Our current setup: Windows 7, MATLAB R2011B, Quanser Q4 terminal board
Hardware setup
Win 7 computer

Analog inputs (not used for us?)

Analog outputs

Encoders

Motor-cart

Springy-cart

Motor-cart

Springy-cart
How to build a simple Simulink model to read the encoders
Welcome to mex -setup. This utility will help you set up a default compiler. For a list of supported compilers, see http://www.mathworks.com/support/compilers/R2011b/win64.html

Please choose your compiler for building MEX-files:
Would you like mex to locate installed compilers [y/n]?

Select a compiler:
[0] None

Compiler: 1

Please verify your choices:
Compiler: Microsoft Visual C++ 2010
Location: c:\Program Files (x86)\Microsoft Visual Studio 10.0

Are these correct [y/n]?  

Warning: MEX-files generated using Microsoft Visual C++ 2010 require that Microsoft Visual Studio 2010 run-time libraries be available on the computer they are run on.
If you plan to redistribute your MEX-files to other MATLAB users, be sure that they have the run-time libraries.

Trying to update options file: C:\Users\jpeerson\AppData\Roaming\MathWorks\MATLAB\R2011b\mexopts.bat
From template: C:\ProgramData\1\MATLAB\R2011b\bin\win64\mexopts\mexv64opt.bat

Done . . .

Warning: The MATLAB C and Fortran API has changed to support MATLAB variables with more than 2^32-1 elements. In the near future you will be required to update your code to utilize the new API. You can find more information about this at:
http://www.mathworks.com/support/answers/en/data/1-5C7TR9?solution=1-5C7TR9
Building with the -largeArrayDims option enables the new API.
This model plots the encoders for 10 seconds.

HW config:
- power the car's wheel encoder is plugged into encoder 0
- springy sort is plugged into encoder 1
- don't even need univ. pwr. module powered to see the encoder's values.
- Moving the encoder about 0 in results in encoder counts of ~5000

The trick: had to enable scopes for logging:
Tools -> External mode control panel -> signals & triggering -> ensure all scopes are "on" (not off) by selecting them and clicking "on" radio button

HIL Initialize
HIL Set Property
HIL Set Encoder
HIL Read Encoder
Scope1
Step
Trigger scope
HIL Initialize HIL-1 (q4.0)
This model plots the encoders for 10 seconds.

HW config:
- powered-cart's wheel encoder is plugged into encoder 0
- springy seat is plugged into encoder 1
- don't even need unix, pwm module powered to see the encoder's values.
- Moving the encoder about 5in results in encoder counts of ~5000

The trick: Had to enable scopes for logging.
Tools -> “external mode control panel” -> signals & triggering -> ensure all scopes are “on” (not off) by selecting them and clicking “on” radio button

Source Block Parameters: HIL Initialize

**HIL Initialize**
Initializes a hardware-in-the-loop card.

**Navigation**
- Goto HIL blocks using this board...

### Main Tab
- **Hardware clocks:** [0:1]
- **Clock mode:** (0=timebase, 1=PWM output, 2=encoder input) [0 0]
- **Initial clock frequencies (specify zero to disable a clock initialy):** [ ]

- **Set clock parameters at model start**
- **Set clock parameters when switching to this model**
- **Set initial clock frequencies at model start**
- **Set initial clock frequencies when switching to this model**

### Clocks Tab
- **Analog input channels:** [0:3]
- **Analog input maximums:**
  - 10
- **Analog input minimums:**
  - -10

- **Set analog input parameters at model start**
- **Set analog input parameters when switching to this model**
Source Block Parameters: HIL Initialize

**HIL Initialize**
Initializes a hardware-in-the-loop card.

**Navigation**
Goto HIL blocks using this board...

### Main
- **Analog output channels:** [0:3]
- **Analog output maximums:** [10]
- **Analog output minimums:** [-10]
- **Analog outputs on watchdog expiry:** 0
- Set analog output parameters at model start
- Set analog output parameters when switching to this model
- Set initial analog outputs at model start
- Set initial analog outputs when switching to this model
- Set final analog outputs at model termination
- Set final analog outputs when switching from this model
- Set the analog outputs when a watchdog timer expires

### Digital Inputs
- **Digital input channels:** 0

### Other Outputs
- Encoder inputs
- PWM Outputs
- Other Outputs
Source Block Parameters: HIL Initialize

**HIL Initialize**
 Initializes a hardware-in-the-loop card.

**Navigation**
Go to HIL blocks using this board...

**Main**
- **PWM Outputs**
  - PWM output channels: [ ]
  - Mode: (0=duty cycle, 1=frequency, 2=period, 3=one-shot, 4=time, 5=encode)
  - Frequency in Hz (mode 0, 3 and 4) or duty cycle (modes 1 and 2):
    - [1/60s-9/1024]
  - Configuration: (0=unipolar, 1=bipolar, 2=paired, 3=complementary)
  - Alignment: (0=leading-edge, 1=trailing-edge, 2=center-aligned)
  - Polarity: (0=active low, 1=active high)
  - Leading-edge deadband (seconds): [ ]
  - Trailing-edge deadband (seconds): [ ]
  - Initial PWM outputs: [0]
  - Final PWM outputs: [0]

**Other Outputs**
- Other output channels: [ ]
- Initial other outputs:
- Final other outputs:
- Other outputs on watchdog expiry:
  - Set initial other outputs at model start
  - Set initial other outputs when switching to this model
  - Set final other outputs at model termination
  - Set final other outputs when switching from this model
  - Set the other outputs when a watchdog timer expires
Configuration Parameters: ece147c_lab0_v2/Configuration (Active)

Select:
- Solver
- Data Import/Export
- Optimization
- Diagnostics
- Hardware Implementation
- Model Referencing
- Simulation Target
- Code Generation
- Report
- Comments
- Symbols
- Custom Code
- Debug
- Interface

QUARC

- MAT-file logging
- Stop model if an overrun occurs
- Allow use of fast system timer (experimental)
- Allow console output (printing to stdout)
- Support dynamic reconfiguration
- Show completion times
- Debug version
- Enable heap checking (debug version only)

Assertions: Stop model with an error

Minimum thread priority: 2

Stack size: 0

Model affinity: []
This model plots the encoders for 10 seconds.

HW config:
- powered-carts wheel encoder is plugged into encoder 0
- springy cart is plugged into encoder 1
- don't even need uni. pwr. module powered to see the encoder's values.
- Moving the encoder about 8in results in encoder counts of ~5000

The trick: Had to enable scopes for logging.
Tools -> "external mode control panel" -> signals & triggering ->
ensure all scopes are "on" (not off) by selecting them and clicking "on" radio button

```
HIL Read Encoder
HIL Read Encoder (HIL-1)
Step

Quarc
```

HIL Initialize HIL-1 (r+c)
>> vray, no errors !
This model plots the encoders for 10 seconds.

HW config:
- powered-cart's wheel encoder is plugged into encoder 0
- springy cart is plugged into encoder 1
- don't even need univers. pow. module powered to see the encoder's values.
- Moving the encoder about 6in results in encoder counts of ~5000

The trick: Had to enable scopes for logging.
Tools -> "external mode control panel" -> signals & triggering -> ensure all scopes are "on" (not off) by selecting them and clicking "on" radio button

```
<table>
<thead>
<tr>
<th>HIL Read Encoder</th>
<th>Scope1</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIL Read Encoder (HL-1)</td>
<td></td>
</tr>
</tbody>
</table>

Step  trigger scope
```

```
<table>
<thead>
<tr>
<th>HIL Read Encoder</th>
<th>Scope1</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIL Read Encoder (HL-1)</td>
<td></td>
</tr>
</tbody>
</table>

Step  trigger scope
```

```
<table>
<thead>
<tr>
<th>Ready</th>
<th>100%</th>
<th>T=0.00</th>
<th>FredStepDiscrete</th>
</tr>
</thead>
</table>

```

```
<table>
<thead>
<tr>
<th>External</th>
<th>100%</th>
<th>T=0.00</th>
<th>FredStepDiscrete</th>
</tr>
</thead>
</table>
```
Wiggle motor-cart's wheel encoder

Wiggle springy-cart's wheel encoder

Note: doesn't show 3 encoders??
>> who
Name         Size      Bytes   Class      Attributes
ScopeData    9001x2    144016  double
ScopeData1   9001x5    360040  double

>> plot(ScopeData1)
Next: move motor
Move cart slightly

This model sends a sine wave command to the motor.

HW config:
- powered-cart's wheel encoder is plugged into encoder 0
- spring cart is plugged into encoder 1
- don't even need univ. pwr. module powered to see the encoder's values.
- Moving the encoder about 6in produces in encoder counts of ~5000
- Sin amplitude is 2, freq is 2*pi. Results in very small 3” wiggles

The trick. Had to enable scopes for logging.
Tools -> "external mode control panel" -> signals & triggering ->
ensure all scopes are "on" (not off) by selecting them and clicking "on" radio button

Note: this is also reqd to add a new scope
Move cart slightly

This model sends a sine wave command to the motor.

HW config:
-powered cart, wheel encoder is plugged into encoder 0
-encoder cart is plugged into encoder 1
-Don't even need uni. pair; encoder powered to sense the encoder's values.
-Model encoder about 617 results in encoder counts of -558
-5x amplitude is 2, freq is 2 Hz. Results in very small 0 in wiggle

The trick:Had to enable access for logging

Tools - external mode control panel - sign & triggering
-ensure all scopes are 'on' (not off) by selecting them and clicking 'on' radio button

Note: this is also read to add a new scope
Troubleshooting:
- is Universal power module on?
- Is analog output cable connecting analog output 0 to UPM's "From D/A"?
- Is cable from UPM's "To Load" to motor-cart's motor plug?
- Is simulink "HIL Write Analog" block set to same channel as the board's analog output?