



Until recently, the use of robots has always required a special CPU. The robot controller often had to be connected to the actual machine controller by a special fieldbus and the programming of the robot required a special software package and a special, usually manufacturer-specific programming language. However, the different controllers with the fieldbus connected in-between allowed no synchronization between robots and Motion Control.

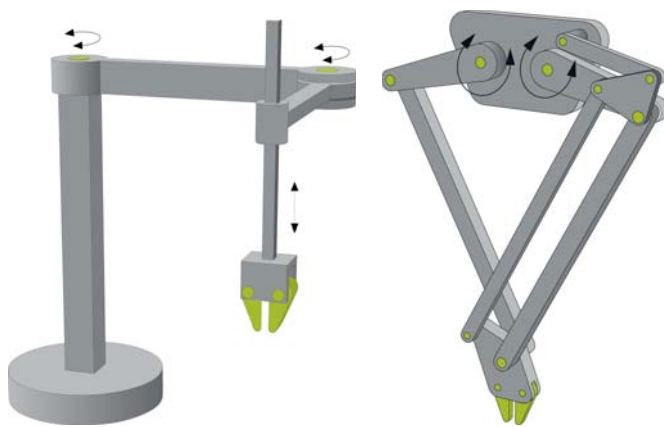
The solution is the realization of the robot in software. This eliminates the costs of the robot controller, the wiring and configuration of the fieldbus and, in particular, the costs of programming the robot. High-precision synchronization can be achieved by means of reducing the communication between the standard controller and robot to pure memory access. This way, a part on a conveyor belt operated with standard Motion Control can be grasped and set aside by the robot 'on the fly.'

### Kinematic transformation

Kinematics (from the Greek 'kinema,' motion) is the study of the movement of points and bodies in space, described by the variables:

- | path  $s$  (change of position coordinates)
- | velocity  $v$
- | acceleration  $a$

The causes of the movement (forces) are not taken into account. In the context of robotics, the term 'kinematics' refers to the different movement possibilities. Since the structure and number of axes determine the



Examples of serial kinematics: SCARA and crane kinematics

workspace of the robot, this workspace is concretely dependent on many parameters: arm lengths, angular range, center of gravity, maximum load, etc. The arrangement of the arms and joints determines the kinematic structure, which is divided into two main classes:

### Serial kinematics:

The current position of any axis is always dependent on the position of the preceding axis. The TCP (Tool Center Point) is changed by each axis in the  $x$ ,  $y$  and  $z$  planes. Examples: SCARA and crane kinematics

### Parallel kinematics:

The current position of the axis is influenced by the preceding axis in one direction only. The TCP is only changed in one plane. Examples: delta kinematics, shear kinematics



Examples of parallel kinematics: Delta and shear kinematics

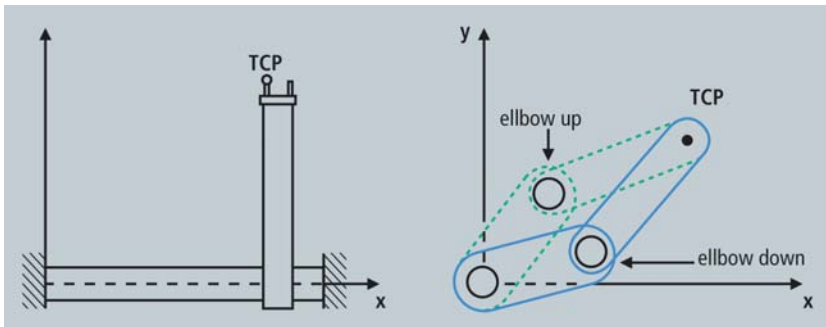
### Coordinate systems

Coordinate systems are required in order to describe the positional behavior of a system. These are placed in the individual active joint axes. To this end one can refer, for example, to Cartesian coordinate systems, which are bound in such a way to the individual bodies that a rotation or a translation takes place around or, as the case may be, towards the coordinate axes. Different coordinate systems can be used as a basis for programming:

- | The Cartesian coordinate system
  - right-handed
  - direction of rotation always positive (counter-clockwise)

- | The Piece Coordinate System (PCS) is independent of the kinematics employed and is used for part-specific orientations.
- | The Machine Coordinate System (MCS) is independent of the kinematics employed and is used for machine-specific orientations.
- | The Axis Coordinate System (ACS) depends on the kinematics employed and is used for referencing or homing.

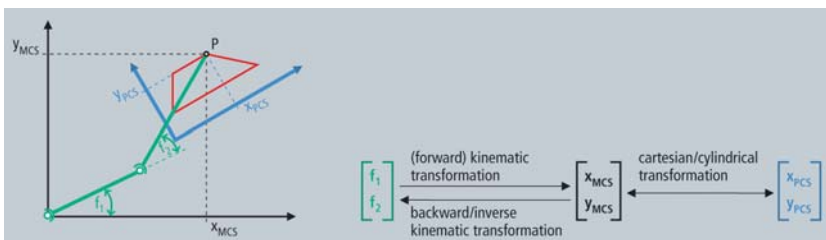
The robots are often programmed in the PCS or MCS, since both systems are very clear due to the Cartesian structure. The type of kinematics in these systems does not need to be taken into account, since the appropriate motion sequences are calculated by a transformation. As opposed to this, when programming in the ACS, it is essential to take into account the structure of the axes, since the movement commands for the axes must be programmed directly. Therefore, this type of programming is used only in exceptional cases.



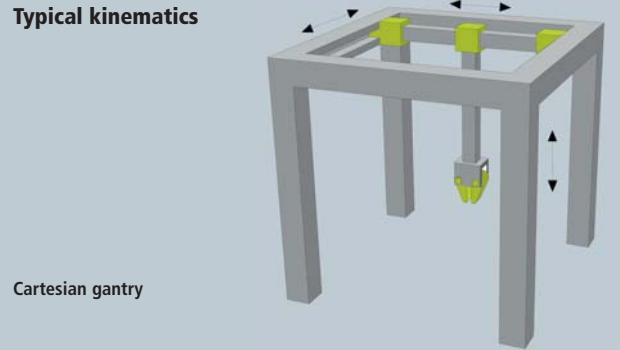
**Transformation**

Kinematics describes the investigation of the possible movements of the individual limbs of the robot in relation to each other. It takes into account the velocities and accelerations that occur during the movement of the joints, but not the forces that occur or the type of drive of the joint (active/passive). A different arrangement of joints and limbs can produce an identical Tool Center Point (TCP) movement path.

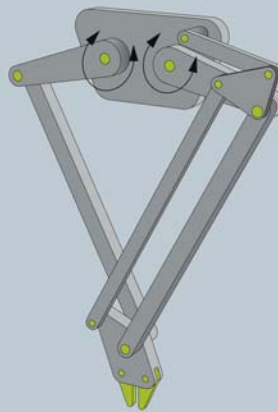
Transformation describes, in the context of the kinematics, the calculation necessary in order to change from one coordinate system to another. There are basically two problems in considering the kinematics of robots. The direct kinematic problem (KP), also called forward transformation, deals with the calculation of the position of the TCP in spatially fixed coordinates from the axis-specific joint coordinates of the robot. The inverse kinematic problem (IKP), also called backward transformation, is the reverse relationship, in which the axis-specific joint coordinates are to be determined from the TCP position. The task of a transforma-



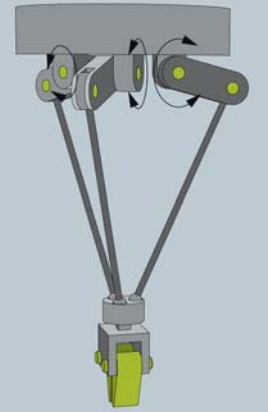
**Typical kinematics**



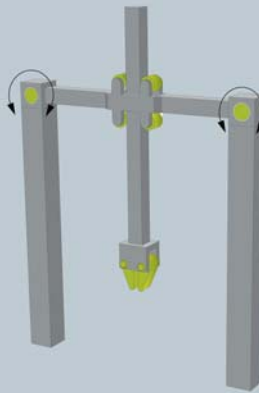
Cartesian gantry



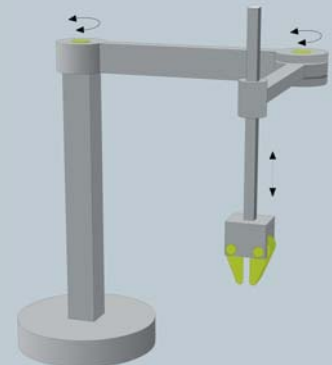
2-D parallel kinematics



3-D delta kinematics



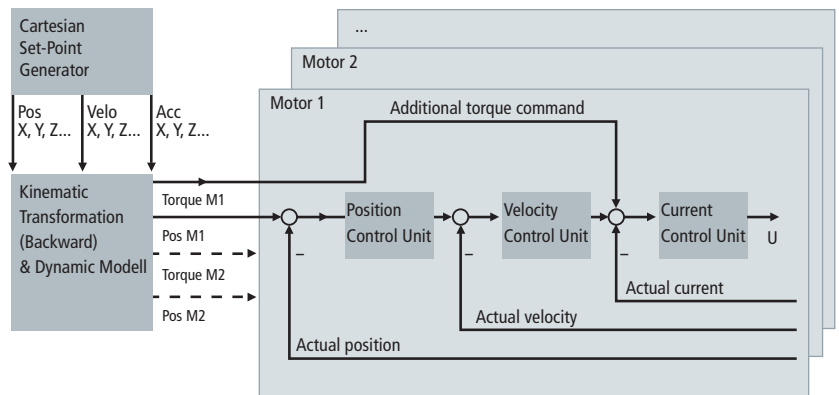
Roller kinematics (H-Bot)



SCARA kinematics



Shear kinematics



Mode of operation of the backward transformation and of the dynamic model

tion is to change the position and orientation of the objects relative to one another so that the TCP traverses the desired movement path.

**Tracking**

Naturally a robot does not exist on its own in the world, but must cooperate with other machine components. A frequent task is driving to a specific position on a moving belt and, of course, synchronous movement along with the belt. For example, the robot is supposed to pick up a part from a specific belt or to set it down in a certain position. The synchronization and movement of the robot along with a moving part – in the translatory or rotary coordinate system – is referred to as tracking.

**Realization in TwinCAT**

TwinCAT Kinematic Transformation integrates itself transparently in the existing motion control world: robotic and motion control functions can be synchronized optimally using TwinCAT NC PTP (point-to-point axis positioning) or NC I (axis interpolation in three dimensions). All PLC and NC characteristics can be combined as desired on a common hardware and software platform.

TwinCAT Kinematic Transformation implements various robot kinematics on the PC. In addition to 2-D kinematics, it can also be used to calculate 3-D bar kinematics, delta kinematics as well as SCARA kinematics. The axes are controlled directly from the TwinCAT Motion Control system. The user thereby programs the robot movement directly in the Cartesian coordinate system. The transformation into the robot coordinate system is calculated in software during each cycle. In order to minimize the oscillation tendency and to increase the positioning accuracy, another current pre-control can be activated for many kinematics. This is naturally only possible if the drive amplifiers and the fieldbus are fast enough and interfaces are available for an additional current pre-control. EtherCAT and the AX5000 Servo Drives from Beckhoff fulfill precisely these requirements.

The robot is configured in the familiar TwinCAT System Manager, the tool for configuring the I/Os, all axes and the robot. The bar and arm lengths as well as kinematic offsets are parameterized here. Masses and mass inertias of the robot can also be specified optionally for the pre-control.

**Outlook**

The TwinCAT Kinematic Transformation package provides the TwinCAT user the possibility to simply integrate a robot of e.g. the type delta or SCARA kinematics. Configuration takes place in the TwinCAT System Manager. The calculation of the dynamic model makes it possible to minimize the path tracking distance. As a result, the position is maintained very precisely with high repeatability.

Further transformations are easy to develop. Simple gantries, cable pull transformation and articulated robots will shortly be on the agenda. Beckhoff showed a new item in a live demonstration at the Automatica trade show (in Munich, Germany): an extension of TwinCAT software by 6-D kinematics for cable robots. The advantages of the cable robot for pick-and-place tasks result from its mechanical structure: Elaborate carbon rods are replaced by economical cables, the arrangement of the servo motors is flexible (e.g. also underneath the robot) and the workpiece gripper can be rotated and tilted.

TwinCAT Kinematic Transformation [www.beckhoff.com/kinematics](http://www.beckhoff.com/kinematics)



Dr.-Ing. Josef Papenfort,  
TwinCAT Product Manager, Beckhoff



Dipl.-Ing. Klaus Bernzen, TwinCAT  
Product Manager for Motion Control &  
Robotics, Beckhoff