

Getting Started with VisSim/Motion

This chapter contains the following information:

- Installation procedure
- Summary of blocks in the Motion block set
- AC induction motor tutorial
- Brushless DC (BLDC/PMSM) motor tutorial

Installation procedure

Before you begin the installation procedure, be sure you have your serial number. This number appears on your manual and on the box cover of your product. If you are installing an evaluation (or trial) copy of VisSim/Motion, you will not need a serial number or activation number. Your evaluation copy will run for 14 days, at which point the software automatically reverts to demo mode. In this mode, all new models you create and all changes you make to existing models will not be saved.

To install VisSim/Motion:

1. Insert the VisSim/Motion CD in your CD drive.
2. The VisSim/Motion installation procedure automatically starts up. If it does not start up, do the following:
 - Click on the Start menu and choose Run.
 - In the Run dialog box, click on the Browse button.
 - In the Browse dialog box, select the CD drive.
 - Select VSMMOTION.EXE and click on the Open button.
 - The Run dialog box re-appears.
 - Click on the OK button to begin the installation.
3. Install asks you for a serial number. Do the following:
 - If you are a licensed user, enter the number on your manual or box cover.
 - If you are installing an evaluation (or trial) copy of VisSim/Motion, enter TRIAL in the Serial Number box.
4. Follow the on-screen instructions.

Summary of Motion block set

The following table categorizes the Motion blocks as they appear in the Motion menu. These blocks are described in detail in Chapter 2, “Block Reference.”

Block category	Block name
Amplifiers	PWM Brush Servo Amplifier
	PWM Brushless Servo Amplifier
Controllers	Commutator (Six Step)
	PID Controller (Digital)
	PID Controller (Fixed Point)
	PID Controller (Ideal)
	Field Orientated Controller (FOC)
	PWM (Dual Phase)
	PWM (Single Phase)
Filters	PWM (Space Vector)
	Low Pass Filter
	Rate Estimation (Rooftop) Filter
Loads	Rotational Load
	Translational Load
Motors	AC Induction Motor (DQ)
	AC Induction Motor (FOC)
	AC Induction Motor (Machine Reference)
	Basic (Permanent Magnet) DC Motor
	Brushless DC (BLDC/PMSM) Motor
Sensors	Frequency Demodulator
	Hall Sensor
	Linear Encoder
	LVDT
	Rotary Encoder
	Rotary Position Sensor
	Rotary Servo Potentiometer
	Rotary Tachometer
Sources	Three Phase AC Source
	Three Phase Square Wave Inverter
	Triangle Wave Generator
Tools	Discrete Integration
Transforms	Clarke Transform
	Inverse Clarke Transform
	Inverse Park Transform
	Park Transform (Stationary Frame)

AC induction motor tutorial: speed control of a machine tool lathe

Tutorial Model: ... \MOTION\TUTORIAL\MACHINE_TOOL_TUTORIAL.VSM

The typical machine tool lathe is operated from a single-speed motor drive, together with multiple gear selection to vary chuck speed. Here a simpler design is considered: one with a single 10:1 gear reducer and a variable speed control drive for a 3-phase AC induction motor.

The lathe is required to operate with the following specifications:

Maximum work piece load: 1 meter by 0.1 meter diameter aluminum bar stock

Chuck speed control range: 30 to 400 RPM

Speed control accuracy: ± 5 RPM from set point steady state

Maximum load torque: not to exceed 0.3 N-m, introduced by cutting tool

The motor specifications are given as:

Motor parameter	Value	Units
Stator resistance	9.4	Ohms
Stator self inductance	0.402	Henries
Stator leakage inductance	0.032	Henries
Rotor resistance	7.1	Ohms
Rotor leakage inductance	0.032	Henries
Number of poles	2	
Rotor inertia	0.001	Kg-m ²
Rotor viscous damping constant	0.0001	Kg-m ² - s

The moment of inertia of the chuck and moving drive assembly is given as 0.1 kg-m². The moment of inertia of the work piece is calculated as:

$$I = \frac{1}{2}Mr^2 = \left(\frac{1}{2}\right)(21.14kg)\left(\frac{0.1m}{2}\right)^2 = 0.026kg\ m^2$$

Since the axes of the chuck and work piece are coincident, they add to total 0.126 kg m².

One very effective way of controlling speed by an induction motor is to control the stator field frequency. Since stator flux is inversely proportional to frequency below the base frequency, it is necessary to adjust voltage proportional to frequency to maintain constant flux. For frequency above the base frequency (power supply limitation), the voltage is kept constant. This method is the basis of the design, with one minor improvement. The constant volts to frequency control mentioned above are used as a feed forward leg of a feed forward – proportional integral controller (PI). The PI component of the control is used to adjust any error that may occur due to motor slip and loading from the cutting tool. Motor speed is estimated from motor shaft position measured by an incremental encoder. To drive the motor, an inverter is used with six-step logic to switch polyphase-rectified voltage producing a balanced 3-phase signal.

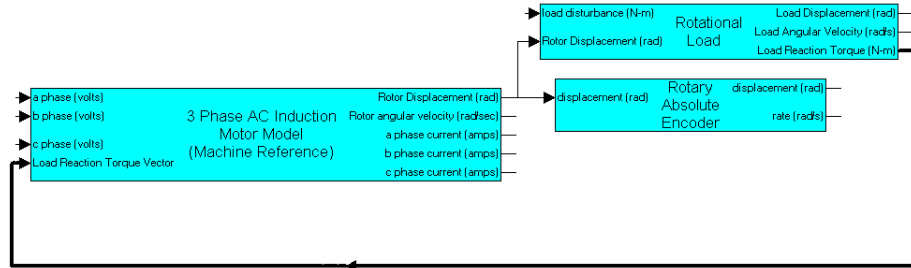
Setting up the motor, load, and encoder

The first step is to place the following Motion blocks in your diagram:

- Rotational Load
- AC Induction Motor (Machine Reference)

- Rotary Encoder

The Rotational Load block is under the Loads category; the AC Induction Motor (Machine Reference) block is under the Motors category; and the Rotary Encoder block is under the Sensors category. Wire the blocks together and use wirePositioner blocks to clearly represent the feedback of the load reaction torque to the motor model.



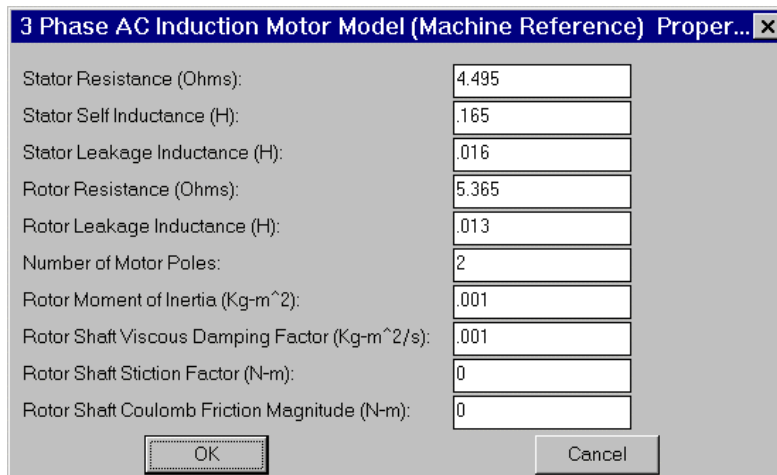
The rotational load model is used to simulate the lathe chuck and work piece. The rotary encoder model input is connected to the motor's rotor shaft displacement output connector. The motor displacement output is also connected to the rotational load model. To complete the dynamic interaction between the motor and load, the load reaction torque output connector must be connected to the load reaction torque vector input of the motor model. Note that this wire is thicker than the other wired connections, indicating that it transmits a vector quantity. The vector contains load dynamic parameters that are reflected back to the motor dynamics through the coupling (linkage) mechanism: in this case, a 10:1 gear reduction.

Setting parameter values

The next step is to enter the parameters for the motor, load and encoder. The parameter values can be changed later to see what affect they may have on the final control solution.

Parameter values for the AC Induction Motor (Machine Reference) block

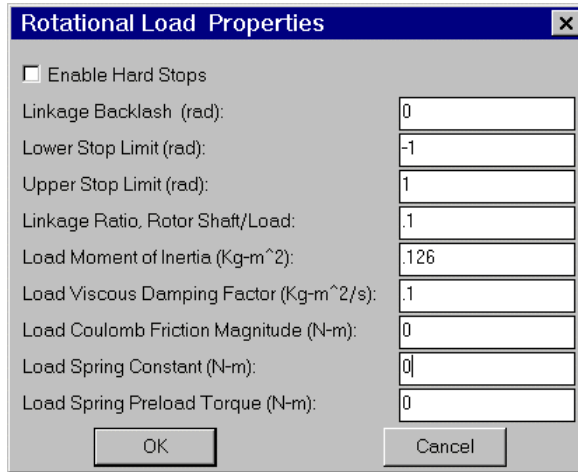
Click the right mouse button over the AC Induction Motor (Machine Reference) block to display its Properties dialog box.



Enter the motor parameters shown above. These parameter values are taken from the motor specifications table shown on page 3.

Parameter values for the Rotational Load block

Click the right mouse button over the Rotational Load block to display its Properties dialog box.



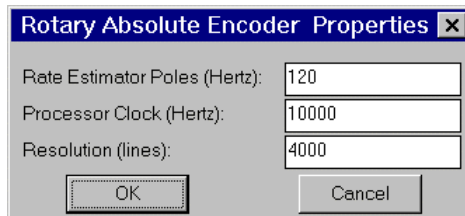
Enter the values shown above.

Note the following:

- The value for the Load Viscous Damping Factor is a rough guess.
- For the linkage ratio (gear ratio for this application), follow this rule: a factor less than 1.0 multiplies torque, and a factor greater than 1.0 multiplies speed; entering 1.0 produces a direct connection between motor and load.
- Default values are shown for the upper and lower stop limits, but since the Enable Hard Stops checkbox is not activated, hard stop limits are not used in the model. Hard stops are useful in position control system applications.

Parameter values for the Rotary Encoder block

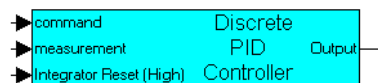
Click the right mouse button over the Rotary Encoder block to display its Properties dialog box.

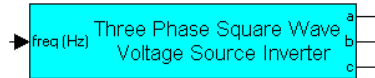


Enter the values shown above.

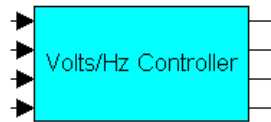
Designing the volts/frequency controller for the motor

In this step, you use the PID Controller (Digital) block (from the Controllers category); the Three Phase Square wave Inverter from the Sources category to design the volts/frequency controller for the motor.



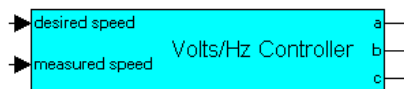


After placing the blocks in your diagram, encapsulate them in a compound block using the Edit > Create Compound Block command. Use the name Volts/Hz Controller for the compound block name.



This block design requires only 2 inputs and 3 outputs. By default, when you create a compound block, VisSim creates and outputs for all the blocks contained in the compound, which may not be appropriate. In this case, you must remove 2 inputs and one output using the Edit > Remove Connector command.

Now label the wire tab connectors by double clicking each tab connector with the left mouse button and entering the names as shown below:



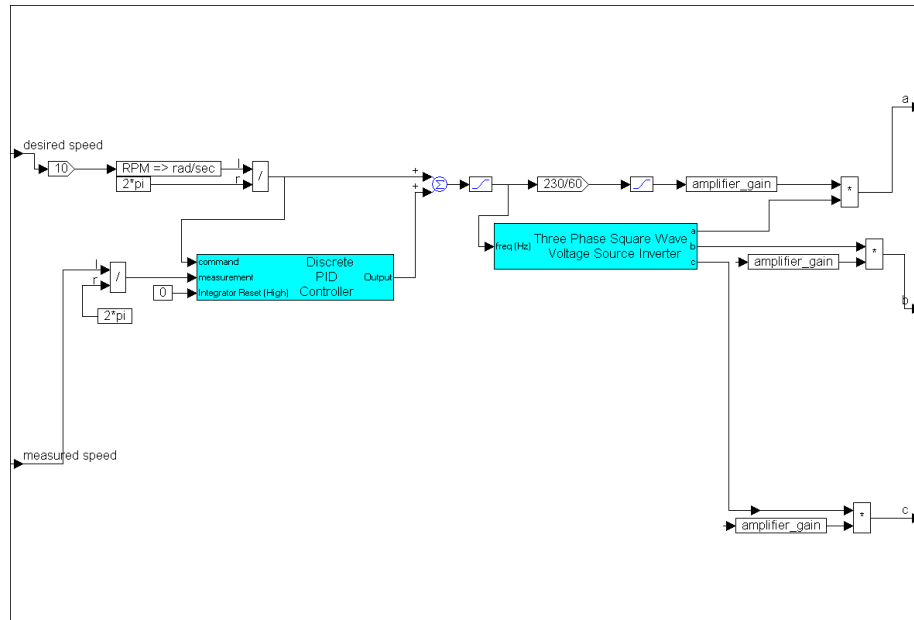
Next, click the right mouse button over the compound block to drill into it. As mentioned before, the process of creating the compound block creates wire connections that may not be needed. Remove all wire connections within the compound by clicking, holding and dragging the wires with the left mouse button to an open space of the desktop and releasing the button.

Customizing the Volts/Hz Controller block

Within the `volts/Hz Controller` compound block, insert the following blocks:

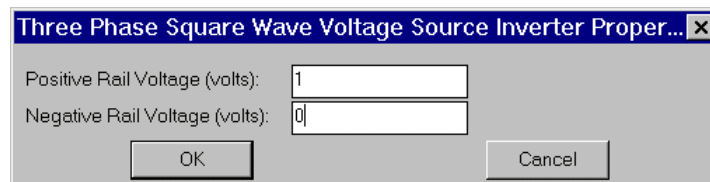
- From the Blocks > Arithmetic category, add a `summingJunction`, `gain`, `unitConversion`, and `/` block.
- From the Blocks > Signal Producer category, add a `const` block.
- From the Blocks > Nonlinear category, add a `limit` block.

Wire the blocks together, as shown on the next page.



The input speed for this block is assumed to be the speed of the chuck; therefore, the `gain` block is needed to scale this speed up by a gear ratio of 10 since this controller affects the speed on the motor side. RPM is then converted to hertz by using a `unitConversion` block set to RPM rad/sec and then dividing the output by 2π . The value 2π is produced by using a `const` block set to $2*\pi$.

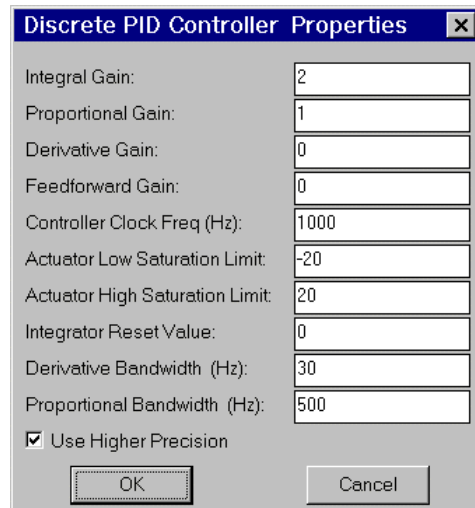
The measured speed comes from the `Rotary Encoder` and is in radians per second. This measurement is converted to hertz simply by dividing by $2*\pi$. The desired speed in hertz is fed into a `summingJunction` block, as well as the command input of the `PID Controller (Digital)` block. The desired speed directly feeds the inverter/amplifier as the feed forward component of the control. `PID Controller (Digital)` block output is used to correct for minor errors in the feed forward component. The sum of these two components is fed to the inverter/amplifier, the sum is limited to 70 hertz to prevent running the motor into its unstable region of control. The output of the `limit` block feeds the `Three Phase Square Wave Inverter` block. The `Three Phase Square Wave Inverter` block rail voltages must be set to 0 and 1, as shown below, to effectively provide logic control rather than bus level voltages:



The output of the control `summingJunction` block is scaled inversely proportional to frequency by using a `gain` block with the factor $230/60$. The output is then limited between 0 and 230 volts, and defined to be a `variable` block with the user-defined name `amplifier_gain`.

Configuring the PID compensator

To configure the PID compensator, enter the following values into the PID Controller (Digital) block:



Parameter	Value
Integral Gain:	2
Proportional Gain:	1
Derivative Gain:	0
Feedforward Gain:	0
Controller Clock Freq (Hz):	1000
Actuator Low Saturation Limit:	-20
Actuator High Saturation Limit:	20
Integrator Reset Value:	0
Derivative Bandwidth (Hz):	30
Proportional Bandwidth (Hz):	500

Use Higher Precision

OK Cancel

Since the feed forward and derivative gain are set to 0, the block is actually configured to operate as a PI controller. Saturation is set to limit the influence of the integral correction to ± 20 Hz. Proportional bandwidth is set at Nyquist frequency ($\frac{1}{2}$ the sampling frequency); derivative bandwidth does not matter in this controller. The Use Higher Precision option is turned ON to allow trapezoidal integration to be used.

Integral reset is not used on this controller, so a `const` block with a value of 0 is fed into `PID Controller (Digital)` to prevent integral reset. The actual values for the proportional and integral gain were determined experimentally in the final configuration to obtain minor overshoot and settling in the control.

This completes the construction of the `Volts/Hz Controller` compound block.

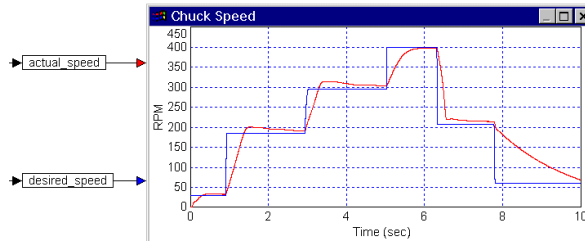
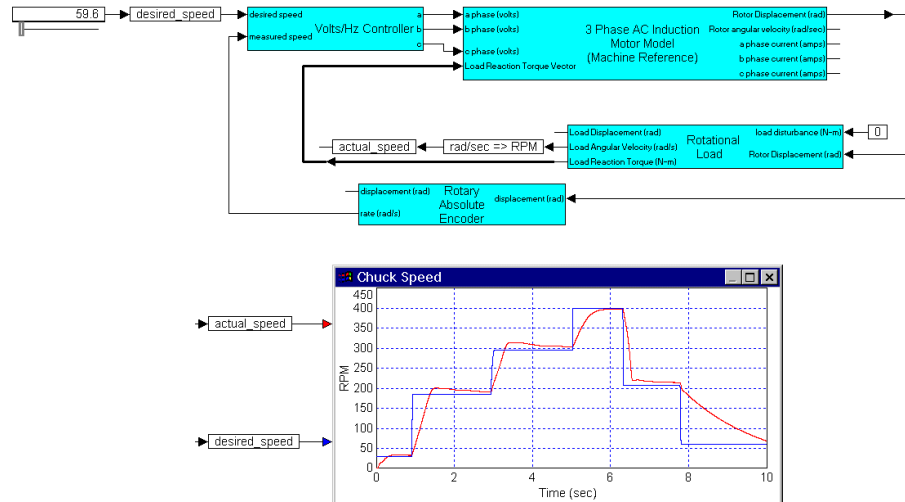
Wiring Volts/Hz Controller to the overall simulation

The three outputs for `Volts/Hz Controller` are connected to the corresponding inputs of the induction motor block. Measured speed from the `Rotary Encoder` block is connected to the measured speed input of the `Volts/Hz Controller` block. A `slider` block, scaled between 30 and 400, is connected to the desired speed input of the `Volts/Hz Controller` block as RPM speed input. A `plot` block is wired to compare the desired and actual speeds. The actual speed is determined by converting load angular velocity to RPM. A `const` block set to 0 is connected to the load disturbance input of the rotational load model. `wirePositioner` and `variable` blocks are used to make the diagram legible.

Before simulating the model, click on the `Simulate > Simulation Properties` command and make the following selections:

- In the Start Time box, enter 0
- In the Step Size box, enter 0.0001
- In the End Time box, enter 10

Through minor exploration, the motor drive is found to have sufficient torque at all speeds to overcome maximum tool exertion.



Now with a working simulation, you have met the design requirements and can now begin playing simulation games to optimize performance and reduce cost. For example, a fairly high-resolution encoder was used for estimating rate. How coarse can the resolution become before performance is degraded? Also the motor may be oversized for the particular application. Surveys show that over 50% of the motors selected in the US are oversized for their application. Simulation provides a lower cost alternative to performing extensive analysis or purchasing a variety of motors to empirically determine which is best suited for an application. This is true for any motion control application; not just limited to machine tools.

AC induction motor application: power train drive system

The process of building the simulation for a machine tool application can be similarly applied to a power train drive system for a passenger electric vehicle. Here the design could start with a 3-phase AC induction motor model, which provides the necessary horsepower and speed to drive the vehicle. Based on the vehicle mass and wheel size, a rotational load model can be configured. Losses due to wheel bearing friction can also be included as a combination of stiction, Coulomb, and viscous friction within the `Rotational Load` block parameters. Using the wheel size, output angular velocity for this model can be converted to vehicle linear speed.

Additional static blocks can be configured to model drag forces on the vehicle that can be fed back into the load disturbance of the rotational load block. Selecting a linkage ratio of 1, and using additional blocks to connect between the motor and load displacement connections can simulate a clutch and automatic transmission. Gravity induced loading can be simulated as a function of road angle and superimposed with other load disturbance inputs. The purpose of such a simulation could possibly focus on optimization of motor parameters or the development of a new method for automatic gear shifting.

Brushless DC (BLDC/PMSM) motor tutorial: target tracking system

Tutorial Mdoel: ...MOTION\TUTORIALS\TARGET_TRACKING_TUTORIAL.VSM

This example simulates a servo-controlled positioning system that maintains focal plane line of sight coincident with target angle. The permanent magnet synchronous motor model is selected as an actuator to provide fast response.

Motor Specifications

Automatically acquiring and maintaining the line of sight of a video camera or focal plane sensor is often required in various aerospace, defense, and security system applications. One way to mechanize such a system is to reflect the field of view through two independently-controlled mirrors that each rotate in axes orthogonal to one another. The object of the control system is to acquire the target, and by controlling rotation of each mirror, move the line of sight coincident with the target angle. This places the virtual image of the target in the center of the focal plane. Once the image of the target is acquired on the focal plane, an error in azimuth and elevation can be determined by a variety of image processing techniques, such as contrasting, differencing, and area parameter calculations.

For this simulation, such a mechanism is assumed, with a pipeline image processor providing direct angular azimuth and elevation measurements. The following design decisions are also assumed:

Motor type: Permanent magnet DC synchronous motor with Hall sensors for commutation sensing and control.

Motor parameter	Value	Units
Operating voltage	28	Volts
Magnetizing inductance	0.0009	Henries
Stator winding self inductance	0.001	Henries
Stator winding resistance	0.5	Ohms
Magnetic flux linkage magnitude	0.069	weber
Number of poles	2	
Rotor moment of inertia	8.5 E-06	kg-m ²
Rotor shaft viscous damping factor	5.695 E-06	kg m ² /s

For the simulation, the `Brushless DC (BLDC/PMSM) Motor` block is used, along with the `Hall Sensor` block for commutation.

Power Electronics: Brushless PWM servo amplifier with speed and current Control. The base frequency of the PWM is 9000 Hz.

For the simulation, the `PWM Brushless Servo Amplifier` block is used

Precision current sense resistors produce voltage that is fed into a processor. An encoder provides motor shaft position and velocity. Encoder angle measurement and phase current measurements are used to obtain direct and quadrature current estimates through Clarke and Park transforms. Current and speed loops are used to set stiff inner loop performance.

Mechanical Load: Precision $\lambda/4$ flat oval mirrors mounted on a gear reducer shaft with rotation center coincident with reflecting surface represent the main load moment of inertia. A torsional spring with preload tension is used to help minimize backlash hysteresis. An optical encoder is provided with 16000 lines to measure mirror angle. PI compensation is used for controlling line of sight. Load parameters are:

Gear reduction	20:1
Backlash	0.0005 radians
Load moment of inertia	0.001 kg – m ²
Load viscous damping	0.01 kg – m ² /s

Load spring constant 0.01 N-m/rad
 Load spring preload 0.1 N-m

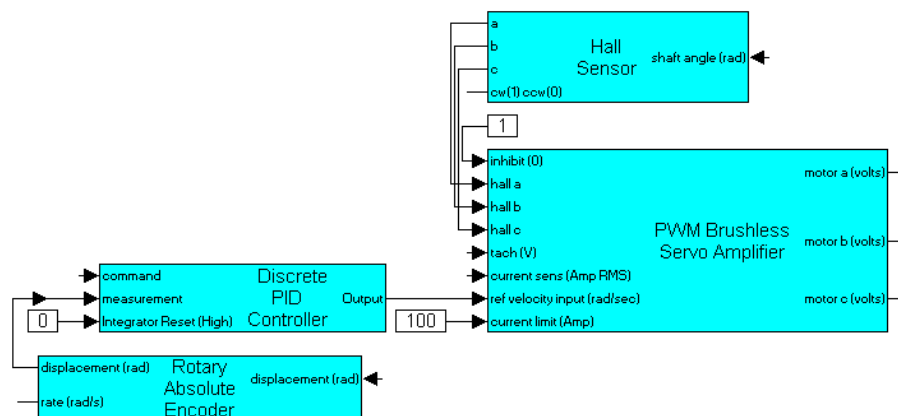
Pipeline Image Processor: Provides 60 Hz frame rate acquisition of target from focal plane array. Pixel resolution is sufficiently higher than expected control requirement of less than ± 3 degrees between target angle and line of sight in both axes. Hierarchical classification and size discrimination of blobs with subsequent calculation of the target centroid determine target position.

Simulation development

From the Motion block set, place the following blocks in your diagram:

- PID Controller (Digital)
- Hall Sensor
- PWM Brushless Servo Amplifier
- Rotary Encoder

Flip the Rotary Encoder and Hall Sensor blocks using the Edit > Flip Horizontal command. Then arrange the blocks and wire them together, as shown below.



Connect the Hall Sensor block outputs to the corresponding inputs of the Brushless PWM Servo Amplifier block. Then, wire the output of the PID Controller (Digital) block to the reference velocity input of the Brushless PWM Servo Amplifier block. Finally, connect the displacement output of the Rotary Encoder block to the measurement input of the PID Controller (Digital) block.

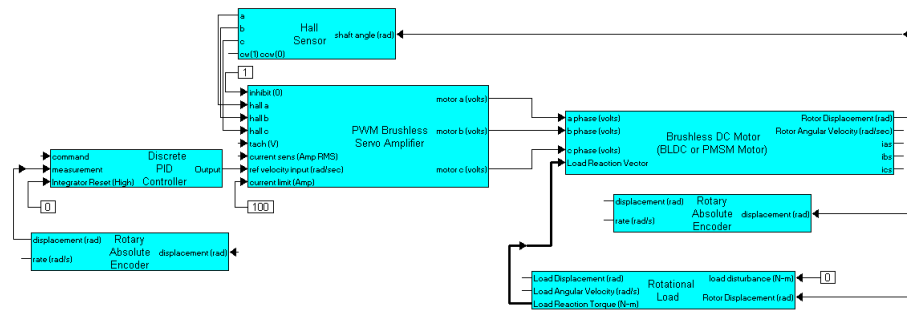
Two const blocks are fed into the Brushless PWM Servo Amplifier block; another const block is fed into the PID Controller (Digital) block:

- Set the value of the const block, wired into inhibit, to 1. This prevents inhibit.
- In this particular application there is no reason to reset the integration of the PID Controller (Digital), so a 0 const is wired to Integrator Reset (High) to disable. In other applications repetitive control may be used, and Integrator Reset (High) may be required to re-initialize the control between repetitions.
- A value of 100 amps is chosen for this example to make certain saturation does not occur. Later on, you might possibly measure currents encountered in this simulation under highest load conditions and set a more appropriate current limit for the final design.

Next, place the following Motion blocks in the diagram:

- Brushless DC (BLDC/PMSM) Motor
- Rotary Encoder
- Rotational Load

Flip the Rotary Encoder and Rotational Load blocks using the Edit > Flip Horizontal command. Then arrange the blocks to the right of the previous construction, as shown in the following diagram:

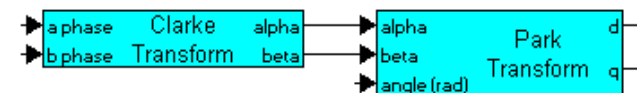


The rotor output displacement of the Brushless DC (BLDC/PMSM) Motor block connects to three other block connections: the displacement input of the Hall Sensor block, the input of the Rotary Encoder, and the rotary displacement input of the Rotational Load block.

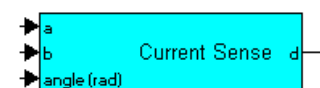
Connect the outputs of the PWM Brushless Servo Amplifier block to the corresponding inputs of the Brushless DC (BLDC/PMSM) Motor block. Connect the load reaction torque vector output connector on the Rotational Load block to the load reaction vector input of the Brushless DC (BLDC/PMSM) Motor block.

Lastly connect a const block with 0 set value to the load disturbance input connector on the Rotational Load block. If there were other torques related to influences that could not be directly represented by the set parameters of the rotary load model, the load disturbance input provides a method for introducing such torques. For the target tracker, it might be conceivable to introduce torque noise induced by structural vibrations of the tracker mount. If the mount were part of a satellite payload, such vibrations could arise from solar array positioning systems. Noise profiles with specific power spectral densities can be generated in VisSim using the Random Generator blocks and transferFunction block. Coefficients of the transfer function are determined by applying spectral factorization techniques to the known PSD.

Next, insert a Park Transform and a Clarke Transform block into the diagram and connect them as shown below:



Encapsulate the blocks in a compound block and name it Current Sense. Then label the input and output connector tabs as shown below:



Flip the block 180° and connect the ias and ibs output connectors of the Brushless DC (BLDC/PMSM) Motor block to the corresponding a and b inputs of the Current Sense

PWM Brushless Servo Amplifier block

PWM Brushless Servo Amplifier Properties	
Supply Voltage (V):	28
PWM frequency (Hz):	9000
Velocity Loop Integral Gain (A/rad):	100
Velocity Loop Proportional Gain (A-s/rad):	3
Current Loop Integral Gain (%mod/A-s):	0
Current Loop Proportional Gain (%Mod/Amp):	2.5
Transconductance Gain (Amp/Volt):	.4
Tachometer Sensitivity (V/rad/s):	1
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

Rotary Encoder block that feeds back to PID Controller (Digital) block

Rotary Absolute Encoder Properties	
Rate Estimator Poles (Hertz):	800
Processor Clock (Hertz):	10000
Resolution (lines):	16000
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

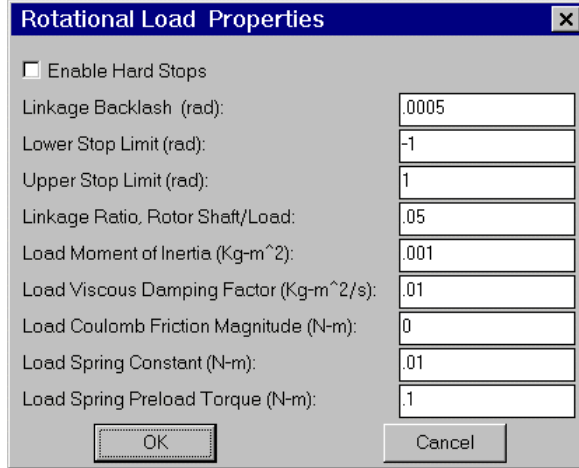
Rotary Encoder block that feeds back into the PWM Brushless Servo Amplifier block

Rotary Absolute Encoder Properties	
Rate Estimator Poles (Hertz):	800
Processor Clock (Hertz):	10000
Resolution (lines):	4000
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

Brushless DC (BLDC/PMSM) Motor block

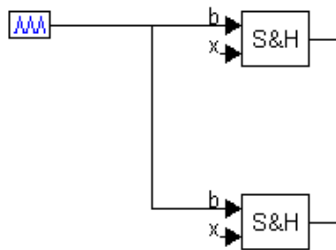
Brushless DC Motor (BLDC or PMSM Motor) Proper...	
Magnetizing Inductance (H):	.0009
Stator Winding Self Inductance (H):	.001
Stator Winding Resistance (ohms):	.5
Magnet Flux linkage Magnitude (Webers):	.069
Number of Poles:	2
Rotor Moment of Inertia (Kg-m ²):	8.5e-6
Rotor Shaft Viscous Damping Factor (Kg-m ² /s):	5.695e-6
Rotor Shaft Stiction Factor (N-m):	0
Rotor Shaft Coulomb Friction Magnitude (N-m):	0
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

Rotational Load block

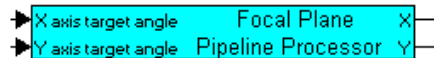


This completes the X-axis of the servo controller. Completing the Y-axis takes only a couple of keystrokes, as all dynamics for this axis are assumed to be equal. Make a duplicate copy of the X Axis Servo block using the Edit > Copy command. In the dialog box for the newly-created X Axis Servo, change the name of the block to Y Axis Servo. At this point, there are two servo controllers in your diagram: an X-axis and a Y-axis servo controller.

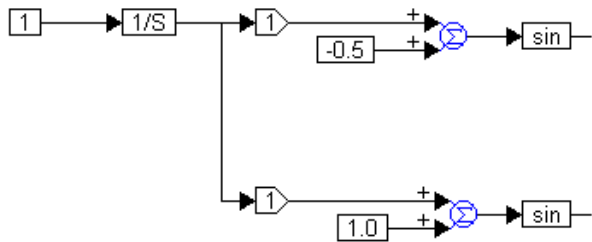
Next, create a simulation of the pipeline image processor. For this processor, the dominant feature is the sample frame rate of 60 hertz. Place two sampleHold blocks (located under Blocks > Nonlinear) and a pulseTrain block (located under Blocks > Signal Producer) in your diagram. Arrange these three blocks as shown below:



In the pulseTrain block, set the time between pulses to 1/60 (0.0167). Then encapsulate the three blocks in a compound block named Focal Plane Pipeline Processor.



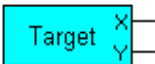
Next, create the following block configuration:



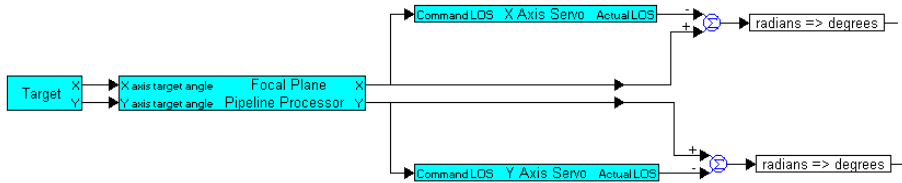
This construction is used to create an elliptical motion for the target in the X-Y plane. Frequency for each axis is the same (1 rad/sec); however, phase differs.

Note that the `integrator` (1/S) and `sin` blocks are located under `Blocks > Integration` and `Blocks > Transcendentals`, respectively.

Enclose this construction in a compound block and name it `Target`:



Connect the compound blocks as shown in the following diagram:

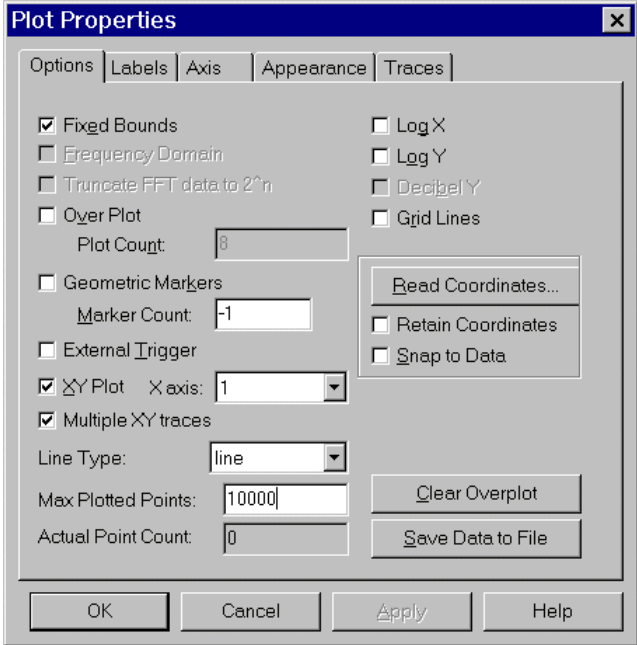


In this construction, command line of sight (LOS) is set to the target angle, which is determined by the pipeline processor. The difference between the target angle and actual line of sight is calculated using `summing junction` blocks, which provide focal plane error. The error is converted into degrees by `unitConversion` blocks set to convert radians to degrees.

Setting up the plot blocks

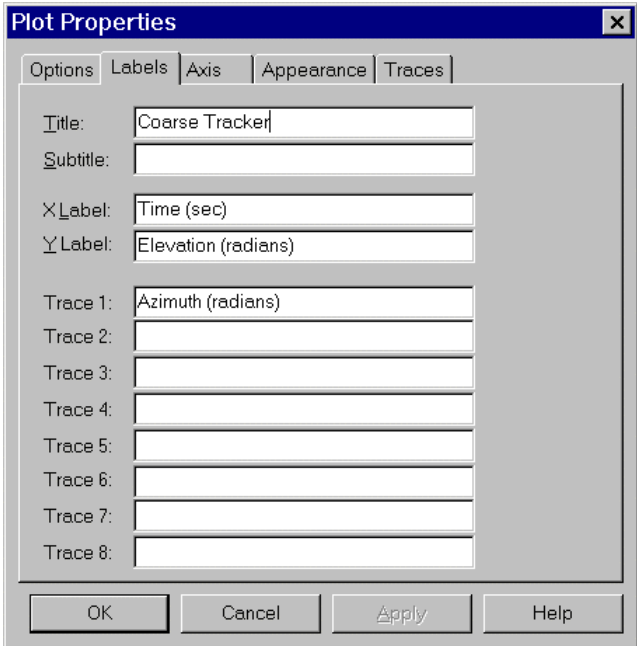
Next, plot blocks are prepared to display simulation output.

Place a plot block (located under Blocks > Signal Consumer) in the diagram; then open its dialog box and under the Options tab enter the following settings:

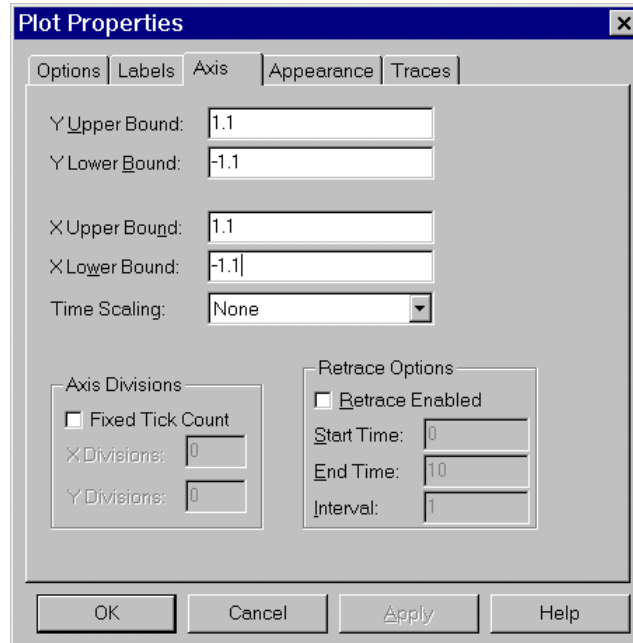


Note that the Multiple XY Traces option is activated. This feature allows the display of the target motion independently from the servo line of sight.

Under the Labels tab, enter the information shown below.



And under the Axis tab, enter the following information:

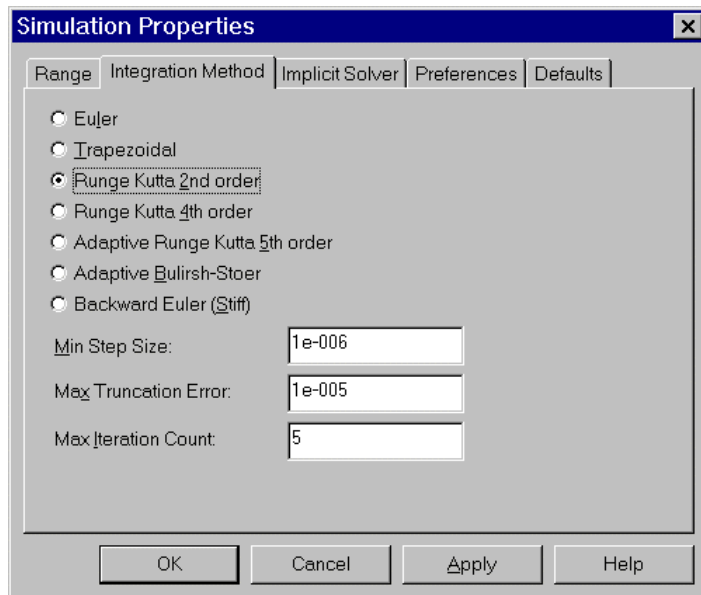
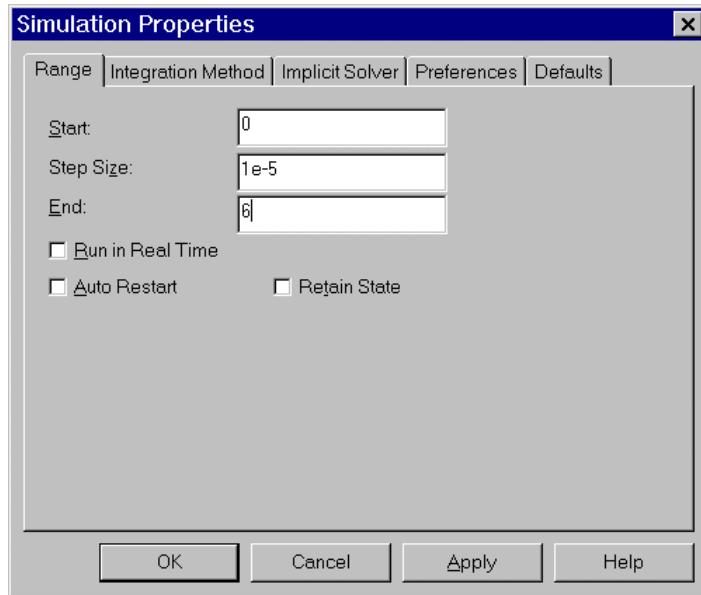


Place another plot block on the diagram. Enter the same parameters as in the previous plot with these exceptions:

- Under the Labels tab, make these changes:
 - In the Title box, enter Focal Plane
 - Enter degrees as units instead of radians
- Under the Options tab, make these changes:
 - Activate the Fixed Bounds option
 - De-activate the Multiple XY Traces option
- Under the Axis tab, make these changes:
 - In the X Upper Bound and Y Upper Bound boxes, enter 5
 - In the X Lower Bound and Y Lower Bound boxes, enter -5.

Setting the simulation properties

Simulation properties are set through the Simulate > Simulation Properties command. Enter the following information to the Simulation Properties dialog box.



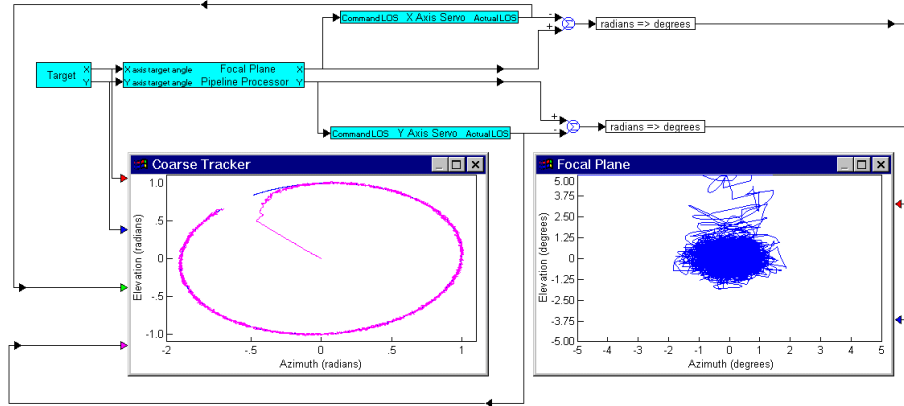
For this simulation, a very small step size is necessary because pulse width modulation is being simulated at 9000 hertz.

Final configuration requirements

Connect the X and Y outputs of the Target compound block to the first two input tabs of the Coarse Tracker plot block and the output X and Y servo compound blocks actual line of sights to the next two input tabs of the same plot block.

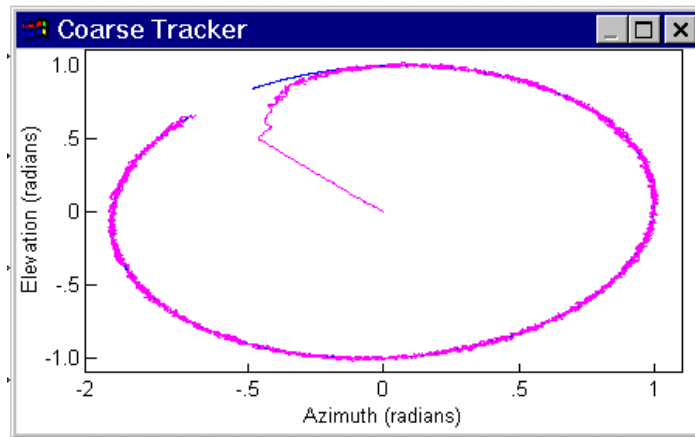
Connect the output of the two unitConversion blocks to the first two input connectors of the Focal Plane plot block.

You are now ready to run the simulation with the Simulate > Go command.

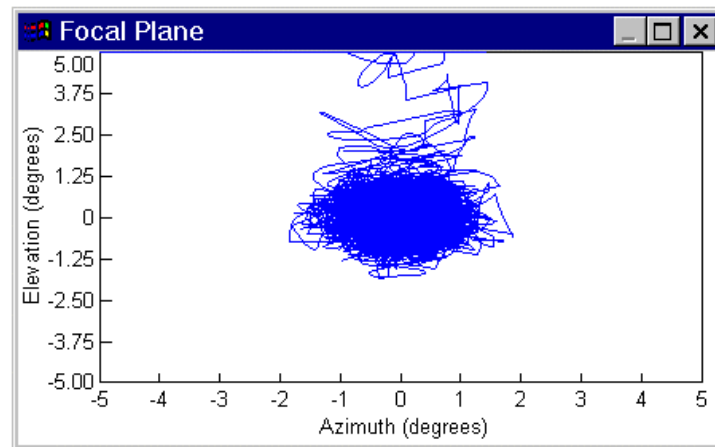


Simulation results

The following plot shows the acquisition and tracking of the actual target's elliptical motion with the servo line of sight:



To better illustrate accuracy, the following plot shows the focal plane error. The darkened circular area represents the time after the servo has acquired the target and begins tracking. These results show errors to be on the order of 1° , exceeding the requirement.



It should be noted that to get to this level of control required tuning of each of the control loops with multiple iterations before an acceptable control was achieved.

Other applications

The process of building the simulation for a tracking system can similarly be applied to a tape drive speed and tension control system. Here the design could start with a permanent magnet DC synchronous motor model providing necessary torque to rapidly accelerate a tape spindle to a desired speed. A model could be developed that provides simulated track positioning information for indexing tape position and controlling start and stop profiling.

Other things you can do with VisSim/Motion

This chapter demonstrated one specific simulation example among many possible control applications using permanent magnet synchronous motors. The Motion block set, together with the basic block set in VisSim, provide all the necessary elements to simulate motion control applications in a variety of disciplines, including:

- Aerospace
- Automotive
- Defense
- Factory automation
- Industrial robotics
- Medical instrumentation and surgical tools
- Office automation and computer peripheral drives
- Optics
- Semiconductor pick-and-place machines