

Sommario

DEVICE EANPOS	3
1. Introduction	3
1.1 Installation	4
1.1.1 Device declaration in the configuration file (.CNF)	4
1.1.2 Example	4
1.2 Operation	4
1.2.1 Position detection	4
1.2.2 Definitions	4
1.2.3 Decimal point	4
1.2.4 Speed	4
1.3 Commands and parameters	5
1.3.1 Symbols used	5
1.3.2 Main controls	5
1.3.3 Description of the epicycloidal motion	6
1.3.4 Profile reduction	7
1.3.5 Type of stop during the acceleration ramp	8
1.3.6 Analogue output calibration	9
1.3.7 Movement	10
1.3.8 PID+FF calibration	10
1.3.9 Preset search	13
1.4 Special functions	15
1.4.1 Multi axis management	15
1.4.2 Change quota and counting in motion	15
1.4.3 Change speed and moving ramp time	15
1.4.4 Delta count	16
1.5 Device error management	17
1.6 Device warning management	17
1.7 Commands and parameters table	17
1.7.1 Symbols used	17
1.7.2 Commands	17
1.7.3 Parameters	19
1.7.4 State	21
1.8 Limitations	22
1.9 Application example	23
1.9.1 Configuration file	23
1.9.2 EANPOS management	23

DEVICE EANPOS

1. Introduction

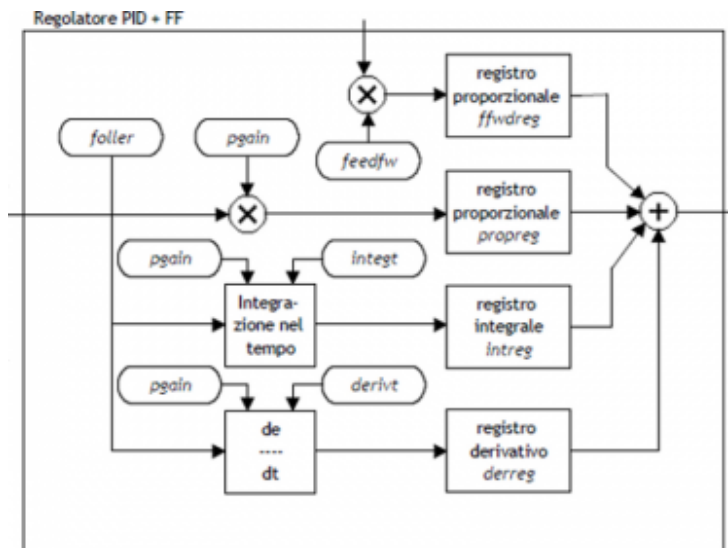
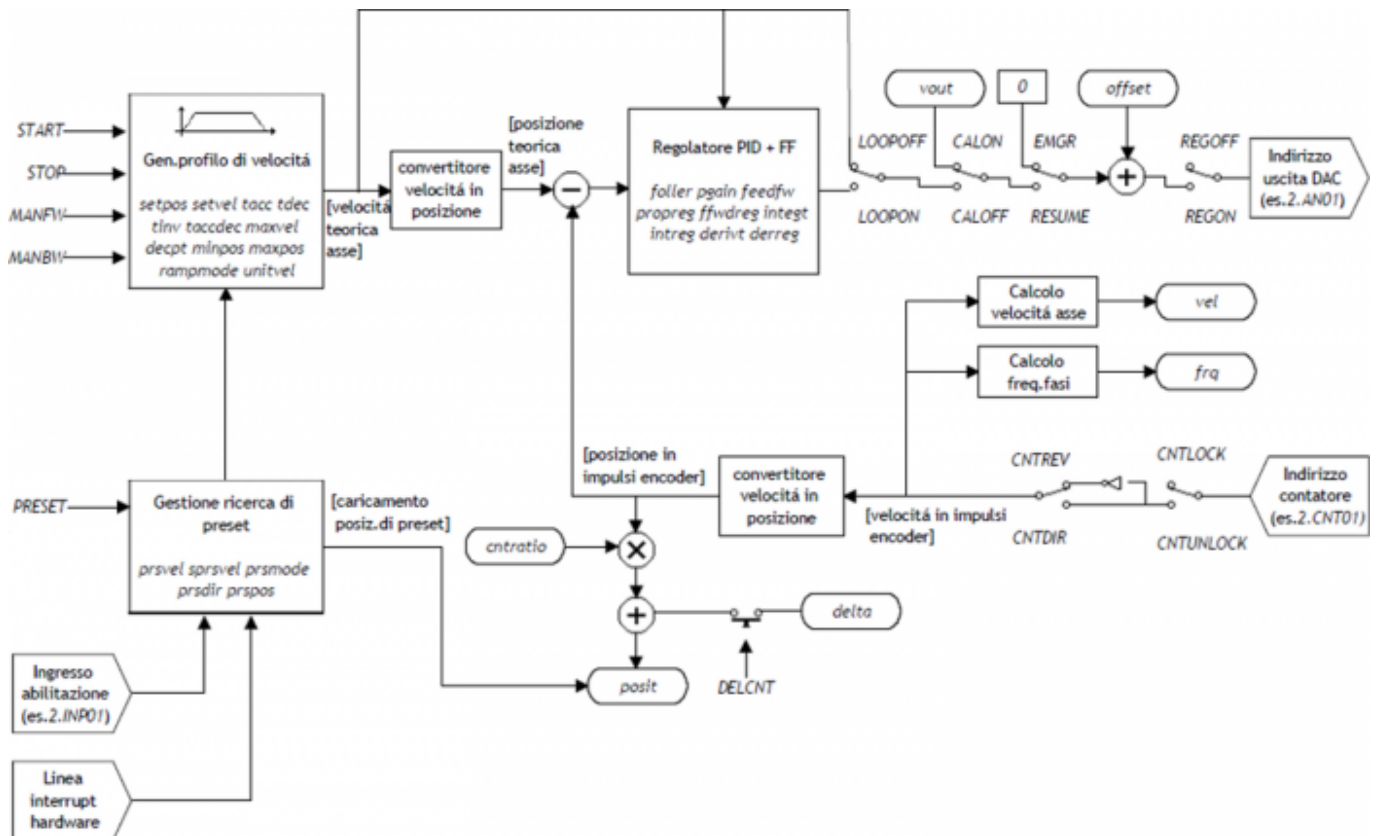
The internal device EANPOS is a tool that resides in the CPU that allows to manage the movement of a mechanical axis with a trapezoidal ramps with analog to trapezoidal or "S" ramp.

The main function of the device is to generate a trajectory or "speed profile" that, suitably converted into a proportional voltage signal, He commands a driving and then the axis attached to it, to bring it to the desired position.

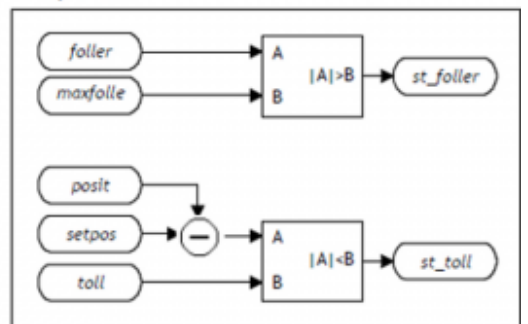
The device also includes a feed-back process on the measurement taken from the transducer. This measure appropriately compared is used to generate the feedback.

The control voltage for operation is generated by a DAC hardware (Digital to Analogic Converter) present on the BUS QMOVE. To sync the device with the actual axis position returned by, is there a special procedure for handling called "searching for presets".

The feed-back is of PID+FF type (proportional, integral, derivative and feed-forward).



Comparazioni



1.1 Installation

1.1.1 Device declaration in the configuration file (.CNF)

In the configuration file (.CNF), the BUS section must be declared so that you have the hardware resources required for the implementation of the EANPOS device. There must be at least a bi-directional counter and a 16-bit resolution analog output. The device can also be used with an input and an interrupt line for preset search functions.

In the INTDEVICE section of the .CNF file must be add the following definition:

```
-----
; Internal device declaration
-----
INTDEVICE
<device_name> EANPOS TCamp ICont IntL IAZero IoutA
```

where:

<device name>	The name assigned to the device
EANPOS	Keyword that identifies the analog positioner device
TCamp	Sample time device (1÷255 ms)
ICont	Bidirectional counter input (to prevent the device uses this resource enter the characters X.X)
IntL	Number of the interrupt line dedicated to the encoder zero pulse during the research phase of presets. Allowed values: 1÷8 (to prevent the device uses this resource, enter the character X)
IAZero	Enable input for transducer zero pulse acquisition during the research phase of presets (to prevent the device uses this resource enter the characters X.X)
IOutA	Hardware address of the DAC from analog output (to prevent the device uses this resource enter the characters X.X)

N.B. It is necessary that each definition are present on the same line. In case you do not want to assign a resource, for example IAZero, you must enter in the appropriate field the string X.X.

1.1.2 Example

```
-----
; Internal device declaration
-----
INTDEVICE
Axis_X EANPOS 2 2.CNT02 3 3.INP01 3.AN01
```

1.2 Operation

1.2.1 Position detection

The EANPOS device cquires the position of via bi-directional transmitter signals; these signals are used by an internal counter. This counter value does not express the position of the axis directly measured in the units that are required by your application. The device does not have within it the cnratio parameter, but let the installer can work with unfinished encoder resolutions by setting the data as space covered in a round encoder (measure) and number of pulses round of encoder (pulse). The relationship between the measure and pulse is the encoder resolution and must have values between 1 and 0.000935.

1.2.2 Definitions

- 1) The parameter measure is placed in units without decimal points (for example 100.0 mm is inserted 1000 tenths of a millimeter).
- 2) The pulse parameter is put into bit encoder x 4 (for example if I connected a 1024 pulses round encoder, is inserted 4096, If the measure is calculated on a round encoder).

1.2.3 Decimal point

If for the selected unit of measure is also provided for the presence of a decimal point, the locations must be represented as an integer and represent space in units without decimal point. The resolution must be calculated with the same formula and measure represent the size without decimal point. The decimal point will then be inserted into the representation of the value-time viewers (example as properties in the terminal operator).

1.2.4 Speed

The speeds are always expressed in whole units of measure in the unit of choice time. From this it emerges that the device must know the location of the decimal point of the unit of measure and this is done with the decpt parameter. This parameter can take values 0÷3.

N.B. To activate this device send the INIT command.

1.2.4.1 Example:

You have to control a rotating table that have the accuracy of 0,1° having an encoder from 1024 pulse for rown directly; you will set the following values:

Axis:measure = 3600

Axis:pulse = 4096

Axis:decpt = 1

To execute the positioning at 4.6 degrees you need to enter:

Axis:setpos = 146

To set a speed of 10 degrees degrees per second, you must type:

Axis:unitvel = 1

Axis:setvel = 10

1.3 Commands and parameters

1.3.1 Symbols used

The name of parameter, state or command are shown to the left of the table.

R

Indicates if its parameter os state is retentive (upon initialization of the device maintains the previously defined), or the state assumes upon initialization of the device.

If the device does not need to initialize the field "R" indicates the value that the parameter or state take to the power up of the card.

R = Retentive

0 = Upon initialization of the device the value is forced to zero.

1 = Upon initialization of the device the value is forced to one.

- = Upon initialization of the device is presented significant value.

D

Indicates the size of the parameter.

F = Flag

B = Byte

W = Word

L = Long

S = Single Float

1.3.1.1 Conditions

Describes all the conditions necessary so that the parameter is considered correct or because the command is accepted.

In some cases, limit values are specified for the acceptance of the parameter: if are introduced the value outside the limits set, the data is however accepted; therefore appropriate controls of the application must be provided to ensure the proper functioning.

To run a command, all conditions must be met; otherwise the command does not run.

A

Indicates the access mode.

R = Read.

W = Write.

RW = Read / Write.

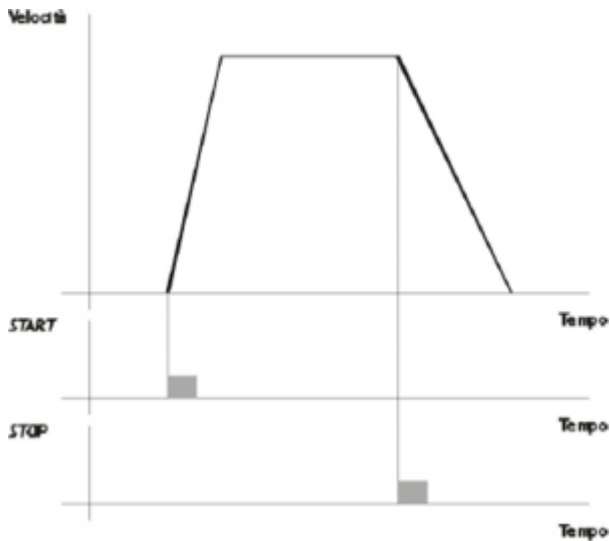
1.3.2 Main controls

1.3.2.1 START

At the START command, the axis is positioned to the declared quota in the setpos variable with speed set in setvel; the placement will run using the acceleration ramp set in tacc parameter and the deceleration ramp in tdec parameter. The type of used ramp (trapezoidal or epicycloidal) is inserted in the ramptype parameter.

1.3.2.2 STOP

If during placement is necessary to stop the axis with a deceleration ramp, it will simply give the command to STOP and the axis decelerates to a stop with the ramp sets in the tdec parameter.



1.3.2.3 EMRG

This command puts the axis in emergency conditions; the st_emrg state is set to one. If the emergency command is sent to the axis during a placement, the movement is interrupted without deceleration ramp, the analog output will be set to zero volts and you dropped the reaction of space. With st_emrg = 1 (emergency condition), you cannot move the axis.

1.3.2.4 RESUME

This command will reset the emergency condition; the axis comes in reaction to space and waits for a command to be able to move (does not automatically resume interrupted positioning).

1.3.2.5 LOPOFF

The LOPOFF command removes the reaction of space without stopping the axis. With st_loopon = 0 the axis handling axis commands accepted, but all placements will be performed without reaction of space. A placement made reaction without loop is comparable to a positioning run without proportional gain (is not guaranteed to arrive in position).

1.3.2.6 LOOPON

The LOOPON command closes the space feedback without stopping the axis. With st_loopon = 1 the axis is moved using all the features of the P.I.D. control.

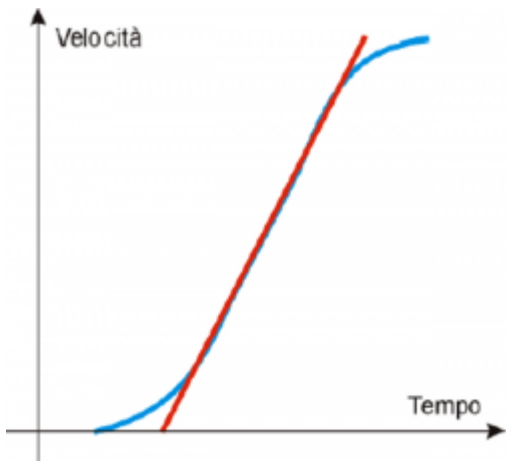
Following is a table summarizing the conditions necessary for the axis in reaction to space and to perform placements.

Loopon	Emrg	Space reaction	Possibility of movement
Yes	No	Yes	Yes
Yes	Yes	No	No
No	No	No	Yes
No	Yes	No	No

1.3.3 Description of the epicycloidal motion

The epicycloidal motion (ramptype = 1) is used to move the axes without sudden variations in speed. The time of positioning of the axis with trapezoidal ramps is less than the same axis moved with the epicycloidal ramps, but the mechanical wear is considerably higher.

For comparison shows the difference of the time of acceleration in the two cases with constant maximum gradient acceleration.



The same goes for the deceleration ramp.

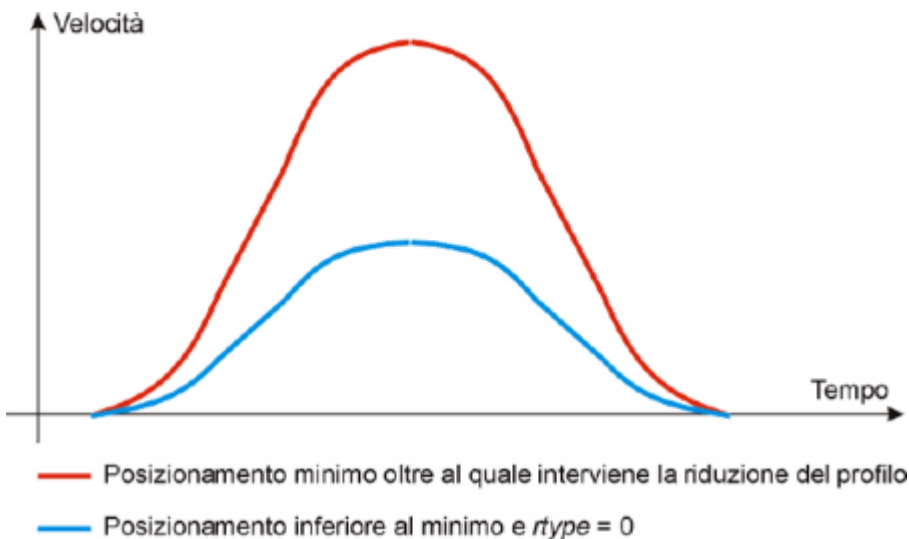
The epicycloidal movement has the ability to behave in different ways in the event of a reduction in profile (rtype) and in the case of stop during acceleration (stopt).

1.3.4 Profile reduction

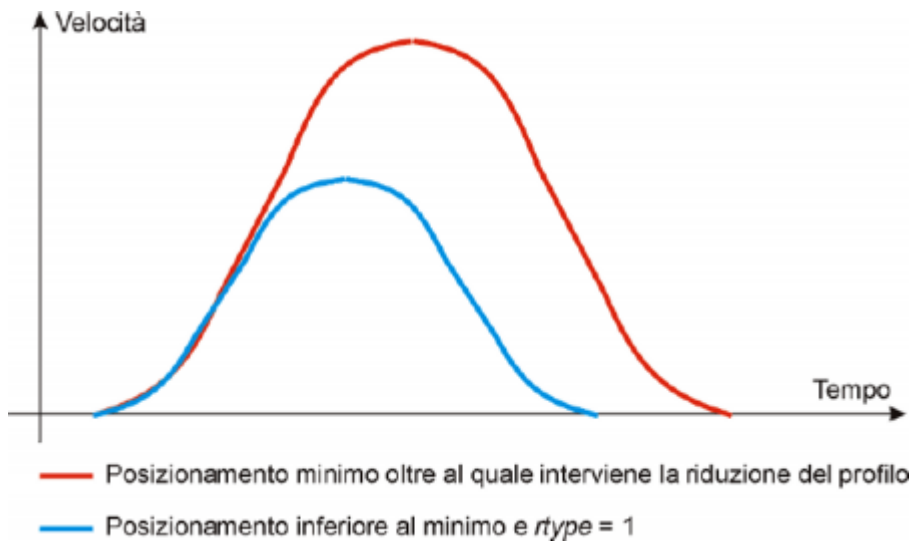
In the event that $\text{ramptype} = 1$. In case the space ahead is less than that which allows to reach the speed set by the acceleration and deceleration ramps, you pass in the phase called "reduction of profile".

You can remain fixed the ramp time, decreasing gradients of the ramps and the speed in proportion (rtype parameter set to 0).

The profile reduction is used only if you are doing a placement.



You can also decrease the time of the ramps while maintaining the gradient of constant acceleration and decrease speed in proportion (rtype parameter set to 1).



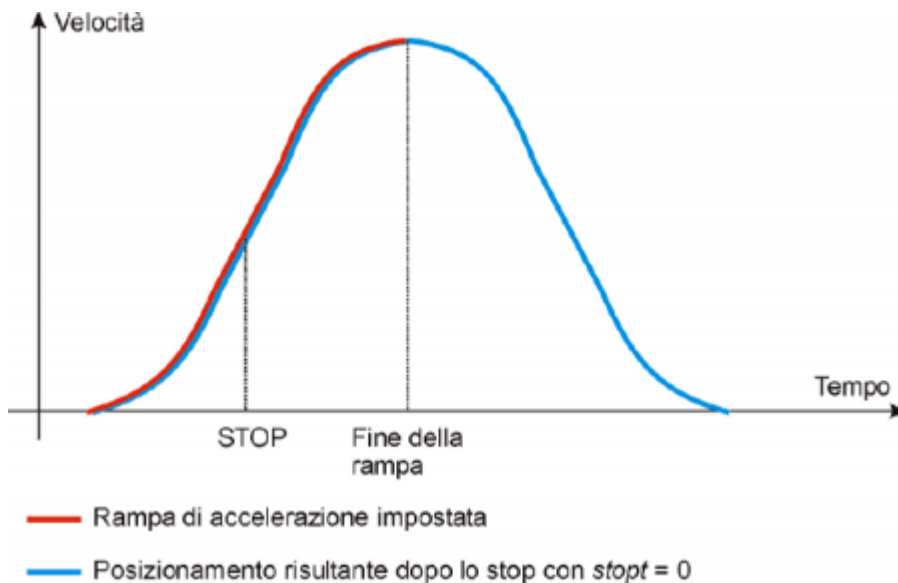
With $rtype$ parameter set to 0 extend considerably the time needed to placements with loss of productivity of the machine, instead of setting it to 1 in case of short placements, shorter times, but keeping the constant gradient you lose the beneficial effect of epicycloidal baseline.

1.3.5 Type of stop during the acceleration ramp

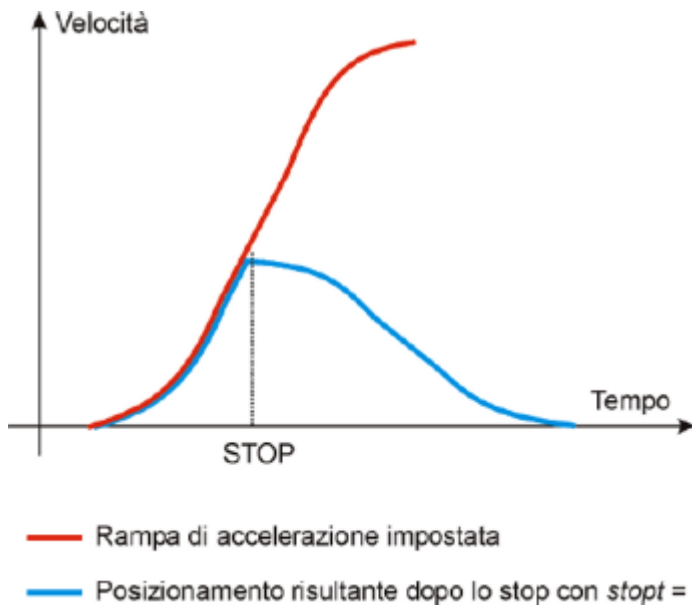
In the event that $ramptype = 1$ and we should curb the axis during acceleration with the STOP command, You must choose whether to complete the ramp, or if you want to abort the ramp and consequently change the epicycloid.

In case you set the $stop$ parameter to 0 is first completed the acceleration ramp and then performed the deceleration ramp.

The type of stop during the ramps is only used if you are doing a placement.



In case you set the $stop$ parameter to 1 stops the acceleration ramp and started immediately the deceleration ramp set.



You immediately notice that there is a substantial difference between the setting of $stop$ to 0 or to 1. To make the choice of what type of stop use, it should be noted that in case of emergency exists the emergency command that instantly locks and without ramp positioning.

1.3.6 Analogue output calibration



Before starting the placements you must make sure that electrical connections and mechanical appliances do not cause malfunctions.

For axis management, the EANPOS device using an analog output with range ± 10 V and resolution 16 bit with sign; with this calibration function with analog output can be driven with a constant value in order to test links and functionality.

1.3.6.1 Preliminary motion

- Remove the emergency condition with the RESUME command.
- The state $st_emrg = 0$
- Enable calibration axis status with the CALON command; the st_cal state must therefore be 1.
- You can now set the analog voltage $vout$ parameter; the value is expressed in tenths of a volt ($-100 \div 100 = 10 \div 10$ V). It is recommended to introduce low values (5, 10, 15 ... equal to 0.5, 1, 1.5 V).
- When the axis is in movement the frq parameter indicates the frequency in Hz of the transducer phases.
- The $posit$ parameter that show the position, varies indicating the space taken by axis. If setting a positive voltage will count decrements, it is necessary to invert the phases of the transducer or reverse the direction in driving.
- You can reverse the direction of the count using the CNTREV command.
- If output voltage equal to zero the axis is not stationary, adjust the offset parameter to correct the tension until movement does not stop. The introduced value (each bit corresponds to approximately 0.3 mV), will be added algebraically to the value of the analogue output; This operation allows to compensate for any drift of electronic component, in out of QMOVE and in the input of the drive. The value is expressed in bits with sign.
For an optimal result of calibration this operation must be performed with the system to temperature regime.
- To disable calibration status send the CALOFF command.
- The state $st_cal = 0$

1.3.6.2 Output settings

The EANPOS device raises the voltage value of the analogue output on the basis of a ratio between the maximum speed of the axis and the maximum output voltage. Proportionality is obtained with the $maxvel$ parameter, that represents the speed of the

axis on the analog voltage (10 V). The axle must behave symmetrical behavior respect to zero value of analog output, therefore the speed must be the same on both the positive and negative voltage at maximum.

Before you determine the value of the maximum speed, we must establish the unit of time to use for the representation of the speed in the device; the unitvel parameter defines the unit of time of the speed (Um/min or Um/s).

1.3.6.3 Theoretical method for the determination of the maximum speed.

The theoretical method is a calculation that was performed on the basis of the maximum speed of the motor. Once established the maximum revolutions per minute declared on the motor, we get the maximum speed.

Introduce in the maxvel parameter the value of maximum speed calculated.

1.3.6.4 Practical method for determining the maximum speed.

The practical approach is based on the reading of the speed detected by the device in the vel variable, giving the drive a known voltage. To provide the voltage to drive the device should be placed in a position of calibration as described in the previous paragraph. If the system permits, supply the 10V voltage to the drive and read the speed value in the vel parameter. If, on the other hand, is an excerpt of the output voltage (1, 2, ... 5 V), calculate the maximum speed with the proportion: $v_{out} : 10 [V] = vel : maxvel$

Enter the maximum speed value found in maxvel parameter.

1.3.7 Movement



Attention! Before to move the axis, check the correct function of emergency and protection devices.

The procedures described here have allowed us to complete the first phase of device parameterizing. Now you can run a simple movement of the axis.

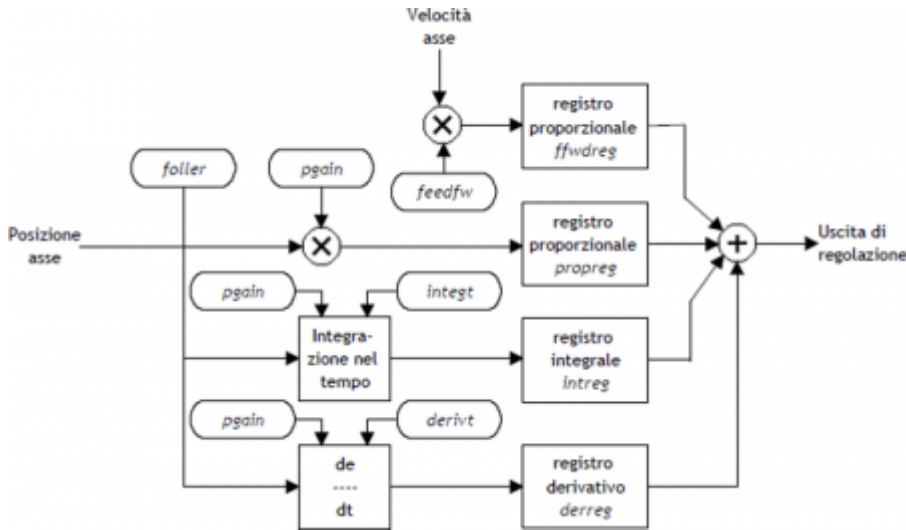
- Move the axis in a position whereby it can fulfill a certain area without touching the maximum quota limit.
- Set the current position to zero axis in parameter posit = 0.
- Set up the parameters that define the position of the software limit switches: minpos = 0 and maxpos to the value of the maximum stroke of the axis.
- Set the parameter that defines the time that the axis to achieve maximum speed taccdec = 100. This parameter is expressed in hundredths of a second (100 = 1 sec.)
- Set the positioning speed with the setvel parameter.
- Set the target quota with the setpos parameter.
- Set the parameter feedforward = 1000 (100%)
- If the device is in emergency state (st_emrg = 1) give the RESUME command.
- Start positioning with the START command. To stop the movement give the EMRG command.

This first movement was done without speed feedback space. The placement may have been executed with some error introduced by the non-linearity of the components or by an imperfection in the calibration of the maximum speed. Subsequently enabling feedback space this error goes away.

1.3.8 PID+FF calibration

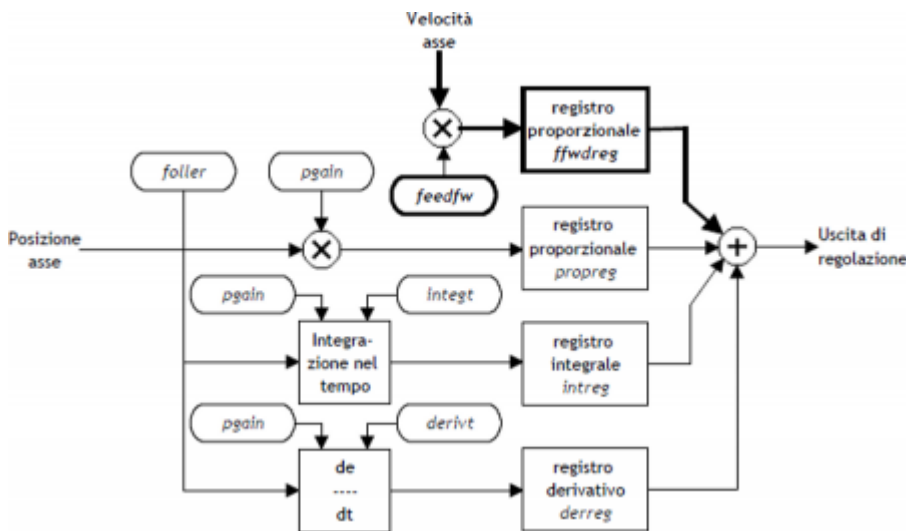
The runs placement in the preceding paragraph has been made without considering any position errors. To check the correct position of the axis continuously and automatically, You must have a feed-back on the position; for this reason introduces the PID+FF control algorithm including proportional, integral, derivative actions and feed-forward; the value of the analog output is given by the summation of the feed-forward, proportional, integrative e derivative actions.

Without going into the merits of a technical description of the theory of regulation, in this section describes a series of actions to adjust the parameters that affect this control. In order to achieve a satisfactory adjustment is sufficient to use only the feedforward and proportional actions; integral and derivative actions are used only for adjustments under special conditions.



1.3.8.1 Feed forward action

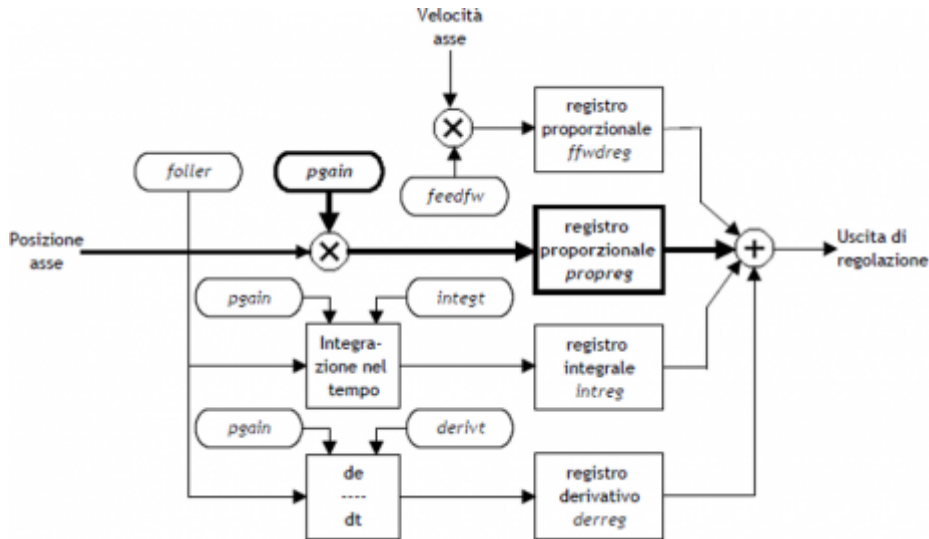
The feed-forward helps make the system more ready on placements, by providing the analog output voltage proportional to the theoretical speed of positioning. In practice it is the component which you performed the placements of the previous chapter. The contribution of this action can be adjusted with the feedfw parameter; this parameter is expressed as a portion of millesimal theoretical speed; so, to introduce such as 98.5% you must set 985 (thousandths).



1.3.8.2 Proportional action

This action provides an proportional output to the instantaneous axis position error. The extent of the proportional action is defined by the **pgain** parameter that defines the sensitivity of the system.

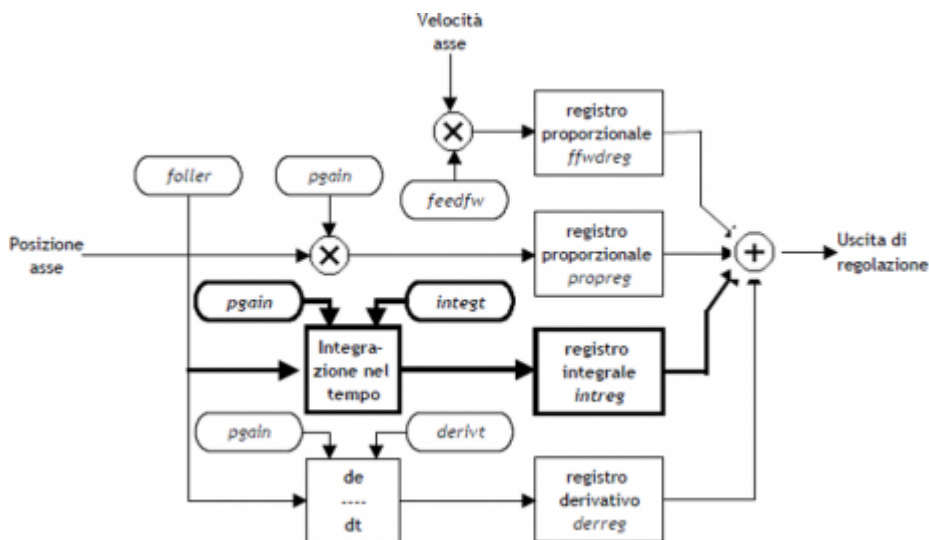
The **pgain** parameter is introduced in thousandths; the unit value of the gain (1000) provides an analog output to maximum value (10 V) concerning the maximum speed error. For maximum speed error means the space taken by axis - at the maximum speed - for the duration of the sampling time of the device.



1.3.8.3 Integral action

Integrates the position error of the system over time set in **integt** parameter updating the release until the error is not canceled.

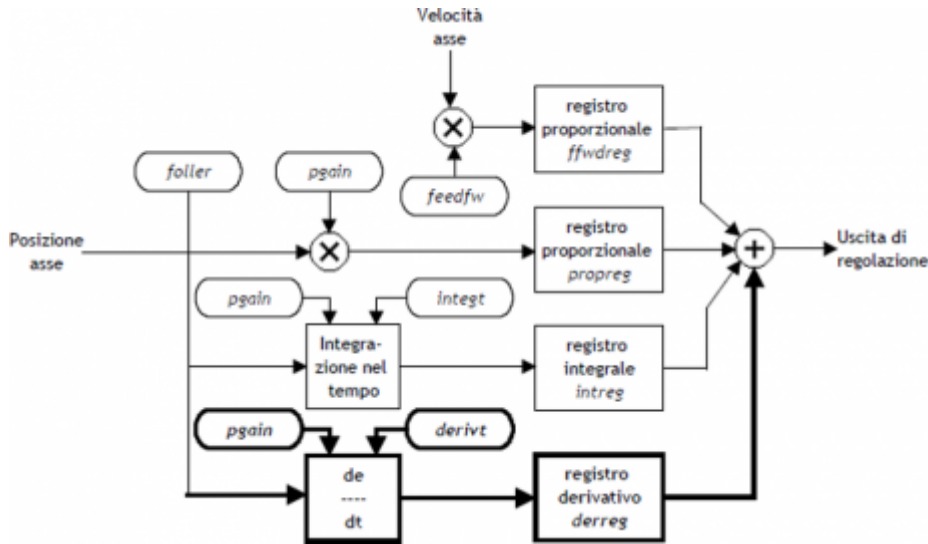
The lower the integration time of the error, the faster the recovery system error, but the system can become unstable, tending to swing.



1.3.8.4 Derivative action

Anticipates the change of the motion of the system tends to eliminate the overshoot of the positioning. The entity of change is calculated over time set in **derivt** parameter.

More higher is the derivation time and more speed is the system in the transient error recovery, but if you enter a value that is too high the system becomes unstable, tending to fluctuate.



1.3.9 Preset search

Searching for presets is a procedure to synchronize the count (posit parameter) with the actual position of the axis. Can be performed in 3 different ways according to the **prsmode** parameter.

1.3.9.1 Preliminary considerations

- In order to perform a search preset is need a input (acquired through the BUS QMOVE) we define “enabling zero transducer”. In its path, the axis must turn this entry into a particular point.
- The preset search procedure is started with the PRESET command; at the initial stage the count acquired from QMOVE can have any value and can be located anywhere.
- The axis must direct towards the cam allows the upload of preset. Not knowing which way to direct the axis to meet the enable input, search for preset lets you choose with the prsdir parameter two solutions:
prmdir = 0 the axis will always a forward movement
prmdir = 1 the axis will always a back movement
- If during the preset search execution is give again the PRESET command, the direction of motion of the axis is reversed.
- It is to be noted that the presets quota contained in the **prspos** parameter must always be between the minimum and the maximum quota and that, after you load the preset quota, the axis is automatically positioned to the preset.
- During the preset search the software limits **maxpos** and **minpos** are not enable.
- When you set the search speed zero pulse - **sprsvel** - we must consider that in Mode 0 the input has a hardware filter that delays the acquisition and so influence the precision of the loading. In the 1 mode, the acquisition is execute on interrupt input so the speed is not determinative. We must ensure, however, that the time of zero impulse duration is sufficient to be acquired from the card. In order to know the acquisition time of the entrance and the minimum time of zero pulse signal refer to the technical documentation of the used cards.

Once you start search presets with the PRESET command, the axis towards the position where the zero pulse enable input transducer with speed defined by **prsvel**. The **st_prson** state you turn to report the current preset. When, in its rush, the active axis this input, reverses direction and starts with speed **sprsvel**.

At this point the behavior depends on the manner of loading selected (Mode 0 or Mode 1).

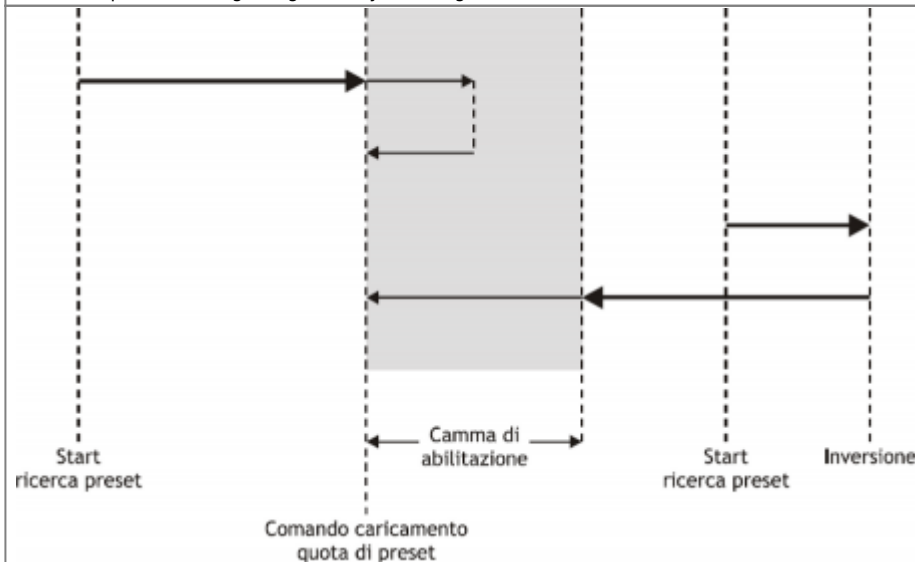
1.3.9.2 Preset search - Mode 0

This procedure involves the movement of the axis and the use of the zero pulse enable input transducer.

To the deactivation of the zero-pulse enable trasducer input, the preset quota is loaded on the counting (parameter posit=prspos).

Next, you disable the **st_prson** status and active the **st_prsok** state to signal the end of search. This State remains active until the starting of a new presets procedure; when the system is powered this State is always zero. See picture 1.

Picture 1: preset loading using the only enabling cam.



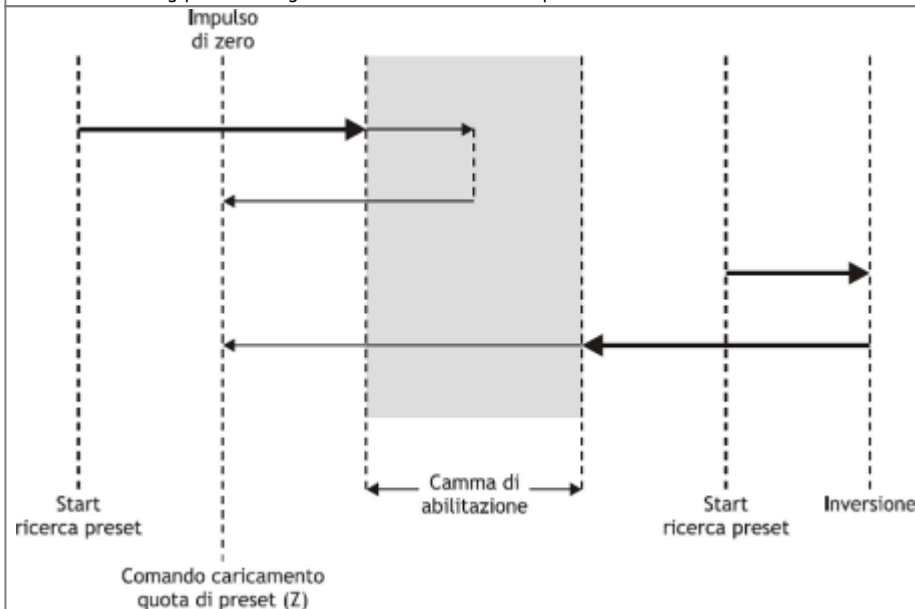
1.3.9.3 Preset search - Mode 1

This procedure involves the movement of axis and using the zero pulse inputs and enabling zero-pulse transducer.

To the deactivation of the zero-pulse enable transducer input, enables reading of the first zero-pulse provided by the transducer and, on this signal, it loads the preset quota in counting (parameter posit = prspos). To allow the load of the preset quota with "Mode 1", the card which is attached to the zero-pulse translator must transmit this signal on one of the 8 hardware interrupt lines. Moreover, the device must have been defined to use the same hardware interrupt line (See definition in the file .CNF).

Subsequently, the axis place at the preset quota, it disable the **st_prson** status and active the **st_prsok** state to signal the end of search; this State remains active until the start of a new process of presets. At the power on of the system **st_prsok** is always to zero. See picture 2.

Picture 2: loading preset using the enable cam and zero-pulse transducer.



1.3.9.4 Search preset - Mode 2

This procedure does NOT provide for the handling of the axis and the use of the zero-pulse transducer enable input.

With this procedure, the preset search does not execute any positioning. The preset quota load command is supply is supply from the activation of the enable input zero encoder and the **st_prsok** state is set to one. If the input is a continuous load remains activated while, if at power up the input is already active, the first upload is executed only after its deactivation and activation.

1.4 Special functions

1.4.1 Multi axis management

In some applications you are prompted to place a large number of axes by moving a single axis at a time. The design choice falls on installing just one drive that depending on the axis to be placed electrically connects the motor interested in positioning; the transducer is always bound to its axis.

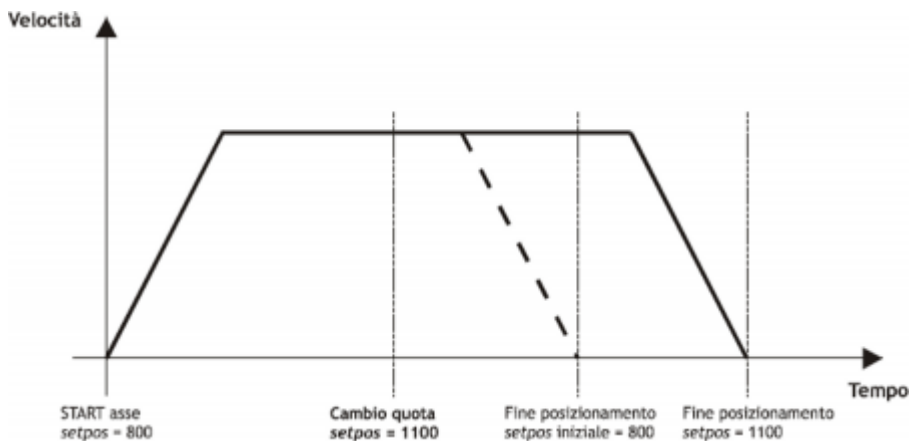
The EANPOS device allows you to control access to the **DAC** device using the **REGON** and **REGOFF** commands. In this way through the QCL application defines a set of device many axes to be placed; in the definition, all devices use the same **DAC** resource (IOutA).

Normally all devices must be in the $st_regoff = 1$ state so you will not have access to the DAC device. Before you start positioning, with the **REGON** command, the device is brought to the $st_regoff = 0$ state. Runs the placement and completion, the device is returned in the state of $st_regoff = 1$ with the **REGOFF** command. For no reason two devices simultaneously must be found in the $st_regoff = 0$ state. When the device is placed in the position of **regoff**, the analog output remains fixed at the last voltage value called before the **REGOFF** command.

1.4.2 Change quota and counting in motion

In some applications you are prompted to define the target quota during placement, according to external events to the device. This means you can also write to the **setpos** parameter with placements occurring.

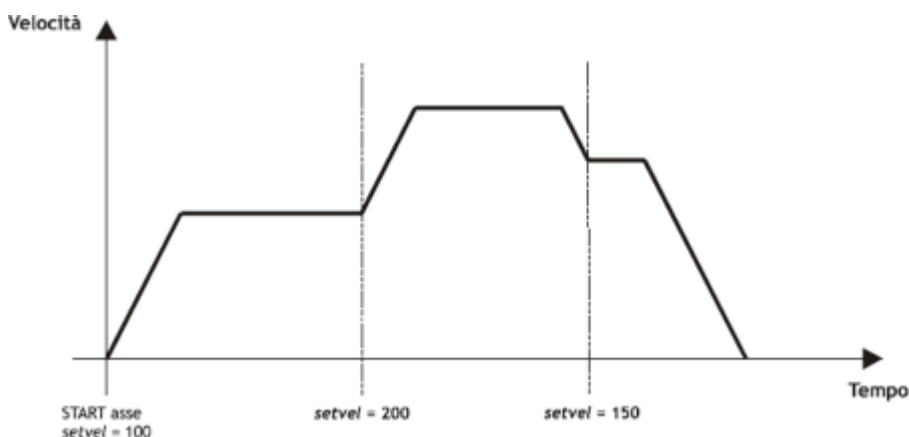
Changing the quota is accepted only if the new location is accessible by the management. When using the epicycloidal ramps the change quota is not accepted if $st_dec = 1$



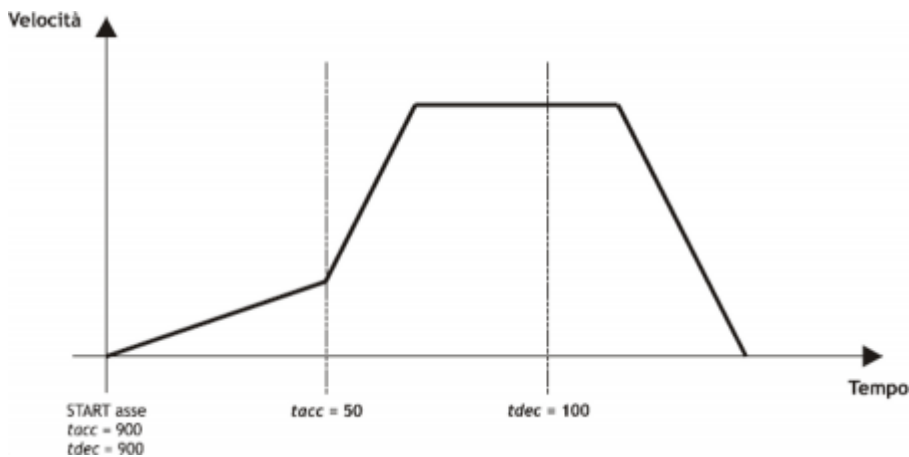
When positioning you can also change the value of the posit count. This function is usually used when a device must, under special conditions, continue a speed profile for a very long time, that exceeds the time axis takes to reach the limit quota (**maxpos** or **minpos**).

1.4.3 Change speed and moving ramp time

When positioning it is possible to vary the speed of the axis without affecting the location to get to. This operation can lead an increase or a decrease of the speed, even more points with the same placement. This is accomplished with new writing in the **setvel** parameter. The speed change is always available except during the deceleration ramp indicated by the state ($st_dec = 1$)



During the placement can be varied even acceleration/deceleration times. For example, the device can start a placement with a long ramp, once you reach the target speed, has varied the **tacc** parameter and run a speed change with a more long ramp. For special applications and trapezoidal ramps, ramp time can be varied even during a speed change, in this case the new time is put into execution immediately.



1.4.4 Delta count

The changing to the value of resolution (**cntratio**) or the writing of the **posit** variable, causes the reset of the remnants of the conversion.

The device always shows the absolute position of the axis; for execute the incremental positioning you must have a tool to subtract or add some value to the count (**posit**) without introducing errors. The count change can also execute with a direct writing to the new value in the **posit** parameter.

Wanting to steal 100 units of measure from the count you can:

Axis:posit = Axis:posit - 100

This introduces an error because it imposes the location "posit = -100", when the axis could have an intermediate position between a unit of measurement and subsequent (example 100.3). This fraction (0.3) is lost and the repetition of these steps will the accumulation of a considerable error.

The **DELCNT** command sum the count an amount equal to the parameter delta without losing the fraction part of the position:

Axis:delta = -100 DELCNT Axis

The **DELCNT** command may be sent only with stopped axis (st_still=1)

1.4.4.1 Example

The reading the position of the axis is 2 units and it is located at point A. You want to add to the count posit three units.

With the instructions:

Axis:posit = Axis:posit + 2

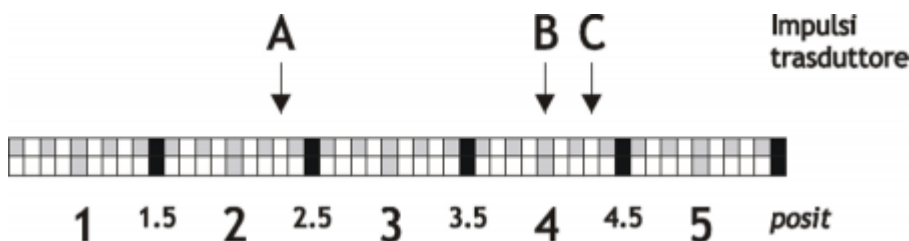
the axis takes the new position B.

With the instructions:

Axis:delta = 2

DELCNT Axis

C position is reached.



Note that with the first instructions the axis took a delta less than 3 units and then introduced an error.

If you need to send DELCNT commands in succession, It is convenient to calculate the magnitude to be summed and send once the command; otherwise beware not to send subsequent commands without a read statement on device parameter.

1.4.4.2 Example

Axis:delta = 3

DELCNT Axis

WAIT Axis:st_init
Axis:delta = 40
DELCNT Axis

1.5 Device error management

The error presence in the device is reported by the **st_error** state.

When **st_error** is equal to 1, are present on the **errcode** variable the type of error occurred (see the table) and in the **errvalue** variable an indication on the cause of the error.

If the device goes in error, to resume the working you must delete the **st_error** state through the RSERR command.

1.6 Device warning management

The presence of a warning system camming is signaled by the **st_warning** state.

Being caused by a minor event and being guaranteed in this situation, the management of the axis, the axis continues his work.

When **st_warning** is equal to 1, are present on the **wrncode** variable the type of the **warning** intervened (see the table) and in the **wrnvalue** variable an indication on the cause that resulted **warning**.

Code	Priority	Description
1	0	Command not executed

To clear the **st_warning** status you must send the **RSWRN** command.

1.7 Commands and parameters table

1.7.1 Symbols used

The name of parameter, state or command are shown at the left of the table.

R

Indicates if its parameter or state is retentive (upon initialization of the device maintains the previously defined), or the state assumes upon initialization of the device.

If the device does not need to initialize the "R" field indicates the value that the parameter or state take to the power up of the card.

R = Retentivce

0 = Upon initialization of the device the value is forced to zero.

1 = Upon initialization of the device the value is forced to one.

- = Upon initialization of the device is presented significant value.

D

Indicates the size of the parameter.

F = Flag

B = Byte

W = Word

L = Long

S = Single Float

1.7.1.1 Conditions

Describes all the conditions necessary so that the parameter is considered correct or because the command is accepted.

In some cases, limit values are specified for the acceptance of the parameter: if are introduced any values outside the limits set, the data is however accepted; therefore must be provided the controls to ensure proper operation.

To run a command, all conditions must be met; otherwise the command does not execute.

A

Indicates the access mode.

R = Read.

W = Write.

RW = Read / Write.

1.7.2 Commands

The available commands to manage the device are listed below in descending order of priority. The device executes all

commands received within the same sampling time starting from the one with the highest priority. For example if the device receives the same sampling time **CNTUNLOCK** and **CNTLOCK** commands, first run the **CNTLOCK** command and then to **CNTUNLOCK** leaving therefore the counter free to count.

Name	Conditions	Description
INIT	st_init = 0	Initialization Initializing command device. If the device is not initialized does not perform the calculations related to the axis and then sits idle. To power on you can download all parameters in memory; Subsequently, with the INIT command the axis will be initialized, by performing the calculations once. Activates the st_init status.
EMRG	st_init = 1	Emergency Put the axis in emergency mode, without deceleration mode, any ongoing positioning. It is also disabled the reaction of the axis space.
RESUME	st_init = 1 st_emrg = 1	Resume Restore of emergency condition of the axis; space reaction is reenabled if disabled. To the start acquisition, the axis resumes the placement.
START	st_init = 1 st_rego = 0 st_emrg = 0 st_still = 1	Start Controls the positioning at the setpos quota and setvel speed.
STOP	st_init = 1 st_rego = 0 st_emrg = 0 st_still = 0	Stop Stops any ongoing axis positioning. The axis stop, occur follows the deceleration ramp in use. The axis remains in reaction to space.
LOOPON	st_init = 1	Loop ON Enables the reaction of space axis. The analog output sharpen every external action that attempts to move the axis from the reached position (derive, user, ...). This operation resets any following error follerr.
LOOPOFF	st_init = 1	Loop OFF Disable the reaction of space axis. The axle can be moved from its position that the analogue output contrasts the movement.
CNTLOCK	st_init = 1	Counter lock Blocks the acquisition of axis count even if the transducer continues to send signals. At this stage the possible displacement of the axis is not detected. Disables loading of presets on share count. Activates the st_cntlock status
CNTUNLOCK	st_init = 1	Counter unlock Unlock the axis count. Resumes reading of the signals sent from the transducer and the updating of the counter. Disable the st_cntlock status.
CNTREV	st_init = 1	Counter reverse Invert the phases of the transducer inside the card. Is then reversed the direction of the count (increase/decrease). Activates the st_cntrev status.
CNTDIR	st_init = 1	Counter direct Counting the axis is not reversed. Disable the st_cntrev status.
PRESET	st_init = 1 st_rego = 0 st_emrg = 0 st_cal = 0	Preset Start search presets axis. Preset search procedure is run in the modality set by the prsmode and prsdir parameters. If the preset search is already running, the command performs the reverse search. Activates the st_prson state and disable the st_prsok status.
CALON	st_init = 1	Calibration ON The analog output is used as a voltage source; in this case you cannot use it for positioning the axis. The output value is settable at will through the vout variable.
CALOFF	st_init = 1 st_cal = 1	Calibration OFF The analog output is not management as voltage generator, so it can be used for managing placements.
MANFW	st_init = 1 st_rego = 0 st_prson = 0 st_still = 1 st_cal = 0	Manual forward Controls the manual forward positioning (towards maxpos) to the set speed. If the count is greater of maxpos, the axis will not move
MANBW	st_init = 1 st_rego = 0 st_prson = 0 st_still = 1 st_cal = 0	Manual backward Controls the manual positioning in back (towards minpos) to the set speed. If the count is less of minpos, the axis will not move
REGOFF	st_init = 1 st_still = 1	Regulation OFF Turn the adjustment and upgrading the DAC, and all the movement commands
REGON	st_init = 1 st_rego = 1	Regulation ON Re-instate the adjustment and updating the DAC, and all the movement commands.

DELCNT	st_init = 1 st_cal = 0 st_prson = 0	Delta counter The count (axis position) is modified by adding algebraically the value specified in the delta variable. This command can be given always, unless with the moving axis command would reverse the direction. In the case of using epicycloidal deceleration ramps you cannot give the command if st_dec = 1	
RSERR	-	Reset error Reset the st_error status.	
RSWRN	-	Reset warning Reset the st_warning status	
RSPRSOK	-	Reset preset OK Reset the st_prsok status	

1.7.3 Parameters

Name	D	R	A	Conditions	Description
measure	L	R	R-W	-	Measure (1÷999999) Indicates the space, in units of measure, cover from the axis to get the encoder pulses * 4 set in the pulse parameter. This parameter is used to calculate the axis resolution with the formula: Resolution = measure* 4 / pulse The resolution must have a value between 0.00374 and 4.00000
pulse	L	R	R-W	-	Pulse encoder (1÷999999) Indicates the pulses multiplied * 4 provided by the encoder to get the space set in the measure parameter. This parameter is used to calculate the axis resolution with the formula: Resolution = measure* 4 / pulse The resolution must have a value between 0.00374 and 4.00000 st_camex = 0 st_prson = 0
maxpos	L	R	R-W	st_still = 1	Maximum position (-999999 ÷ 999999) Defines the maximum quota reached by the axis; the value set is to be considered also as an upper limit for the introduction of work quotas. Value expressed in units of measurement (Um).
minpos	L	R	R-W	st_still = 1	Minimum position (-999999 ÷ 999999) Defines the minimum quota reached by the axis; the set value is to be considered also as the lower limit for the introduction of work quotas. Value expressed in units of measurement (Um).
prspos	L	R	R-W	st_still = 1 st_prson = 0	Preset position (minpos÷maxpos) Defines the value that is loaded on the counting with the preset search procedure.
maxvel	L	R	R-W	st_still = 1	Maximum velocity (0 ÷ 999999) Defines the maximum axis speed (related analogue reference of +/-10V). The input value is per unit of time of the set speed unit Velocity. Value expressed in Um/s o Um/min.
prsvcl	L	R	R-W	st_still = 1 st_prson = 0	Preset velocity (0 ÷ maxvel) Defines the speed of the axis during the search procedure of presets. The input value is per unit of time of the set speed (unit Velocity).
sprsvcl	L	R	R-W	st_still = 1 st_prson = 0	Preset search velocity In the preset search procedure, defines the speed of the axis in the acquisition phase of zero-pulse. The input value is per unit of time of the set speed (unit Velocity).
toll	L	R	R-W	-	Tolerance (-999999 ÷ 999999) Defines a count range around the positioning quotas. If the placement ends within this range, is considered correct. Value expressed in units of measurement (Um).
maxfollerr	L	R	R-W	-	Maximum following error (0 ÷ 2147483648) Defines the maximum acceptable deviation between the theoretical position and the actual position of the axis. It is used for the management of st_foller. Value, in bits, transducer.
rampmode	B	R	R-W	If st_still = 1 writing always enable, if st_still = 0 the new value is stored but prosecuted only if the new times of acceleration and deceleration allow it possible to reach the quota set.	Acceleration/deceleration time mode Used for choosing the same or differentiated between acceleration and deceleration ramps. Allowed values: 0 = equal ramps, 1 = differentiated ramps.
ramptype	B	R	R-W	st_still = 1	Ramp type Defines the type of ramp run: 0 = Trapezoidal acceleration and deceleration, 1 = Acceleration and deceleration of epicycloidal type, 2 = Trapezoidal acceleration and epicycloidal deceleration type, 3 = Epicycloidal acceleration and trapezoidal deceleration.
taccdec	W	R	R-W	if st_still = 1 writing always enable, if st_still = 0 the new value is stored but prosecuted only if the new time allows you to reach the quota set.	Acceleration/deceleration time (from 0 to 999) Is the time required to switch between 0 speed to maximum speed and vice versa; the parameter is used if rampmode = 0. Value expressed in hundredths of a second.

tacc	W	R	R-W	if st_still = 1 writing always enable, if st_still = 0 the new value is stored but prosecuted only if the new time allows you to reach the quota set.	Acceleration time (da 0 a 999) Defines the time it takes to accelerate from zero-axis (condition of axis stopped) to the maximum speed. Value expressed in hundredths of a second.
tdec	W	R	R-W	if st_still = 1 writing always enable, se st_still = 0 the new value is stored but prosecuted only if the new time allows you to reach the quota set.	Deceleration time (da 0 a 999) Defines the time necessary for the axis to decelerate from maximum speed to zero (condition of axis stopped). Value expressed in hundredths of a second.
rtype	B	R	R-W	ramptype=1	Reduction type Profile reduction type in case the space covere let not fully execute the acceleration and deceleration ramps. 0= Time fixed ramps with reduction of slope, 1 = Fixed slope with reduced time ramps.
Stopt	B	R	R-W	ramptype=1	Stop type Type of stop during acceleration 0 = The acceleration ramp is completed, and then it starts the deceleration ramp, 1 = The acceleration ramp is stopped stops and start the epicicloidal deceleration ramp.
tinv	W	R	R-W	-	Direction inversion delay (from 0 to 999) Is used to avoid mechanical stress caused by too rapid direction changes movement. Value expressed in hundredths of a second.
toldly	W	R	R-W	-	Tolerance delay (da 0 a 999) Defines the time between the arrival of the axis in the tolerance range and its status report. Value expressed in hundredths of a second.
pgain	W	R	R-W	-	Proportional gain (from 0 to 32767) By setting the value 1000, the coefficient is 1.000. Is the factor that is multiplied against the tracking error produces the proportional portion of the control.
feedfw	W	R	R-W	-	Feed forward (from 0 to 32767) By setting the value 1000, the percentage is 100%. Is the coefficient percentage multiplied by the speed, generates the feed-forward control output.
integt	W	R	R-W	-	Integral time (from 0 to 32767) Is the time, in milliseconds, that produces the coefficient of integrating the following error. The integration of that error multiplied by that coefficient raises the integral part of the control output.
derivt	W	R	R-W	-	Derivative time (from 0 to 32767) Is the time, in milliseconds, that produces the derivative of the following error. The derivation of this error multiplied by that coefficient generates the output control derivative part.
prsmode	W	R	R-W	st_prson = 0	Preset mode Definisce il type of preset search: 0 = For research enabling zero-pulse, the axis movement begins in rapid, meet the enabling cam, reverses the direction of slowing and, on the falling edge on the cam signal, load the preset quota. 1 = For research enabling zero-pulse, the axis movement begins in rapid, meet the enabling cam, reverses the direction and slowly acquires the first zero-pulse (after disabling the cam signal), 2 = Does not activate the search procedure presets with axis movement. The count is updated to the presets to the activation of the enabling zero-pulse.
prmdir	B	R	R-W	st_prson = 0	Preset direction Definisce the direction of movement of the cam research axis zero pulse enable. 0 = the axis goes forward, 1 = the axis goes back.
unitvel	B	R	R-W	st_still = 1	Velocityc unit Definisce If the unit of time is expressed in minutes or seconds. 0 = Um/min, 1 = Um/sec.
decpt	B	R	R-W	st_still = 1	Decimal point (0 ÷ 3) Defines the accuracy with which you wish to set the presets and view the counts in relation to axis.
offset	W	R	R-W	-	Offset (-32768 ÷ 32767) Offset DAC output in bits. Defines the value in bits of the correction concerning the analogue output to compensate for any drift in the system. Value expressed in bits DAC.
setpos	L	R	R-W	If st_still = 0, see conditions in paragraph "Special functions - change dimensions and counts in motion"	Setted position (minpos ÷ maxpos) Defines the placement quota reached from axis with the setvel speed. N.B. Il value is altered if you tell the enquiry procedure presets with prsmode = 0 or 1.

setvel	L	R	R-W	If st_still = 0, see conditions in paragraph "Special functions - speed change ramp time in motion"	Setted velocity (0 ÷ maxvel) Is the value of positioning speed. Value expressed in Um/s or Um/min. N.B. If value is altered if you tell the enquiry procedure presets with prsmode = 0 or 1.
vout	B	0	R-W	st_cal = 1	Volt out (-100 ÷ 100) Is the value of the output voltage, measured in tenths of a volt, to be used in the calibration procedure.
vel	L	0	R-W	-	Velocity Is the value of the instantaneous speed of the axis. The update is performed every 250 milliseconds. Value expressed in Um/s or Um/min.
frq	L	0	R-W	-	Frequency Is the value of the instantaneous frequency of the axis. The update is performed every 250 milliseconds. Value expressed in Hz.
posit	L	R	R-W	If st_still = 0, see conditions indicated in dedicated paragraph	Actual position Is the value of the instantaneous position of the axis. Value expressed in units of measurement (Um).
encoder	L	R	R	-	Encoder Is the value of the instantaneous position of the axis. Value expressed in bits encoder * 4.
follerr	L	0	R	-	Following error Is the instantaneous value of the tracking error. Value expressed transducer in bits.
ffwdreg	L	0	R	-	Feed-forward register Is the instantaneous value of the register of feed-forward, expressed in bits.
propreg	L	0	R	-	Proportional register Is the instantaneous value of the proportional register expressed in bits.
integreg	L	0	R	-	Integral register Is the instantaneous value of the integral register expressed in bits.
derreg	L	0	R	-	Derivative register \\Is the instantaneous value of the derivative register expressed in bits
delta	L	0	R-W	-	Delta counter (-999999 ÷ 999999) Is the value that is added to the count when is sent a DELCNT command. Value expressed in units of measurement (Um).
errcode	B	0	R	-	Error code (0 ÷ 100) Indicates the type of error in the system. The code is valid only if st_error = 1 (See dedicated chapter).
errvalue	B	0	R	-	Error value (0 ÷ 100) Indicates the cause of the error in the system. The code is valid only if st_error = 1 (See dedicated chapter)
wrncode	B	0	R	-	Warning code (0 ÷ 100) Indicates the type of warning in the system. The code is valid only if st_warning = 1 (See dedicated chapter)
wrnvalue	B	0	R	-	Warning value (0 ÷ 100) Indicates the cause of the warning in the system. The code is valid only if st_warning = 1 (See dedicated chapter)

1.7.4 State

Name	D	R	A	Conditions	Description
st_init	F	0	R	-	Initialization Reporting of device initialized. 0 = device is not initialized, 1 = device initialized. Value of zero is loaded by default at power up.
st_emrg	F	0	R	-	Emergency Emergency signaling axis. 0 = axis not in emergency, 1 = axis in emergency.
st_toll	F	0	R	-	Tolerance Signaling axis in tolerance in relation to the quota put in execution from the START command. Shaft tolerance reporting can be delayed with toldly parameter. 0 = axis not in tolerance, 1 = axis in tolerance. Value of zero is loaded by default at power up.
st_prsok	F	0	R	-	Preset Ok Preset search reporting was successfully executed. 0 = presets don't execute or not concluded correctly, 1 = preset executed and concluded correctly. Value of zero is loaded by default at power up.

st_still	F	0	R	-	Still Signalling of stationary axis. 0 = axis in moving, 1 = axis stopped. To power up by default loads the value to 1.
st_prson	F	0	R	-	Preset ON Reporting of research ongoing axis preset. 0 = preset search not in progress, 1 = search for prest in progress. Value of zero is loaded by default at power up.
st_movdir	F	0	R	-	Movement direction Indication of direction of movement. 0 = forward, 1 = backward. Value of zero is loaded by default at power up.
st_loopon	F	0	R	-	Loop ON Reporting of the axis in reaction to space. 0 = axis not in reaction to space, 1 = axis in reaction to space. Value of zero is loaded by default at power up.
st_foller	F	0	R	-	Following error Axis tracking error signal (withholding 500 ms). 0 = axis not in following error, 1 = axis in following error. Value of one is loaded by default at power up.
st_regoff	F	0	R	-	Regulation Indicates that the setting is disabled and that upgrading the DAC is not execute.
st_cal	F	0	R	-	Calibration Reporting of device voltage generator. 0 = voltage axis generator disable, 1 = active axis voltage generator. Value of zero is loaded by default at power up.
st_cntlock	F	0	R	-	Counter locked Reporting of unlocked axis. 0 = axis count unlocked, 1 = axis count blocked.
st_cntrev	F	0	R	-	Counter reversed Inverted axis count report. 0 = Axis count not inverted, 1 = Inverted axis count. At the power up is maintained the state present at the power off.
st_acc	F	0	R	-	Acceleration Reporting of axis during acceleration. 0 = axis not in acceleration, 1 = axis acceleration. Value of zero is loaded by default at power up.
st_dec	F	0	R	-	Deceleration Reporting of axis in slowing down phase. 0 = axis does not decelerating, 1 = axis in decelerating. Value of zero is loaded by default at power up.
st_vconst	F	0	R	-	Costant speed Reporting of axis during constant speed. 0 = axis not in constant speed, 1 = axis in constant speed. Value of zero is loaded by default at power up.
st_error	F	0	R	-	Status of device error Indicates the status of error in device. To decode the error you must refer to the errcode and errvalue variables. 0 = Error not present, 1 = Error present. To the power up by default is set to zero.
st_warning	F	0	R	-	Status of device warning Indicates the status of warning in device. To decode the warning you have to refer to the wrncode and wrnvalue variables. 0 = Warning not present, 1 = Warning present. To the power up by default is set to zero.
st_int	F	0	R	-	Status of interrupt line Indicates the status of the interrupt line. 0 = Interrupt input off, 1 = Interrupt input on. To the power up by default is set to zero.

1.8 Limitations

No limitation.

1.9 Application example

1.9.1 Configuration file

```

*****
; Module name:      Ex Eanpos.CNF      Project:      Ex EANPOS
; Autor:           QEM srl             Date:         01/05/99
; System:          QMove1 / QCL3       Library:       1LIB4001
; Functionality:    EANPOS management example Release:  0
;----- Note -----
; [1] - Sample application using device EANPOS
*****
;-----
; Constant definitions
;-----
CONST
;-----
; Defining SYSTEM Variables
;-----
SYSTEM
slQuotaPos L          ;Variable for placement quota
slVelAsse L          ;Variable for axis speed
;-----
; Defining GLOBAL Variables
;-----
GLOBAL
gfMovMan F            ;Flag signalling ongoing
                    ;manual movements
gfMovAuto F           ;Flag signalling ongoing
                    ;automatic movements
;-----
; Defining TIMER Variables
;-----
TIMER
;-----
; Defining DATAGROUP
;-----
DATAGROUP
;-----
; Bus Configuration
;-----
BUS
1      1CPUD 01
2      1MIXA 00
3      .
4      .
;-----
; Defining INPUT Variables
;-----
INPUT
ifAvMan   F    2.INP01 ;Forward manual input
ifInMan   F    2.INP02 ;Backward manual input
ifStart   F    2.INP03 ;START axis input
ifStop    F    2.INP04 ;STOP axis input
;-----
; Defining OUTPUT Variables
;-----
OUTPUT
ofToll    F    2.OUT01 ;Output of axis in tolerance
ofAxeFermo F    2.OUT02 ;Output of axis stopped
;-----
; Internal Device declaration
;-----
INTDEVICE
;Name      Tipo  TCamp      Counter      Inter      AbilZero      OutAnal
Axis       EANPOS 0004      2.CNT01      1           2.INP01       2.AN01
END

```

1.9.2 EANPOS management

```

*****
; File Name : TASK_00.MOD
; Project   : EX EANPOS
; Description: PoSitioning Control
*****
;-----
; Initialization work Axis
;-----
Axis:measure = 10000 ;Space in 1 around encoder(Um)
Axis:pulse = 40000 ;pulses/Rev encoder
Axis:maxvel = 100000 ;Maximum speed = Axis speed with
                    ;analog output to 10V (CALON/vout=100)
Axis:maxpos = 999999 ;Maximum quota
Axis:minpos = -999999 ;Minimum quota
Axis:maxfollerr = 10000 ;Following error
Axis:unitvel = 0 ;Unit time speed
Axis:decpt = 0 ;Decimal digits
Axis:rampmode = 0 ;Type of ramps used
Axis:taccdec = 100 ;Acceleration and deceleration time
Axis:tacc = 0 ;Acceleration time (not used)
Axis:tdec = 0 ;Deceleration time (not used)
Axis:tinvt = 0 ;Inversion time axis
Axis:toll = 5 ;Tolerance
Axis:toldly = 10 ;Delay activation time tolerance
Axis:prsmode = 0 ;Preset search type
Axis:prspos = 0 ;Preset quota
Axis:prsdire = 0 ;Preset search direction
Axis:prsvl = (20 * Axis:maxvel)/100 ;Preset speed (20% of the maximum speed)
Axis:sprsvl = (10 * Axis:maxvel)/100 ;Zero pulse search speed
                    ;(10% of the maximum speed)
Axis:offset = 0 ;Offset voltage
Axis:pgain = 10 ;Proportional gain
Axis:feedfw = 1000 ;Feedforward
Axis:integt = 0 ;Integral time
Axis:derivt = 0 ;Derivative time
INIT Axis ;Initializes the device
WAIT Axis:st_init ;Wait until the device is initialized
CNTUNLOCK Axis ;Unlock count
WAIT NOT Axis:st_cntlock ;Wait until the count is unlocked
CNTDIR Axis ;Sets the direction of the count
WAIT NOT Axis:st_cntrev ;Wait until it is set the direction
                    ;of the count
REGON Axis ;Enable the setting
WAIT NOT Axis:st_regoff ;Wait the enable for the RESUME axis adjustment

```

```

;Remove the axis from the State of emergency
WAIT NOT Axis:st_emrg ;Wait until the axis is not in emergency
LOOPON Axis ;Hook the loop of axis reaction
WAIT Axis:st_loopon ;Wait until it is hooked on the
;the reaction axis loop
IF (slVelAxis EQ 0) ;In case the set of speed
;the axle is zero
slVelAxis = 50 ;Sets a positioning speed
ENDIF
IF (slQuotaPos EQ 0) ;In the case that the axis quota
;of placement is zero
slQuotaPos = 2000 ;Sets a quota of positioning
ENDIF

-----
; Positioning tasks
-----
;----- variables used -----
; slVelAxis: Adjustable variable that represents the axis speed
; (expressed in % of the maximum speed)
; slQuotaPos: Adjustable variable that represents the quota of axis positioning
;----- flag used -----
; gfMovMan: manual operation in progress
; gfMovAuto: automatic operation in progress
-----
MAIN:
-----
; Outputs managing
-----
ofToll = Axis:st_toll ;Forced out of tolerance as
;the tolerance state
ofAxeStopped = Axis:st_still ;Set the stopped axis as
;the stopped axis state

-----
; Automatic movements managing
-----
IF ifStart ;Waits for the START input
IF NOT gfMovMan ;Check that there are no manual movements
IF Axis:st_still ;Check that the axis is stopped
Axis:setvel=(slVelAxis*Axis:maxvel)/100 ;Sets the speed of the axis
Axis:setpos = slQuotaPos ;Sets the placement quota
START Axis ;Run the start of axis
gfMovAuto = 1 ;Automatic movement in progress reports
ENDIF
ENDIF
IF ifStop ;Waits for the STOP input
IF NOT Axis:st_still ;Check that the axle is not stopped
STOP Axis ;Run the stop of the axis
ENDIF
ENDIF
IF gfMovAuto ;Check reporting movement
;automatico in corso
IF Axis:st_still ;Check that the axis is stopped
gfMovAuto = 0 ;Reset the automatic movement state
ENDIF
ENDIF

-----
; Manual movements managing
-----
IF ifAvMan ;Awaiting the entry of manual operation
IF NOT (gfMovAuto OR gfMovMan) ;Check that there are no movements
;automatics or manuals
IF Axis:st_still ;Check that the axis is stopped
Axis:setvel=(slVelAxis*Axis:maxvel)/100 ;Sets the speed of the axis
MANFW Axis ;Forward axis in manual
gfMovMan = 1 ;Manual movement in progress reports
ENDIF
ENDIF
IF ifInMan ;Awaiting the manual operation input
IF NOT (gfMovAuto OR gfMovMan) ;Check that there are no movements
;automatics or manuals
IF Axis:st_still ;Check that the axis is stopped
Axis:setvel=(slVelAxis*Axis:maxvel)/100 ;Sets the speed of the axis
MANBW Axis ;Forward axis in manual
gfMovMan = 1 ;Manual movement in progress reports
ENDIF
ENDIF
IF gfMovMan ;If the axis moves in manual
IF NOT (ifAvMan OR ifInMan) ;If the forward and backward inputs
;manual is to OFF
STOP Axis ;Stop the axis
gfMovMan = 0 ;Remove the reporting of axis
;manual movement
ENDIF
ENDIF

-----
; Final operations
-----
WAIT 1
JUMP MAIN
END

```

Documento generato automaticamente da **Qem Wiki** - <https://wiki.qem.it/>

Il contenuto wiki è costantemente aggiornato dal team di sviluppo, è quindi possibile che la versione online contenga informazioni più recenti di questo documento.