

Method of developing parametric machine cycles for modern CNC systems using high-level language

Roman Pushkov^{1,*}, Evgeniy Salamatin¹, and Svetlana Evstafieva¹

¹Computer Control Systems department, Moscow State Technological University “STANKIN”, Moscow, Russia

Abstract. The article deals with the questions connected with the technique of developing standard technological cycles by the example of milling a circular spigot. The sequence of cycle development is given, key development moments are highlighted, examples of implementation are given. An interface has been implemented for the possibility of convenient input of cycle parameters.

1 Introduction

When machining various parts on CNC machine-tools, repetitive machining elements are encountered, such as pockets, spigots, holes, grooves, and the like. To process these elements, standard technological cycles are used. Depending on the purpose of the cycles, they can be conditionally divided into 3 large groups:

- turning cycles;
- milling cycles;
- drilling cycles.

When writing a control program, the user calls the necessary technological cycle, sets the cycle parameters, and the CNC system processes this element in accordance with the logic of the subroutine of the cycle. In various NC systems, the parameters and logic of the cycles can be different.

Using cycles provides the following benefits:

- the size of the control program is reduced. This is achieved by transferring part of the nc-program code to the cycle subroutine, in the main program there is only a call to the loop;
- simplification of the development for the control program. The programmer does not need to think through the logic of the cycle - using accelerated displacements, taps / outlets of the tool, changing the feed values for plunging and machining, changing coordinate systems, etc.

All this leads to a wide use of cycles in control programs written for different systems of numerical control.

* Corresponding author: pushkov@ncsystems.ru

The cycle itself can be written in the line of the control program: in this case, the name of the cycle is specified in the main control program, and then the parameters for calling this cycle are written. In general, the number of parameters can be any. So, for example, to call the drilling cycle, you must specify 4 parameters, to call the cycle of milling a circular spigot more than 14 parameters. In this case, there are a number of restrictions on the parameters: the parameters must be written in a strictly defined sequence, taking into account the restrictions on entering the allowed parameter values (maximum, minimum parameter value, non-zero parameter value, the parameter is specified in the relative or absolute coordinate system, etc.). Given the presence of a large number of technological cycles, the task of entering a cycle with an indication of its parameters becomes even more complicated. The programmer must use the documentation for the corresponding CNC system [1].

The second way to set the cycle is to use a special dialog window with a graphical representation of the processing cycle, with the definition of all parameters and information with restrictions on entering values for these parameters. In the dialog window, it is also possible to pre-check the correctness of the entered cycle parameter values.

The development of the technological cycle consists of the following stages:

- determination of the input parameters of the cycle;
- creation of the algorithm of the cycle;
- creation and testing of a cycle in a high-level language;
- creation of a dialog window for entering cycle parameters and checking the operation of the cycle.

Let's consider an example of development of a technological cycle of processing of a circular spigot.

2 Determination of input cycle parameters

The circular spigot and the input parameters for the description of the technological cycle are shown in the figure 1:

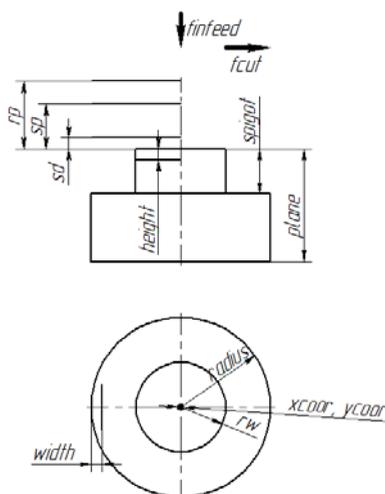


Fig. 1. Circular spigot and input parameters for the description of the cycle.

The input parameters for the cycle are:

- rp – the tool retraction plane is defined in the absolute coordinate system, (mm);

- sp – a safe plane, is defined in the absolute coordinate system, (mm);
- sd – safety distance is specified in the relative coordinate system, (mm);
- fcut – feed at processing, (mm/min);
- finfeed – infeed feed, (mm/min);
- plane – the coordinate of the surface of the part, is given in the relative coordinate system, (mm);
- spigot – height of the spigot, is given in the relative coordinate system, (mm);
- height – the value of one pass (depth) of cutting along the height is given in the relative coordinate system, (mm);
- width – the value of one pass in the plane of radius processing is given in the relative coordinate system, (mm);
- radius – the initial radius of the workpiece is specified in the absolute coordinate system, (mm);
- rw – radius of the spigot, is given in the absolute coordinate system, (mm);
- xcoor, ycoor – the coordinates of the center of the spigot along the x and y axes, respectively, are given in the absolute coordinate system, (mm).

3 Creating an algorithm for the cycle

The algorithm for processing a circular spigot is shown in the figure 2.

At the first stage, the user in the called cycle function specifies the specific values of the cycle parameters in accordance with the drawing data. When the control program is executing, the circular spigot processing function is called with the processing parameters specified by the user in the main program. Such parameterization of the processing cycle and reuse of the code is possible due to the use of a high-level language, in which it is possible to create parameterized programs and work with subroutines.

Then, the processing cycle starts: the inner parameters of the cycle such as the number of machining passes along the radius, the height of the spigot and the size of the residual material layer are calculated. The tool is moved to the starting point for processing. The starting point for processing is outside the body of the part at a safe distance from it. The tool is lowered to the first depth for processing and processing takes place in the radial plane until the number of passes calculated previously is made. If part of the material in the plane remains untreated, this residual material is processed. Then, the tool is accelerated to the start point of the next pass through the depth. The treatment is repeated until the spigot is fully processed [2,3].

4 Creating and testing a cycle in a high-level language

The cycle is called in the main control program as shown on figure 3.

The name of the spigot processing cycle with parameters is specified. Since a high-level language is used, the program can be parametric, i.e. changing the parameters in the called function (highlighted in the figure), you can process the spigot with other dimensions.

The specified parameters are transferred to the subroutine, which describes the technology of spigot processing. The parameters are accepted by the subroutine and the internal parameters of the subroutine are calculated, such as the number of processing

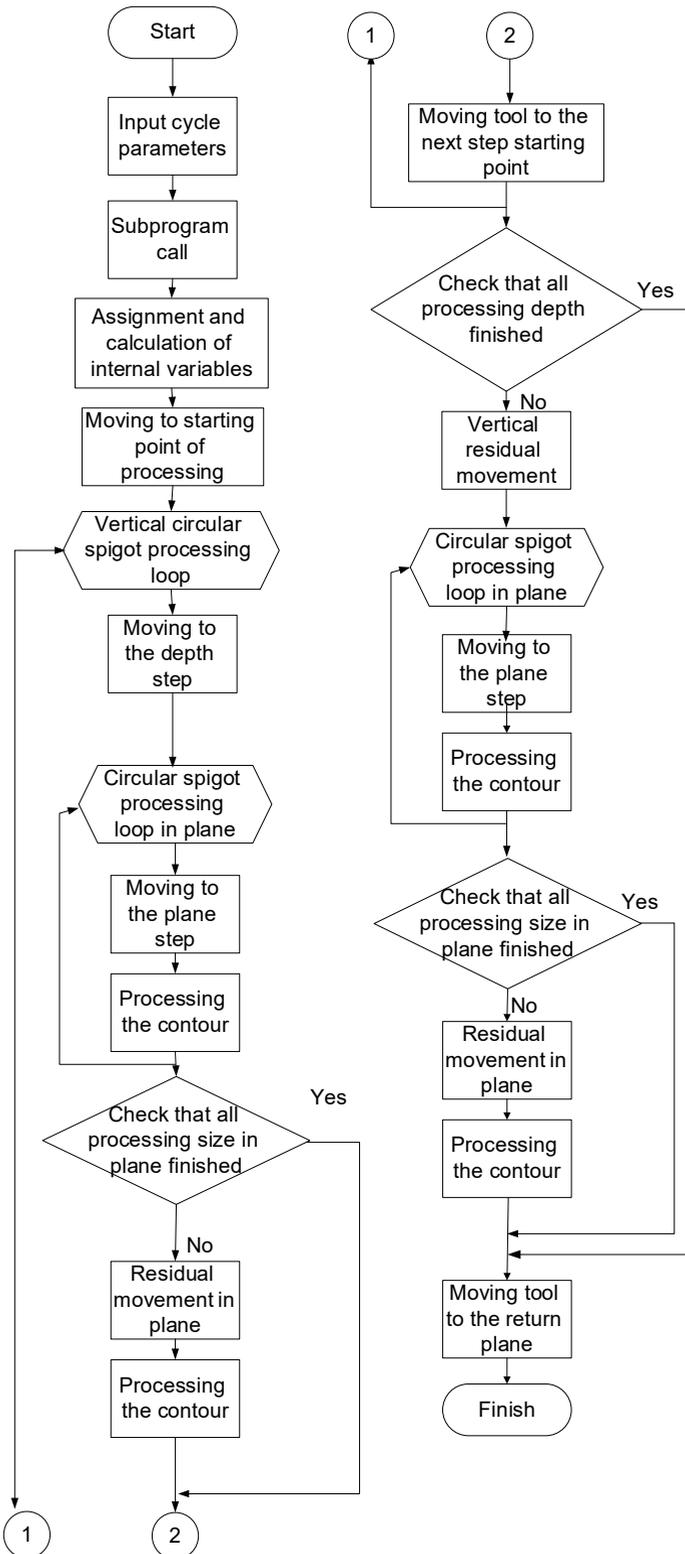


Fig. 2. Algorithm for processing a circular spigot.

steps in terms of radius and depth, the value of the last pass along the radius and depth:

```
double zsteps=floor(spigot/height);  
double last_zsteps=spigot-(zsteps*height);  
double rsteps=floor((radius-rw)/width);  
double last_rsteps=((radius-rw)-(rsteps*width));
```

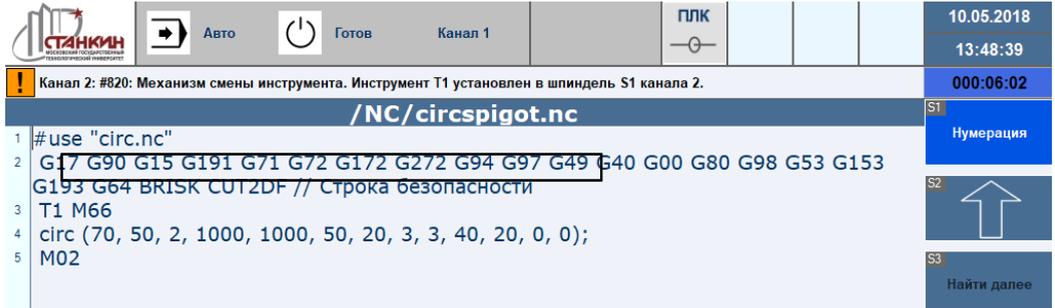


Fig. 3. Calling the cycle of processing a circular spigot.

The tool is moved to the starting point to begin processing the first depth and machining along the spigot radius begins. If there is residual material after the radial machining, then the remaining material will be removed:

```
for (kk=0;kk<rsteps;kk=kk+1)  
{g91  
if (kk==0)  
{g91 g01 x=sd F=finfeed}  
g91 g01 x=width F=finfeed  
g90 g02 g190 I=xcoor J=ycoor F=fcut}  
if (last_rsteps>0)  
{g91  
g01 x=last_rsteps F=finfeed  
g90 g02 g190 I=xcoor J=ycoor F=fcut}
```

After the fully processed first depth, the tool is quickly retracted to the point for the beginning of the next processing and processing is performed at the next depth until all passages calculated at the beginning of the subroutine are processed. If there is residual material in depth, the tool moves not by the depth, but by the amount of residual material, and processing in the plane:

```
if (last_zsteps>0)  
{g91 g00 z=-last_zsteps-sd F=finfeed}
```

After the processing is completed, the tool is retracted to the retraction plane.

5 Creating a dialog box for entering cycle parameters

A large number of technological cycles can be implemented in the CNC system. Despite their variety in the functional plan, the interface for entry data in the cycle will have the same appearance. In order not to create several identical dialog boxes, you can use the following approach: one common XSD-scheme is created that describes the structure of the window. Depending on the type of the called cycle, parameters in the XML-description of the dialog window that are specific for this cycle [4].

Based on the developed unified XSD - scheme for implementing the dialog window, the structure of the XML-document as following:

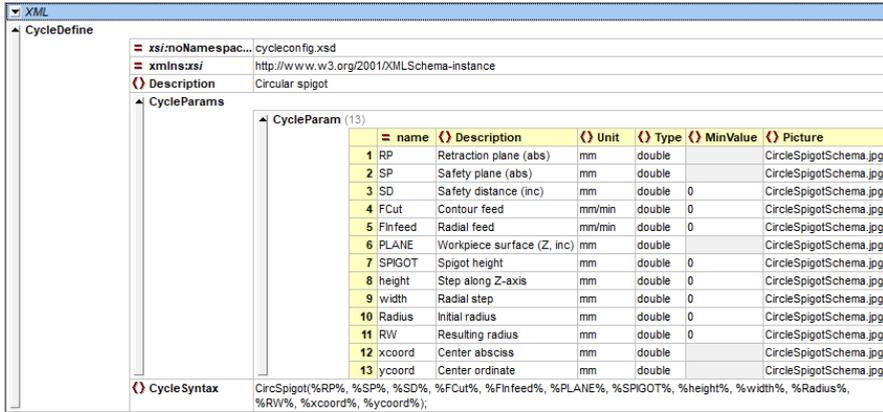


Fig. 4. Implementing a dialog window for entering cycle parameters using XML.

In XML-document the parameters is being described, their purpose, the coordinate system used (relative or absolute), units of measure, type of parameters, constraints on possible values, and the associated picture. The lower part of the figure shows the general form of the syntax of the cycle: the names of the variables, their order of recording.

6 An example of a cycle

In the example, processing of a circular spigot with the following parameters shown in the figure 5. To process such spigot, the user in the dialog window indicates the spigot parameters, and the CNC takes the data entered in this window and collects them in the next line:

```
circ (70, 50, 2, 1000, 1000, 50, 20, 3, 3, 40, 20, 0, 0);
```

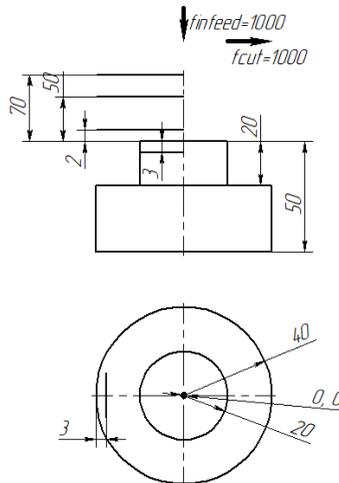


Fig. 5. Circular spigot parameters.

The path of tool movement, when processing a circular spigot with the specified parameters, is shown in the figure 6:

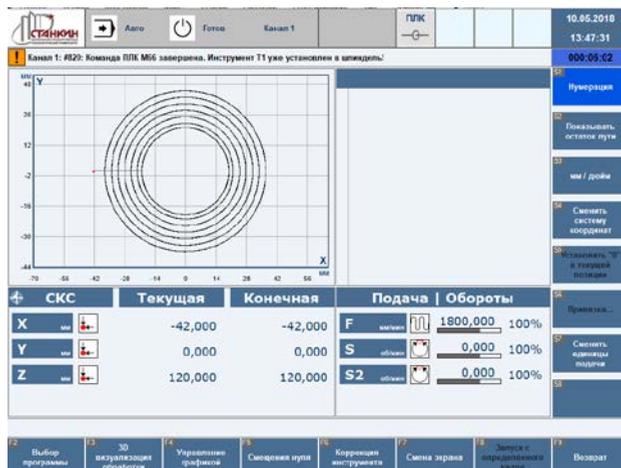


Fig. 6. Processing cycle testing.

This research was supported by the Ministry of Education and Science of the Russian Federation as a public program in the sphere of scientific activity (N 2.1237.2017/4.6)

References

1. L. I. Martinova, N. V. Kozak, R. A. Nezhmetdinov, R. L. Pushkov, A. I. Obukhov, *The Russian multi-functional CNC system AxiOMA control: Practical aspects of application*, Automation and Remote Control, Volume 76, Issue 1, pp 179-186 (2015)
2. S.V. Evstafieva, *Razrabotka konfiguriruemyykh ekranov stanochnykh tsykhlov dlya terminala SCHPU*, Materialy mezhdunarodnoy nauchno-prakticheskoy konferencii "Mashinostroenie: innovacionnye aspekty razvitiya".1, p.59-63, (2018)
3. R.L. Pushkov, E.V. Salamatina, S.V. Evstafieva, *Primeneniye yazyka vysokogo urovnya systemy CHPU "Axioma Control" dlya realizacii gruppovoy obrabotki*, Trudy XVII mezhdunarodnoy nauchno-prakticheskoy konferencii "Sistemy proektirovaniya, tehnologicheskoy podgotovki proizvodstva i upravleniya etapami zchiznennogo cikla promyshlennogo produkta". p.94-98, (2017)
4. R.L. Pushkov, E.V. Salamatina, S.V., *Evstafieva Prakticheskie aspekty primeneniya yazyka vysokogo urovnya v systeme CHPU dlya realizacii gruppovoy obrabotki, Avtomatizatsiya v promyshlennosti*, 5, p.31-34, (2018).