

# Learning On-Line

Tech Tip # 4

November 2002

Written by:

Phil Rasey

Stiles Education's  
Weeke Specialist

*Training Your  
Knowledge Workers  
For Tomorrow*

**STILES EDUCATION**

3965 44th Street S.E.

Grand Rapids, MI 49512

(616) 698-7500

fax (616) 698-0196

e-mail [classes@stilesmachinery.com](mailto:classes@stilesmachinery.com)

## WoodWOP Programming:

- Merge Programs
- Utilizing Components
- Utilizing Block Macros

**What are the advantages / disadvantages to each method?**

Every WoodWOP program has its own special needs and design issues. At times it would be nice to have a ready made pattern to place anywhere on a main panel program. Common patterns include, but are not limited to:

- Hardware mounting locations
- Hole boring locations
- Entire cabinet side panels to be nested on another panel
- Commonly utilized or repeated routing contours

When programming a panel design that includes some or many of these ideas, the keys to productivity are keeping design time to a minimum and to send the program into production as quickly as possible.

**Example:** Your panel design requires specific bolt hole patterns and these patterns must be placed in five different locations. This pattern fits a specific line of hardware and may be used on numerous products within your company.

## WoodWOP Programming

### Possible resolutions:

- 1) Create the bolt hole patterns one at a time (very time consuming).
- 2) Create the bolt hole pattern as an MPR and merge it into the panel MPR.
- 3) Create the bolt hole pattern as a component and place each component on the panel MPR.
- 4) Create the bolt hole pattern as a block macro and place the block macro on the panel MPR.

**The real question at this time would be, which option is the most appropriate for this situation?**

### OPTION 1: MERGE PROGRAMS (See Figure 1)

**Creating a panel program by merging programs:** We can create an MPR that has a specific design and save that as a normal MPR in the C:\WW4\A1\MP4 folder. This MPR will be available to the programmer to merge with any other MPR.

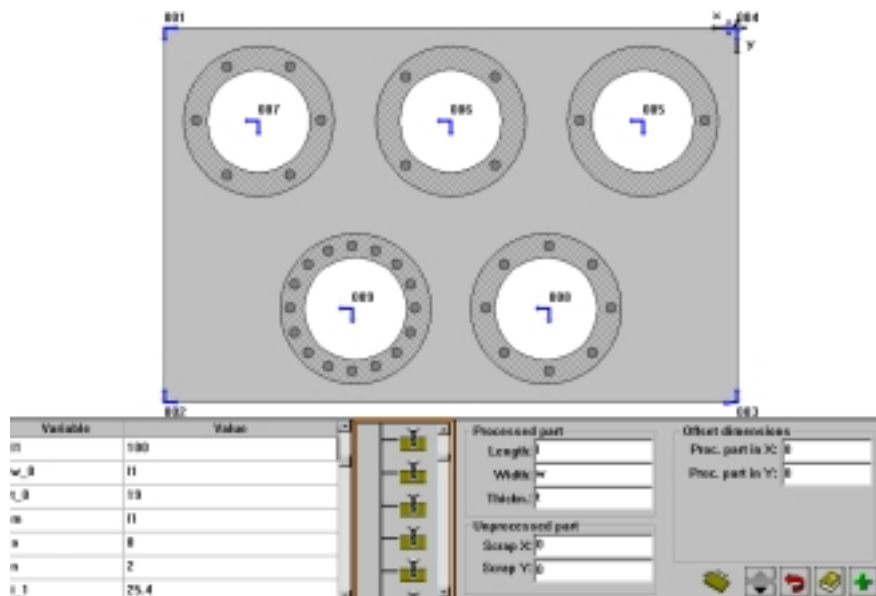


Figure 1

## WoodWOP Programming

Page 3

### Advantages to merging MPR's:

- 1) Automatic assignment of new coordinate system to each merged MPR. This allows the operator to easily adjust the placement of each merged MPR.
- 2) Ability to rotate a merged MPR. Provides the ability to use the material to the best efficiency. (See Figure 2)
- 3) Ability to mirror a merged MPR. Provides the programmer the ability to place mirrored parts in a nest for nested manufacturing or any other required mirror function. (See Figure 2)
- 4) Selection of automatic optimizing of operations. Allows the main panel program to optimize all merged programs for efficient use of the routers, limiting the number of tool changes.
- 5) Programmer can select the option for the program to automatically adjust the panel size based on space occupied by the merged programs.
- 6) Options for multiple lists of variables, selected by the programmer.
- 7) Merging MPR's is the most common practice for customers with Nested Manufacturing machines. (See Figure 2)

## WoodWOP Programming

### Disadvantages to merging MPR's:

- 1) If the merged MPR changes in design, every program that uses that MPR will need to be adjusted individually and manually.
- 2) If the option for new coordinate system is not selected, the programmer does not have any means to easily relocate the merged MPR and it is often simpler to start over.
- 3) Numerous merged MPR's can lead to a lengthy list of variables, unless the programmer utilizes global variables.

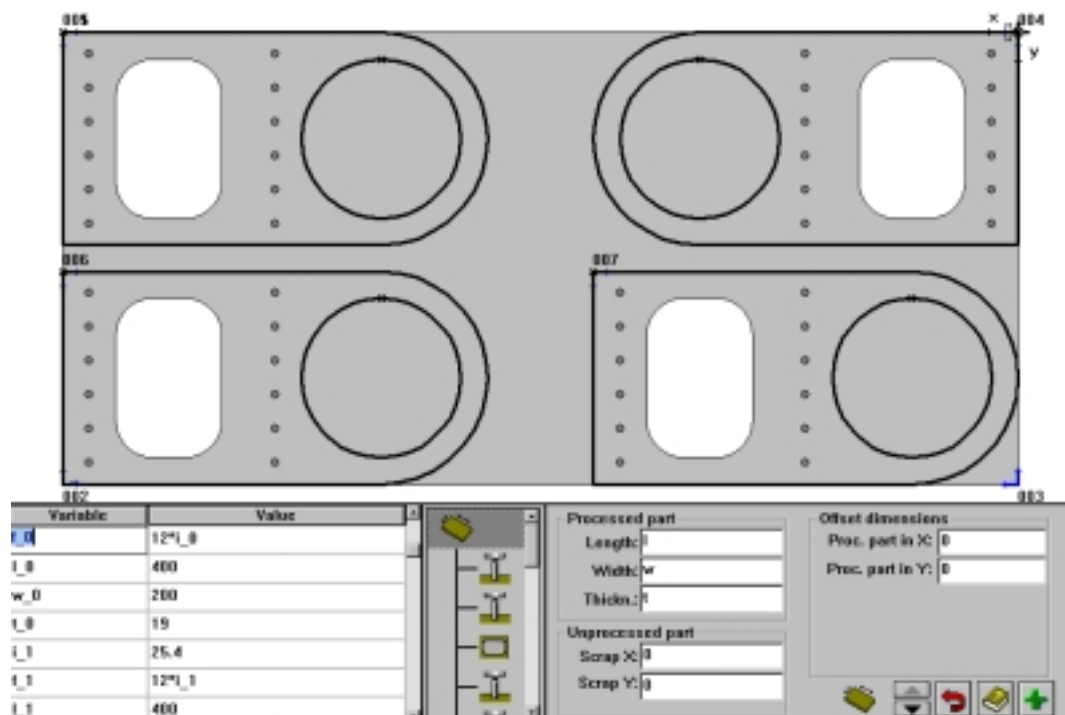


Figure 2

## WoodWOP Programming

### OPTION 2: UTILIZING COMPONENTS (See Figure 3)

**Creating a program with Component macros:** We can create individual components that have the required designs and then place each component on the panel using the X, Y, and Z-axis offsets.

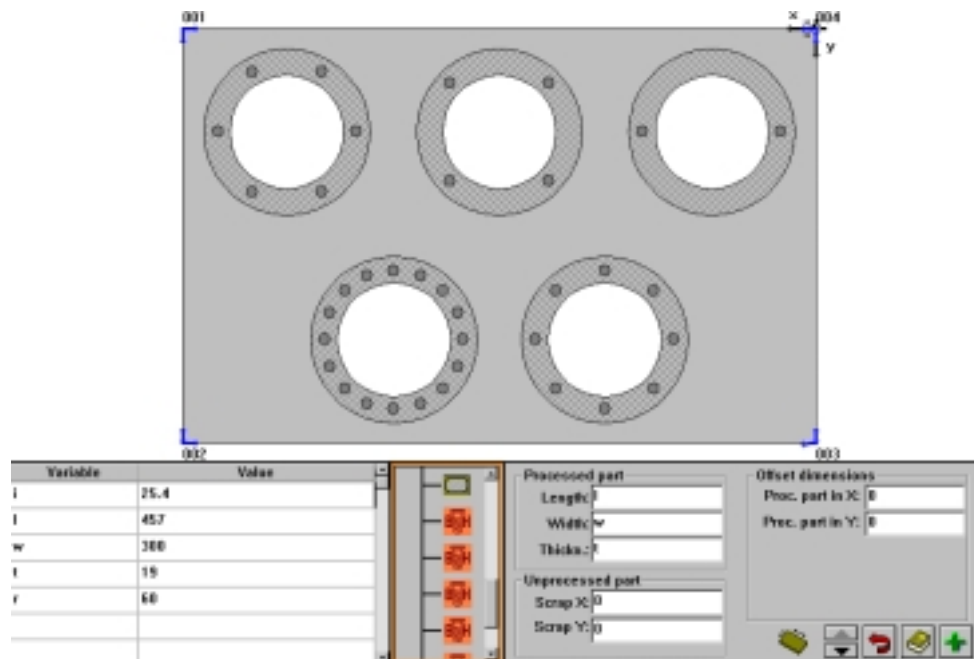


Figure 3

### Advantages to creating panels using Components:

- 1) Components may be repositioned by changing the X, Y, and Z-axis offsets for each component.
- 2) If a component design changes, ALL programs using that component will automatically change, system wide.

## WoodWOP Programming

Page 6

- 3) An individual component may be “locked” from future changes using the EMBED command.
- 4) Components can store any machine function; vertical and horizontal boring, saws, pockets, contour routing, etc...

### **Disadvantages to creating panels with components:**

- 1) Components cannot be mirrored or rotated on a panel.
- 2) Components take up more memory and too many components may cause the program to load and process slowly.
- 3) The program processes each component before moving on to the next processing macro.
- 4) Changes to the order of operation within the component may only be performed within the parent component, not in the panel program level.

## WoodWOP Programming

Page 7

### OPTION 3: UTILIZING BLOCK MACROS (See Figure 4)

Creating a program with Block Macros:

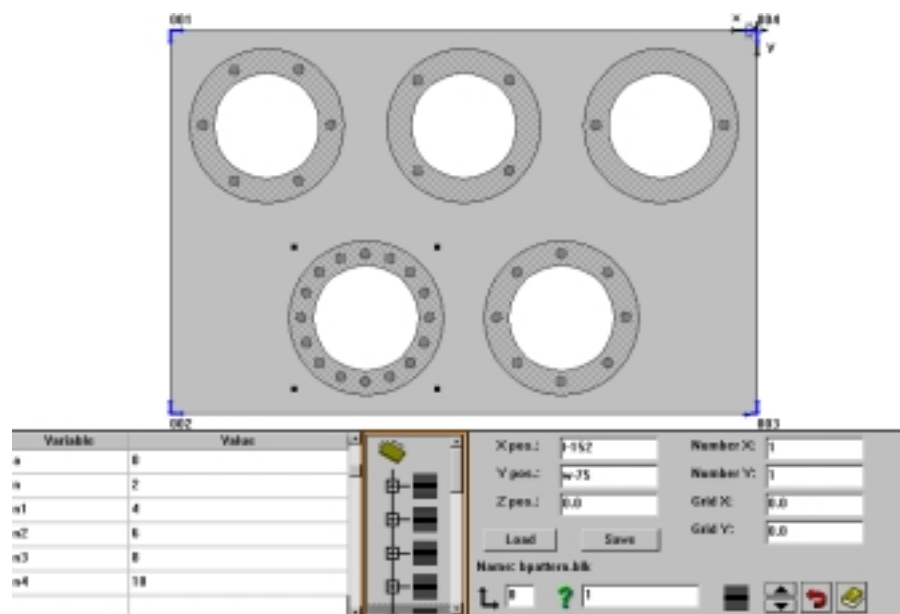


Figure 4

### Advantages to Block Macros:

- 1) Individual macros may be moved within a block macro or completely moved out of the macro, on the main panel program.
- 2) Individual macros may be “X” out using the space bar to eliminate it from the NC generated program, but maintain the macro in the list for future use.

## WoodWOP Programming

Page 8

### Disadvantages to Block Macros:

- 1) Block macros do not understand contour routing functions and therefore the contours are not available.
- 2) Variables used in the original block macro do not come forward to the main panel program variable list. In the above example, the number of holes (n) must be inserted into the variable list as: (n, n1, n2, n3, and 4) to achieve the desired number of holes in each location.
- 3) Block macros are not system wide. If the original block macro changes all the programs that use it must be changed individually and manually.

### CONCLUSION:

Each programming option discussed (**merging, components, and block macros**) has its advantages and disadvantages. Some programs are generated in a style to restrict the number of variables the machine operator can adjust, and others require the operators to adjust more of the program's variables based on the production run and end product design. *As a programmer, you must decide which option fits the needs of the entire panel program and what elements of the program you wish to control, as well as future production designs.*

### Want to learn more?

*Stiles Education offers an Advanced Weeke Machining Center Programming course (MC300) that addresses these techniques in-depth along with advanced parametric programming fundamentals and importation of DXF program formats.*

**Call us at 616/698-7500 for further information or to register!**