Spline interpolation with the APCI-8001 / APCI-8008

PCAP programming

The following functions can be used in PCAP programming to run spline traversing profiles.

Important: These functions can only be used if an RWMOS.ELF with the option "SPLINE" is used. The options of RWMOS.ELF can be seen after booting in fwsetup.exe.

LdSSplineA – Load Spooled Spline Absolut

Transfer data record for spline interpolation (spline vector) with absolute coordinates to the control unit. Call like for SMLA.

LdSSplineR – Load Spooled Spline Relativ

Transfer data record for spline interpolation (spline vector) with relative coordinates to the control unit. Call like for SMLR.

ExeSSpline – Execute Spooled Spline

Execute loaded spline data records. This command corresponds to the SSMS command for linear interpolation. The target velocity specified in the "tvl" element of the "Imp" parameter is executed in the direction of the last sub-profile.

Spline interpolation using rw_SymPas

SMSA

Load absolute spline record (parameter as for SMLA).

SMSR

Load relative spline record (parameter as for SMLR).

SSMS, SSMSW

Calculate and run spline. Velocities, accelerations and units are specified analog to the traversing commands for linear interpolation. The corresponding system variables are as follows:

TRVL	Trajectory velocity
TRTVL	Trajectory target velocity
TRAC	Trajectory acceleration
PU	Position unit for velocity and acceleration
TU	Time unit for velocity and acceleration

Velocity profiles

Normally, a spline contour is run with a trapezoidal velocity profile. Acceleration, maximum velocity and target velocity are specified in the "Imp" parameter of the "ExeSSpline" function. If the maximum velocity cannot be reached, the velocity has a triangle-shaped traverse, comprising an acceleration and a deceleration phase. Trajectory velocity and trajectory acceleration are based on the trajectory to be run, i.e. the geometric sum of the individual traverse paths. Non-feedrate axes can take part in the interpolation but are not used to calculate the trajectory traverse path.
Here, it should be noted that the actual trajectory velocity on a spline curve is not constant but is subject to certain fluctuations around the programmed value due to the curvatures of the individual traverse paths.
From RWMOS.ELF V2.5.3.90, it is also possible to specify a velocity and an acceleration for each spline vector. For this, the LookAhead bit must be set in the MODEREG register.
In this case, the acceleration and the velocity for each spline vector are taken into account in the "Imp" parameter (ac and vl) with the functions "LdSSplineA" and "LdSSplineR". If the velocity of a vector is increased, then from this vector, acceleration is increased using the specified acceleration ac until the desired traversing speed is reached. The acceleration process can also span multiple spline vectors. The specified acceleration is then taken into account in each vector.
If the velocity of a vector is reduced, then the velocity in the preceding vectors is reduced so that the trajectory velocity on the relevant vector is decelerated to the specified velocity. This deceleration process can span multiple vectors with the specified accelerations.

Additional notes on running splines

Within a spline contour, the same axes must always be used when programming the contour and at the start of the execution. These splines are natural splines. The programmed spline vectors are executed on the contour to be run.
When ExeSSpline is called, the spline function is calculated and executed along the entire previously programmed contour. This means that by default 1000 calibration points can be connected using one spline function.
If more than 1000 spline calibration points are to be executed in a contour, a valid value must be assigned to the system variable *SplineBlockSize*. This means that individual parts of the spline contour are processed and executed block by block. Reasonable values for this block size are between 20 and 300.
The system variable SplineBlockSize is programmed with the PCAP command *wrSplineBlockSize()* or *rdSplineBlockSize()* for reading back. In SAP programming, the variable *SPLINEBLOCKSIZE* can be used directly.
The spline interpolation can be run on any number of axes.