Introduction to the Use and Functions of Xilog Plus

This manual has been designed to provide information relative to the use of Bore-Milling machines manufactured by the SCM Group.

The information contained in this manual is the property of the SCM Group and may not be reproduced or disclosed without authorization.

SCM Group cannot be held liable for incorrect use of the information contained in the manual.

Index

1. How to Use the Computer

1.1 Using Keyboard and Mouse

In order to use Xilog Plus, it is necessary to become familiar with two instruments that are essential for "speaking" with the computer: the keyboard and the mouse.

How to use the keyboard

The keyboard can be used to send commands by pressing a single key or a combination of keys.

Single key. To use a single key just press that key and then release it. If the key is held down for a few seconds, in many cases the effect is repeated: for example, if we want to write the letter A and we hold key **A** down instead of pressing it just one time, the screen will show a series of A: AAAAAAA.

Combination of keys. For some commands to be sent using the keyboard more than one key must be pressed at the same time. Lets take, for example, the key combination **CTRL** and **X:** in this case the **CTRL** key must be pressed down and the **X** key released. This instruction is indicated as: "press keys **[CTRL]** + **[X]** ".

How to use the mouse

The mouse is a device with a small rubber ball underneath and usually two keys on top.
By dragging the mouse over a hard surface the movement of the ball causes the pointer (which is usually shaped like an arrow) to move over the computer screen. In this manner, the cursor can be moved over particular graphic elements that appear of the screen (for example, drawings, menu items, etc.), each of which corresponds to a command: by pressing one of the mouse keys (usually the left key) the program will receive a command that corresponds to the graphic element over which the cursor is located. This operation if referred to by the expression "to click".
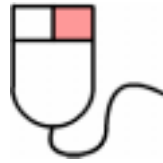The instruction to "click on the **OPEN** button" will therefore be interpreted as moving the mouse pointer over the graphic called **OPEN**...

and then pressing the left mouse button.

In most cases the left mouse key is pressed, but some commands can also be sent to the computer by pressing the right key.

1.2 Starting a Program: the Start Menu

Xilog Plus must be installed on a computer equipped with Windows operating system. To start the program in Windows, usually the **START** menu that appears in the lower left-hand corner is used.
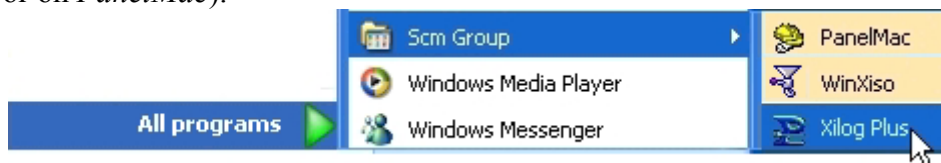Click on **START**…

then click on the item **ALL PROGRAMS**…

then click on the name of the program (for Xilog Plus click on *Scm Group,* then click on Xilog Plus or on *PanelMac*).
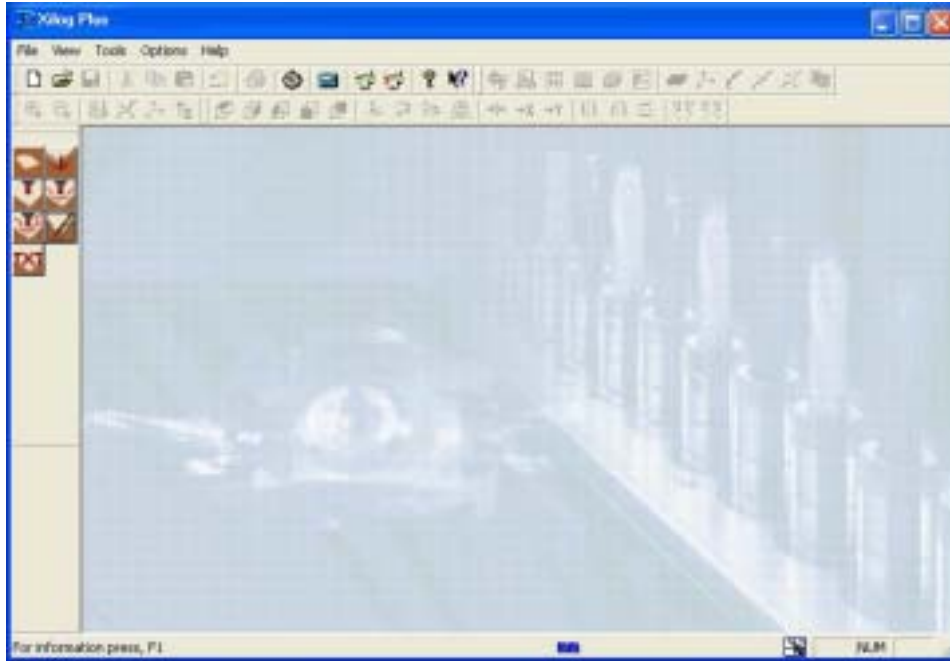

The **START** button is located on the bar called *Applications Toolbar.*

1.3 The "Window" Interface

The type of graphic interface used by Xilog Plus is called "window" (the same as the Windows operating system).
Windows are rectangular spaces of modifiable size that group Xilog Plus functions and work environments.

When we are inside a window, we can give the computer commands with the mouse, by clicking on the buttons and menus, writing in the appropriate spaces with the keyboard, reading information, etc. More than one window can be opened at the same time.

All Window style windows have three buttons in common that are located in the top right-hand corner.
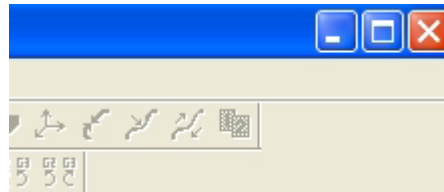
The **first button** (from the left: REDUCE TO ICON button) serves to temporarily hide a window. To open the window again, just click on the rectangle containing the window name that appears on the *Application toolbar* below.

The **second button** (ENLARGE button) serves to *enlarge* the window to *full screen* (so that it occupied the entire computer screen) or *reduce* it. To go from one condition to the next just click on the button.

When a window is *open at full screen* the icon for that button looks like this:

However, when the window is reduced, the icon looks like this:

In this second case, the size of the window can be changed as necessary. Bring the mouse pointer to the edge of the window so that the arrow becomes a double arrow, then keep the left mouse key keep pressed and move the mouse: the edge of the window will move. To change both the vertical and horizontal size of a window *at the same time* repeat the operation

outlined about by moving the mouse pointer over one of the window corners.

The window can also simply be moved around on the screen without changing its size. Move the mouse pointer over the blue bar on top (called *title bar,* because it usually shows the name of the program and the title of the open file), keep the left mouse key pressed and move the mouse: the window will move.

The **third button** (CLOSE button, marked with an X) is used to close the window. When the main window of a program is closed this will cause the program to close down as well.
The move, reduce and close window functions are also available from a *menu,* that is, a drop down list of commands: click on the title bar with the <u>right</u> mouse key and using the left mouse key click on the item desired.
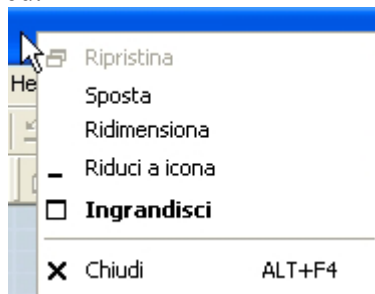
As shown, the writing "ALT+F4" appears near the word CLOSE to indicate that it is also possible to close the window using the keyboard, by pressing the keys [ALT] + [F4]**.**
This is a feature of the windows interface: there are many commands that can be sent to the computer in various ways. For example, as seen above, to close a window you can:

- click on the close button (the one in the upper right-hand corner marked with an X);
- click on the work Close in the menu of the title bar;
- press keys [ALT] + [F4] on the keyboard**.**

Choose the method you prefer!
Therefore, in a windows interface we have various methods for sending commands to a program. Here are some examples taken from the Xilog Plus interface.
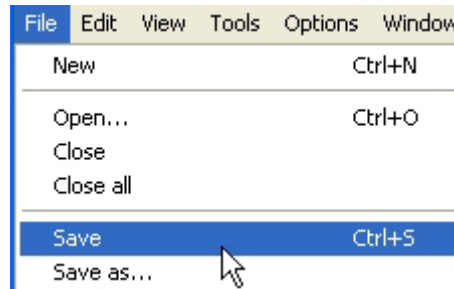
Menu

The **menus** group the commands into lists. For example, by clicking on FILE with the mouse, a list of commands will appear.
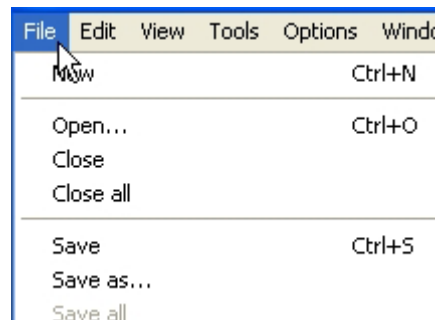
To select a command from the list just click on the command name. Hence "click on
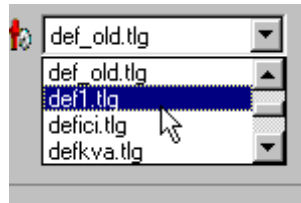
FILE/SAVE" indicates that it is necessary to click on the **FILE** menu and then click on **SAVE** in the commands list.

At times the menus are contained in a box. In this case, it is necessary to click on the black arrow so that a list of the available items appears

then click on one of the items to select it.

Icons

Many of the commands found in the menu are also represented by *icons,* that is, by buttons with a drawing to represent the command to be activated. The icons can be grouped in bars.


In this case as well, to implement a command using the corresponding icon, just click with the mouse on that icon.

Icons facilitate and speed up the selection of a command. To learn to recognize the command associated with each icon, it may serve to bring the mouse pointer over the icons for a few seconds: the name of the relative command will appear.
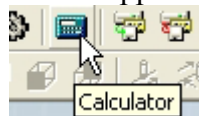
The interface can also signal particular program or machine conditions. For example, the letters "**mm**" that appear on the status bas (below) indicate that millimeters is the set unit of measure in Xilog Plus (instead of using indications expressed in inches).
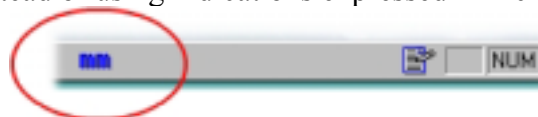
Buttons

The buttons may not be occupied by icons but words that directly indicate some of the more common functions. For example, many windows have an **OK** button that serves to confirm an action and a **CANCEL** button that serves to cancel an action. The figure below shows these buttons in the window that appears when we give the command to create a new program in Xilog Plus: when the **OK** button is clicked this creates a new program (if **PROGRAM** has been

selected in the window) and if the **CANCEL** button is clicked the window simply disappears.
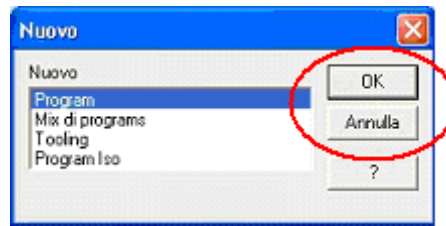
Test boxes

The text boxes serve to enter text using the keyboard. To write inside a text box click above the box with the mouse: an intermittent bar will appear to indicate that it is possible to insert text.
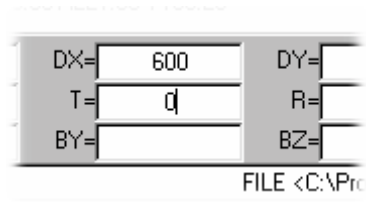
1.4 Files and Folders

1.4.1 Xilog Plus Files

Work programs written with Xilog Plus: are saved (that is, stored by the computer) under the form of *files*. A file is nothing more than a bunch of data which the computer stores in its memory. In this manner, after a program has been written and saved in memory, it can be used again at another time. In the Windows operating system, files are represented by icons. For example, the icon for a Xilog Plus program is represented in this manner:

Xilog Plus actually uses various types of files: mix, program, tooling files, etc. Each type of file is identified by its own icon.

Another important feature that distinguishes one type of file from another is the *extension*, that is, a three-letter abbreviation that is added to the file name, separated by a dot.
For example, if a work program is created and called DOOR, being a program (PGM extension) its complete name will be DOOR.PGM; if it is a mix (MIX extension), its complete name will be DOOR.MIX; if it is a tooling file (TLG extension), its complete name will be DOOR.TLG, and so on.
To simply things, files are collected together into *folders*. The folders are represented with this icon:

We can imagine the folders as empty boxes used to collect files in an orderly manner. Once installed, Xilog Plus makes available a folder into which to collect the various types of files: the JOB folder.

1.4.2 Opening and Closing a File

File opening and closing operations are carried out inside appropriate windows. For example, to open a stored file, after clicking on the **FILE/OPEN** menu this window will appear on the screen:
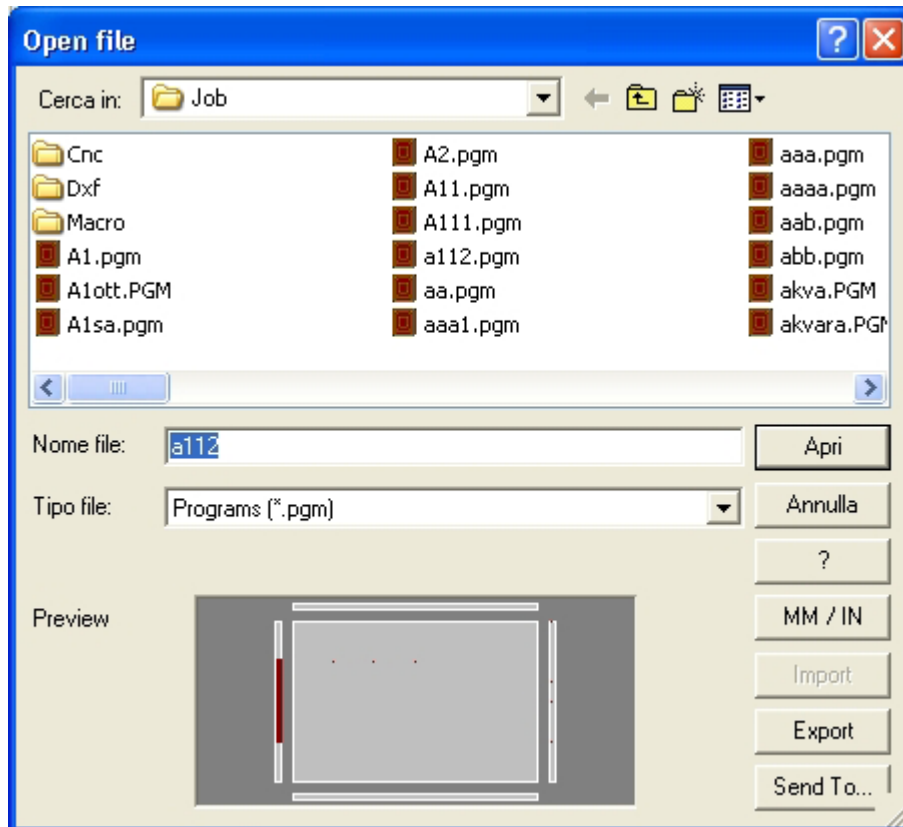
To open a Xilog Plus file stored in the JOB folder we will have to:

1) if the JOB folder is not already selected, find it in the **SEARCH IN** drop down menu and click on the relative icon to select it;
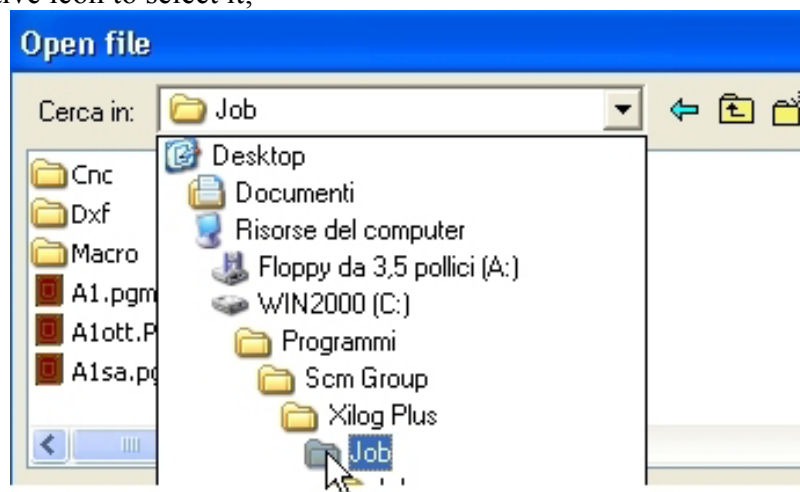
2) click on the **OPEN** button.
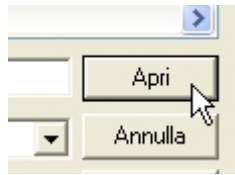
3) The files contained in the folder will appear. Now select the file you want to open and click again on the open button: the file will open in Xilog Plus.

N.B. The drop down menu of the **FILE** type menu displays only some types of files. For example, if **PROGRAM (*PGM)** is selected only those types of files will be displayed. To display all files, select **ALL FILES.**
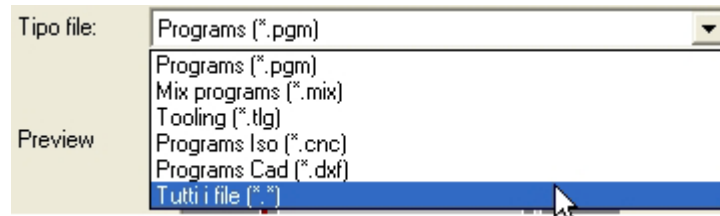
The *file save* window is similar to the file open window. Xilog Plus programs must be saved in the JOB folder that will eventually be selected and opened. Xilog Plus will automatically assign the correct extension to the file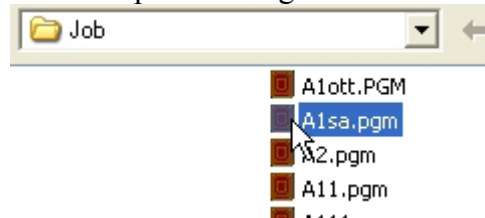; all that needs to be done is to give the file a name by writing in the **FILE NAME** box and saving it by clicking on the **SAVE** button.
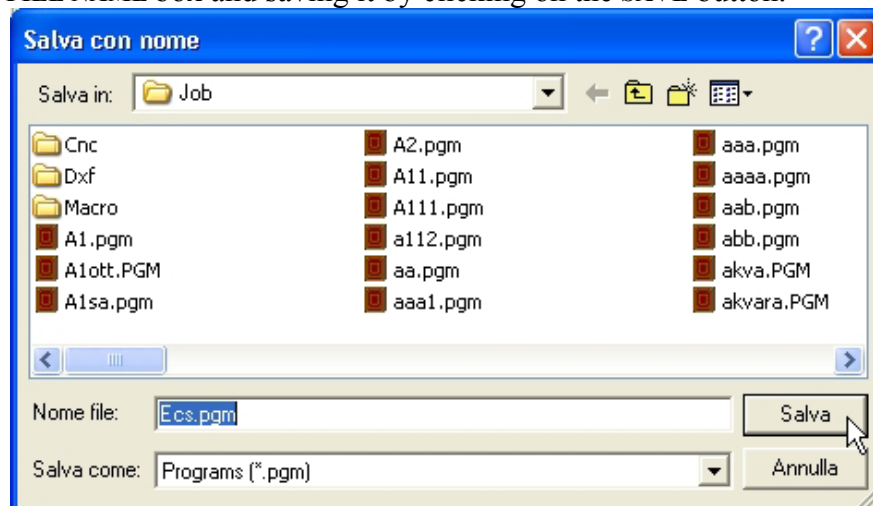
N.B. DXF files imported in Xilog Plus can be saved both in PGM format and in XXL format: in this case just select one of the two from the **SAVE AS** box (see the manual by the Xilog Plus editor – Importing a DXF file).

WARNING. Each file must have its own name which must be different from other same types of files contained in the same folder. If, for example, the PGM folder already contains a file called DOOR.PGM and the file is saved as DOOR.PGM the old file could be cancelled by the new one!
To save a file, click on **FILE/SAVE** or **FILE/SAVE AS** from the menu.
The difference between these two commands is:

- <u>SAVE</u>. When we save a file for the first time we click on the **FILE/SAVE** menu, the save window opens as shown above, and we assign a file name so that the file can be stored. The next time the file is saved with the **FILE/SAVE** menu the save window will not appear again: if changes are made, the old version will be saved with the same name and in the same location as the previous one.
- <u>SAVE AS</u>. With the **SAVE AS** function the save window appears each time. This is very useful if we want to change the name or storage location for a file already saved: in this manner the change file will be saved without canceling the original file.

In the same manner as the Xilog Plus JOB folder, the floppy disk and CD-ROM can also be selected from the **SEARCH IN** drop down menu. If, for example, we want to view a program contained in a floppy disk in Xilog Plus, insert the floppy disk into its drive (the appropriate slit found on the computer) and click on the **FILE/OPEN** menu, select the floppy disk icon in the **SEARCH IN** drop down menu and click on the **OPEN** button to view the contents.
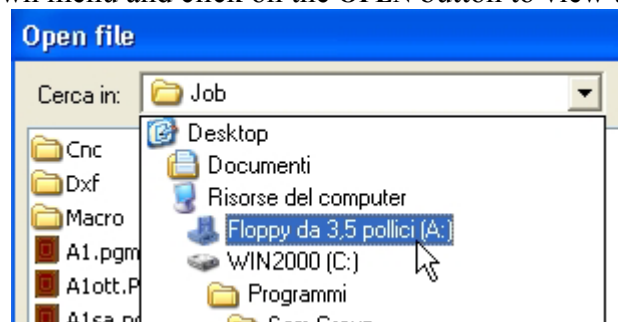
1.4.3 Creating a Folder, Deleting and Renaming a File

Both the **OPEN FILE** and **SAVE AS** windows have a button for creating a new folder (that can also be created inside other folders). Just click on the button…

and give the folder a name by typing it on the keyboard.
Once created, files and folders can easily be deleted or renamed. Click on the icon for a file or folder using the <u>right</u> mouse key: a quick choice menu will appear from which it is possible to click on **CANCEL** (to delete) or **RENAME** (to rename).

2. Introduction to Xilog Plus

2.1 What is Xilog Plus

The Xilog Plus software of the SCM Group is dedicated to managing bore-milling machines with numeric control. It is made up of two software packages that can either be installed together or separately: the Programs Editor and the Machine Panel.

The **Programs Editor** is dedicated to creating and changing programs containing instructions that control the function and behavior of the machine and its parts. It can also be installed on a normal office PC. Other editors suitable for creating programs can be used in its place.

With **Machine Panel** it is possible to run programs created by the editor. It also enables semi-automatic and/or manual management of machine movements. It is installed on the PC hosted in the electrical cabinet.

2.2 The Language

The machine controlled by Xilog Plus carried out the work by interpreting instructions contained in the programs. The instructions generally contain information on the operation to be carried out and a series of parameters that specify the manner in which the work must be performed.
The first thing to get to know is the system for orienting the boring – milling machine used by Xilog Plus in the work space, which makes it possible to instruct the machine where to move the milling heads. This system takes into consideration three directions that go straight across the work surface: the length, the width an the depth. The three directions are indicated with three letters, according to the Cartesian axis system:

X (for the length, left-right direction);
Y (for the width, front-back direction);
Z (for the depth, up-down direction).
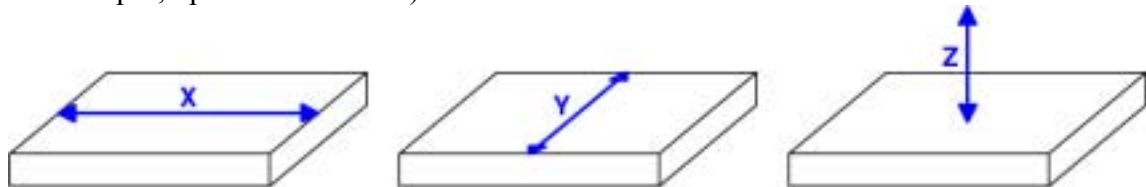
Consequently, in order to tell the machine how to move the milling heads, it is necessary to measure the distance of the movements in the space according to these three axis, starting from a starting point called **axis origin**.

To better understand this system, lets take the game of chess as an example. The chessboard is divided into columns (vertical) and lines (horizontal). All the squares are located at a cross point between a line and a column. For example, to indicate the square in which the rook is located in the following picture, we would say that the rook is located in square C6, that is, at the crossing between column C and line 6.
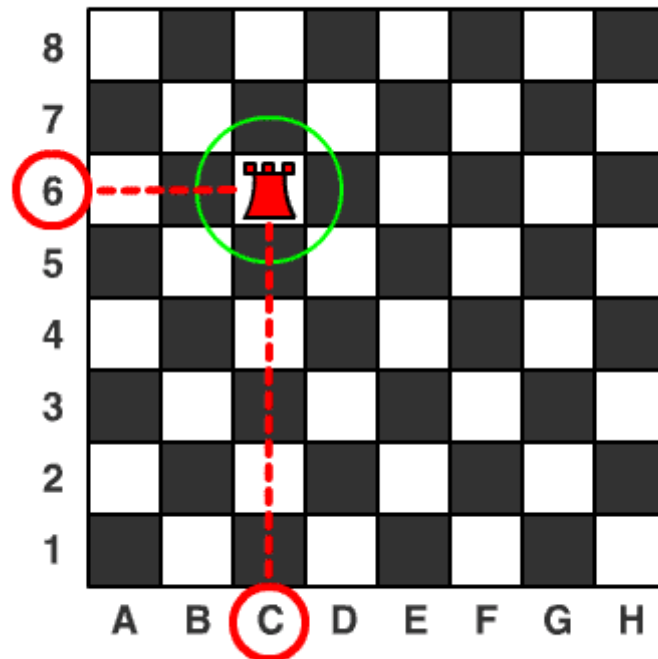
In this case, lines and columns are numbered starting from a point of origin that is found in the lower left hand corner of the chessboard.

In a Cartesian axis system we could say that the rook is in position X=3 and Y=6, to indicate a distance of three squares along the X axis (the columns) and 6 squares along the Y axis (the lines).
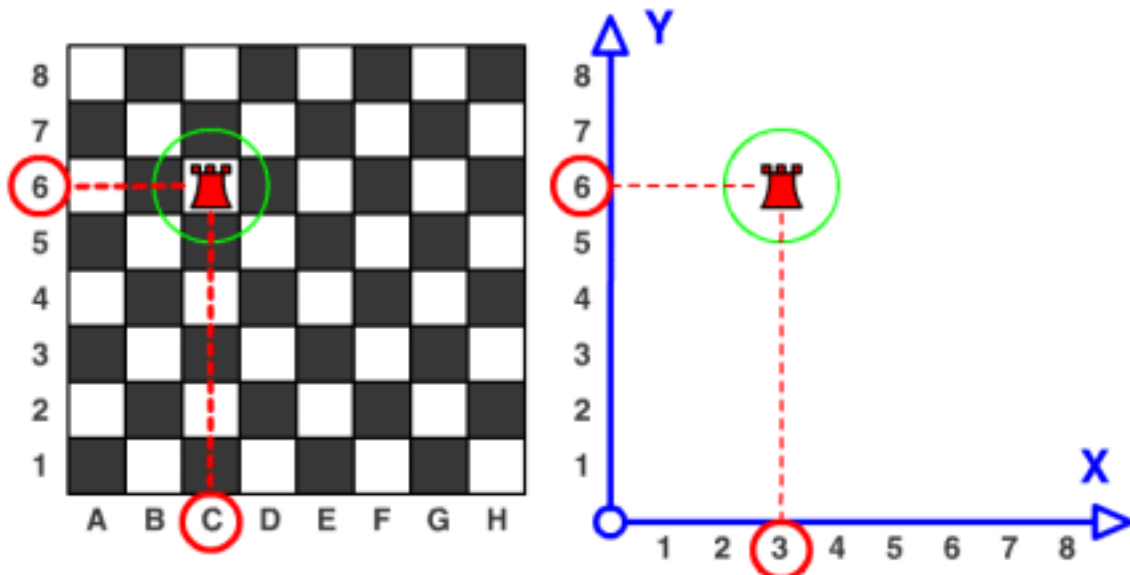
These two coordinates are sufficient to indicate a square on the chessboard. In the case of our bore-milling machine it is necessary to add a third coordinate that indicates the vertical movement along the depth. This third coordinate is indicated as one third of the axis called **Z**. We would therefore have this type of coordinates diagram:
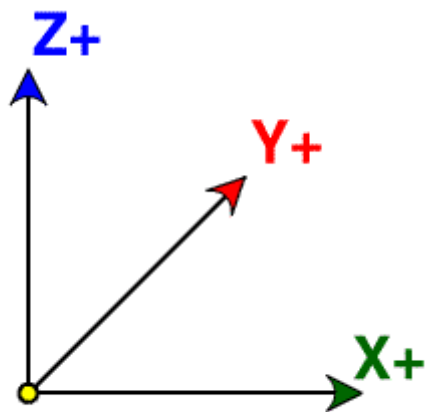
These coordinates are indicated with positive values. We could however also move along the three axis from the side opposite side to the movement starting point. In this case, the distance is always measured starting from the origin, but is indicated with negative values.
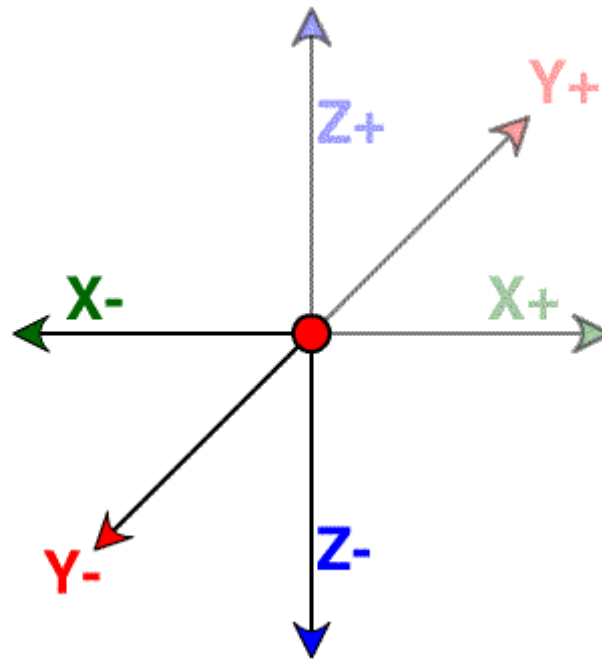
With this system we can therefore indicate any point in a space. For example, to indicate the point on a wooden panel where the machine must make a hole, we will consider that point as the result of intersection between coordinates X, Y and Z.

In the example illustrated in the figure below, the machine makes a hole on the surface of the panel, in the point indicated by coordinates X=100, Y=100, Z=-10.
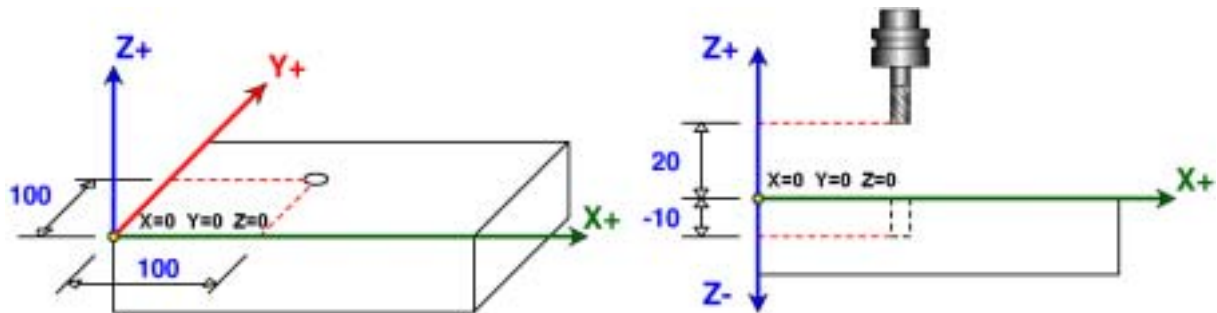
It should be noted that, in this example, depth coordinate Z is indicated by a negative number.

N.B. The position of the axis origin and the manner of use of the Z coordinate can vary depending on the type of machine used. See specific information in the Xilog Plus manuals.

In the previous example, the milling head will complete four distinct movements:

1. one movement to position itself over the point to be bored;
2. one movement to move closer to the point to be bored;
3. one movement to make the hole;
4. one movement to exit the hole once made and return to the position at point 2.

The program language for these movements corresponds to something like the following type of instructions:

XB X=100 Y=100 Z=-10 D=20

where:

XB	Name of the instruction that indicates the boring operation.
X=100	Position of the center of the hole compared to the X axis (in millimeters).
Y=100	Position of the center of the hole compared to the Y axis (in millimeters).
Z=-10	Position of the center of the hole compared to the Z axis (in millimeters).
D=20	Outside work quotas: indicates the point in which the boring head positions itself before making the hole and after completing it (in millimeters)

3. Working with Xilog Plus

3.1 The Work Phases

Using Xilog Plus means organizing the work into a series of phases. In general, a work cycle includes:

1) Turning on and **calibrating** the machine. After the machine has been switched on and the systems loaded, the operator must proceed with *calibrating* the machine, which automatically prepares it to run the programs. This phase is only carried out once, unless the power is switched off from the entire machine, in this case calibration must be repeated after the machine is switched on again.

▶ *see: Machine Panel manual*

2) Creating a **tooling file**. Once calibration is performed, it is necessary to tell the machine which tools have been loaded and their characteristics. This operation is called *tooling*. The tooling can be stored and used again various times.

▶ *see: Xilog Plus editor manual*

3) Creating a **program**. The *program* is a list of working instructions written with an editor (either the Xilog Plus editor or another editor) and stored in a file.

▶ *see: Xilog Plus editor manual*

4) Program **Test**. Before working on the piece, it is highly recommended that a complete program simulation be performed to check for any errors or irregular machine behavior (for example, collision of tools with surface or erroneous tool movements over the piece).

▶ *see: Machine Panel manual*

5) Actual program **implementation**. Once the test is passed, proceed with actual program implementation by once again verifying that the first work was correct.

▶ *see: Machine Panel manual*

3.2 A Concrete Example: Making a Door

Now that we have had an overview of the phases in which the work generally takes place, lets examine a concrete example in greater detail: the preparation of a door.

N.B. The following examples of machine programming are purely indicative and only serve to clarify the methods for use for Xilog Plus software and making them more understandable.

Our scope is to produce a door. The first thing to do will be to find a drawing of the door with all the measurements. Lets suppose therefore, that we have the drawing shown below:
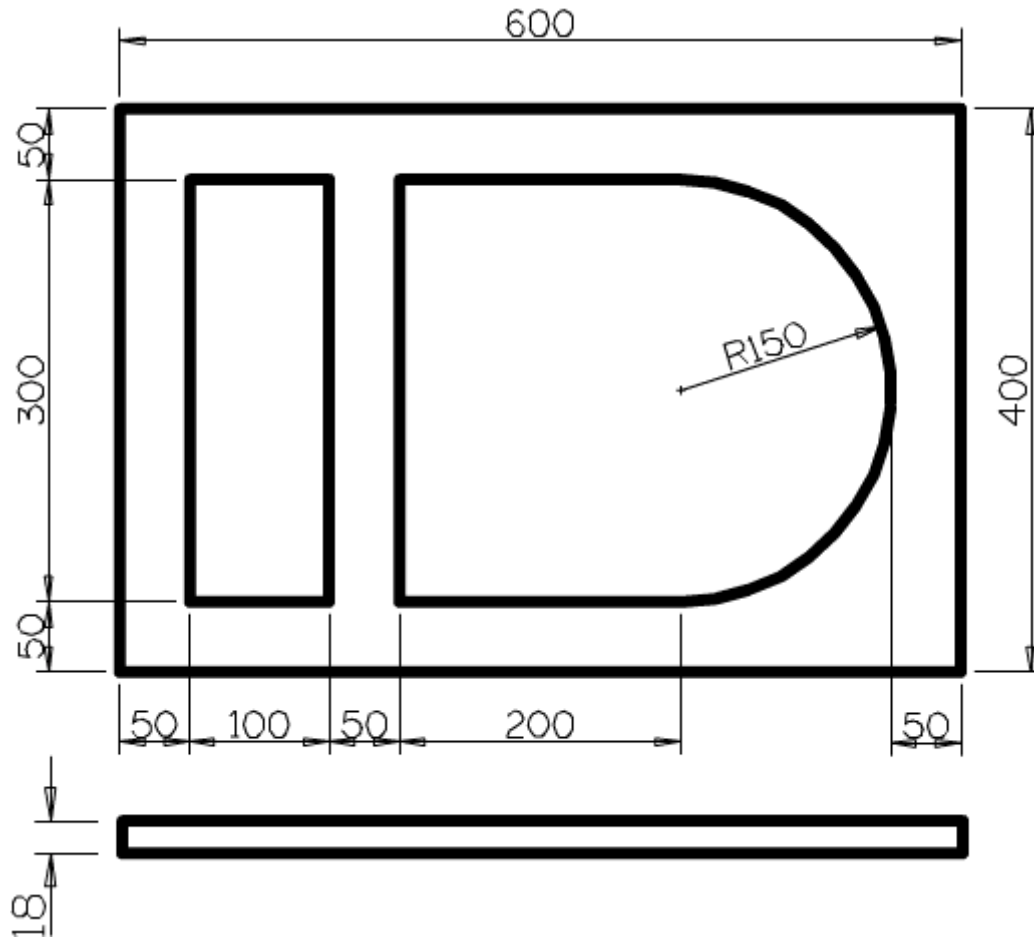
The nest step consists in deciding which work tools must be loaded onto the machine. For example, lets suppose we need three milling devices:

- a cylinder cutter to roughen the panel;
- a shaped cutter to trim the external edge of the panel;
- a second shaped cutter to trim the inside of the panel.
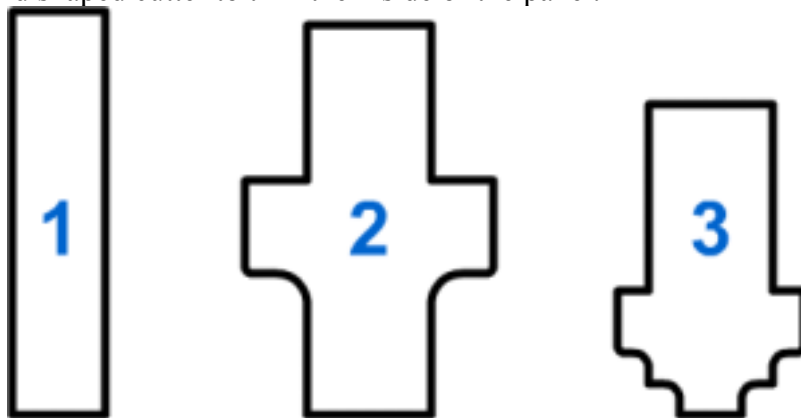
These three tools must be mounted onto appropriate cones and positioned in the store of the machine.

After being positioned, Xilog Plus must be advised which tools are present in the store and what their characteristics are. This operation, called "tooling", takes place in the following manner:

- first of all, the tools must be measured according to certain criteria;
- the characteristics of the tools must then be entered into a tooling file;
- the file must be saved in the computer's memory bank, so that it can be used when a program is created to work the panel.

For example, we will take the following measurements:

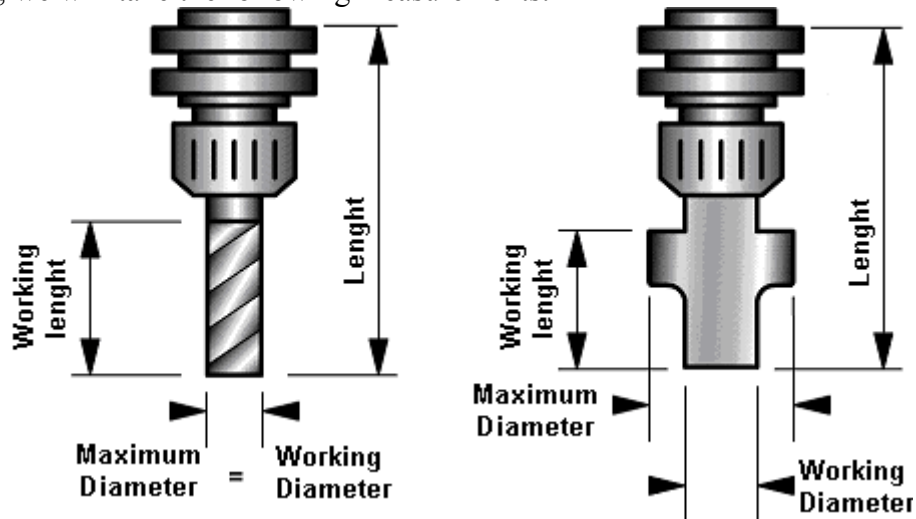

This information will be entered in the tooling file together with other parameters (such as rotation speed, feed-path speed, etc.).

List parameters	Value
Type (VUOTO=N.D.,P,F,D,T,S):	F
Cutter Length:	+100.00
Cutter Diameter:	+20.00
Working Length:	+60.00
Working Diameter:	+20.00
Wear coefficient in Length:	+0.00
Wear coefficient in diameter	+0.00
Max wear in Length :	+0.00
Max wear in diameter:	+0.00
Correct Length:	+0.00
Correct Diameter:	+0.00
Max. speed (m/m):	+10.00
Standard speed(m/m):	+10.00
Max.Rotation(g/min):	+18000.00
Rotation standard (rpm):	+18.000
Direction of rotation (+=dx, -=sx):	+
Speed G0/B (m/m):	+3000.00
Tool Counter position in X:	E1

At this point we are ready to write the program which will tell the machine which operations to perform on the panel to produce the door

What does writing a program with Xilog Plus mean? To understand the concept of "program" lets imagine that we want to but a newspaper. To do this we have to perform as series of specific actions:

1. go out of the house;
2. get into the car and turn on the engine;
3. drive to the newsstand;
4. ask the newsagent for the paper;

5. pay;
6. get into the car and turn the engine on;
7. drive home;
8. park the car;
9. enter the house;
10. comfortably read our paper.

For us these actions are in fact obviously a habit, so we perform them without thinking about it. To make our door however, we have to explain to the machine, in detail, the action to be taken. This is done precisely by writing a *program*, in other words, a detailed list of operations that we want the machine to perform for us.

A program written with Xilog Plus presents itself as a series of text lines, each of which corresponds to an instruction. Each instruction has its own name, expressed as an abbreviation, and serves to indicate to the machine one or more operations to be performed.

We can distinguish come of the parts inside the program:

- **Heading**, or header. The heading starts with the letter H and is the first instruction (mandatory) in a program. It contains some general information: for example, the panel measurements, the unit if measure used in the program (millimeters or inches), etc.
- A series of **profiles**. The profile is a geometric route which the tool must follow in a continuous manner. Normally, each profile starts with instructions for automatic entrance (GIN or XGIN instructions: indicate the manner in which the tool must near the panel to start working), followed by instructions to start milling (G0 or XG0 instructions: indicate the point in which the tool starts to cut the panel), and ends with instructions for automatic exit (GOUT or XGOUT instructions: indicate the manner in which the tool moves away from the panel at the end of the profile). Instructions for working the profile, which can be of various types: milling, boring, movements, are written between the automatic entrance and exit.

```
00001   H DX=600.00 DY=400.00 DZ=20 -A C=0 V10 T=0 R=1 *MM /"antina.tlg"
00002   ;**** RIFILATURA DEL PANNELLO ****
00003   XGIN Q=0 R=2 G=1# C=1
00004   XG0 X=0 Y=0 Z=20 D=-20 T=101 F=1 K=0 P=0
00005   XL2P X=0 Y=400 K=0
00006   XL2P X=600 Y=400 K=0
00007   XL2P X=600 Y=0 K=0
00008   XL2P X=0 Y=0 K=0
00009   XGOUT Q=0# G=1# F=1 C=0 K=0 P=0
00010   ;**** INCISIONE RETTANGOLARE ****
00011   XGIN Q=1# G=1# F=1 C=2 K=0 P=0
00012   XG0 X=50 Y=200 Z=10 D=-20 T=102 F=1 C=2 K=0 P=0
00013   XL2P X=50 Y=350 K0
00014   XL2P X=150 Y=350 K0
00015   XL2P X=150 Y=50 K0
00016   XL2P X=50 Y=50 K0
00017   XL2P X=50 Y=200 K0
00018   XGOUT Q=1# G=1# L=10 F=1 K=0 P=0
00019   ;**** INCISIONE A BUGNA ****
00020   XGIN Q=1# G=1# F=1 K=0 P=0
00021   XG0 X=200 Y=200 Z=10 D=-20 T=102 F=1 K=0 P=0
00022   XL2P X=200 Y=350 K0
00023   XL2P X=400 Y=350 K0
00024   XAR2 X=400 Y=50 K0 r=150 G=3#
00025   XL2P X=200 Y=50 K0
00026   XL2P X=200 Y=200 K0
00027   XGOUT F1 C0 K0 P0 Q=1# G=1# L=10
00028   ;**** FINE PROGRAMMA ****
00029   XN X=2000 Y=0 T=101
00030
```

When program writing is complete, we can pass to actual working.

To carry out the program it is necessary to use the Xilog Plus Machine Panel. We will have to:

- from the Machine Panel we will have to select the tooling to be used to produce the door;
- load the Machine Panel program (directly from the computer memory or by transferring it by floppy disk, if it was written on an office computer).

At this point the machine is ready to produce the door.

Before implementing the program, work simulation is suggested (without a panel on the work surface) to check that the program has been written properly and that work has been programmed as desired. On the Machine Panel we can verify if the machine movements are correct or whether they can cause damage (for example, collide with the work surface); in case of dangerous movements, we can slow down or interrupt the work and adjust the machine's movement speed.

After this test has been performed, the panel to be worked must be arranged on the work surface in the desired position and the work can begin. In this case as well, it is suggested to control that the work is being performed properly before giving way to a series.

3.3 Running More than one Program at the Same Time: Program Mixes

In the previous paragraph we saw, with the example of the door, all the phases of a complete production cycle, from the creation of a program to its execution on the machine. Often there is a need to produce various products, each of which requires the creation of an appropriate program. In this case, instead of implementing each different program separately, with Xilog Plus it is possible to create a list that includes them all: in this manner it will be sufficient to have the machine run the list to automatically obtain the implementation, in series, of all the programs.

These lists of different program are called *program mix*. The program mix is a type of Xilog Plus program but, unlike a normal program, the mix contains only one type of instruction which serves to recall the programs to be carried out: when we create a mix therefore, we must simply tell Xilog Plus, using the appropriate command, which programs are to be included in the list.

▶ *see: Xilog Plus editor manual*

3.4 Saving Time: Optimizing the Programs

When Xilog Plus has to carry out various work (for example, in case of a mix), moving tools over the work surface can be repetitive and time wasting. To remedy this inconvenience a program can be *optimized.*
Program optimization reorders tool movements so that work is carried out in as little time as possible and the need to stop the machine to change tools is reduced to a minimum.
Optimizing a program with Xilog Plus is very simple: just select the optimization command and indicate that type of optimization desired, Xilog Plus will carry it out automatically, without further intervention by the user.

▶ *see: Xilog Plus editor manual*

3.5 Personalizing the Program: Parametric Programming

Supposing we created a program, HOLES. PGM, to make four holes in the corners of a rectangular panel that measures 600 millimeters long and 400 millimeters wide. The holes are all at a distance of 50 millimeters from the panel edges.
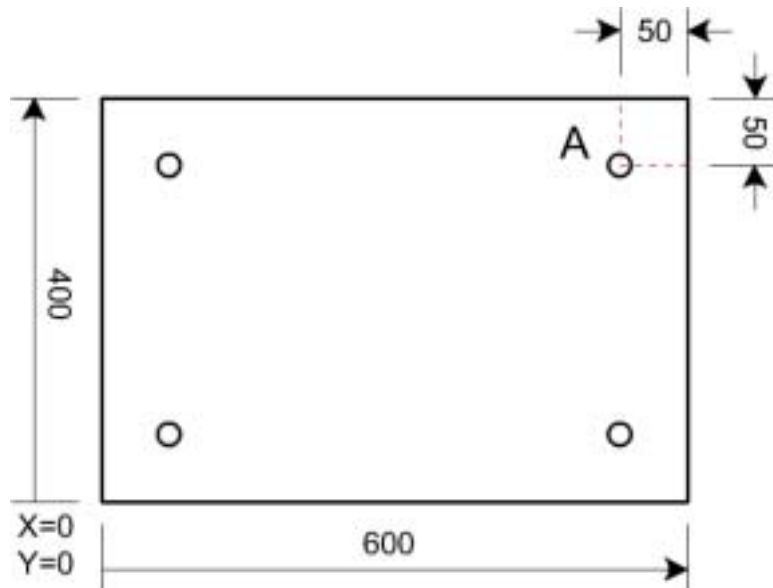
If we want to run the HOLES.PGM program on a panel of a different size, to maintain the same distance between the holes and the edges, we need to change the program by entering new hole coordinates recalculated on the measurements of the new panel and save a second version of the program with, for example, the name HOLES2.PGM.

Often there is a need to carry out the same work on different panels of various sizes. In this case Xilog Plus allows us to create a single program that can be simply and quickly adapted to the various needs. This type of program is called *parametric program.*

Lets see an example of how a parametric program functions in the case of the four holes mentioned above. In this case, the boring instructions will not contain absolute coordinates for the holes (for example, information that hole A is found at coordinates X=550 and Y=350), but will tell the program that hole A must always be 50 millimeter from the edges. When the program is run on a panel with different measurements, *Zilog Plus* will automatically calculate the distance for positioning the holes correctly.

In the parametric program, instead of the absolute coordinates X and Y for the hole, we will enter a formula where the distance X from the edge is indicated, for example, by the variable DISTANCEX and the distance Y from the edge is indicated by the variable DISTANCEY. The value for both variables will be calculated by Xilog Plus based on a mathematical formula:

- DISTANCEX=DX-50 (to subtract 50 millimeters from the length of the panel, shown in the program header as parameter DX);
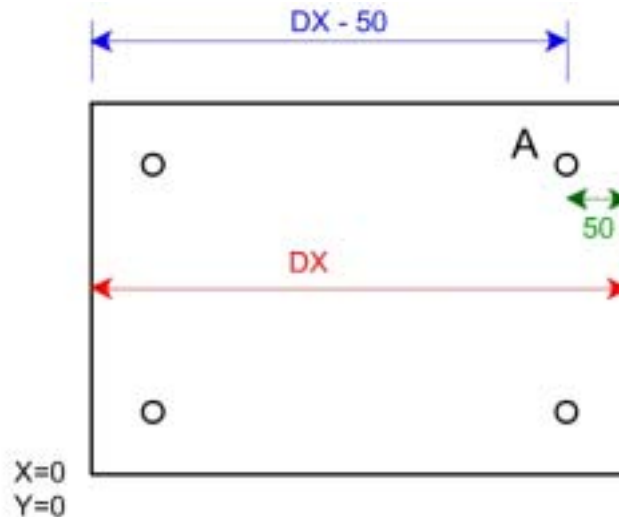
- DISTANCEY=DY-50 (to subtract 50 millimeters from the width of the panel, shown in the program header as parameter DY.

▶ *see: Xilog Plus editor manual*

3.6 Adding Our Personal Instructions to Xilog Plus: the Macros

The instructions for a Xilog Plus program (for example, the XBR instruction that allows boring) appear to the user as simple commands: a XBR instruction, entered into a program with the relative parameter, is in fact sufficient to have a hole made by the bore-milling machine which for this purpose will perform a series of movements. Actually, the XBR instruction is a real parametric program transformed into an instruction: in other words, the XBR instruction is a *macro.*

In Xilog Plus the term *macro* indicates parametric programs that are transformed into instructions. When Xilog Plus is installed it appears already equipped with a set of instructions for boring, milling, etc. These instructions can easily be entered into a program by clicking on the icon that represents them, collected together in the instructions toolbar.
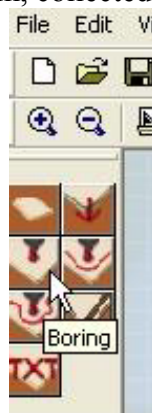
Xilog Plus allows the user to create his own macro. For example, we an transform the parametric program into a macro to make the four holes mentioned above. For this purpose, it will be necessary to also create an image in bitmap format to be associated with the program. Once the procedure for conversion into macro is completed, the image which we associated

with the hole work program will also appear on the instructions toolbar, and it will be possible to use the program with a simple click, just like any other instruction.

▶ *see: Xilog Plus editor manual*