
SIEMENS

Ingenuity for life

# Circular Motion on the Basis of Cam Discs "MoveCircle2D"

TIA Portal V14, S7-1500T

https://support.industry.siemens.com/cs/ww/en/view/109742306

Siemens Industry Online Support

# Warranty and liability

**Note**

The Application Examples are not binding and do not claim to be complete regarding the circuits shown, equipping and any eventuality. The Application Examples do not represent customer-specific solutions. They are only intended to provide support for typical applications. You are responsible for ensuring that the described products are used correctly. These Application Examples do not relieve you of the responsibility to use safe practices in application, installation, operation and maintenance. When using these Application Examples, you recognize that we cannot be made liable for any damage/claims beyond the liability clause described. We reserve the right to make changes to these Application Examples at any time without prior notice.
If there are any deviations between the recommendations provided in these Application Examples and other Siemens publications – e.g. Catalogs – the contents of the other documents have priority.

We do not accept any liability for the information contained in this document.
Any claims against us – based on whatever legal reason – resulting from the use of the examples, information, programs, engineering and performance data etc., described in this Application Example shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act ("Produkthaftungsgesetz"), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract ("wesentliche Vertragspflichten"). The damages for a breach of a substantial contractual obligation are, however, limited to the foreseeable damage, typical for the type of contract, except in the event of intent or gross negligence or injury to life, body or health. The above provisions do not imply a change of the burden of proof to your detriment.
Any form of duplication or distribution of these Application Examples or excerpts hereof is prohibited without the expressed consent of the Siemens AG.



**Security information**

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks.
In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions only form one element of such a concept.
Customer is responsible to prevent unauthorized access to its plants, systems, machines and networks. Systems, machines and components should only be connected to the enterprise network or the internet if and to the extent necessary and with appropriate security measures (e.g. use of firewalls and network segmentation) in place.
Additionally, Siemens' guidance on appropriate security measures should be taken into account. For more information about industrial security, please visit http://www.siemens.com/industrialsecurity.

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends to apply product updates as soon as available and to always use the latest product versions. Use of product versions that are no longer supported, and failure to apply latest updates may increase customer's exposure to cyber threats.
To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed under http://www.siemens.com/industrialsecurity.

# Table of contents

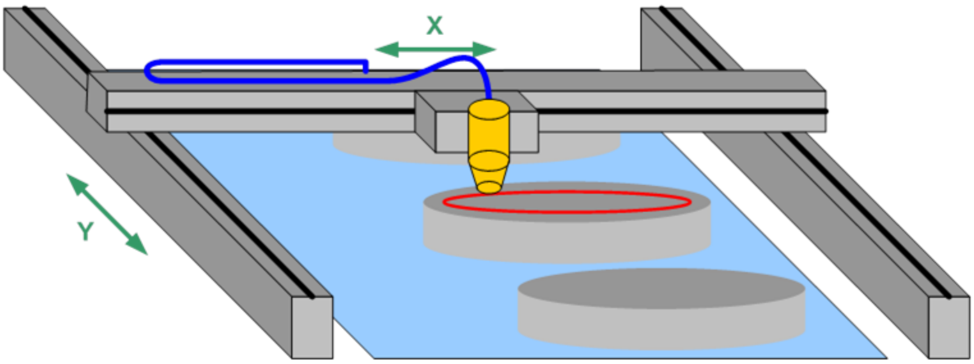
# 1 Introduction

In production engineering there are many applications where a product is processed on a circular or elliptical path, for example, applying an adhesive onto a carrier.

## 1.1 Overview

This application example describes how you can realize an interpolated, even circular or elliptical path with a S7-1500T. The individual steps of the motion are programmed in the (FB) "MoveCircle2D" function block. The desired circular or elliptical motion can be configured via the block parameters. You can specify the parameters via a user-defined website. You can also execute and monitor the function via the website.

Figure 1-1: Example of a two-dimensional motion



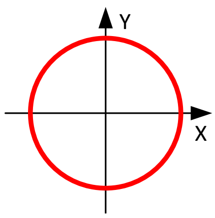
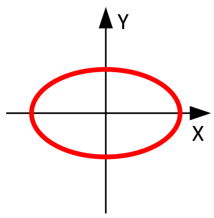



## 1.2 Mode of Operation

A circular motion in a Cartesian coordinate system is achieved by the overlapping of a sinusoidal motion (X axis) and a cosinusoidal motion (Y axis). Both axes are coupled to a virtual master axis.

The motion of both axes is created via cam disc synchronism. The cam disks required for this are generated by the "MoveCircle2D" function block, based on the parameters at program runtime.

Table 1-1: Examples of possible trajectories

| Circle | Ellipse |
|---|---|
|  |  |

### Properties of the FB “MoveCircle2D”

The following features have been included in the realization of the function block and can be utilized when applied in a user program.

- **Simple configuration of circle and elliptical trajectories**
  The desired circular or elliptical trajectory can be specified via the function block parameters.
- **Starting the current position**
  The circular or elliptical trajectory always starts at the current position of the two axes in the plane. This saves additional configuration of the starting position or possible required calculation of the starting point of the circular or elliptical trajectory.
- **Start with defined starting angle**
  The position of the circle or the ellipse in the plane only depends on the orientation of the X and Y axis of the coordinate system. In order to be able to start at a defined position on the circle or the ellipse, a starting angle can be specified on the circle or the ellipse via the function block parameters.
- **Motion on definable segments**
  The desired arc length of the circle or the ellipse can also be specified as a swept angle via parameters on the function block. This enables realizing parts of a circle or an ellipse, as well as whole circles and ellipses, with several “revolutions”.

### Restrictions

The following properties were not considered in the implementation of FB "MoveCircle2D".

- **No guarantee and monitoring of path accuracy**
  The function block does not monitor the compliance with the traversing motion for the specified trajectory. Particularly when the dynamic restrictions of the axes are not complied with, deviations from the specified trajectory may occur.
- **No absolute specification of trajectory velocity**
  The function block cannot be given the absolute trajectory velocity on the circular or elliptical trajectory. The parameters velocity, acceleration and jerk only refer to the respective master axis of the cam disc synchronization. The parameters therefore only define the relations of the angle motion on the circular or elliptical trajectory.

### Application environment

The function block was especially designed for the application with the Cartesian kinematics. This means that two linear axes (X and Y axis) can be controlled that are positioned at a right angle to the plane.

## 1.3 Components used

This application example was created with the following hardware and software components:

Table 1-2: Hardware and software components

| Component | Numbe r | Article number | Note |
|---|---|---|---|
| CPU 1515T-2 PN | 1 | 6ES7515-2TM01-0AB0 | Alternatively, any other CPU of the S7-1500T product family can also be used. |
| Memory card, 24 Mbytes | 1 | 6ES7954-8LF02-0AA0 | - |
| STEP 7 Professional V14 SP1 | 1 | 6ES7822-1..04-.. | - |
| Internet Explorer 11 | 1 | - | Required for user-defined web pages of this application example. |

This application example consists of the following components:

Table 1-3: Components of the application example

| Component | File name | Note |
|---|---|---|
| Documentation | 109742306_MoveCircle2D_DOC_v11_en.docx | - |
| STEP 7 project | 109742306_MoveCircle2D_CODE_v11.zip | Test project and library |

# 2 Engineering

## 2.1 Interface description

The "MoveCircle2D" function block of the application example takes on the complete execution of an interpolated circular or elliptical motion. To be able to use this function block you have to call it and configure it in your user program.

For you to be able to control the interpolated motion, the “MoveCircle2D” function block has the following parameters:

Figure 2-1: Parameters of the "MoveCircle2D” block

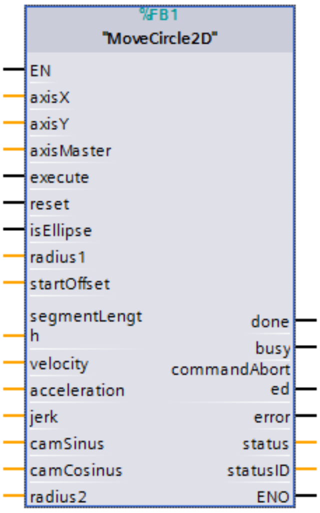



Table 2-1: Parameters of the block interface

| Parameter | Data type | Start value | Description |
|---|---|---|---|
| **Input parameters** | | | |
| axisX | TO_SynchronousAxis | | Synchronized axis for the motion in the X Y plane in the X direction. |
| axisY | TO_SynchronousAxis | | Synchronized axis for the motion in the X Y plane in the Y direction. |
| axisMaster | TO_PositioningAxis | | Virtual axis as master axis for executing the motion via the cam discs. |
| execute | BOOL | False | The execution of the circular or elliptical motion is started with a positive edge on this input. |
| reset | BOOL | | Resetting an error. |
| isEllipse | BOOL | False | Selection of the desired trajectory:<br>False = circular motion<br>True = elliptical motion<br>If the elliptical motion is selected, “radius2” parameter is evaluated in addition to the “radius1” parameter. |

| Parameter | Data type | Start value | Description |
|---|---|---|---|
| radius1 | REAL | 0.0 | Circle radius is specified for a circular motion or the radius of the ellipse in Y direction for an elliptical motion. |
| startOffset | REAL | 0.0 | Starting angle of the circular or elliptical motion in the coordinate system of the circle or ellipse.<br>The positive angle count occurs **clockwise**. |
| segmentLength | REAL | 0.0 | Angle length of the circle or elliptical trajectory from the starting angle on.<br>The positive angle count occurs **clockwise**. |
| velocity | LREAL | -1.0 | Maximum angle velocity for circular or elliptical motion |
| acceleration | LREAL | -1.0 | Maximum angle acceleration for circular or elliptical motion |
| jerk | LREAL | -1.0 | Maximum angle jerk of the circular or elliptical motion. |
| camSinus | TO_Cam | | Cam disc in which the sinus function is stored. |
| camCosinus | TO_Cam | | Cam disc in which the cosinus function is stored. |
| radius2 | REAL | 0.0 | Radius of the ellipse in X direction for elliptical motion. |
| **Output parameters** | | | |
| done | BOOL | False | The block has finished processing.<br>The traversing motion was performed completely. |
| busy | BOOL | False | The block is busy processing. |
| commandAborted | BOOL | False | The technology functions used in the block and thus the actual block have been replaced by a technology function outside the block. |
| error | BOOL | False | An error has occurred while processing the block.<br>Further information on the localization of the error cause is provided via the “status” and “statusID” outputs. |
| status | WORD | 0 | Error code of the block or of a technology function that has been called internally. In addition, it is possible to locate the error within the block via the "statusID” output. |
| statusID | WORD | 0 | Specification of an additional error code for localizing the error cause within the block. |

**Assignment of the parameters “velocity”, “acceleration” and “jerk”**

The parameters “velocity”, “acceleration” and “jerk” directly affect the technology function “MC_MoveAbsolute” of the master axis.

The master axis provides the angle value of the swept segment angle for the circular or elliptical motion. Thus the dynamic parameters "velocity", "acceleration" and "jerk" only have an influence on this angle change. In contrast, the actual velocity of the reference point on the circular or elliptical trajectory (trajectory velocity) depends on the specified radius for the circle or the radii of the ellipses.

**Note** The trajectory velocity on the elliptical trajectory resulting from the angle speed can only be approximated mathematically.

## 2.2 Integration into the user project

To be able to integrate the function into your STEP 7 project, you have to generate, configure and interconnect the required technology objects (for example, axis and cam discs) in the S7-1500T.

### 2.2.1 Creating the virtual master axis "AxisMaster"

The virtual axis “AxisMaster” has to be created as follows:

**Configuration**

Table 2-2: Configuration "AxisMaster"

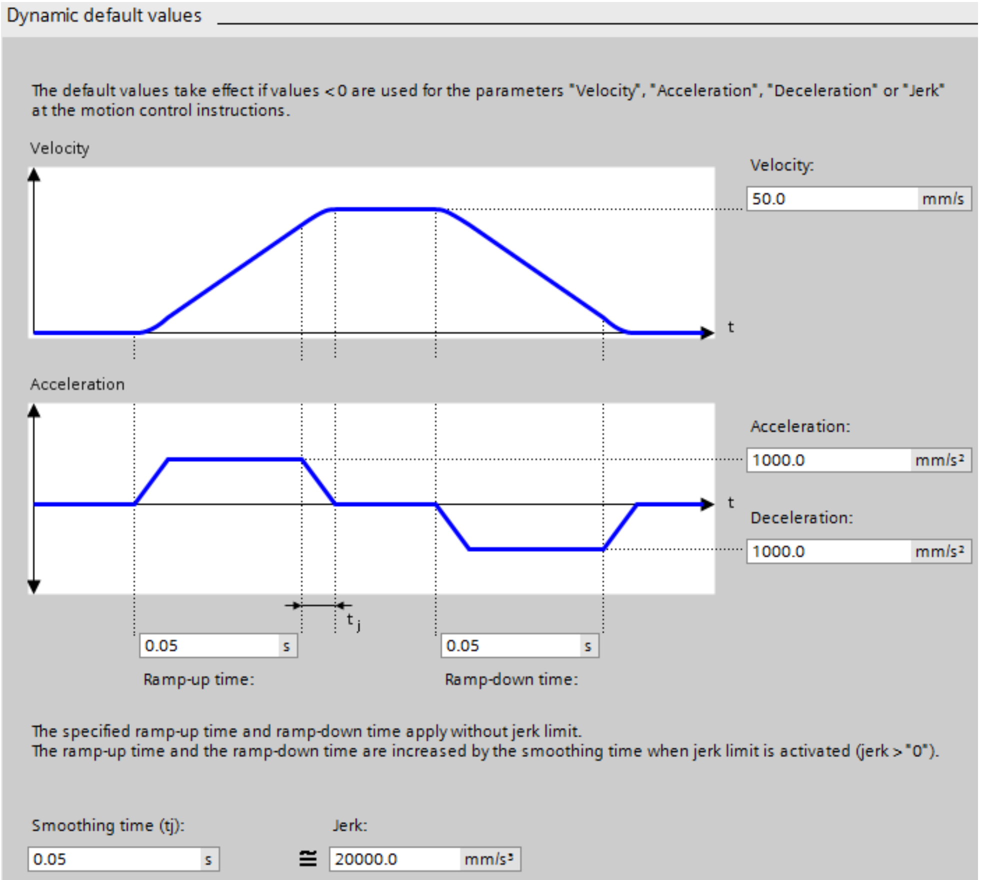
| Parameter | Setting | Note |
|---|---|---|
| Technology object | Positioning axis | Alternatively, you can also use a synchronized axis. |
| Axis type | Virtual axis, linear | - |
| Units of measure | - | All axes have to be set to the same system of units. |
| Modulo | Inactive | - |
| Drive assignment | None | It is mandatory to create the "AxisMaster” axis as virtual axis. |

## Default setting

The dynamic default setting of “AxisMaster” determines the dynamic properties of the axis.

For “acceleration” and “jerk”, the desired values for “AxisMaster” must be entered whilst taking into account the cam discs and the mechanical conditions on axes X and Y.

Figure 2-2: Dynamic default setting "AxisMaster"

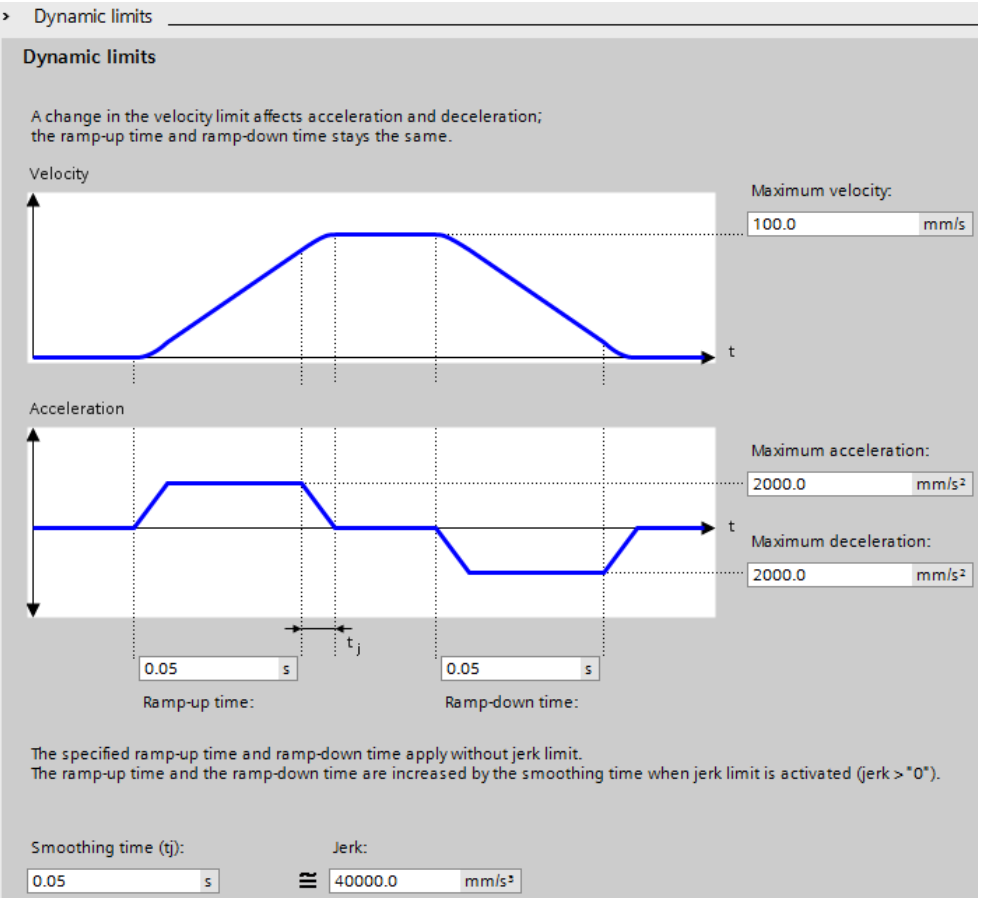


### Limitations

When you are setting the velocity, acceleration and jerk limits, you also have to consider the mechanical conditions for axes X and Y, as well as the cam discs.

Select the limits of the “AxisMaster” in a way so that the set limits for axes X and Y are not exceeded in any operating status. Otherwise there may be an axis error and thus a deviation from the desired trajectory.

Figure 2-3: Limits on “AxisMaster"



### 2.2.2 Creating of the real traversing axes "AxisX" and "AxisY"

Create the traversing axes as follows:

#### Configuration

Table 2-3: Configuration of the traversing axes "AxisX" and "AxisY"

| Parameter | Setting | Note |
|---|---|---|
| Axis technology | Synchronous axis | Required for the cam disc synchronization as slave axis. |
| Axis type | Linear | |
| Units | | All axes have to be set to the same system of units. |
| Modulo | Inactive | |
| Drive assignment | Drive module<br>(Real axis) | Alternatively, the traversing axes can also be operated as virtual axes (see test program). |

#### Interconnecting conductance

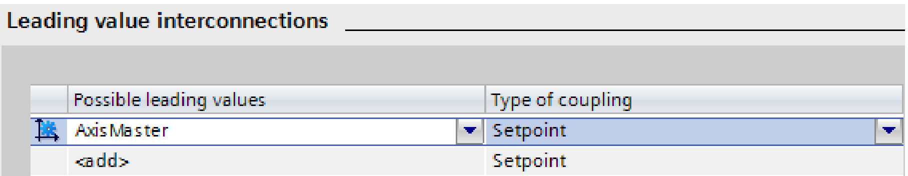
Once the axes “AxisX” and “AxisY” have been generated as synchronized axes, the master axis for the synchronous coupling has to be specified. The following table shows the possible settings for the virtual and real axes:

Table 2-4: Possible settings of the axes

| Axis | Axis technology | | | Axis type | Modulo | Drive | |
|---|---|---|---|---|---|---|---|
| | Speed | Position | Synchronous operation | | | Real | Virtual |
| **Master** | - | yes | yes | Linear | - | - | yes |
| **X** | - | - | yes | Linear | - | yes | yes |
| **Y** | - | - | yes | Linear | - | yes | yes |

As coupling type between master axis and slave axis, the “setpoint coupling” is to be selected.

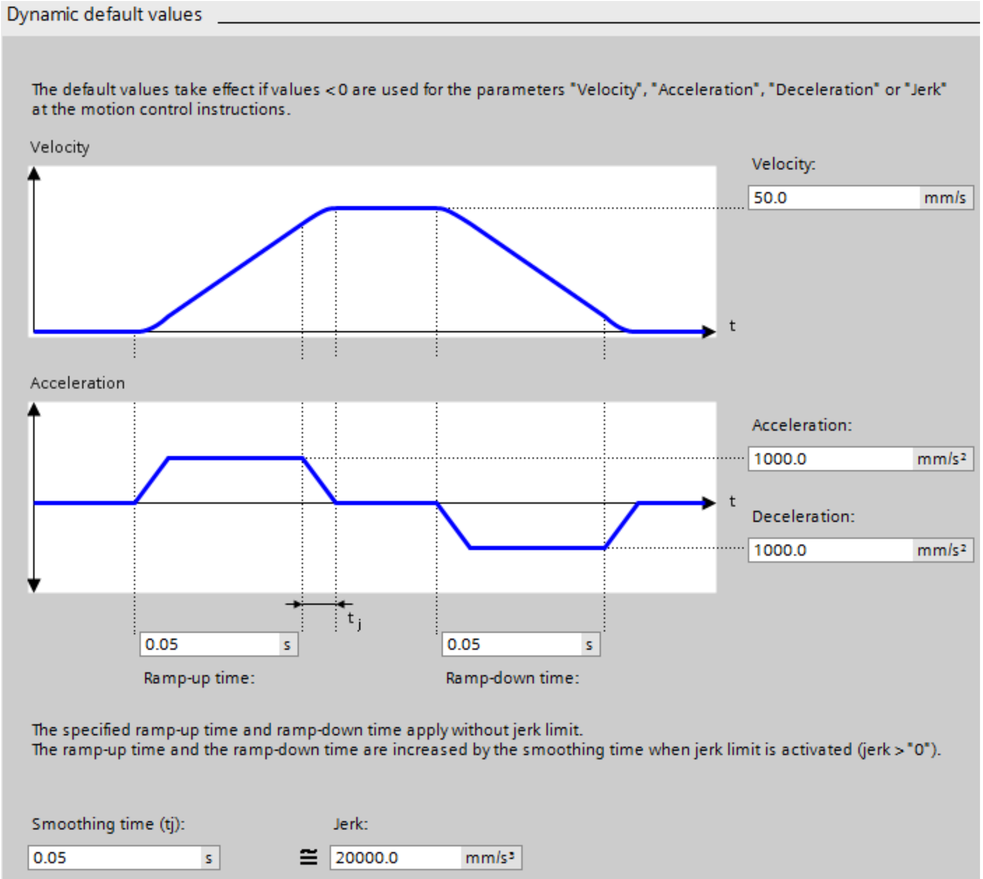
Figure 2-4: Interconnecting conductance on "AxisX"



### Dynamic default setting

If the traversing axes are in motion when block “MoveCircle2D” is called, these axes are stopped by the block to achieve a defined initial status. The dynamic values that define the dynamic default settings are used to keep the axes in this status.

Figure 2-5: Example of the dynamic default setting of the traversing axes

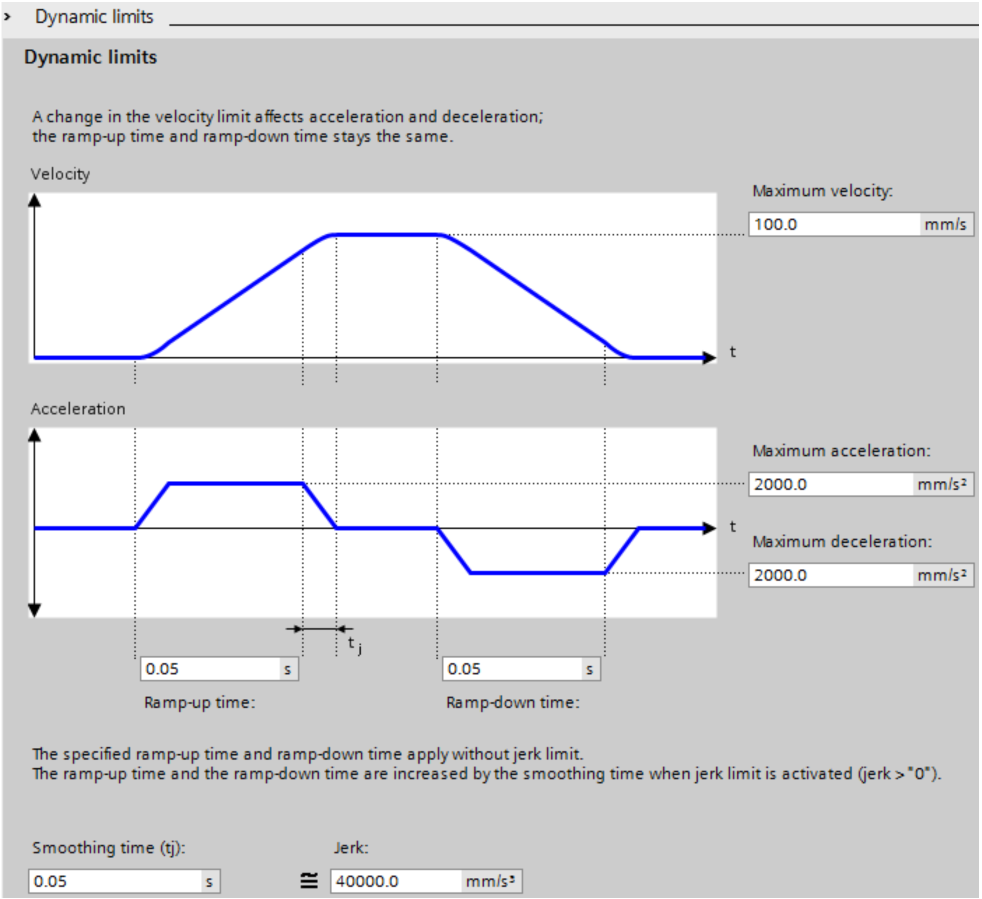



Adjust the settings to your application. The settings do not have an influence on the behavior of the axes during the interpolated motion.

## Limitations

The velocity, acceleration and jerk limits must be set according to the mechanical conditions of the X and Y axis.

Figure 2-6: Example of the dynamic limitations of the traversing axes




| | |
|---|---|
| **CAUTION** | If the motion of at least one axis is limited, this may cause a deviation of the desired trajectory. This is why you should make sure that the limitations of the traversing axes are not reached during the interpolated motion. |

### 2.2.3 Creating the cam disks "CamSinus" and "CamCosinus"

The cam discs only have to be created as technology objects. The definition of the actual cam discs is created by FB "MoveCircle2D" using the specified parameter values on the input variables at runtime.

Figure 2-7: Configuration of the "CamSinus" cam discs


### 2.2.4 Integrating the block

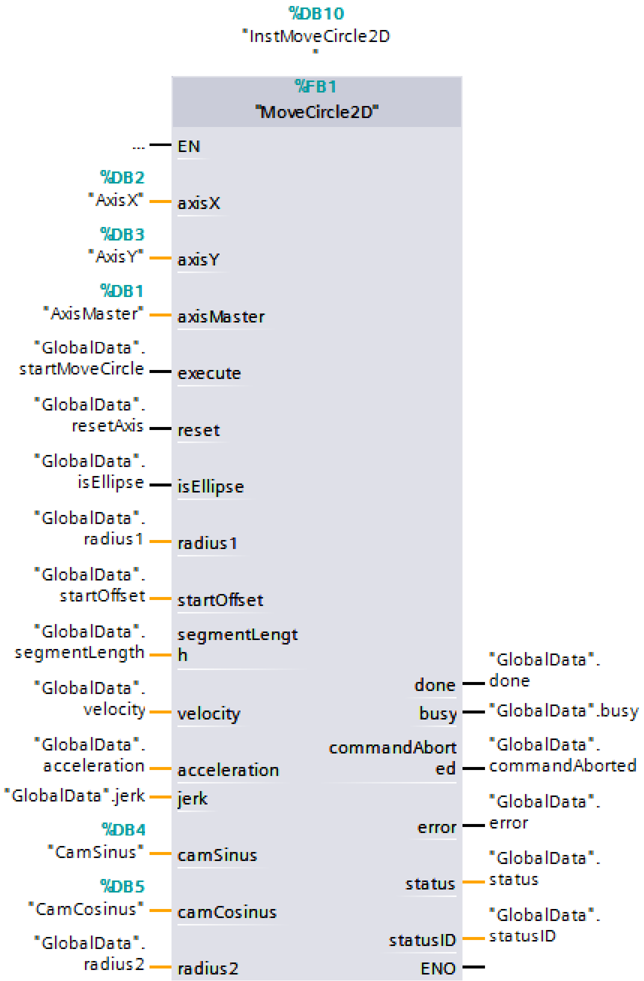
To transfer the function into your user program, proceed as follows:

1. Open the "LMC2D" library in the TIA Portal.
2. Drag the "MoveCircle2D" block from the library into the "Program blocks" folder, using drag-and-drop.

### 2.2.5 Calling the “MoveCircle2D” block in the user program

The “MoveCircle2D” function block is integrated into the processing sequence of the user program by a block call and the transfer of the required parameters. The function block has to be called in a cyclically starting OB or FB block. A timer-interrupt controlled processing (for example, in OB 35) is possible.

Figure 2-8: Calling the "MoveCircle2D” block




**Note** For each interpolated traversing motion with two axes that is performed in parallel to another interpolated traversing motion, you have to assign the “MoveCircle2D” function block to a separate instance data block or separate multi-instance.

### 2.2.6 Managing the axes and cam discs

**Traversing axes "AxisX" and "AxisY"**

The “MoveCircle2D” block takes on the complete control of axes for executing the circle or elliptical motion. You have to manage the axes yourself in your user program.

To manage the axes you require the following technology functions in the user program:

- Release of the axes via “MC_Power”
- Acknowledgement of errors on the axes via “MC_Reset”
- Homing of the axes via "MC_Home"

This ensures that the user program keeps the control over all axes and cam discs and that it can start and implement specific measures in case of emergency (for example, emergency stop, disabling the axis release).

| **Note** | The “ControlAxis” block manages the axes in the test program. |
|---|---|

**Virtual master axis "AxisMaster"**

The "MoveCircle2D" function block fully takes on the management of the virtual master axis.

The "MoveCircle2D" function block takes on the release of the axis and the acknowledgment of errors on the axis.

The virtual master axis must not be influenced from the user program.

**Cam discs "CamSinus" and "CamCosinus"**

The "MoveCircle2D" function block takes on the management of the two required cam discs.

The cam discs must not be influenced by the user program.

## 2.3 Operation

### 2.3.1 Test program

You can test the functionality with the "MoveCircle2D_V11" project. Enter the values on the interface of the "MoveCircle2D" function block via a web user interface. You can test the responses and options of the block with it.

You do not need real machine axes for the test program. The program is instantly runnable on a S7-1500T. The axes are simulated as virtual axes in the controller.

| **NOTICE** | The test program enables you to familiarize yourself with the principle of operation and the reactions of the "MoveCircle2D" block.<br>The test program is not intended for use in real machines and thus not released. |
|---|---|

The following technology objects (axes and cam discs) have already been created:

Table 2-5: Technology objects (axes and cam discs)

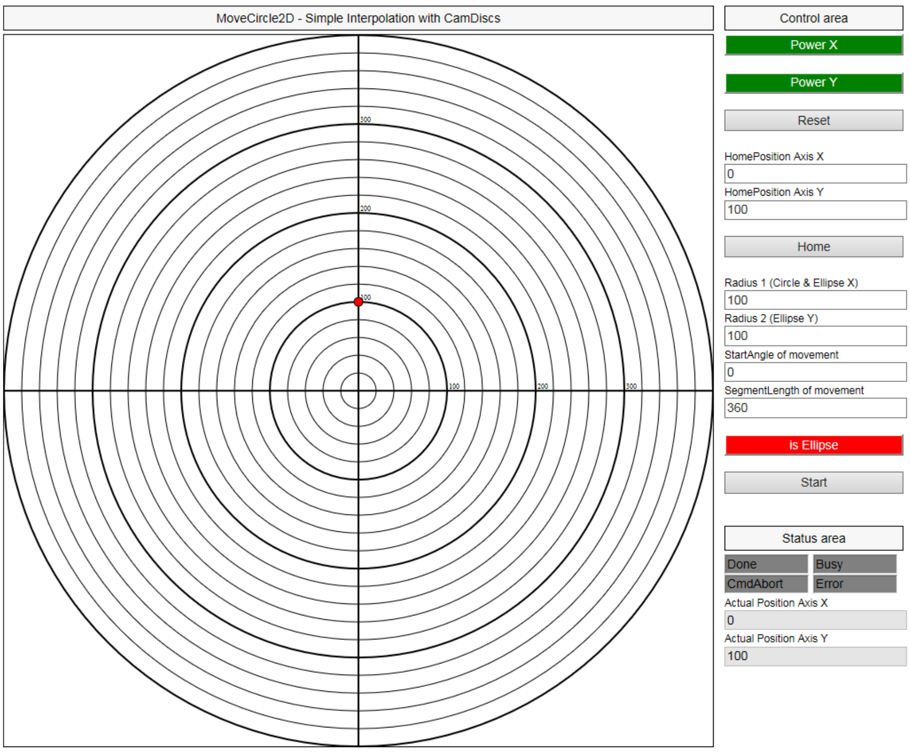
| Technology object | Function | Note |
|---|---|---|
| "AxisX" | Synchronous axis | Slave axis of X-Y plane for executing the circular motion. |
| "AxisY" | Synchronous axis | |
| "AxisMaster" | Positioning axis | Master axis that is managed in the "MoveCircle2D" block. |
| "CamSinus" | Cam disc | Cam disc for generating the sinus through the "MoveCircle2D" block. |
| "CamCosinus" | Cam disc | Cam disc for generating the cosinus through the "MoveCircle2D" block. |

## 2.3.2 The web user interface of the test program

### Overview

The following figure shows you the user-defined web page with which you can operate the test program.

Figure 2-9: Web user interface of the test program




Different operating screens are available to you for operating the test program.

The red dot in the coordinate system shows the current position of the traversing axes X and Y.

In the “Control area” section, the following functions are available:

- Releasing, resetting and homing the axes
- Configuration of the "MoveCircle2D" function block
- Controlling the function block

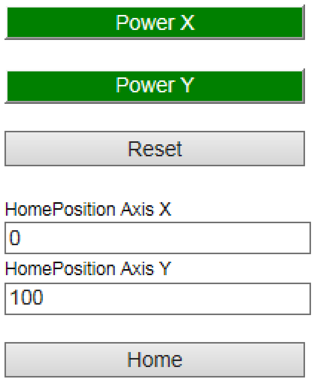
The "Status area" section provides the following information:

- Status of the block outputs
- Current position of the traversing axes X and Y.

### Operating the axes

The axes can be activated, reset and homed to a specified position via the operating screen.

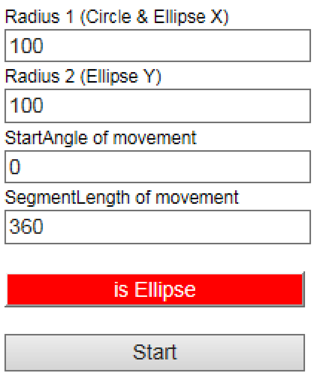
Figure 2-10: Operating the axes




### Operation and configuration of the "MoveCircle2D" block

The parameters of the “MoveCircle2D” block can be controlled via further control elements.

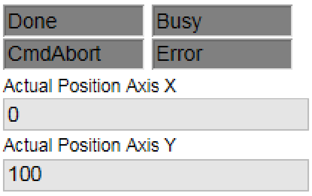
Figure 2-11: Operation and configuration of the block



You can switch between circle and ellipsis, and specify the radius. The start angle and the segment length can also be specified. When you are clicking on the "Start" button, the cam discs are created and the execution of the interpolated motion is started.

The following figure shows the status of the “MoveCircle2D” block and the current position of the axis.

Figure 2-12: Block status

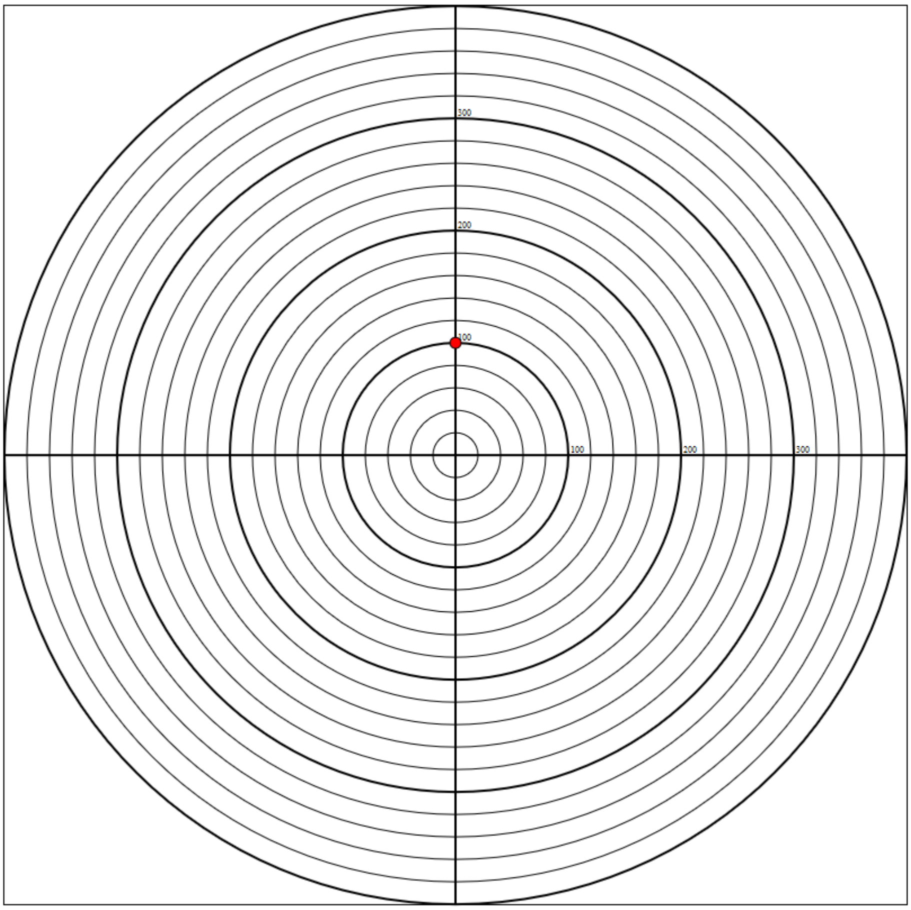



## Monitoring the traverse motion

The execution of the traversing motion can be directly observed on the web interface in the X-Y plane.

Compliance with the defined radii for the circular and elliptical motions can easily be checked via concentric circles on the motion area.

Figure 2-13: Depiction of the traversing motion in the X-Y plane



### 2.3.3 Operating the test program

Proceed as follows to test the "MoveCircle2D" function block:

Table 2-6: Operating the test program

| | Procedure | Remark |
|---|---|---|
| 1. | Load the test program to the controller and set the controller to RUN. | |
| 2. | Start a web browser, for example, the Internet Explorer.<br>Enter the IP address of the CPU as the address, for example, http://192.168.0.1 | The user-defined webpage of the test program opens.<br>(See Figure 2-9: Web user interface of the test program) |
| 3. | Enable the two axes via the respective buttons "Power X" or "Power Y". | See Figure 2-10: Operating the axes |
| 4. | If there are errors, reset them with the “Reset” button. | See Figure 2-10: Operating the axes |
| 5. | Enter the desired starting position via the input fields “HomePosition Axis X” and “HomePosition Axis Y”. Home the axes to this position via the “Home” button. | See Figure 2-10: Operating the axes |
| 6. | Configure the circular motion with the desired values. | **Example:**<br>• Radius 1 = 100.000<br>• StartAngle = 0.000<br>• SegmentLength = 360.000<br>(See Figure 2-11: Operation and configuration of the block) |
| 7. | Start the circular motion via the “Start” button and monitor the traversing motion. | See Figure 2-13: Depiction of the traversing motion in the X-Y plane |
| 8. | Enable the elliptical motion with the “is Ellipse” button. | |
| 9. | Configure the elliptical motion with the desired values. | **Example:**<br>• Radius 1 = 50.000<br>• Radius 2 = 100.000<br>• StartAngle = 0.000<br>• SegmentLength = 360.000<br>(See Figure 2-11: Operation and configuration of the block) |
| 10. | Start the elliptical motion via the “Start” button and monitor the traversing motion. | See Figure 2-13: Depiction of the traversing motion in the X-Y plane |

**Note**

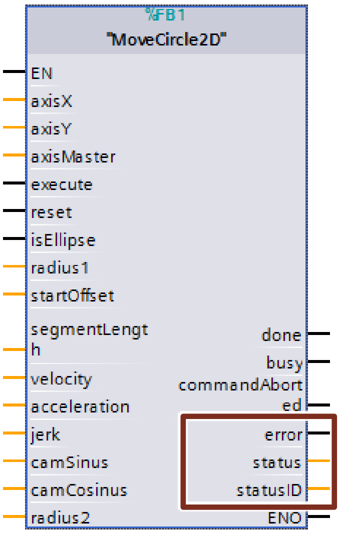
Other examples of circular and elliptical motions can be found in chapter 3.5 Execution examples for circular and elliptical trajectories

# 2.4 Error handling

## 2.4.1 Output of error messages

Errors occurring within the block are signaled via the outputs "error", "status" and "statusID" of the “MoveCircle2D” block.

Figure 2-14: Interfaces for error messages on “MoveCircle2D”




The parameters selected transfer the information on the occurred events to the user program in order to be able to take appropriate measures, if required.

## 2.4.2 Error concept

The display of error messages is based on the following concept:

- **“error” output:**
  If this output is set, the occurred event is an error. The error cause is output on the "status" and "statusID" outputs.
- **“status” output:**
  The error code is output on this output. The following error codes are possible:
  - **Error on a technology function:**
    If an error occurs in the block during the use of a technology function, the error code of the technology function (ErrorID) is directly transferred to the "status" output of “MoveCircle2D” block.
    The error codes of the technology functions and their remedy are described in the TIA Portal V14 online help.
  - **Error in block "MoveCircle2D":**
    If an error occurs in the “MoveCircle2D” block which was not caused by the use of a technology function, the "16#8200" error code is output.
- **“statusID” output:**
  On this output the respective error source of the error codes is output on the “status” output. With this error code you can diagnose the cause of error.

The error sources are described in the following table.

Table 2-7: Error source on the “statusID” output

| statusID | Meaning | Note |
|---|---|---|
| 0 | No error | |
| 1 | Error when plausibility check of the block parameter is carried out. | |
| 2 | Error when resetting or acknowledging the error of the "camSinus" cam disk. | “MC_Reset” |
| 3 | Error when resetting or acknowledging the error on the "camCosinus" cam disk. | “MC_Reset” |
| 4 | Error when interpolating the cam of "camSinus" cam disc. | "MC_InterpolateCam" |
| 5 | Error when interpolating the cam of "camCosinus" cam disc. | "MC_InterpolateCam" |
| 6 | Error when activating the virtual “AxisMaster” axis. | "MC_Power" |
| 7 | Error when resetting or acknowledging the error on the "AxisMaster" axis. | “MC_Reset” |
| 8 | Error when homing of the virtual "AxisMaster" axis. | "MC_Home" |
| 9 | Error when synchronizing the cam disc with the "AxisX" axis. | "MC_CamIn" |
| 10 | Error when synchronizing the cam disc with the “AxisY" axis. | "MC_CamIn" |
| 11 | Error when traversing motion of the virtual "AxisMaster" axis is carried out. | "MC_MoveAbsolute" |
| 12 | Error when stopping of "AxisX" axis. | "MC_Halt" |
| 13 | Error when stopping of “AxisY" axis. | "MC_Halt" |
| 14 | Error when traversing motion of the virtual "AxisX" axis is carried out. | "MC_MoveAbsolute" |
| 15 | Error when traversing motion of the virtual "AxisMaster" axis is carried out. | "MC_MoveAbsolute" |
| 16 | Error when reading the following value from cam disc of "AxisX" axis. | "MC_GetCamFollowingValue" |
| 17 | Error when reading the following value from cam disc of "AxisY" axis. | "MC_GetCamFollowingValue" |

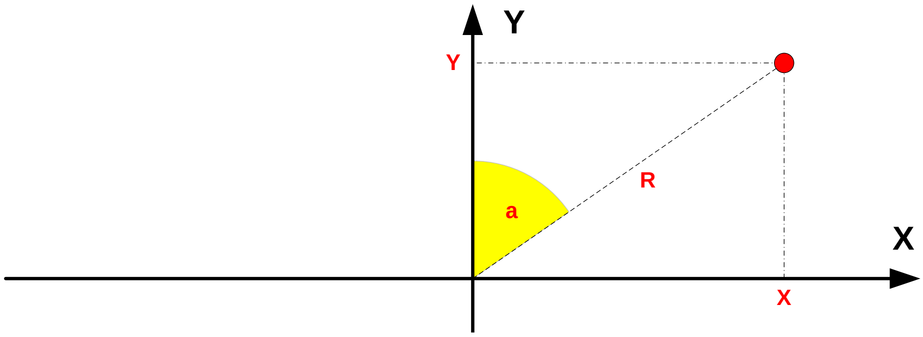
# 3 Valuable Information

## 3.1 Location of a point in the XY plane

You can locate a point in the XY plane in two different ways:

- **Location via the X-Y coordinates**
  The precise location of the point is determined by specifying the coordinates in X and Y direction of the coordinate system. This depiction is the common display method for points in a plane.
- **Location through radius and angle**
  The location of a point can also be specified via the radius “R” and the angle “a”. “R” is the distance between the point and the origin of the coordinate system. “a” is the angle between positive Y axis and the connecting line of the coordinate origin to the point. This display is suitable to specify a point on a circular path.

Figure 3-1: Location of a point in the plane






### Unit system of the axes in the XY plane

To be able to perform a defined motion on a circular or elliptical trajectory, all axes must have the same unit system. Furthermore, the axes have to be designed in a way so that the motion does not reach the set limit values.

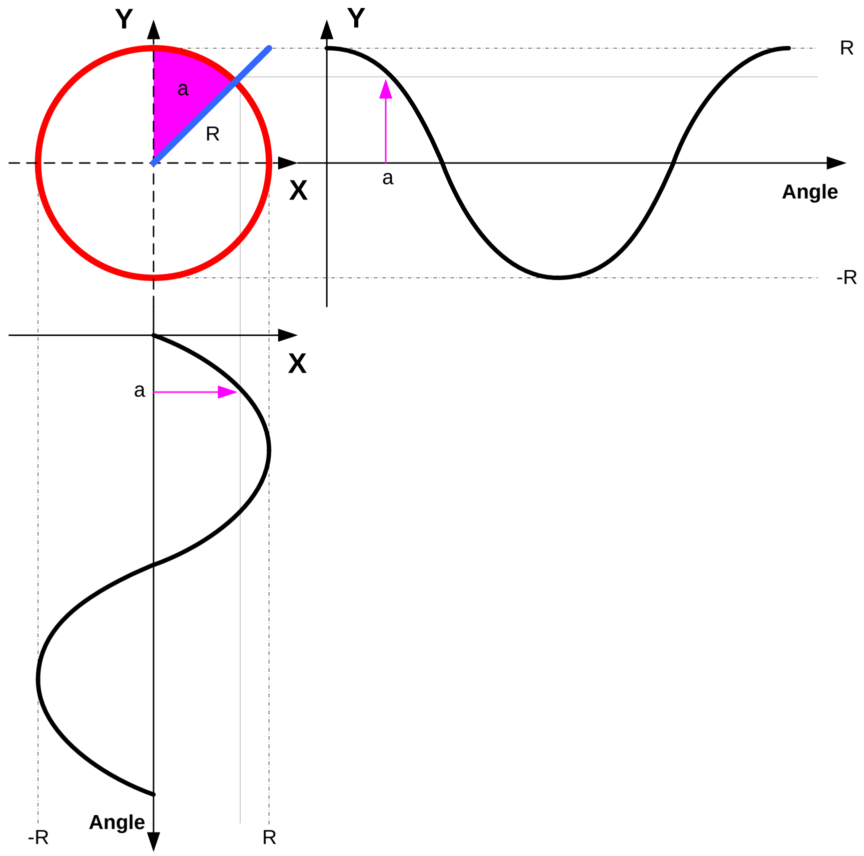
## 3.2 Realizing a circular and elliptical motion

To be able to understand the structure and function of the application example, this chapter explains the fundamental realization of a circular and elliptical motion.

### 3.2.1 The circle

A circular motion in the X-Y plane can be reached by superimposed motion of X and Y axis of a machine, by one axis performing a sinus motion, while the Y makes a cosinusoidal motion.

Figure 3-2: Circular motion



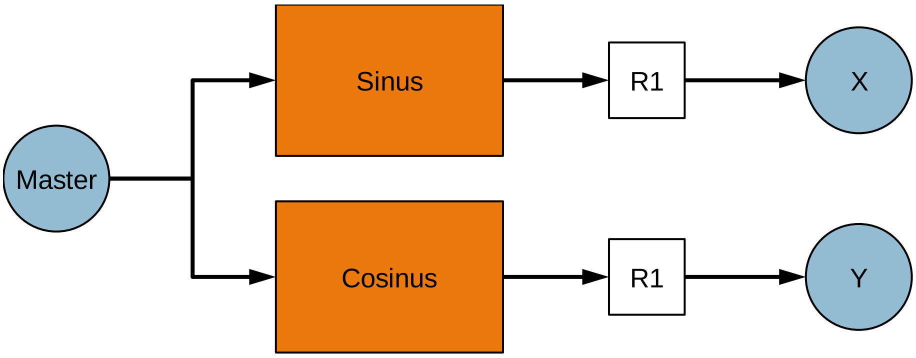

The overlapping of the two motions is based on the segment angle that is overswept when moving on the circular path. The angle count starts on the positive Y axis of the coordinate systems and runs in clockwise direction along the circular path. Depending on the size of the angle specification, either only a section of the circular arc is covered or up to several “revolutions" of the circle can be reached.

For the circular path the amplitudes (the maximum deflection of sinus and cosinus) depend on the desired radius of the circle.

### Realization in SIMATIC

In the SIMATIC, this overlapping is realized with two cam discs which synchronize axes X and Y with a master axis. The master axis represents the angle that is overswept during the circular motion.

Figure 3-3: Realization of a circular motion with the technology CPU

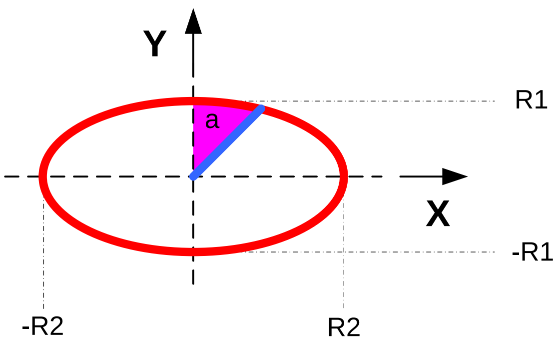



Sinus and cosinus are formed via the respective cam discs. The master axis provides the current angle as an input value. The multiplication of the output values of the cam discs with radius "R" of the circle is reached via scaling the slave values of the cam discs during synchronization via the technology function "MC_CamIn".

## 3.2.2 The ellipse

The ellipse differs from the circle by the fact that a respective different radius can be specified for the X and Y axis.

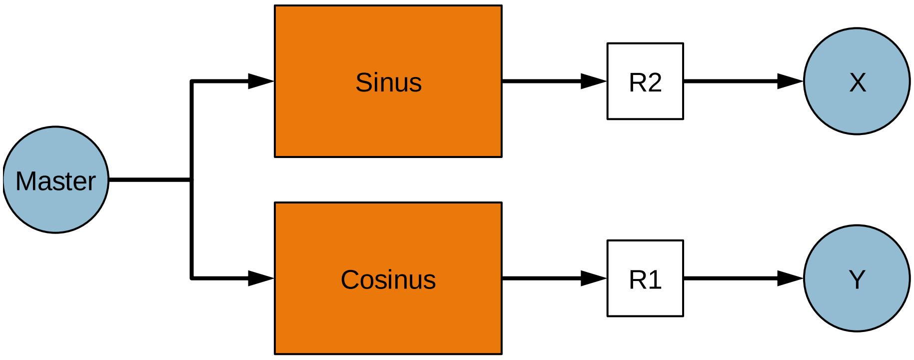
Figure 3-4: Radii of the ellipse



**Realization with the SIMATIC**

In the SIMATIC the same setup of axes and cam discs can be used for the realization of an elliptical trajectory. Only the scaling of both cam discs must be performed with the appropriate radii during synchronizing via the “MC_CamIn” technology function.

Figure 3-5: Realization of an elliptical motion with the technology CPU




## 3.3 Handling of the curve discs

### 3.3.1 Cam disc definition via polynomial assignment

For this form of cam disc definition, the cam disc is specified via the coefficients of a mathematical polynomial with linear and trigonometric fraction. The polynomial is as follows:

$$Y = a_0 + a_1 \cdot X + a_2 \cdot X^2 + a_3 \cdot X^3 + a_4 \cdot X^4 + a_5 \cdot X^5 + a_6 \cdot X^6 + \\ + B_0 \cdot \sin((B_1 \cdot X) + B_2)$$

| | | |
|---|---|---|
| with | $a_0$ to $a_6$ | Coefficients of the linear fraction |
| | $B_0$ = sineAmplitude | Amplitude of the sinus curve |
| | $B_1$ = sinePeriod | Period of the sinus curve |
| | $B_2$ = sinePhase | Phase of the sinus curve |

The course of the curve for the determination of the coefficient of the linear and trigonometric fraction must always be specified in standardized form. This means that the coefficients of the polynomial must be determined in a way so that the desired course of the curve is in the value range of **0 ≤ X ≤ 1** of the X variable.

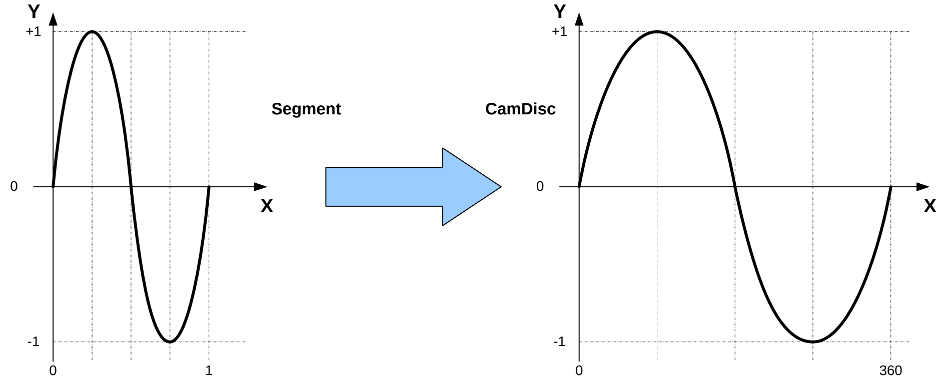
The transformation of the standardized curves in the desired target value area is then reached by the specification of the start and end coordinate "xmin" and "xmax" of the segment.

**Note**

The trigonometric fraction of the polynomial has to be specified in radian measure. The conversion factor between degree and radian measure is:

2π / 360° ≈ 0.01745329

Figure 3-6: Example for the transformation of the segment in the curve disc




The variable structure of a segment is defined as follows.

Table 3-1: Variable structure for a segment of the curve disc

| Variable | Description |
|---|---|
| xmin | Start coordinate of the segment |
| xmax | End coordinate of the segment |
| a0 | Coefficient A0 for $x^0$ the polynomial for the segment (linear fraction) |
| a1 | Coefficient A1 for $x^1$ the polynomial for the segment (linear fraction) |
| a2 | Coefficient A2 for $x^2$ the polynomial for the segment (linear fraction) |
| a3 | Coefficient A3 for $x^3$ the polynomial for the segment (linear fraction) |
| a4 | Coefficient A4 for $x^4$ the polynomial for the segment (linear fraction) |
| a5 | Coefficient A5 for $x^5$ the polynomial for the segment (linear fraction) |
| a6 | Coefficient A6 for $x^6$ the polynomial for the segment (linear fraction) |
| sineAmplitude | Amplitude of the sinus element (trigonometric fraction) |
| sinePeriod | Period length of the sinus element in radian measure (trig. fraction) |
| sinePhase | Phase shift of the sinus element in radian measure (trig. fraction) |

The definition of the coefficient for the linear fraction and the trigonometric fraction of the polynomial is done segment by segment in the above shown variable structure. This is where the target area is also specified for the respective segment by definition of the start and end coordinate "xmin" and "xmax" of the segment in X direction.

Afterwards the segment has to be activated. To do this the "ValidSegment[n]" variable of the "n" segment used is set to "True".

### 3.3.2 Generating a cam disc for sinus and cosinus

Generating a circular motion requires cam discs in form of a sinus and a cosinus. These cam discs can simply be generated via the polynomial segments of a cam disc.

The sinus cam disc can be defined directly via the trigonometric part of the polynomial segment.

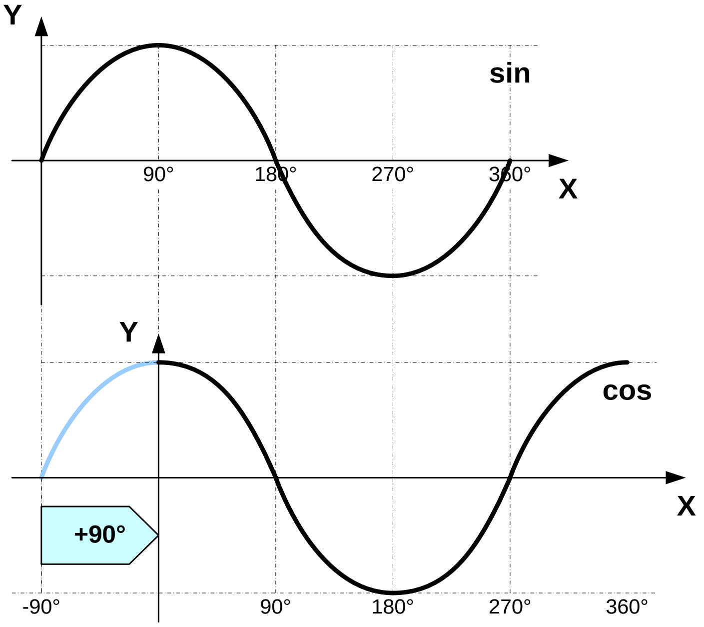
$$Y = B_0 \cdot \sin(B_1 \cdot X) \qquad B_2 = 0$$

The cosinus cam disc is generated by shifting the sinus in X direction by 90°, this can be achieved by specifying the phase $B_2$ of the sinus.

$$Y = B_0 \cdot \cos(B_1 \cdot X) = B_0 \cdot \sin((B_1 \cdot X) + \frac{\pi}{2}) \qquad B_2 = +\frac{\pi}{2}$$

| | | |
|---|---|---|
| with | $B_0$ = sineAmplitude | Amplitude of the sinus curve |
| | $B_1$ = sinePeriod | Period of the sinus curve |
| | $B_2$ = sinePhase | Phase of the sinus curve |

Figure 3-7: Generating the cam disc for sinus and cosinus




The curves for sinus and cosinus each have to be standardized for the range **0 ≤ X ≤ 1** in the trigonometric fraction of the segment. The function value of the two angle functions have to be in the range of **-1 ≤ Y ≤ +1**. Thus the following assignment results for the two missing parameters of the angle functions:

$$B_0 = 1 \quad \text{and} \quad B_1 = 2\pi$$

**Note**

The trigonometric fraction of the polynomial has to be specified in radian measure. The conversion factor between degree and radian measure is:

2π / 360° ≈ 0.01745329.

### 3.3.3 Considering the start angle

If the motion on the circular or elliptical trajectory is to be executed from a defined starting angle on, this starting angle must already be considered in the generation of the cam discs for sinus and cosinus. The master axis is then only moved by the desired segment length, starting from zero.

To do this, the cam discs have to be shifted by the given starting angle. This is done via the "sinePhase" parameter of the polynomial segment of the cam disc. A shift of the cam disc can be achieved via this parameter just as for forming the cosinus. To do this, the "startOffset" angle is converted to the radian measure and added to the value of the "sinePhase" parameter.

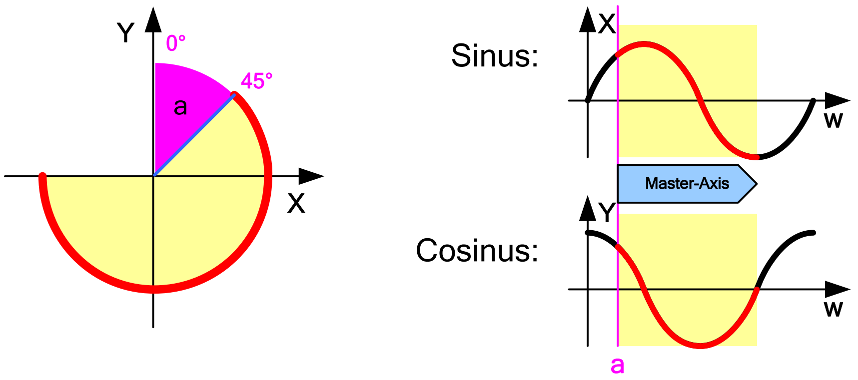
For the sinus, the "sinePhase" phase shift is calculated as follows:

$$sinPhase = 0 + startOffset \cdot \frac{2\pi}{360°}$$

For the cosinus, the "sinePhase" phase shift is calculated as follows:

$$sinPhase = \frac{\pi}{2} + startOffset \cdot \frac{2\pi}{360°}$$

Figure 3-8: Starting angle of the circular motion



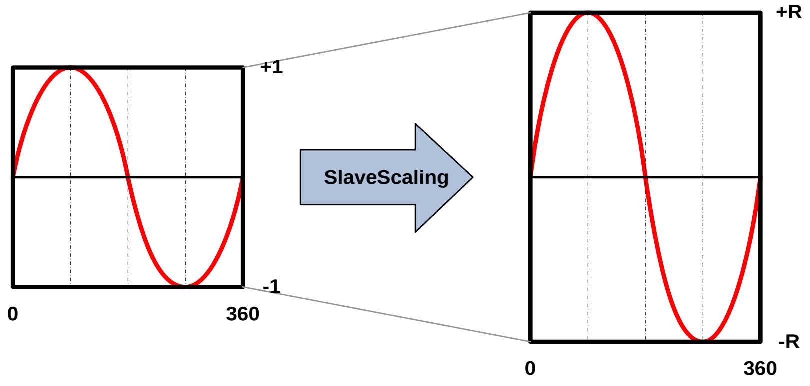

### 3.3.4 Scaling the cam disc

The scaling of the cam disc is adjusted to the desired circular radius via the circular motion. To do this, the desired circular or elliptical radius is specified as scaling factor for the slave axis on the "SlaveScaling" parameter of the "MC_CamIn" technology function (cam disc synchronism).

The following applies:

- Scaling X axis: camInX.SlaveScaling = radius1
- Scaling Y axis: camInY.SlaveScaling = radius2

Figure 3-9: Scaling of the "CamSinus" cam disc

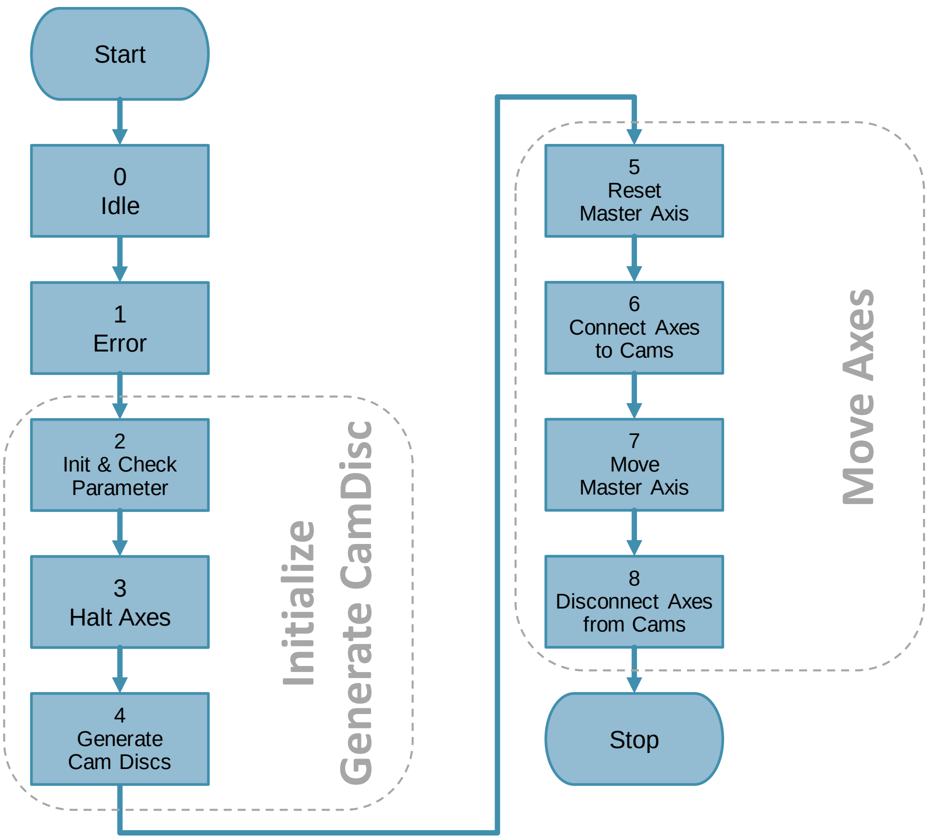



## 3.4 Possible operating states

To execute an interpolated motion on a circular or elliptical trajectory, the “MoveCircle2D” block successively executes several functions which are displayed in the figure below.



Figure 3-10: Possible statuses of FB "MoveCircle2D"



The individual steps of the block have the following function:

Table 3-2: Description of the statuses of FB "MoveCircle2D"

| Step | | |
|---|---|---|
| 0 | Idle | No action. |
| 1 | Error | If an error occurs while executing the individual functions of the block or a technology function of the block is superseded, the block branches into this status and outputs an error code for exact localization of the cause of the error. |
| 2 | Initialize & Check Parameter | In this step the internal variables of the block and the functions called in the block are set to defined values and the input parameters of the block are checked. |
| 3 | Halt Axes | Prior to executing the following functions the axes are brought to a defined status. |
| 4 | Generate Cam Discs | During this step, the cam discs required for the interpolated motion are generated in the integrated technology of the CPU via mode selection and input parameters. |
| 5 | Reset Master Axis | The master axis is reset and brought to a defined status for executing the desired motion. |
| 6 | Connect Axes to Cams | The X and Y axes are synchronized to the master axis via the basic synchronism with the generated cam discs. |
| 7 | Move Master Axis | The master axis is started and moves along the configured segment length so the desired circular or elliptical motion is executed. |
| 8 | Disconnect Axes from Cams | The cam disc synchronization with the X and Y axis is cancelled. |
| 9 | End | The motion on the circular or elliptical trajectory is cancelled and the block waits for a new traversing task. |



**Determining the current block status**

The current block status or the current step in which the "MoveCircle2D" block is currently located can be determined via the integer "sequencer" variable in the instance data block of FB "MoveCircle2D".

## 3.5 Execution examples for circular and elliptical trajectories

This chapter introduces some examples that can be realized with the “MoveCircle2D” block.

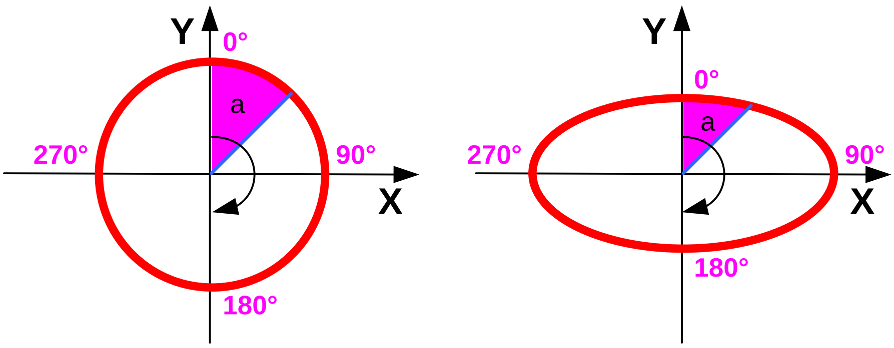
### 3.5.1 Starting point of the circle or ellipse

The circle or the ellipse always starts at the current position of the X and Y axis. The circle or ellipse can be aligned at the starting point via the starting angle of the circle.

### 3.5.2 Orientation of circle and ellipse

The angle count within the circle or ellipse starts at the positive Y axis or the X-Y plane. The positive direction of rotation is in clockwise direction from the positive Y axis to the positive X axis.

Figure 3-11: Orientation of circle and ellipse



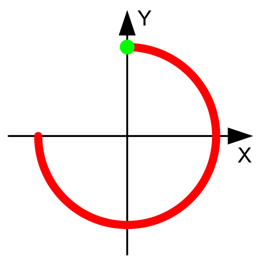
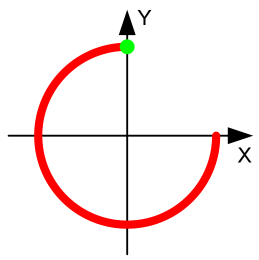

### 3.5.3 Examples for circular motions

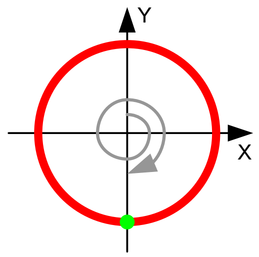
Here some examples for circular motions, which can be performed with the "MoveCircle2D" block.

**Simple motion**

A simple motion is performed via the “Start” button of the web interface.

Table 3-3: Examples for simple circular motions

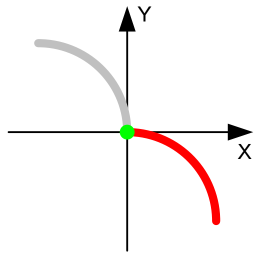
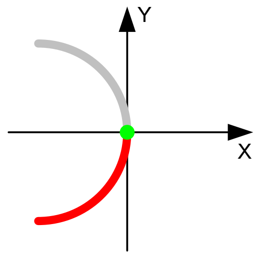
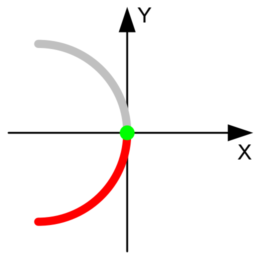
| Description | Figure |
|---|---|
| **Parameter:**<br>Radius = 100 mm<br>Starting angle = 0°<br>Segment length = +270°<br><br>**Motion:**<br>Starting at the current position of the axes, a clockwise motion on a ¾ circle is performed. |  |
| **Parameter:**<br>Radius = 100 mm<br>Starting angle = 0°<br>Segment length = -270°<br><br>**Motion:**<br>Starting at the current position of the axes, an anti-clockwise motion on a ¾ circle is performed. |  |

| Description | Figure |
|---|---|
| **Parameter:**<br>Radius = 100 mm<br>Starting angle = 0°<br>Segment length = +540°<br><br>**Motion:**<br>Beginning from the starting point of the circular motion, a 1.5 times rotation around the center of the circle is performed. |  |

## Chained motions

A chained motion is performed through the multiple start of the "MoveCircle2D" block via the “Start” button of the web interface. The axes stop between the individual motions. Each motion is realized via an independent start of the block.

Table 3-4: Examples for chained circular motions

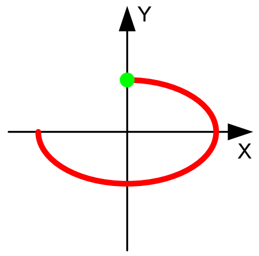
| Description | Figure |
|---|---|
| **Parameter:**<br>Radius = 100 mm<br>Starting angle = 0°<br>Segment length = +90°<br><br>**Motion:**<br>At the end of the previous motion, a clockwise ¼ circle is attached. |  |
| **Parameter:**<br>Radius = 100 mm<br>Starting angle = +90°<br>Segment length = +90°<br><br>**Motion:**<br>A semi-circular motion with stop point half way is performed by attaching a ¼ circle in clockwise direction. |  |
| **Parameter:**<br>Radius = 100 mm<br>Starting angle = 0°<br>Segment length = +90°<br><br>**Motion:**<br>The previously described motion can also be reached by rotating the coordinate system. The basic configuration of the previous circular motion may be retained and only the rotation of the coordinate system must be adjusted. |  |

### 3.5.4 Example for elliptical motions

This section gives an example for an elliptical motion, which can be performed with the "MoveCircle2D" block.

As soon as the "elliptical motion" function has been selected, two radii for both main axes of the ellipse or the web interface can be defined.

Table 3-5: Example for elliptical motion

| Description | Figure |
|---|---|
| **Parameter:**<br>Radius X = 100 mm<br>Radius Y = 50 mm<br>Starting angle = 0°<br>Segment length = +270°<br><br>**Motion:**<br>Starting at the current position of the axes, a ¾ clockwise motion on an ellipse is performed. |  |

# 4 Appendix

## 4.1 Service and Support

### Industry Online Support

Do you have any questions or need support?

Siemens Industry Online Support offers access to our entire service and support know-how as well as to our services.

Siemens Industry Online Support is the central address for information on our products, solutions and services.

Product information, manuals, downloads, FAQs and application examples – all information is accessible with just a few mouse clicks at https://support.industry.siemens.com.

### Technical Support

Siemens Industry's Technical Support offers quick and competent support regarding all technical queries with numerous tailor-made offers – from basic support to individual support contracts.

Please address your requests to the Technical Support via the web form: https://support.industry.siemens.com/My/ww/en/requests.

### Service offer

Our service offer comprises, among other things, the following services:

- Product Training
- Plant Data Services
- Spare Parts Services
- Repair Services
- On Site and Maintenance Services
- Retrofit & Modernization Services
- Service Programs and Agreements

Detailed information on our service offer is available in the Service Catalog: https://support.industry.siemens.com/cs/sc

### Industry Online Support app

Thanks to the "Siemens Industry Online Support" app, you will get optimum support even when you are on the move. The app is available for Apple iOS, Android and Windows Phone. https://support.industry.siemens.com/cs/sc

## 4.2 Links and Literature

Table 4-1

| No. | Topic |
|---|---|
| \1\ | Siemens Industry Online Support<br>https://support.industry.siemens.com |
| \2\ | Link to this entry<br>https://support.industry.siemens.com/cs/ww/en/view/109742306 |
| \3\ | STEP 7 Professional V14.0 System Manual<br>https://support.industry.siemens.com/cs/ww/en/view/109742272 |

## 4.3 Change documentation

Table 4-2

| Version | Date | Modifications |
|---|---|---|
| V1.0 | 11/2016 | First version |
| V1.1 | 04/2017 | • Upgrade to STEP 7 Professional V14 SP1<br>• Constants for "statusID"<br>• Home-Position Master before MasterStartDistance<br>• Disconnect with MC_MoveAbsolute instead of MC_Halt<br>• Calculations with set positions instead of actual positions |