time the program needs to execute as many loops as it is defined by the dialog (test using ... loops). On multithread systems like Windows this time may not be fixed and usually it is underestimated if other threads are not taken into account. On the NT/2000 the program gets information from the system how long time processor was executing corresponding thread, therefore the time is more reliable. On the 95/98 the time includes all other threads. In both case it is highly recommended to use as much as possible loops for test until the speed does not vary from test to test. If the '*Auto-test*' is enabled the program makes the test each time

4. MOTOR

c812.dll is loaded into memory. If the number of loops is small (in order to reduce test time) it may give wrong speed. Therefore, **it is recommended to disable 'Auto-test' and make loop-speed test with sufficient number of loops until the speed is stabilized.** The program will use the calculated speed (stored in motor.ini file) each time the library is loaded into memory.

The '*Communication test*' sends several commands to the board and analyses reply. It is repeated as many times as defined by the corresponding dialog box. The tests is disabled if there is no communication with board

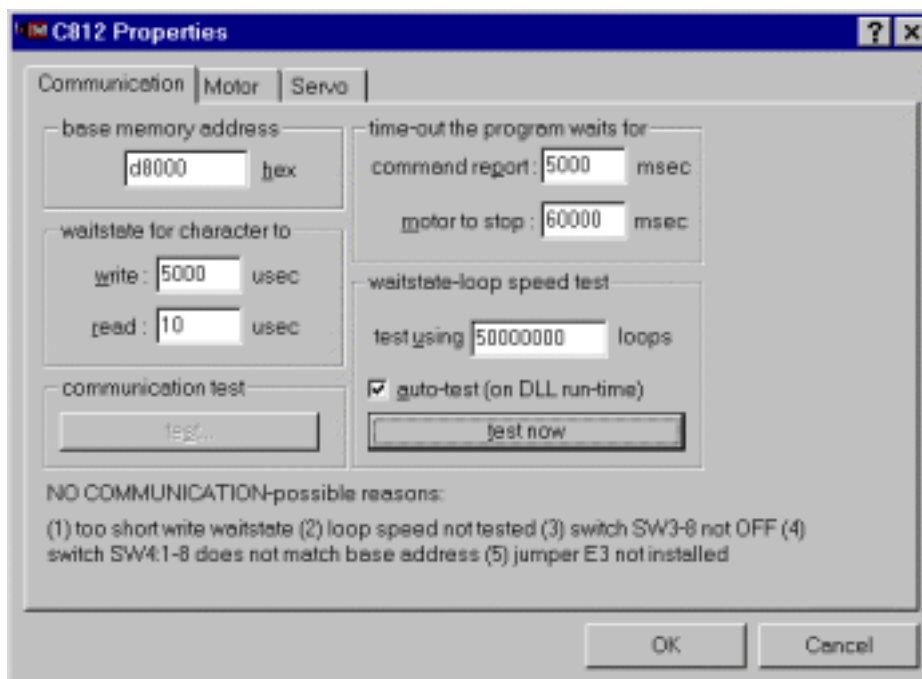


Figure 4.3.1.1.3 The '*Communication*' dialog for ISA channel (PI C812 board).

Motors

The '*Motors*' dialog box defines actuator's related parameters, which includes motors, encoders and screw drive.

The '*Precision*' defines acceptable difference between a target and current position (maximal error). If the differences are less then values specified by the dialog box, the program does not wait for motors to go to the target position nor tries to move motors there. In this case a new command may be accepted that move motors to the new position before they were stopped. The values are given in steps, that is, encoder counts. Use calibration defined in the '*Motor*' dialog to estimate these values in millimeters. To small values may cause an extra dead time in scanning.

The '*Calibration*' The calibration defines the number of encoder counts need to move a sample for one millimeter or one degree. The sign should be set according to the definition of the coordinate system we use. See the '*Coordinate system*'. All default values are positive except for the x motor. In this case, it is assumed that motor steps increase when a sample is moving to the left, that is, in opposite then it is defined by the orientation of the x-axis in the above coordinate system. For default values see 1.2 '*Setup*'. The '*Range*' defines maximal distance that motors are supposed to move sample.

4. MOTOR

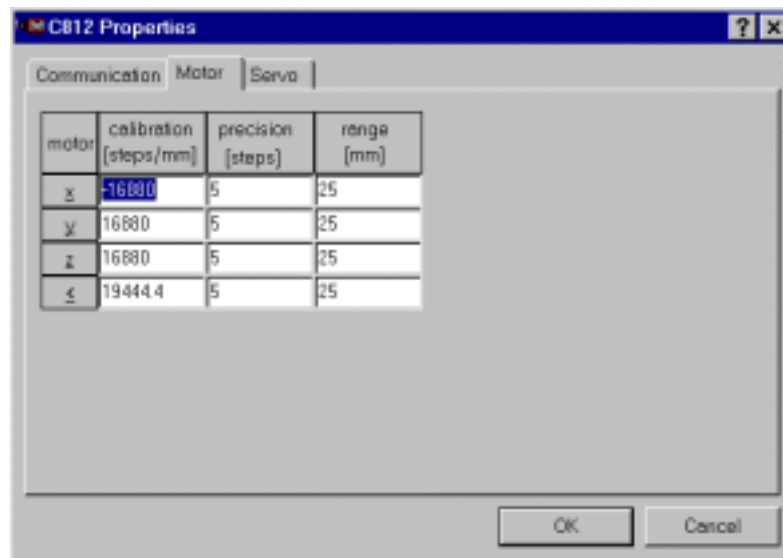


Figure 4.3.1.1.4 The 'Motor' dialog (PI C812 board).

The program uses this information only to indicate that entered target position is eventually out of the range. This is a kind of 'software' protection to saturate motors. Note that you should use sensors for the full protection.

Servo

The servo dialog is used to define basic servo control parameters. The wrong settings may cause that motor does not move correctly under some circumstances (for example when changing position for a few micrometers).

The 'Maximal speed' defines maximal programmed motor velocity during trajectory generation. You can define up to ten different speeds and use it latter for scanning. They are expressed in arbitrary units 'a.u.'. Each unit corresponds to certain number of encoder units per board's (fixed, internal) sampling time. You can adjust this number for each motor according to desired speed in mm/sec.

For other parameters please refer to the 'Information MS 26E C-804, C812' page A38. In the most cases the default values shown on the Fig 4.3.1.1.5 are correct.

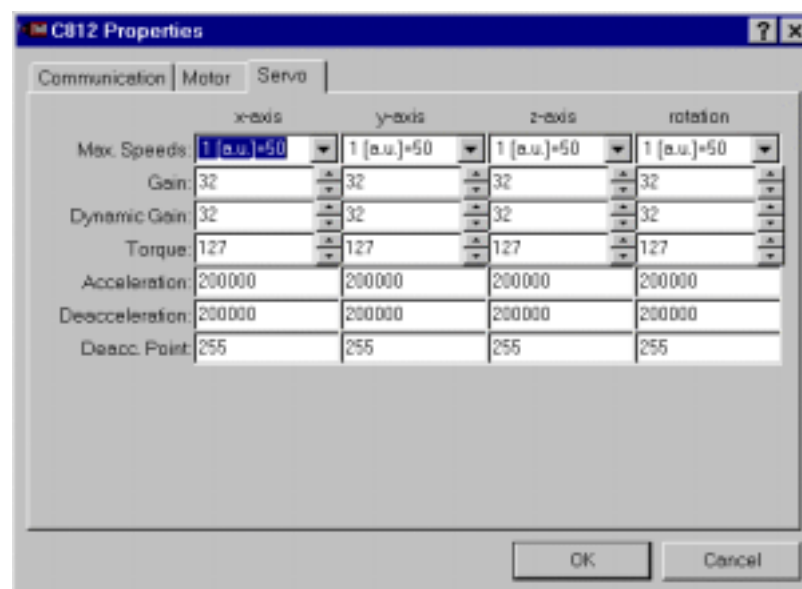


Figure 4.3.1.1.5 The 'Servo' dialog (PI C812 board).

4. MOTOR

4.3.2. Refresh

The command Reads current motor position and updates all values.

Under some circumstances it may happen that motor does not show a correct current position (for example after stopping motors the program can show position from time the motors were still moving, because internal watch thread exits before motors stop). Therefore it is recommended to apply this command. For C842 controller it is not necessary.

4.3.3. Go there

The command moves sample to the position defined by the position controls. If any of these values is equal to the current position, the sample is not moved in the corresponding direction. Each time you enter a value that exceeds the range defined in the 'Motor dialog', the program displays a warning.

4.3.4. Go to zero

The command moves sample to the origin. The origin is position at which internal motor coordinates are zero for all axes. The origin is set by the 'Reset motors' command in the application's system menu.

4.3.5. Reset board

The command restarts the controller by restoring all default hardware conditions. It should be used with a care if you have made any changes since powering up the controller (position, speed,). The command is available through the application's system-menu box.

4.3.6. Reset motors

The command sets position of one or more motors to zero. It brings up the 'Reset Motors' dialog box, which enables selecting an appropriate motor(s). Any motor or a group of motor may be selected. The command is available through the application's system-menu box.

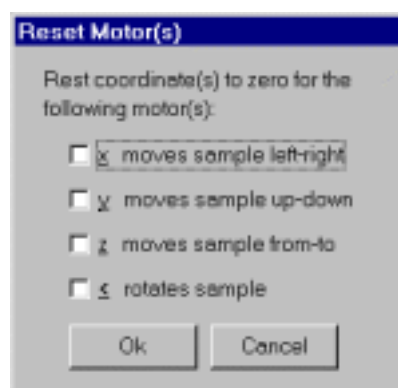


Figure 4.3.6.1. The 'Reset Motors' dialog box.

4.3.7. Recover origin

The command restores last known good position.

On the power-up, the controller always sets encoder counts to zero for all motors, regardless of values on the last power-down. Therefore, it can't remember positions before power-down. For this reason, the program saves encoder counts onto the disk, each time the motors stop. This command uses the last stored values to update positions. It is available through the application's system-menu box.

4.3.8. Application properties

The *'show axes image'* flag, when checked, causes that program shows image with definition of coordinate system. You should restart the program in order the

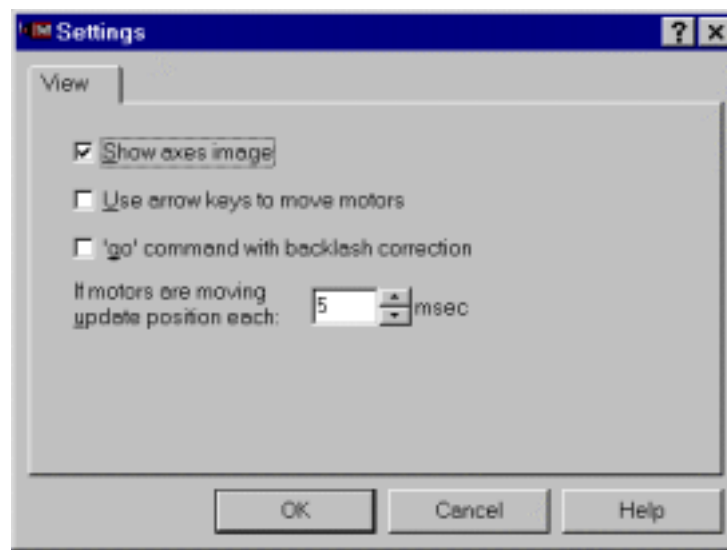


Figure 4.3.8.1 The 'View' dialog.

change takes effect. The *'Use arrow keys to move motor'*, when checked, enables usage of keyboard keys defined in table 4.2.1. It should be used with care because it may overlap with position controls (while in editing mode).

The *'go command with backlash correction'*, when checked, enables compensation of backlash after applying *'goto zero'* and *'go there'* commands. The backlash causes that error in motor position depends on direction of motion. The compensated is performed on such a way that motor always approach the target position from the same direction. Note that compensation is not applied after *'Stop'* command.

The *'update position'* defines period of updating position during motor motion. It is originally implemented for C812 ISA bus channel to decrease frequency of sending commands to the board.

Appendix

1. Common video-signal formats and standards

Video signal is a common name for electrical signals which carry all information necessary to reproduce 2-dimensional images collected in a well-defined time sequences. The most conventions in video techniques come from the time of the first commercial TVs and are still valid (to keep compatibility). For example, it is common that images are analyzed (in cameras) by scanning in horizontal lines, starting from the left-top corner¹. Each line is scanned always from left to right. This means that one characteristic of the video-signal is the horizontal analysis time. It defines the time when a line is started and when it terminates. The first TV used cathode ray tube (CRT) which needs some time to skip to the next line. This is included in the horizontal analysis time and called *horizontal blanking interval*. At all, for each line the video signal needs at least three timing information: start of the line, end of the line and blanking interval, which all together form *horizontal synchronization*. This is the first characteristic of any video signal. The second characteristic is the end of one frame and start of the next one. Here also should be taken into account that after one full frame the CRT skips again to the left top corner of the screen. It needs some time called *vertical blanking interval*. This vertical timing is called *vertical synchronization*. It is the second major video signal characteristic. The third and last is color information. For each line in the frame, the intensities of red (R), green (G) and blue (B) light components² are collected and then converted to voltage. For monochrome image, brightness is accumulated instead of color components.

1.1. RGB video-signal format

The *RGB video format* uses one wire for each of color components (R, G, and B) and two for synchronization. It is common video format for computer monitors. For example, a standard VGA monitor cable uses pins 1, 2 and 3 for Red, Green and Blue components, respectively and pins 13 and 14 for horizontal and vertical synchronization, respectively. There are subsets of this format which use only one wire for both synchronization signals (4 wires at all) or inserts them into all 3 color channels (3 wires at all). The Matrox Meteor and Matrox Meteor/RGB are supplied by DB-9 female connector that uses pins 1,2, and 3 for Red, Green and Blue components respectively, and pin 4 for both synchronization signal. This format is not suitable to transfer on longer distance because it needs too many wires or too much frequency bands for wireless transfer.

¹One full-analyzed image is called *frame*. In some cases the frame is divided into two equal parts called *fields* such that first part contains odd and the second even horizontal scan lines. A frame is analyzed and transferred by fields, that is, first are coded even and then odd lines. The systems which uses 2 fields are called *interlaced*, while the other, without fields, *non-interlaced*. The video signal used in TV is *interlaced*.

²Color representation using RGB system is natural because human eye has three types of cones sensitive to wavelengths that approximately correspond to red, green and blue lights. In addition, it is convenient for CRT displays where proportions of excitation of red, green and blue emitting phosphors produce colors when visually fused. There are many other color systems for example: CMY (cyan/magenta/yellow) and CMYK (black is added) used in printer because it refer to pigments, XYZ, Luv, Lab, HIS (hue/ saturation/ intensity)...All they can be connected to RGB by a transformation formula.

1.2. Composite video-signal format – monochrome

Composite video-signal format uses only one wire to carry all information (75 Ω coaxial cable). This is done by ‘analog coding’ of the natural RGB signals. The simplest case is a monochrome video signal, which transfers only brightness. Timing and amplitude of the signal are defined by standards. There are few basic standards for monochrome composite video-signal: *RS170* in the USA, *RS-330* in Canada, *RS-343* in Japan and *CCIR* in the Europe. The signal structure is more all less the same for all standards. Fig 1.2.1 shows composite monochrome video-signals without vertical synchronization part, defined by the R170 standard.

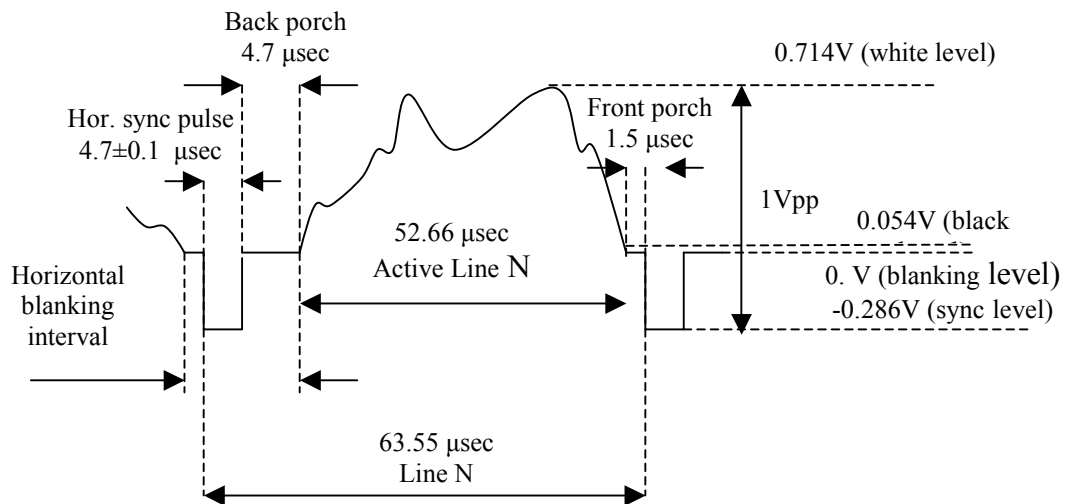


Figure 1.2.1. The RS170 video-signal specification.

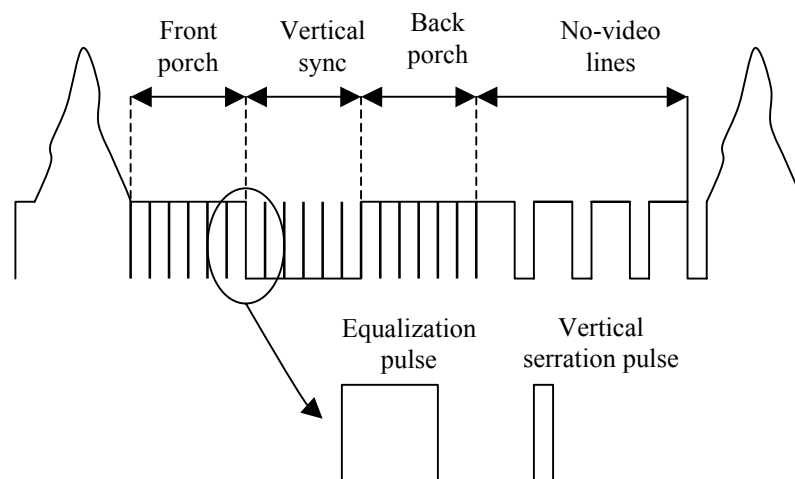


Figure 3.1.2.2 Vertical blanking interval for non-interlaced video signal.

The horizontal *synchronization pulse* indicates the beginning of the line. The video information starts after *back porch* and remains valid before beginning of the *front porch*. We see that only a part of the line contains valid data. The rest is called *horizontal blanking interval*. The *vertical blanking interval* (fig 3.1.2.2) lies between two frames. The name blanking comes from the fact that conventional cathode ray tube (CRT) displays must be extinguished (blanked) during this time interval. The front porch terminates on the next synchronization pulse. The frame terminates on vertical synchronization pulse. The RS170 video signal transfers 30 frames per second which means that number of lines per frame is $525 = (1/63.56\mu\text{sec})/(30*1/\text{s})$. The line period of 63.56 μsec corresponds to scan frequency of 15.734kHz. If the active, line which lasts after 52.15 μsec , is sampled (A/D conversion) with frequency of 12.2576MHz horizontal resolution is of $640 = 52.15 \mu\text{sec} * 12.2576 \text{ M/sec}$ pixels, as is proposed by standard. Because of the vertical blanking interval, which is cca 8% of total frame time, 485 from 525 vertical lines are active.

Characteristics of the CCIR format are as follows. The total amplitude is also 1Vpp (peak-to-peak), but sync level is at -0.3V and white level at 0.7V. The timing is as follows: the scan puls is $4.7\pm 0.2 \mu\text{sec}$, back porch 5.8 μsec , front porch $1.5\pm 0.3 \mu\text{sec}$, active line 52.0 μsec . All this gives 64 μsec for one scan line. The standard defines 25 frames per second which gives $(1/64 \mu\text{sec})/(25*1/\text{s})=625$ lines per frame. Horizontal resolution depends on sampling rate and for 14.77 MHz gives 768 pixels ($52.0 \mu\text{sec} * 14.77 \text{ M/s}$) as is proposed by standard. Because of the vertical blanking interval, there are 575 of 625 active lines. That is, the resolution is 768*572. This is summarized in the table 1.2.1.

Table 1.2.1 Basic parameters specified by RS170 and CCIR standard

	RS-170	CCIR	units
Interlaced	Yes	Yes	
Lines per frame	525	625	lines
Displayed lines per frame	485	575	lines
Effective horizontal resolution	640	768	pixels
Vert. sync. front porch	3.0	2.5	lines
Vert. sync. back porch	14	20	lines
Vert. sync. equalization pulse width	2.3	2.35	lines
Vert. sync. pulses	2.0	2.5	lines
Vert. sync. back porch	14	20	lines
Vert. sync. blanking	20	25	lines
Line frequency	15.734	15.625	kHz
Line duration	63.556	64.000	μs
Line active	52.66	52.0	μs
Line blanking	10.9	12.0	μs
Hor. sync. front porch	1.5	1.5	μs
Hor. sync. pulse width	4.7	4.7	μs
Hor. sync. back porch	4.7	5.8	μs
Output voltage	1	1	Vpp
Impedance	75	75	Ω

1.3. Composite video-signal format - color

The standards established for monochrome video signal are extended to comprise color feature. The RS170 is extended to RS170A which is well known as NTSC while CCIR is extended to PAL. The extensions are performed on the similar way. Again, the extended video standards defines signal which use only one wire,

that is, Composite video signals. The monochrome component is retained and it is called here luminance³. Usually it is denoted with Y. It is formed from so called γ corrected R, G and B which are denoted by R', G' and B'. This correction is made from the following reason: a human eye is more sensitive on the changes of luminance which is closer to black than to white. For example if luminance is represented by integers from 0 to 255 (where 0 is dark and 255 white) then codes 25 and 26 are much more distinguishable than codes 200 and 201. The lightness sensation can be calculated as intensity raised to a power of 0.33 to 0.45. So, the perception requires to transform (R,G,B) to (R'=R ^{γ} , G'=G ^{γ} , B'=B ^{γ}). This transformation is done on the camera side, before coding video signal. As x ^{γ} function is used, the transformation is called γ -correction. A similar γ -correction is made on the display side after decoding for the following reason: it is well known that CRT is a non-linear device, that is, the line output is not proportional to voltage input V (this non-linearity depends on the construction of electronic gun and not on the CRT's phosphor). The light output is approximately proportional to V ^{γ} , where γ is 2.5. So, at display side, one should make a process of precompensation, that is to calculate voltage from intensity by taking 1/ γ -power function. The value for γ depends on standard. The NTSC uses $\gamma=2.2$ due to a perception effect called 'dark surround' while the PAL uses a little bit larger value $\gamma=2.8$. Hence, the correction at display is 0.45- and 0.36-power functions for NTSC and PAL respectively.

The γ -corrected R', G', B' are used to form luminance Y' with the following formula (for both NTSC and PAL)

$$Y' = 0.299 R' + 0.587 G' + 0.114 B'$$

Such luminance is often called luma, to point out that it is derived from gamma corrected (R, G, B). The coefficients chosen for the Y' equation are related to the eye's sensitivity to the RGB colors. This luma signal maintains most of the bandwidth of the original RGB signals so that the details, represented by the highest frequencies, are retained in the monochrome portion of the picture. The three separate RGB signals must eventually be regenerated in the monitor to drive the red, green and blue CRT guns. To make this possible, three equations are required to recover the three signal variables. Two intermediate, color-difference signals, R'-Y' and B'-Y' are created. In the NTSC standard system these color difference signals are further combined into two color signals called I and Q, the PAL uses U and V. they are defined as follows

$$\begin{aligned} I' &= \cos(33^\circ) (R'-Y') - \sin(33^\circ) (B'-Y') = 0.596 R' - 0.274G' - 0.322 B' \\ Q' &= \sin(33^\circ) (R'-Y') + \cos(33^\circ) (B'-Y') = 0.212 R' - 0.523G' + 0.311 B' \\ U' &= -0.147R' - 0.289G' + 0.436B' \\ V' &= +0.615R' - 0.515G' + 0.100B' \end{aligned}$$

The (I', Q') or (U', V') are combined into chrominance signal by a technique called quadrature modulation. Similar to luma, here is used word chroma to point out that is derived from gamma corrected (R, G, B). NTSC and PAL standards use for the modulation a carrier of 3.58MHz and 4.43MHz respectively. The chroma signal is superimposed on the luma so they can be transferred by one wire. This is made at expense of introducing a certain degree of mutual interference. The synchronization portion is made on the similar way to a monochrome video signal with exception that

³ Luminance is luminous flux per unit solid angle and per unit projected area, in a given direction.

a *color burst* is added to the back porch. The color burst contains cca 3 μ sec of carrier signal. Its amplitude is $\frac{1}{4}$ of the level of white.

1.4. S-video-signal format

The *S-video* (also called *Y/C*) format is made of two parts, luminance Y and chrominance C. The synchronization signals are part of luminance Y in a similar way to Composite video signal. These format uses cable terminated at each end with a four-pin DIN connector. Although it may appear to be a single cable, internally it has two 75 Ω coaxial or twisted pair cables to carry the separate Y and C signals.

1.5. Component video-signal format

The *Component video* format treats luminance, R'-Y' and B'-Y' as a separate signals. It requires three 75 Ω coaxial cables, which carry the Y', R'-Y', and B'-Y' signals.

2. Processes and threads

A process is a program that is loaded into memory and prepared for execution. Each process has a private virtual address space. A process consists of the code, data, and other system resources.

Threads are the basic entity to which the operating system allocates CPU time. Each process is started with a single thread, but additional independently executing threads can be created. All threads of a process share the virtual address space and can access the global variables and system resources of the process.

Table 2.2.3.2.1. The Windows priority classes and levels.

Level	Priority class	Priority value	Level		
1	High	Idle	10	Normal (frg)	Above normal
	Normal	Idle	11	High	Lowest
	Idle	Idle		Normal (frg)	Highest
2	Idle	Lowest	12	High	Below normal
3	Idle	Below normal	13	High	Normal
4	Idle	Normal	14	High	Above normal
5	Normal (bkg)	Lowest	15	High	Highest
	Idle	Above normal		Normal	Time critical
6	Normal	Below normal		Idle	Time critical
	Idle	Highest	16	Real time	Idle
7	Normal (frg)	Lowest	22	Real time	Lowest
	Normal (bkg)	Normal	23	Real time	Below normal
8	Normal (frg)	Below normal	24	Real time	Normal
	Normal (bkg)	Above normal	25	Real time	Above normal
9	Normal (frg)	Normal	26	Real time	Highest
	Normal (bkg)	Highest	31	Real time	Time critical

The system scheduler controls multitasking by determining which of the competing threads receives the next slice of the processor's time. Each time slice consists of a fixed number of clock ticks. The currently executing thread is suspended when its time slice elapses, allowing another thread to run. The scheduling dynamic priority of each thread is determined by the following criteria:

- ‘*priority class*’ of its process (real time, high, normal, or idle)
- ‘*priority level*’ of the thread within the priority class of its process (lowest, below normal, normal, above normal, highest)
- dynamic priority boost, if any, the system applies to the thread's base priority

The ‘*base priority*’ of each thread is the thread's priority level within the priority class of its process. Its values are shown in the table 3. The scheduler uses this base priority to determine a thread's dynamic priority, which it uses to make scheduling decisions. A thread's dynamic priority is never less than its base priority.

CONTRIBUTORS TO DRAFTING AND REVIEW

Bogovac, M. Ruder Bošković Institute, Croatia

Bamford, S. International Atomic Energy Agency

Chinea-Cano, E. International Atomic Energy Agency

Markowicz, A. International Atomic Energy Agency

