After that, the coordinates of neighboring points to Pi(xi,yi) of the arc are determined:

$$x_{i=}x_{i-1}+\frac{\Delta x(x_2-x_1)}{l}-\frac{\Delta y(y_2-y_1)}{l}; \quad (9)$$

$$y_{i=}y_{i-1}+\frac{\Delta y(x_2-x_1)}{l}+\frac{\Delta x(y_2-y_1)}{l}. \quad (10)$$

Figure 5 shows a fragment of a program for calculating the coordinates of Pi points be longing to an arc. The variables Xr, Yr determine the coordinates (xi,yi) of the current arc point. The coordinate values of the current point are found incrementing along the arc of the coordinates of the starting point, denoted by the variables X1,Y1. After the calculation, the coordinate values of the current point are assigned to the new starting point and denoted by the times X2,Y2.The calculation cycle is repeated Koef times. Using the Write operator, the calculated coordinates of the points of the circle arc c by means of an operator Write are saved to a file that is passed to the integrated CAD system.

```
Xr:=X1+Xrel*((X2-X1)/l)-Yrel*((Y2-Y1)/l);
Yr:=Y1+Xrel*((Y2-Y1)/l)+Yrel*((X2-X1)/l);
for n:=1 to Koef do
  Begin
    X2:=Xr;
     Y2:=Yr;
     Write(Inp_f,'"');
     Write(Inp_f,X2_p:cod:10,',');
      Write(Inp_f,Y2_p:cod:10);
       WriteLn(Inp_f,'"');
  End;
```

Figure 5 – Fragment of the program for calculating the coordinates of pointsofan arc

Conclusion. The developed software module is designed to calculate the coordinates of points located on the arc of a circle with a given step, and determine the trajectory of the actuator movement.

1. Война, В.С. Разработка и реализация алгоритма строчки «программируемый зигзаг» / В.С. Война, Т.В. Буевич, А.Э. Буевич / Материалы докладов 50 Международной научно-технической конференции преподавателей и студентов, посвященной году науки / УО «ВГТУ». – Витебск, 2017. - С. 206–208.

REALISATION OF THE ALGORITHM OF DIVISION OF A LINE SEGMENT INTO EQUAL SEGMENTS

Wang Wenfeng,

master's student, VSU named P.M. Masherov, Vitebsk, Republic of Belarus

Scientific supervisor – Buyevich A.E., candidate of engineering sciences, docent

Keywords. Algorithm, coordinates, computer-aided design, control programme.

The paper is devoted to the development and implementation of an algorithm for dividing a straight line segment into equal sections. The proposed software module can be used for obtaining trajectories of movement of the actuator according to the given coordinates on laser complexes, cutting machines, sewing semiautomatic machines in light industry.

Target of the research – development and realization of the algorithm of line division into elements for integrated CAD.

Material and methods. The work is based on the results of analysis of scientific and technical information on computer-aided design systems, integrated systems, automated equipment; experimental work on the study of technological processes of parts processing on automated equipment; use of computer modeling methods.

Results and their discussion. In control programmes for mechatronic systems, the required trajectory of movements of actuators is specified by coordinates of points. We propose an algorithm for dividing the trajectory in the form of a line segment into nodes (points) at a given distance from each other with equal spacing. The line segment when represented in vector form is described by the coordinates of the start and end points. Figure 1 shows the calculation scheme of the algorithm for dividing a line into fragments of equal length. The coordinates of the start point 1 are labelled (x_1 , y_1); the coordinates of the end point 2 are labelled (x_2 , y_2). In Figure 1 are also marked: *l* - the length of the line segment, *Δl* - the refined distance between the points on the line segment P_i , $Δl_x$ and $Δl_y$ – projections on the X and Y coordinate axes of the segment *Δl.*

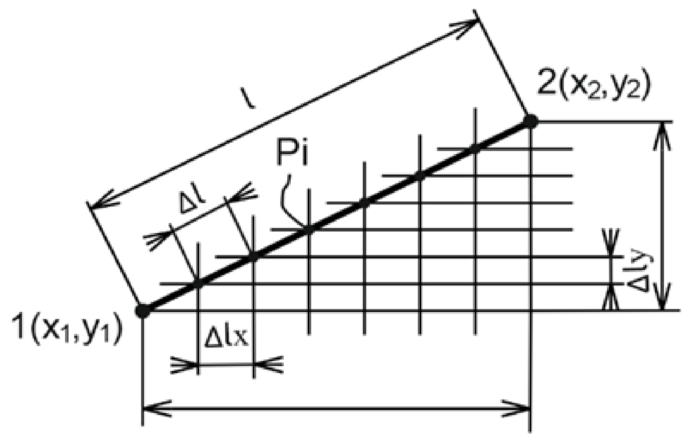

Figure 1 – Calculation scheme

Information about vector image of a line segment in AutoCAD system is contained in a special format ".dxf" drawing exchange file. The ".dxf file" contains text information about coordinates of points 1 and 2 of the beginning and end of the graphical primitive "line segment" (polyline) in groups defined by corresponding codes. The group code " 10" indicates the primary X coordinate; the group code " 20" indicates the primary Y coordinate. The X and Y coordinate values of a point follow each other directly. The fragment of the programme for determining the initial data about the line segment is shown in Figure 2.

```
...
If st=' 10' then
 Begin
    Readln(Var_f,st);
    Val(st,cr,cod);
     x2:=cr;
 end;
...
x1:= x2;

If st=' 20' then
 Begin
    Readln(Var_f,st);
    Val(st,cr,cod);
     y2:=cr;
 end;
...
y1:= y2;
...
```

Figure 2 – Fragment of a programme for determining points 1 and 2 of a line segment

The distance n_0 between points P_i is pre-defined by the user. Then the number N of segments of length n_0, which fit in the line length *l*, is calculated. The number N is calculated by the expression and rounded to integer:

$$N = \left\lceil \frac{\sqrt{((x2-x1)^2+(y2-y1)^2)}}{n0} \right\rceil. \tag{1}$$

Figure 3 shows the implementation of the algorithm for determining the number N of segments of length n_0 . The variable *l* describes the length of the line segment, the variable Lim describes the specified distance n_0 between points P_i , the variable Koef describes the refined number of segments between points 1 and 2. The programme code calculates the real value of Koef, which is rounded to integer.

```
...
Begin
  l:=sqrt(sqr(x2-x1)+sqr(y2-y1));
  Koef:=Floor(l/Lim);
  Koef:=abs(Koef);
end;
...
```

Figure 3 – Program fragment for dividing a line segment into equal sections

If the adjusted distance between points $P_{i,}$ is less than the user-defined distance n_0 , the variable Koef is set to 1. Then the refined distance Δl between the points dividing the line segment into equal segments:

$$\Delta l = \frac{l}{Koef}. \tag{2}$$

The projectionsΔ *lx* andΔ *ly* are calculated respectively from the expressions:

$$\Delta lx = \frac{(x2-x1)}{Koef}; \tag{3}$$

$$\Delta ly = \frac{(y2-y1)}{Koef}. \tag{4}$$

The implementation of the algorithm for calculating the projections Δlx and Δly is shown in Figure 4.

```
...Begin
    if Koef<=1 Then Koef:=1;
    dX:=(x2-x1)/Koef;
    dY:=(y2-y1)/Koef;
end;
...
```

Figure 4 – Fragment of the programme for calculating the projections Δlx and Δly

After that, the coordinates (xi,yi) of the points Pi are determined:

$$Pi(xi, yi) = ((x_{i-1} + \Delta lx), (y_{i-1} + \Delta ly)). \tag{5}$$

Figure 5 shows a fragment of the programme for calculating the coordinates (xi,yi) of the points Pi belonging to the line segment. Variables X1, Y1 define the coordinates (xi,yi) of the current point of the arc. The values of the coordinates of the current point are found by increments dX,dY of the coordinates of the initial point, denoted by variables X1,Y1. The calculation cycle is repeated Koef times. The calculated coordinates of the points of the line segment are saved to a file using the Write operator, which is transferred to the integrated CAD system.

```
…
for n:=0 to Koef do
 begin
   X1:=X1+dX;
   Y1:=Y1+dY;
   Write(Inp_f, '"');
   Write(Inp_f, x1:cod:10);
   Write(Inp_f, ',');
   Write(Inp_f, y1:cod:10);
   Writeln(Inp_f, '"');
 end;
…
```

Figure 5 – Fragment of the programme to calculate the coordinates (xi,yi) of the points Pi

Conclusion. The developed algorithm is implemented and designed to calculate the coordinates of points dividing a line segment into sections of equal length and to determine the trajectory of movement of the actuator of electronically controlled technological equipment.

1. Война, В. С. Разработка и реализация алгоритма строчки «программируемый зигзаг» / В. С. Война, Т. В. Буевич, А. Э. Буевич / Материалы докладов 50 Международной научно-технической конференции преподавателей и студентов, посвященной году науки / УО «ВГТУ». – Витебск, 2017. – С. 206–208.
2. Буевич, Т. В. Алгоритм оптимизации траектории векторного контура для лазерной перфорации кожи / Т. В. Буевич, А. Э. Буевич / Материалы докладов 53-й Международной научно-технической конференции преподавателей и студентов : в 2 т. / УО «ВГТУ». – Витебск, 2020. – Т. 2. – С. 5–8.