Two-story system bond graph.

\[ \begin{align*}
\dot{p}_2 &= m_2 \ddot{v}_2 \\
F_{R2} &= C \cdot \ddot{v}_2 \\
F_{b2} &= -b_2 \dot{v}_2 \\
p_1 &= m_1 \dot{v}_1 \\
p_2 &= m_2 \dot{v}_2 \\
\end{align*} \]
State equation derivation

\[ \dot{x}_{k_1} = v_1 \checkmark \]
\[ \dot{v}_1 = -\frac{1}{m_1} F_{k_1} - \frac{1}{m_1} F_{b_1} + \frac{1}{m_1} F_{k_2} + \frac{1}{m_1} F_{b_2} \]
\[ \dot{v}_2 = -\frac{1}{m_2} F_{k_2} = -\frac{1}{m_2} F_{b_2} \]

\[ F_{k_1} = k_1 x_{k_1} \]
\[ F_{k_2} = k_2 x_{k_2} \]
\[ F_{b_1} = b_1 v_1 \]
\[ F_{b_2} = b_2 (v_2 - v_1) \]

\[ \dot{v}_1 = -\frac{k_1}{m_1} x_{k_1} - \frac{b_1}{m_1} v_1 + \frac{k_2}{m_1} x_{k_2} + \frac{b_2}{m_1} v_2 - \frac{b_2}{m_1} v_1 \]
\[ = -\frac{k_1}{m_1} x_{k_1} \left(1 - \frac{b_2}{m_1}\right) v_1 + \frac{k_2}{m_1} x_{k_2} + \frac{b_2}{m_1} v_2 \checkmark \]

\[ \dot{v}_2 = -\frac{k_2}{m_2} x_{k_2} - \frac{b_2}{m_2} (v_2 - v_1) \]
\[ = -\frac{k_2}{m_2} x_{k_2} + \frac{b_2}{m_2} v_1 - \frac{b_2}{m_2} v_2 \checkmark \]
\[
\begin{align*}
\dot{\mathbf{x}} &= \begin{bmatrix} \dot{x}_1 \\ \dot{v}_1 \\ \dot{x}_2 \\ \dot{v}_2 \end{bmatrix} \\
\mathbf{x} &= \begin{bmatrix} x_1 \\ v_1 \\ x_2 \\ v_2 \end{bmatrix} \\
\mathbf{y} &= \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \\
A &= \begin{bmatrix} -k_1/m_1 & -(b_1+b_2)/m_1 & k_2/m_1 & b_2/m_1 \\ 0 & -1 & 0 & 1 \\ 0 & b_2/m_2 & -k_2/m_2 & -b_2/m_2 \end{bmatrix} \\
B &= \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\
C &= \begin{bmatrix} -k_1/m_1 & -(b_1+b_2)/m_1 & k_2/m_1 & b_2/m_1 \\ 0 & 0 & b_2/m_2 & -k_2/m_2 \end{bmatrix} \\
D &= \begin{bmatrix} 0 \\ 0 \end{bmatrix}
\end{align*}
\]

**NOTE CORRECTION:** To make the outputs actually in 'g', define a variable as 'g', give it a value of 9.817, and then divide each term in the C matrix by g.
NI_LinSys_Model Creation.lvlib:CD Construct State-Space Model.vi

Sampling Time (s)
Symbolic A
Symbolic B
Symbolic C
Symbolic D
error in (no error)
Variables

Creates a deterministic state-space representation of a system using the matrices A, B, C, and D, and the Sampling Time (s). You must manually select the polymorphic instance to use.
Calculates the natural or zero-input response of the system. This VI returns the output of the system due to the initial states only. Wire data to the State-Space Model input to determine the polymorphic instance to use or manually select the instance.

**Initial Response Graph** returns an XY graph that shows the unforced response of the system when the initial conditions are X0.

**State Trajectory Graph** is an XY graph that shows the value of each state as a function of time. This VI computes the magnitude by solving the first order difference equation of the system.

**Initial Response Data** returns information about the initial response. To access the Initial Response Data, use the CD Get Time Response Data VI. When you use the CD Get Time Response Data VI to access the Initial Response Data, you do not need to specify an input value.

- **Time** is the uniformly-spaced time vector against which this VI plots the initial response and the state trajectories.
- **Outputs Data** returns data about the time response of the outputs to the inputs.
- **States Data** returns data about the time response of the states to the inputs. For transfer function and zero-pole-gain models, this array is empty.
**Get Time Response Data (Input-Output Pair)**

**Time Response Data** contains information about the time response of a model. Refer to the [Details](#) section for more information about the time response data.

- **Time** is the uniformly-spaced time vector against which this VI plots the impulse, initial, or step response and the state trajectories.
- **Outputs Data** contains data about the time response of the outputs to the inputs.
- **States Data** contains data about the time response of the states to the inputs. For transfer function and zero-pole-gain models, this array is empty.

**Input** determines the index number of each input for which you want to obtain data.

**Output** determines the index number of each output for which you want to obtain data.

**error in** describes error conditions that occur before this node runs. This input provides standard error in functionality.

**Type of Response Data** specifies the type of response data you obtain.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Outputs (default) — Specifies to obtain the time response of the input-output pair(s).</td>
</tr>
<tr>
<td>1</td>
<td>State — Specifies to obtain the time response of the input-state pair(s) for a state-space model. If you specify this <strong>Type of Response Data</strong> for a transfer function or zero-pole-gain model, this VI returns the time response data as an empty array.</td>
</tr>
</tbody>
</table>

**Time** returns the time at which this VI generates the response data. **Time** is the uniformly-spaced time vector against which this VI plots the impulse, initial, or step response and the state trajectories.

**Response Data** returns a vector that provides the time response data for the input-output pair you select.
NI_MAP.ro.lvlib:Extract Multiple Tone Information.vi

time signal in
export mode
max num tones
error in (no error)
output sorting

exported signals
multiple tone information
error out

Returns the frequency, amplitude, and phase for each signal tone whose amplitude exceeds a specified threshold. Wire data to the time signal in input to determine the polymorphic instance to use or manually select the instance.

**multiple tone information** returns the frequency, amplitude, and phase of each tone that this VI extracts. Each element of the array is one single tone.

- **detected frequency** is the frequency of the detected single tone in Hz.
- **detected amplitude** is the peak amplitude of the detected single tone.
- **detected phase** is the phase of the detected single tone in degrees.

Use the Wrap Angle VI to change the units of detected phase. Wire detected phase to the angle in input on the Wrap Angle VI, and select degree in, degree out or degree in, radians out for the angle units input.

Build Waveform

waveform
waveform component

Builds an analog waveform or modifies an existing waveform. If you do not wire the waveform input, the function creates a new waveform based on the components you wire. If you wire the waveform input, the function modifies the waveform based on the components you wire.
See correction on previous slide. These accelerations are in m/s² not 'g'.