Design Trends

A quarterly publication brought to you by Motion Designs Inc. May 2010

In this issue of Design Trends:

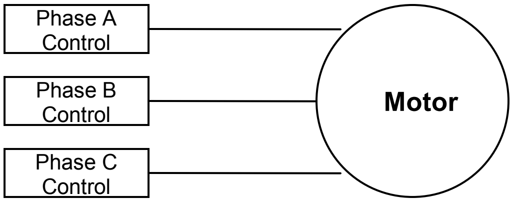
Vector Control and SVM

Vector control (also referred to as FOC – Field Oriented Control) of electric motors has allowed a true breakthrough in efficient motor control. Vector control is applicable to any multi-phase motor and provides an elegant mechanism to control motor currents. When augmented by SVM (Space Vector Modulation), electric motors can be controlled in the most optimal way.

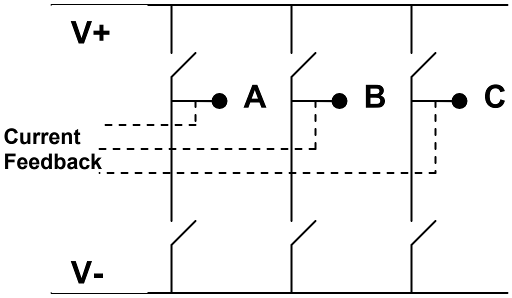
Traditional Current Control and PWM

Traditionally, 3-phase motor currents have been controlled by independently controlling 2 of the 3 currents, and letting the 3rd depend in some fashion.

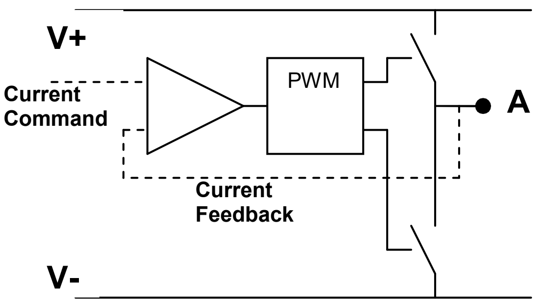
Dependency of the 3rd current is typically based on the sum of the currents being zero as well as maintaining a voltage balance across the motor phases.



With a typical H-bridge, the control topology looks as follows:



The switches of course include fly-back diodes for current reversal. The top and bottom switches for each leg would be modulated by the current controller for each leg, based on the current feedback from that phase.



If the 3 current commands are perfect and the 3 current measurements are perfect, then in principal this works fine. Unfortunately, this is rarely the case. Even if the commands and control loops are implemented with digital circuits, the current feedback is typically an analog measurement with offsets and non-linearities, resulting in an imbalance.

Also, the current in each phase is the result from all three phases' impedance and Back-EMF. However, the current controller for each leg only affects the duty cycle for that leg. So clearly this is not a direct method to control phase currents.

Lastly, and this will become relevant in comparison to vector control, the central point of the motor Wye winding is always around half the DC bus voltage.

Vector Control

Rather than treat each phase individually and modulate each leg based on a single current measurement, vector control takes a more holistic approach.

First the 3-phase currents are transformed to a single rotating current vector via the (2 step) Clarke-Park transformation:

$$i_\alpha = \frac{2}{3} \cdot i_a - \frac{1}{3}(i_b - i_c)$$

$$i_\beta = \frac{2}{\sqrt{3}}(i_b - i_c)$$

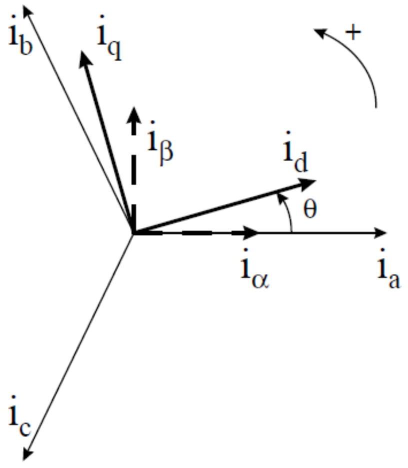
And

$$i_{sd} = i_\alpha \cdot \cos(\theta) + i_\beta \cdot \sin(\theta)$$

$$i_{sq} = -i_\alpha \cdot \sin(\theta) + i_\beta \cdot \cos(\theta)$$

The first transformation creates a rotating vector from the 3 phase currents, represented in a stationary reference frame with an α and β component (horizontal and vertical axis). Essentially the 3 phase current amplitudes are plotted along 3 vectors with 120 degree angle. The sum of those 3 vectors is a rotating vector if the 3 currents are sinusoidal with 120 degree phase shift. This resulting vector will rotate at fixed speed corresponding to the current frequency. The second transformation represents that same vector in a coordinate system rotated over angle θ, with a d and q component.

Graphically this corresponds to:



The same process applies to the phase voltages.

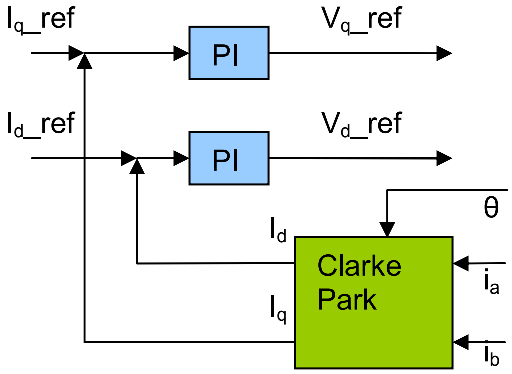
An inverse transformation from (d.q) to (α,β) and to phase a, b and c is of course also possible.

The angle θ is chosen to be the motor rotor position, which means that the current vector created by the stator currents is represented with its d and q components in a reference frame attached to the rotor.

Commutation

Because we have “translated” the 3 phase motor currents to a single current vector relative to the rotating motor rotor (i.e. the net flux producing current as seen from the rotor perspective) and because we know that commutation corresponds to maintaining a 90 degree angle between rotor and stator field, we simply need to ensure that the i_d component is held at zero (we have chosen the d-axis to lie along the rotor magnetic axis) and the i_q component simply becomes the amplitude of the current we whish to maintain. In addition, if we wanted to create a phase lead or lag between the stator flux and rotor flux (to get either motor torque or speed) we would simply adjust the i_d component to a non-zero value.

So we have essentially “bundled” current control and commutation into control of the d and q component of a current vector. Schematically this becomes:

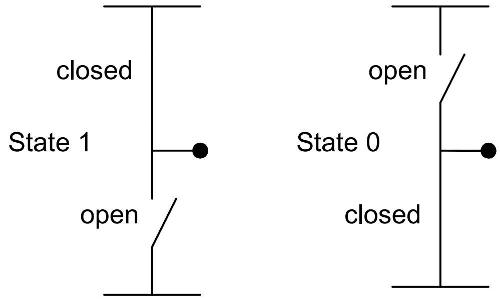


In most applications the PI controller for both the d and q component can use the same gains, so there is only one loop to tune.

Space Vector Modulation

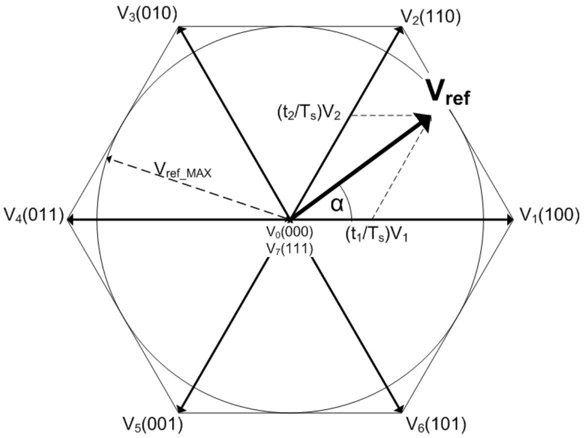
Now that we have simplified the current control and commutation, we will consider how we control voltage to the motor.

If we go back to the structure of a typical 3-phase H-bridge, we can observe that relative to switch status, there are just 8 possibilities. It is custom to label the state of one specific leg as either 1 or 0. A 1-state corresponds to the top switch being ON and the bottom switch being OFF, while a 0-state corresponds to the top switch being OFF and the bottom switch being ON (note that both switches ON would cause a short and both switches being open would leave the phase open).



If we represent the overall bridge state with a 3-digit number, then for example "100" means that the top switch of phase A is ON and the bottom switches of phase B and C are ON. A "011" would mean the bottom switch of A is ON as well as the top switches of B and C. Clearly there are 8 unique states: 000, 100, 010, 001, 110, 011, 101 and 111.

Each of these states creates a specific voltage pattern. For example "010" means that phase A and C are connected to V- and phase B is connected to V+. If we represent this again as a vector the way we did with currents (the phase to α and β transformation), we obtain the following diagram:



Voltage vectors 100, 010 and 001 lie along the 3 vectors at 120 degrees. Vectors 110, 011 and 101 are a combination of the first three. The 000 and 111 vectors are shown in the origin because they result in a net zero voltage (either all top switches are ON or all bottom switches are ON).

To create any other voltage vector (i.e. any vector at a certain angle and with certain amplitude, we simply modulate between the 2 adjacent vectors in the hexagram. As shown above, to create V_{ref} we would modulate between V_1 and V_2 (i.e. between the 100 and 110 state). Note: more precisely we also modulate between one of the null-vectors, V_0 and V_7.

Although this switching scheme appears somewhat convoluted at first, it has some significant advantages:

- More direct way to control the output voltages from the above vector control method.
- A 15% increase in effective voltage across each phase.
- Lower switching losses in the output stage.

We could have taken our above vector control and derive with the inverse Clarke-Park transformation the phase voltages for each and modulated them accordingly. However, with SVM we need only go through the inverse Park in order to obtain the voltage vector in the α and β reference frame.

The 15% increase in effective voltage comes from the fact that the voltage of the winding center point is no longer fixed at half the DC bus but actually moves up and down (requires extensive analysis, just the result is stated here). This increase in effective voltage allows higher speed operation (to counter motor back-EMF) as well as higher current loop bandwidth.

When compared to traditional sinusoidal PWM, the number of switch transitions is lower with SVM. For example as shown above, during modulation between V_1 (100) and V_2 (110) only one leg (phase B) switches state. Due to fewer switch transitions SVM results in significantly lower switching losses in the output stage.

Conclusion

The combination of vector control and SVM creates a holistic method of controlling motor currents and voltages. Rather than controlling individual phases, this approach looks at the motor and output stage as a single unit controlled by a single vector. It then considers the components of this vector in a reference frame attached to the rotor and further reduces the current control and commutation to a single scalar value.

The Space Vector Modulation scheme further enhances current control by again considering the output stage as a single entity to control a voltage vector.

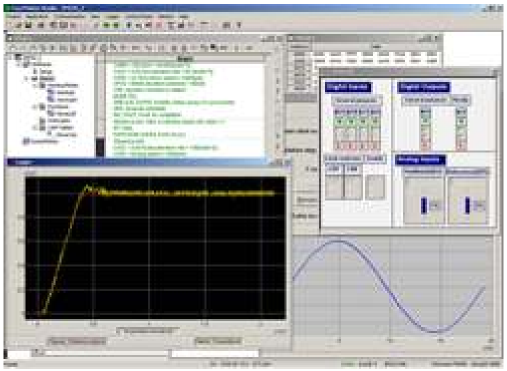
Lastly, both methods apply equally well to flux vector control of asynchronous motors or any other poly-phase motor.

Although the algorithm is more complex than traditional sinusoidal PWM, the use of digital signal processors (DSP’s) has accelerated the use of this control method in almost all motor control applications.

Technosoft ISD720/860

ISD720 and **ISD860** are fully digital high-precision servo drives with embedded intelligence and a built-in power amplifier (720, or 860 W). Used to control brushless DC, brushless AC (vector control), DC brush and linear motors, these high-performance intelligent servo drives combine motion controller, power amplifier and PLC functionality in a single, compact unit.
The on-board CAN-bus interface allows this type of drive to be used as an intelligent axis in a distributed-intelligence, multiple-axis network (up to 256 axes).

- Fully digital servo amplifier with embedded motion controller
- Controls brushless (sinusoidal / trapezoidal), DC brush, or linear motors
- Single-axis, operating in stand-alone or slave mode (distributed intelligence)
- Control modes: position, speed, or torque
- Motion modes: S-curve profiles, 1st / 3rd order PT/PVT interpolation, electronic gearing and electronic camming
- External reference: analog or digital (2nd encoder, or pulse & direction)
- Powerful TML instruction set including: motion commands, program flow control, I/O handling, arithmetic and logic operations, remote control from another drive
- Axis synchronization between all drives
- Output current: 20 A (ISD720 model) or 12 A (ISD860 model) continuous; 49.5 A peak (ISD720) or 31 A peak (ISD860)
- Supply voltage: 12-36 V for logic; 12-36 V (ISD720) or 12-72 V (ISD860) for motor
- Open-frame design (136 x 84 x 26) mm
- DSP controller based on the Technosoft MotionChipTM technology
- 10-bit PWM at 20 KHz
- RS-232 serial communication, up to 115 k baud rate
- CAN bus 2.0B with TMLCAN or CANopen protocols, up to 1 Mbit/s
- Operation in multiple-axis structures, via CAN communication channels



Product Feature: NPM Linear Shaft Motor Currents

Introduction

Linear brushless servo motors are mostly used in applications where high speed, high precision or both combined are required. Linear servo motors create a direct force on the load thereby eliminating any rotary to linear mechanical transmission. This avoids any speed limitations, mechanical wind-up (due to stiffness) or backlash imposed by the mechanical drive train. The Linear Shaft Motor (LSM) from Nippon Pulse is a very optimal implementation of a linear brushless motor, consisting of a stainless steel shaft with rare-earth magnets and an iron-less 3-phase winding which completely captures the shaft magnet fields.

Either the shaft can be held in place and the forcer (coil set) can be moved or vice-a-versa.

Motor Ratings

Linear shaft motor ratings are very similar to rotary brushless motor ratings, in that all electrical parameters of the 3-phase winding are detailed. Below is an example of an S080 motor datasheet:

Electrical Specs	S080D	S080T	S080Q
Continuous Force	1.8N	2.7N	3.5N
Continuous Current	0.8Arms	0.8Arms	0.8Arms
Peak Force	7.2N	10.8N	14N
Peak Current	3.4Arms	3.4Arms	3.4Arms
Force Constant (Kf)	2.1N/Arms	3.2N/Arms	4.2N/Arms
Back EMF (Ke)	0.7V/m/s	1.1V/m/s	1.4V/m/s
Resistance 25°C	4.7Ω	6.8Ω	9.0Ω
Inductance	0.7mH	1.0mH	1.3mH
Electric Time Constant	0.149ms	0.147ms	0.144ms
Fundamental Motor Constant (Km)	0.98N√W	1.23N√W	1.39N√W
Magnetic Pitch (North-North)	30mm	30mm	30mm

The S080D, S080T and S080Q are 3 different forcer coils (with double, triple and quadruple winding respectively).

Instead of have ratings in rotary units (e.g. a brushless rotary motor back-EMF is specified in V/krpm), units are of course in linear units (e.g. back-EMF is V/m/s). The resistance and inductance values are phase-to-phase.

Continuous Ratings

In the table above, the current ratings require some additional detail.

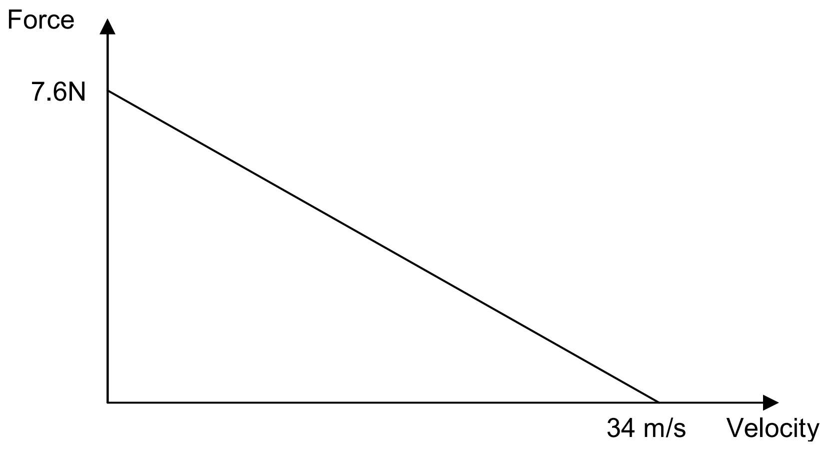
The continuous current (and hence continuous force) rating is based on no heat sinking and an ambient temperature of 25C. This rating corresponds to holding the forcer in open air and allowing the coil temperature to reach its maximum value.

In real applications of course, the forcer (i.e. coil) is attached to something (either the part being moved or a plate in case the shaft is moved). Hence there is always some heat sinking. If one assumes typical heat sinking then the continuous rating can be increased by 20 to 40%. The exact number depends on the specific mounting of the forcer coil to the system. This can either be established empirically or can be modeled by using the thermal resistance of the coil assembly (detailed in the datasheet). For maximum protection and optimal performance, a temperature sensor can be added to the system to monitor the forcer surface temperature.

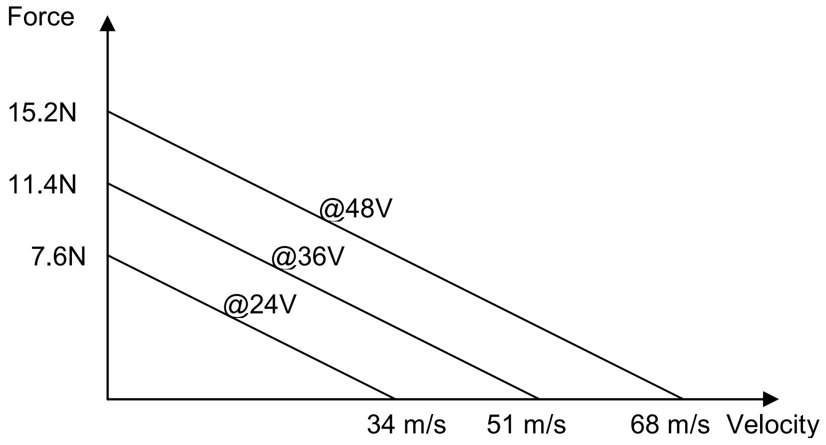
Peak Ratings

The peak current (and hence peak force) rating in the table above is specified for up to a 40 second duration. Higher values are allowed for shorter durations (e.g. short acceleration bursts).

Ignoring inductance for illustrative purposes, the S080D for example has the following force versus velocity curve (with a 24VDC bus):



This is simply based on the motor voltage = current*resistance + back-EMF equation. In this case the 7.6N peak force is only slightly higher than the continuous rating. However at 36 and 48VDC the curves would look as follows:



Of course the maximum velocity increase is of no real use as these velocities are already extremely high (the back-EMF of the motor is relatively low). What is of interest of course is the increase in force, which can help achieve higher accelerations.

To estimate the time duration for these higher forces (and currents), we can simply employ an "I^2-t" rule (i.e. allow same heat dissipation). If a current of 3.4Arms is allowed for 40 seconds then a current that is 50% higher can be allowed for $40/(1.5^2)$ or 17 seconds. A current that is 2 times higher can be allowed for 10 seconds. A current that is 3 times higher can be allowed for 4 seconds. For short accelerations, these time durations are often more than adequate. From the above graphs however, one can immediately see that higher bus voltages will be required in order to generate such currents in the motor. Since the Linear Shaft motors can run from 110VAC or 220VAC sourced drives this is actually not a bottleneck.

As accelerations are increased, and acceleration times are decreased, it might be tempting to also allow this to increase throughput. However, one most continue to respect the overall average current and not exceed continuous ratings.

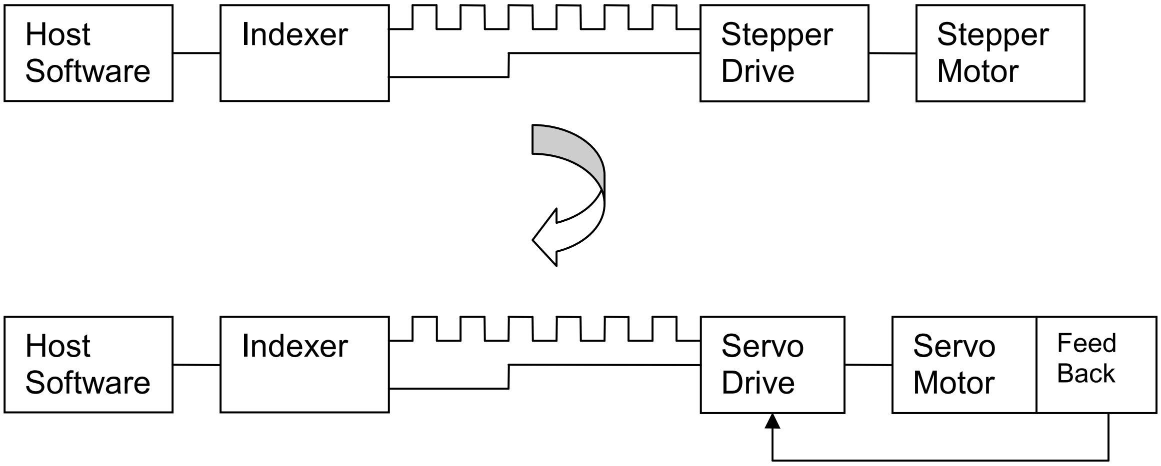
Application Solution: Step and Direction Input Servo

Stepper motor driven systems are extremely prevalent and for many applications, they allow for cost-effective and easy to implement position control.

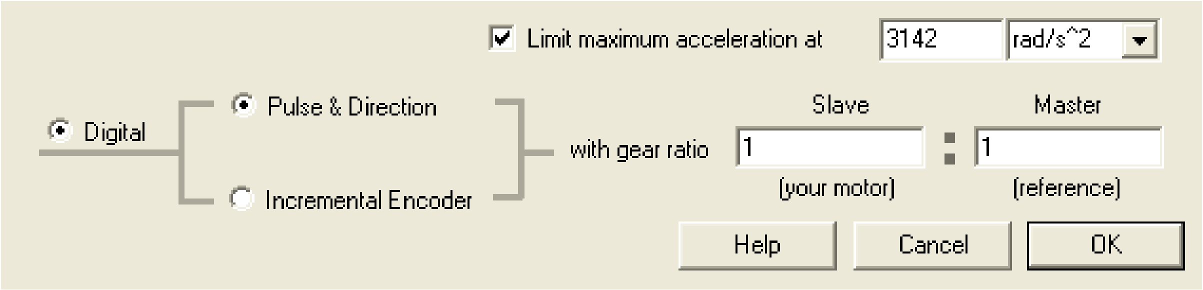
Sometimes however the application requirements evolve over time and higher speed and/or more accurate control is required. This can lead to a very costly re-design of the system, especially if a large amount of software has been developed around an existing stepper-based architecture.

To avoid such an extensive re-design, users can consider just replacing the stepper motor and drive with a servo motor and drive instead. This applies equally well to rotary as well as linear servos.

Basically, instead of sending the step and direction signals from the indexer to the stepper driver, those signals now go to the servo drive and directly control the position target.



One potential difficulty with this scenario is matching up the initial stepper system resolution with the position resolution of the servo. With a servo system, the final positioning resolution will be determined by the motor feedback resolution. Technosoft drives have a straightforward mechanism to match up both resolutions:



☑ Limit maximum acceleration at 3142 rad/s^2

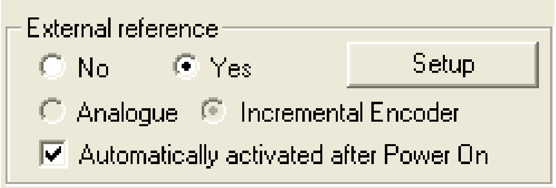
● Digital
- ● Pulse & Direction
- ○ Incremental Encoder

with gear ratio Slave 1 (your motor) : Master 1 (reference)

The user can specify an exact ratio between incoming pulses. For example if the original stepper has 400 steps per revolution and the servo feedback is a 1,024 line encoder (i.e. 4,096 counts per revolution), then the ratio can be set to 4096:400. Although this is not an integer value, the drive maintains any fractional parts and avoids any rounding errors.

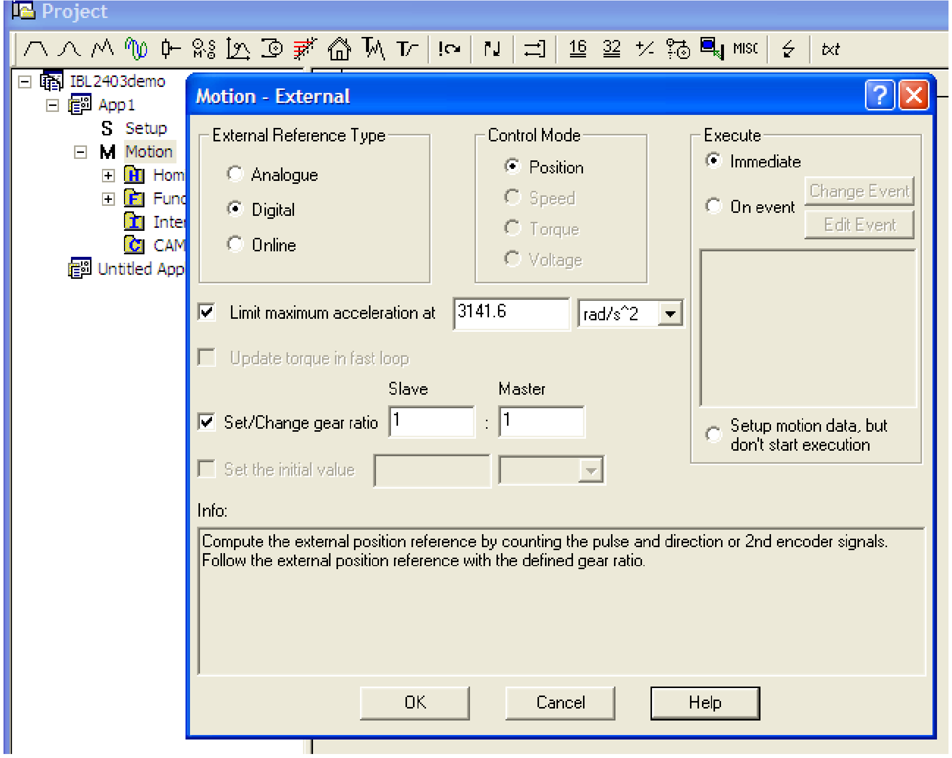
As can be seen above, the drive can also super-impose an acceleration limit for pulse trains that do not have accurate rate changes.

Although the drive can power-up in step and direction input mode:



External reference
○ No ● Yes
○ Analogue ● Incremental Encoder
☑ Automatically activated after Power On

(by checking Automatically activated after Power On), alternatively this can also be enabled programmatically from a TML program:



Motion - External

External Reference Type
○ Analogue
● Digital
○ Online

Control Mode
● Position
○ Speed
○ Torque
○ Voltage

Execute
● Immediate
○ On event
○ Setup motion data, but don't start execution

☑ Limit maximum acceleration at 3141.6 rad/s^2

☐ Update torque in fast loop

☑ Set/Change gear ratio Slave 1 : Master 1

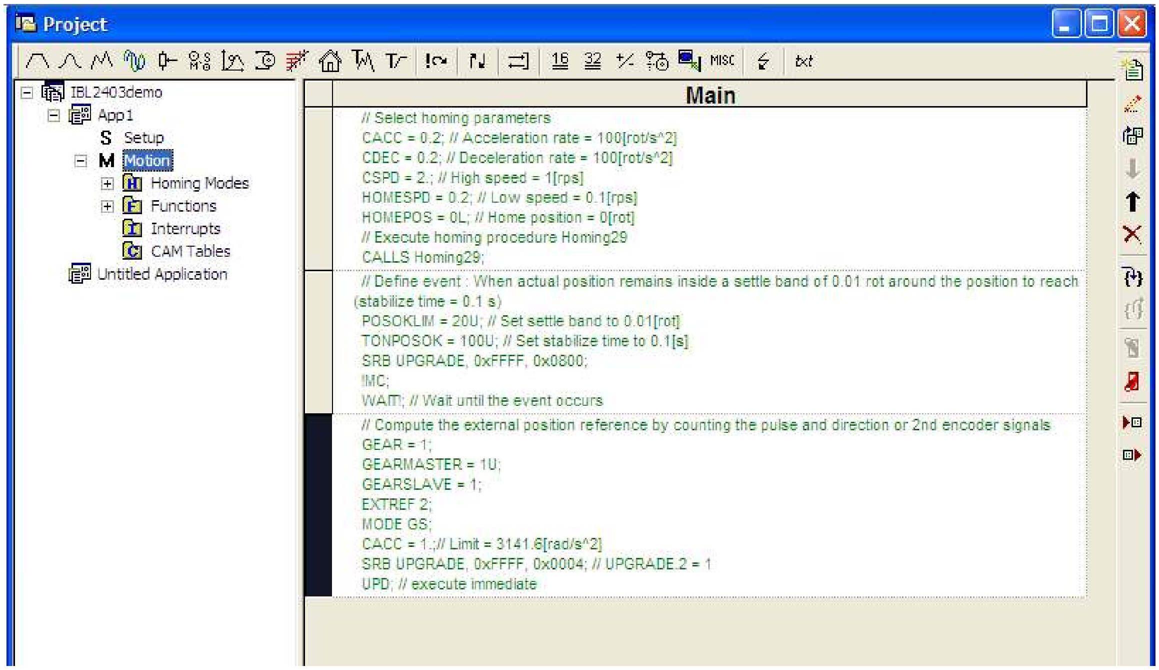
☐ Set the initial value

Info:

Compute the external position reference by counting the pulse and direction or 2nd encoder signals. Follow the external position reference with the defined gear ratio.

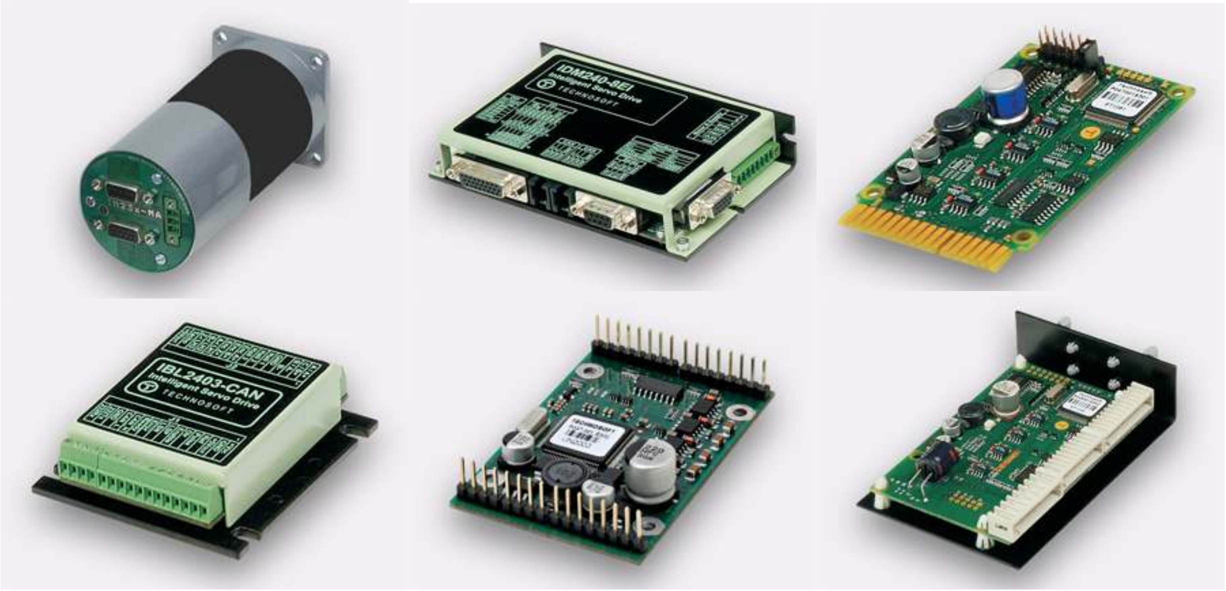
This means that the user can perform some other functions on power-up prior to activating step and direction mode.

For example, the drive can execute a homing routine (e.g. home to a linear encoder index pulse), prior to following a step and direction signal.



```
// Select homing parameters
CACC = 0.2; // Acceleration rate = 100[rot/s^2]
CDEC = 0.2; // Deceleration rate = 100[rot/s^2]
CSPD = 2.; // High speed = 1[rps]
HOMESPD = 0.2; // Low speed = 0.1[rps]
HOMEPOS = 0L; // Home position = 0[rot]
// Execute homing procedure Homing29
CALLS Homing29;
// Define event : When actual position remains inside a settle band of 0.01 rot around the position to reach
(stabilize time = 0.1 s)
POSOKLIM = 20U; // Set settle band to 0.01[rot]
TONPOSOK = 100U; // Set stabilize time to 0.1[s]
SRB UPGRADE, 0xFFFF, 0x0800;
!MC;
WAIT!; // Wait until the event occurs
// Compute the external position reference by counting the pulse and direction or 2nd encoder signals
GEAR = 1;
GEARMASTER = 1U;
GEARSLAVE = 1;
EXTREF 2;
MODE GS;
CACC = 1.;// Limit = 3141.6[rad/s^2]
SRB UPGRADE, 0xFFFF, 0x0004; // UPGRADE.2 = 1
UPD; // execute immediate
```

Step and direction input operation is possible on all Technosoft drives as well as integrated servo motors.



For more information about any of the above topics or general questions or comments, please contact us:

Motion
Designs
Inc.

Motion Designs
contact@motion-designs.com
Tel 805.504.6177

Motion Designs is a technical sales and engineering company with extensive machine and motion control experience. We work with some of the best manufacturers in the industry as witnessed by our present line card:

- www.amosin.com: AMO manufactures induction based precision linear and angle measurement encoders.
- www.arcus-technology.com: Arcus Technology manufactures stepper motor, drive and controller technology, providing USB, Ethernet and Mod-Bus connectivity.
- www.nipponpulse.com: Nippon Pulse manufactures the unique linear shaft motor, a direct drive linear brushless servo motor.
- www.stegmann.com : Stegmann is a leader in high performance motor feedback solutions.
- www.technosoftmotion.com : TSM is a leading DSP motion control technology company specialized in the development, design and manufacture of digital motor drive products and custom motion systems.

amo

ARCUS
Technology

NPM Your Partner in Motion Control

SICK | STEGMANN

TECHNOSOFT
MOTION TECHNOLOGY