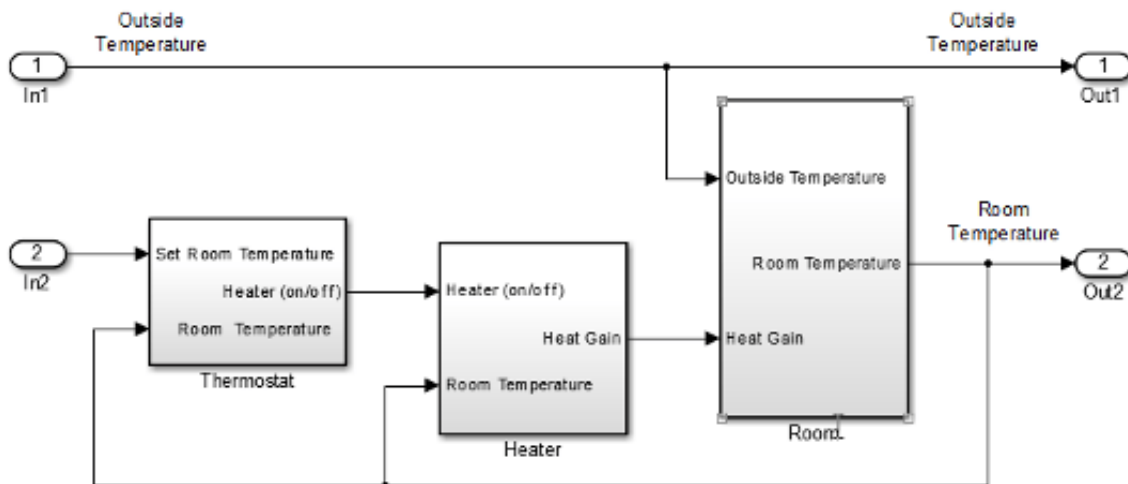


Model A House Heating System

This tutorial shows how to model and simulate a dynamic system using Simulink® software. The model is for a heating system that includes a heater (plant model), controlled by a thermostat (controller model), to heat a room (environment model) to a set temperature. While this is simple model, the processes for creating model structure and algorithm design are the same processes you will use for more complex models.

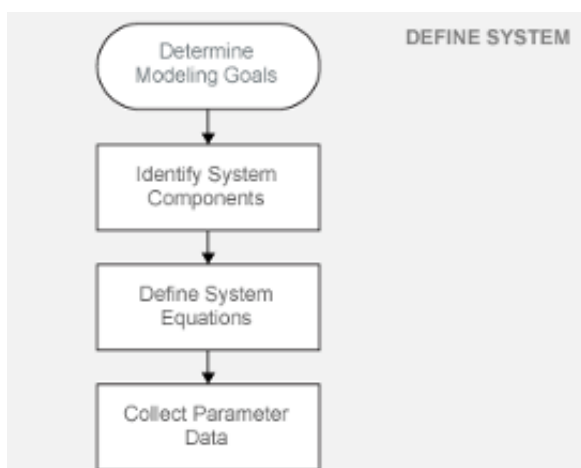
To review a completed model, in the MATLAB® Command Window, enter

```
open_system(fullfile(matlabroot,...
    'help', 'toolbox', 'simulink', 'examples', 'ex_househeat_modeling'))
```



Define a House Heating System

Modeling begins with completion of tasks that are outside of the Simulink software environment. Define model requirements and derive mathematical equations. Collect data for model parameters and output signal data measurements to validate simulation results.



Determine Modeling Goals

Before designing a model, consider your goals and requirements. The goals for modeling the house heating system are:

- Observe how the changing outdoor temperature affects the indoor temperature.
- Examine the effect of changing parameters on the indoor temperature.

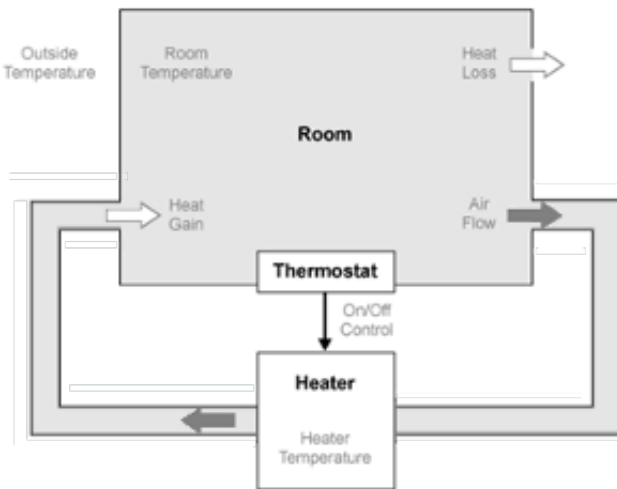
Identify System Components

Once you understand your modeling requirements, you can begin to identify the components of the system.

The house heating system in this tutorial defines a heating system and its relationship to a room. It includes:

- Thermal characteristics of a house
- Thermal characteristics of a heater
- A thermostat to control the heater
- Outdoor environment
- Indoor environment

The thermostat monitors the room temperature regularly and turns the heater on or off, depending on the difference between the set temperature and the room temperature.



The model for this system includes three components: heater, thermostat, and room.

Define System Equations

Three time-dependent variables define the heat exchange in the room:

- Temperature of the room (T_{room})
- Heat gain: Thermal energy transferred from the heater (Q_{gain}) to the room
- Heat loss: Thermal energy transferred from the room (Q_{loss}) to the outdoor environment

A differential equation defines the relationship between these variables, but since heat transfer is defined in terms of changing temperature, only room temperature is a state variable.

Rate of Heat Gain Equation

The temperature of the air in the heater is constant at T_{heater} and the room temperature is T_{room} . Thermal energy gain to the room is by convection of heated air from the heater, with a heat capacity of c_{air} . Heat gain for a mass of air in the heater, $m_{heaterair}$, is proportional to the temperature difference between the heater and the room:

$$Q_{gain} = m_{heaterair} c_{air} (T_{heater} - T_{room}).$$

The rate of thermal energy gain from the heater is

$$\frac{dQ_{gain}}{dt} = \frac{dm_{heaterair}}{dt} c_{air} (T_{heater} - T_{room}).$$

A fan takes room air, and passes it through the heater and back to the room. A constant amount of air, $M_{heaterair}$, flows through the heater per unit time, and replacing $dm_{heaterair}/dt$ with that constant simplifies the equation to

$$\frac{dQ_{\text{gain}}}{dt} = M_{\text{heaterair}} c_{\text{air}} (T_{\text{heater}} - T_{\text{room}}).$$

Rate of Heat Loss Equation

Thermal energy loss from the room is by conduction through the walls and windows, and is proportional to the temperature difference between the room temperature and the outside temperature:

$$Q_{\text{loss}} = \frac{kA(T_{\text{room}} - T_{\text{outside}})t}{D}.$$

The rate of thermal energy loss is

$$\frac{dQ_{\text{loss}}}{dt} = \frac{kA(T_{\text{room}} - T_{\text{outside}})}{D}.$$

Replacing kA/D with $1/R$ where R is the thermal resistance simplifies the equation to

$$\frac{dQ_{\text{loss}}}{dt} = \frac{(T_{\text{room}} - T_{\text{outside}})}{R}.$$

Changing Room Temperature Equation

Define the rate of temperature change in the room by subtracting the rate of heat loss from the rate of heat gain:

$$\frac{dT_{\text{room}}}{dt} = \frac{1}{m_{\text{roomair}} c_{\text{air}}} \left(\frac{dQ_{\text{gain}}}{dt} - \frac{dQ_{\text{loss}}}{dt} \right).$$

Collect Parameter Data

Most of the parameter values needed for the house heating model are published in standard property tables. The flow rate for the heater is from a manufacturer data sheet.

List the variables and coefficients from your equations and check for dimensional consistency between the units. Since the unit of time for the model is hours, convert published values for the thermal property of materials from units of seconds to hours.

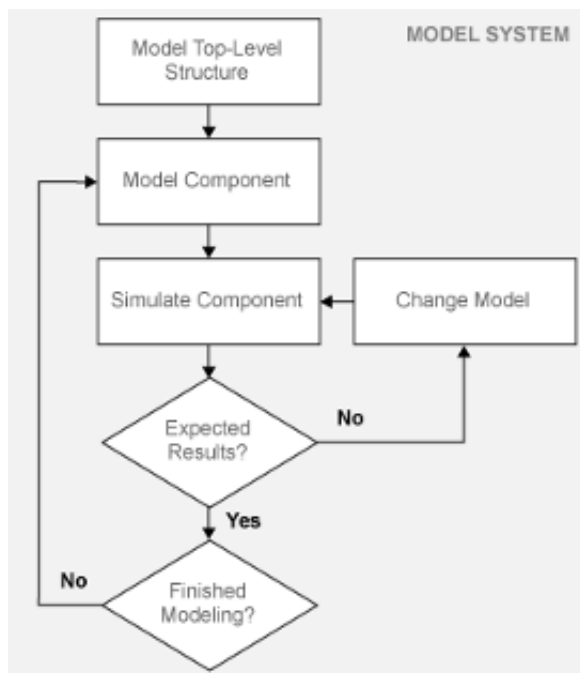
Equation Variables and Constants. You can use the constant names and values in this table when building the model.

Equation Variable or Coefficient	Description	Units
A	Area of wall or window surface $A_{\text{wall}} = 914$, $A_{\text{window}} = 6$	square meter
D	Depth of wall or window $D_{\text{wall}} = 0.2$, $D_{\text{window}} = 0.01$	meter
Q	Thermal energy transferred	joule
dQ/dt	Rate of thermal energy transferred	joule/hour
k	Thermal conductivity; property of a material to conduct heat transfer $k_{\text{fiberglass}} = 136.8$, $k_{\text{glass}} = 2808$	joule/meter·hour·degree
r	Thermal resistivity; property of a material to resist heat transfer $r = 1/k$	meter·hour·degree/joule
R	Thermal resistance $R = D/kA = (T_1 - T_2)/Q$	hour·degree/joule

	$R_{\text{wall}} = 1.599\text{e-}6, R_{\text{window}} = 5.935\text{e-}7$ $R_{\text{equivalent}} = (R_{\text{wall}} * R_{\text{window}}) / (R_{\text{wall}} + R_{\text{window}}) = 4.329\text{e-}7$	
m	Mass of air in the room or heater $m_{\text{room_air}} = 1470$ The mass of the heater $m_{\text{heater_air}}$ is not needed for this model.	kilogram
dm/dt	Rate of air mass passing through the heater	kilogram/hour
M	Constant rate of air mass passing through the heater $M_{\text{heater_air}} = 3600$	kilogram/hour
c	Specific heat capacity $c_{\text{air}} = 1005.4$	joule/kilogram· degree
T_{heater}	Constant air temperature from heater $T_{\text{heater}} = 50$	degree Celsius
T_{room}	Initial air temperature of room $T_{\text{roomIC}} = 20$	degree Celsius

Model House Heating System

Model the top-level structure with components that including interfaces for passing data between individual components. Your model should be organized into a hierarchical structure that corresponds to the components of the system.



Model Top-Level Structure

At the top level of the house heating model, use Subsystem blocks to organize your model and create the structure. The model includes the subsystems Thermostat, Heater, and Room.

1. Open a new Simulink model: [Open New Model](#).
2. Open the Library Browser: [Open Simulink Library Browser](#)
3. Add Subsystem blocks. Drag three Subsystem blocks from the Ports & Subsystems library to the new model in the Simulink Editor.



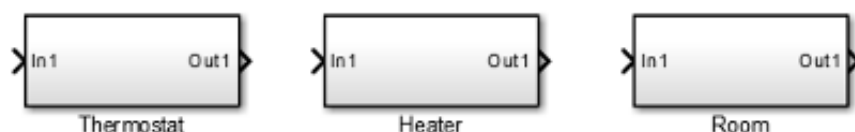
4. Open a Subsystem block. Double-click the block.



Each new Subsystem block contains one Inport (In1) and one Outport (Out1) block. These blocks define the signal interface with the next higher level in a model hierarchy.

Each Inport block creates an input port on the Subsystem block, and each Outport block creates an output port. Add more blocks for additional input and output signals.

5. On the Simulink Toolstrip, click the **Navigate Up To Parent** button  to return to the top level. Rename the Subsystem blocks as shown. Double-click a block name and type the new name.



For each component, model the equations, define parameters, prepare the subsystem for simulation, and simulate to verify its behavior.

Model Heater Component

Let's start by modeling the heater system component. The heater model:

- Takes the current temperature from the room and a control signal from the thermostat as inputs
- Calculates the heat gain from the heater
- Outputs the heat gain when the control signal is on

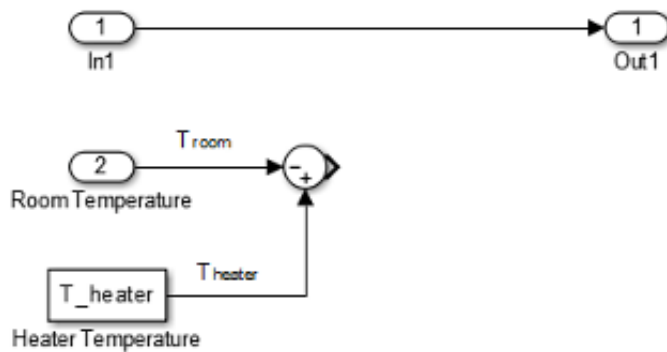
To model the heater subsystem, model the rate of heat gain equation with Simulink blocks:

$$\frac{dQ_{gain}}{dt} = M_{heaterair} c_{air} (T_{heater} - T_{room}).$$

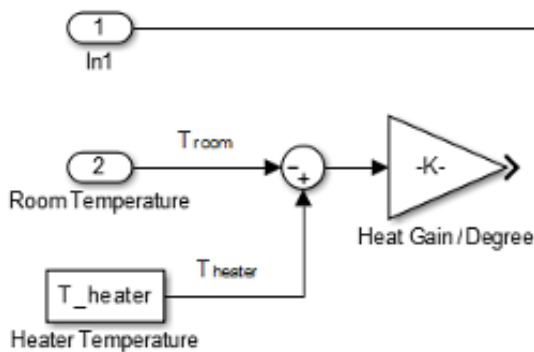
Subtract Room Air Temperature from Heater Air Temperature. The temperature difference is the current room temperature subtracted from the constant temperature of the heater (T_{heater}).

1. Open the Heater subsystem.
2. Click the model and type Sum to display a list of blocks with Sum in the name. Click the Sum block on the list. When prompted for a list of signs, type `| - +` to place - and + input ports on the block, and press **Enter**.
The vertical bar (|) changes the position of input ports by inserting spaces between the ports. A vertical bar at the beginning of the sign list, places a space at the top of the block and shifts the ports counter clockwise.
3. Add a Constant block to model the constant air temperature from the heater. Set the block **Constant value** parameter to T_{heater} . You will define the value of T_{heater} in the Model Workspace.
If the block displays $-C-$, resize the block to display the variable name.
4. Add a second Inport block to take the room temperature signal from another part of your model.
5. Add a Gain block to the Heater subsystem. Set the **Gain** parameter to $M_{heater_air} * c_{air}$. You will define the values of these variables in the Model Workspace.
6. Connect the output of the Sum block to the input of the Gain block.

7. Add labels to the signal lines to help trace model components to the equations and model requirements.
Double-click above a signal line and enter a label.



8. Rename the blocks and connect them as shown in the figure.

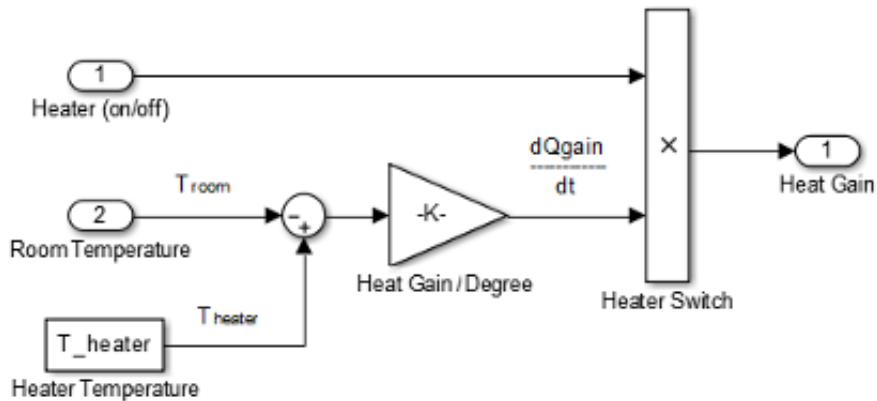


Model a Heater Switch. The thermostat sends an on/off signal equal to 1 (on) or 0 (off) to the heater. Because the input signal is binary, you can use a Product block to model a switch.

1. Remove the connection between the In1 and Out1 blocks. Select the line and press **Delete**.
2. Add a Product block. Resize the block vertically to align the block in your diagram. Connect the In1 block to the first block input and the block output to the Out1 block. Rename the blocks as shown.



3. Connect the output from the Gain block to the second input. Move all the connected blocks together. Draw a selection box around the blocks you want to move, and then drag them to the new location.
4. Rename blocks and add labels to signals as shown in the figure.



The Inport and Outport blocks create ports that connect this subsystem to other subsystems in your model.

Define Heater Model Parameters. You can define parameters in the MATLAB Workspace and then enter their names in the block parameter dialog boxes. However, a more robust method is to use the Simulink Model Workspace because variable values are saved with the model.

1. In the Simulink Editor, on the **Modeling** tab, under **Design**, click **Model Workspace**.
2. In Model Explorer, select **Add > MATLAB Variable**. In the middle pane, click the new variable Var and enter the variable name for a block parameter. For this example, enter T_heater.

Name	Value	DataType	Min	Max	Dimensions
T_heater	0	double (auto)			


3. Click the 0 value and enter the value for this variable. For this example, enter 50 degrees.

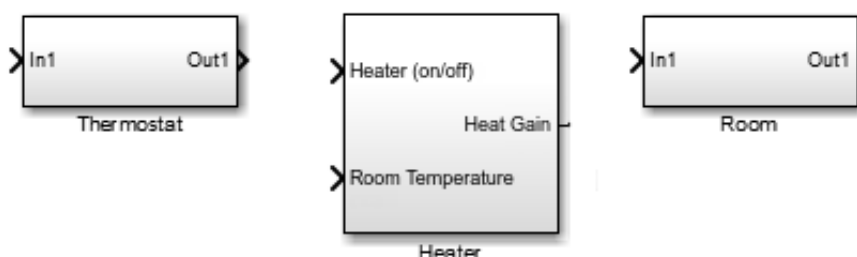
Name	Value	DataType	Min	Max	Dimensions
T_heater	50	double (auto)			

4. Using the same approach, add the variable M_heater_air with a value of 3600 kilogram/hour and c_air with a value of 1005.4 joule/kilogram· degree.

Prepare Heater Model for Simulation. Set up the heater model for simulation. Think about the expected behavior and how you can test that behavior with a simulation. When the thermostat output is 1 (on), and assuming constant room temperature of 25, the expected output from the gain is $(50 - 25) \times 3600 \times 1005.3 = 9.05 \times 10^7$. Verify this output by running the model with these inputs:

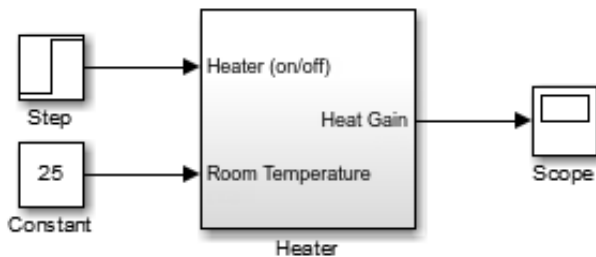
- Heater on/off signal that changes from 0 to 1 after the 4th hour
- Room temperature constant at 25

1. From the Heater subsystem, click the **Navigate Up To Parent** button  to navigate to the top level of your model. You can resize the Heater block as shown in the figure.




Notice the Heater block has a second input port and that each port corresponds to an Inport block or Outport block in the subsystem.

2. Add a Constant block to represent the room temperature, and set the value to 25 (degrees Celsius). Add a Step block for a temporary Heater (on/off) signal. Set **Step time** to 4.
3. Add a Scope block and connect it to the Heat Gain output.

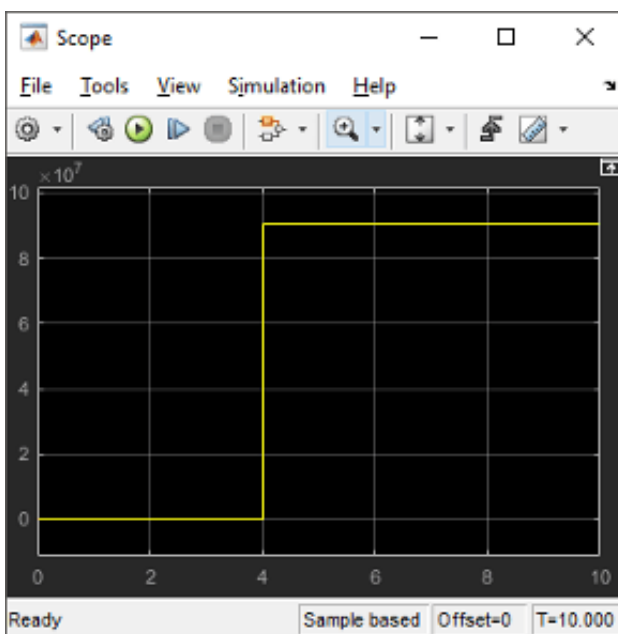


Simulate Heater Model and Evaluate Results. Use the default simulation settings to validate your model design.

1. Double-click the Scope block to open it.
2. Simulate the model. Click the **Run** button .

As the simulation runs, the Scope plots the results.

3. View the scope trace.



4. Determine if this result is what you expected.
When the heater on/off signal flips from 0 to 1 at 4 hours, the heater outputs 9.05×10^7 joule/hour. The simulation validates the expected behavior.
5. Remove Constant, Step, and Scope blocks you added for testing the Heater component.

Model Thermostat Component


You can model a thermostat without using system equations. Requirements for this component:

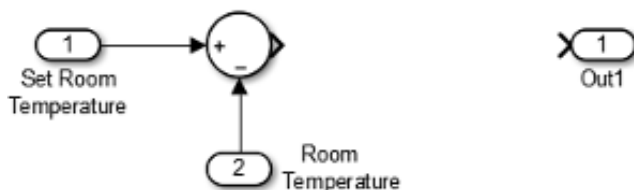
- When the room temperature is below the set temperature, heater is on and the control signal equals 1. When the room temperature is above the set temperature, the control signal equals 0.
- To avoid repeated switching around the set temperature, the thermostat allows a hysteresis of 2 degrees Celsius around the temperature set point. If the thermostat is on, the room temperature must increase 2 degrees above the set temperature before turning off. If the thermostat is off, the room temperature must drop 2 degrees below

the set temperature before turning on.

This component models the operation of a thermostat, determining when the heating system is on or off. It contains only one Relay block but logically represents the thermostat in the model.

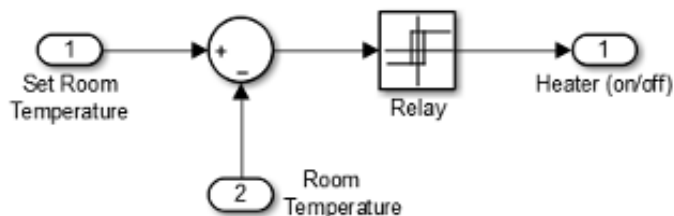
Subtract Set Room Temperature from Room Temperature. If the set room temperature is warmer than the room temperature, the thermostat model sends an “on” signal to the heater model. To determine if this is the case, begin by subtracting the room temperature from the set temperature.

1. Open the Thermostat subsystem. Add a Sum block. Set the parameter **List of signs** to $|+-$.
2. Connect the Inport block to the + input of the Sum block. The Inport block sets the room temperature.
3. Add a second Inport block and connect it to the – input of the Sum block. This second Inport block is the current room temperature from the room subsystem. Move the output port to the top of the block. Select the block. In the toolbar, on the **Format** tab, select **Rotate 90 counterclockwise** . If you want, you can reshape the block as shown in the figure by dragging the handles.
4. Rename the blocks as shown.



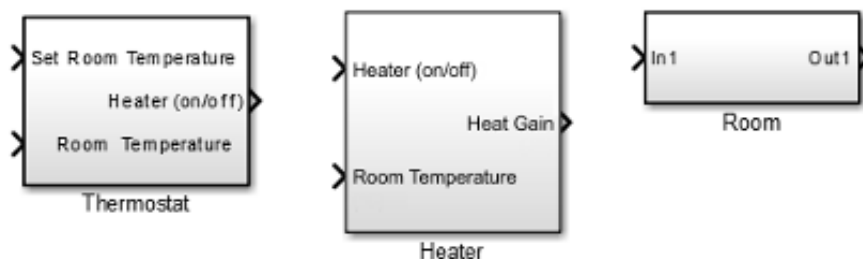
Model Thermostat Signal. Model the signal from the thermostat with a hysteresis value of 2 degrees Celsius.

1. In the Thermostat subsystem, add a Relay block. Set the **Switch on point** parameter to 2, and the **Switch off point** parameter to -2 .
2. Connect and rename the blocks as shown in the figure.



Prepare Thermostat Model for Simulation. Prepare the Thermostat subsystem for simulation. Think about the expected behavior of the thermostat and how you can test that behavior with a simulation. When the room temperature rises above the thermostat setting by 2 degrees, the thermostat output is 0. When the room temperature moves below the thermostat setting by 2 degrees, the thermostat output is 1.

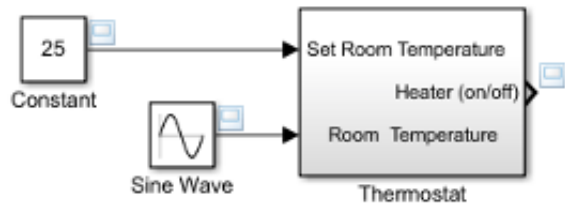
1. From the Thermostat subsystem, click the **Navigate Up To Parent** button  to navigate to the top level of your model. Resize the Thermostat block as shown in the figure.



Notice the Thermostat subsystem now has a second input port. Each input port corresponds to an Inport block in the subsystem.

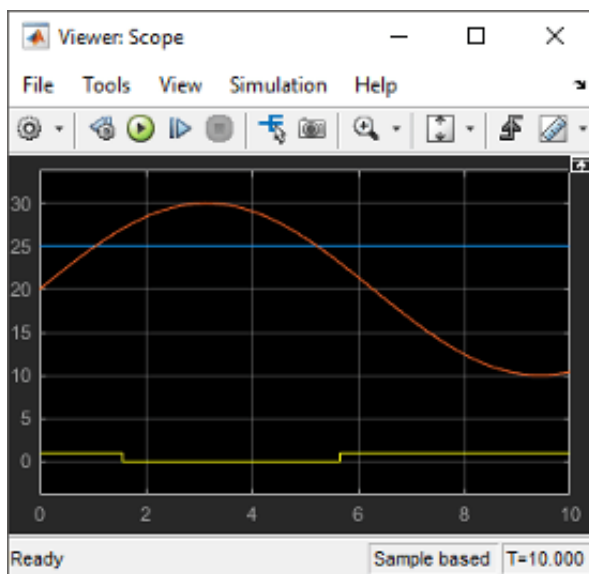
2. Add a Constant block for the set temperature. Set the **Constant** parameter to 25 (degrees Celsius).

3. Add a Sine Wave block to represent the changing room temperature. Set the **Amplitude** parameter to 10, the **Bias** to 20, and the **Frequency** to 0.5. These parameters give a variation above and below the temperature set point of 25.
4. Create and connect Scope Viewer at the Heater port. See [Add Signal Viewer](#).
5. Connect the two input signals to the Scope Viewer.



Simulate Thermostat Model and Evaluate Results. Use the default simulation settings to validate your model design.

1. Simulate the model. As the simulation runs, the Scope Viewer plots the results.
2. Open the Scope to view the scope trace.



3. Determine if this result is what you expected.

Initially the room temperature is below the set temperature and the relay is on. When the room temperature reaches the set temperature, the relay continues to be on until the room temperature increases by 2 more degrees. Simulation validates the expected behavior.

Model Room Component

Inputs to the room component are heat flow from the heater component and the external air temperature. The room component uses these inputs to compute heat loss through the walls, heat loss through the windows, and the current room temperature.

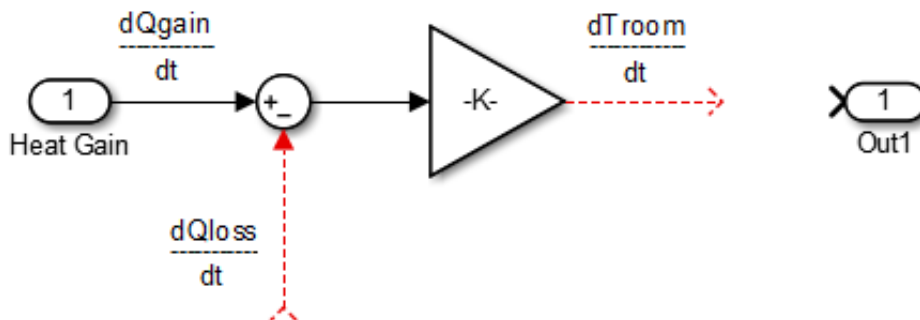
To design the room subsystem, use the Rate of Heat Loss equation and the Changing Room Temperature Equation.

Model Changing Room Temperature. The rate of temperature change in the room (dT_{room}/dt) is defined by the equation

$$\frac{dT_{room}}{dt} = \frac{1}{m_{roomair}c_{air}} \left(\frac{dQ_{gain}}{dt} - \frac{dQ_{loss}}{dt} \right)$$

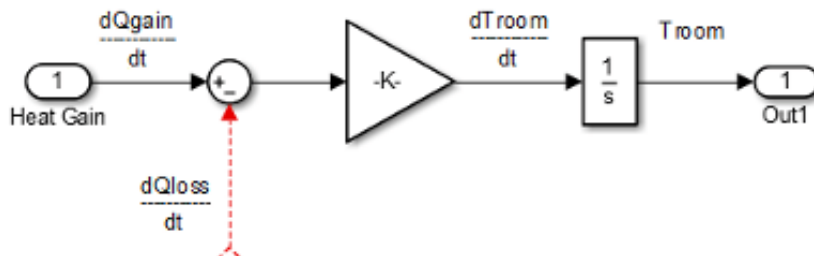
The term dQ_{gain}/dt is a signal from the Heater subsystem.

1. Open the Room subsystem block. In the Room subsystem, add a Sum block. Set the **List of signs** parameter to $|+-$.
2. Connect In1 to the + input. The input is the heat gain (dQ_{gain}/dt) from the heater component. The – input connects to the heat loss (dQ_{loss}/dt) from the room.
3. Add a Gain block. Set the **Gain** parameter to $1/(m_{room_air}*c_{air})$. Connect the output of the Sum block to the input of the Gain block. Label signals as shown in the figure. Dotted signal lines are signals you will connect later.





Model Room Temperature. The output of the Gain block is the change in room temperature (dT_{room}/dt). To get the current room temperature (T_{room}), integrate the signal.

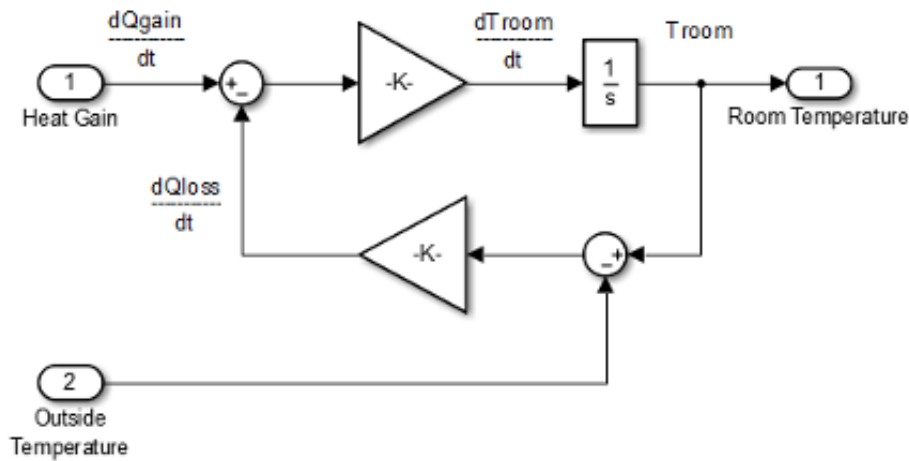
1. Add an Integrator block. Set the **Initial condition** parameter to T_{room_IC} .
2. Connect the output of the Integrator block to Out1 as shown.



Model Heat Loss Through Walls and Windows. This equation is the rate of thermal energy loss through the walls and windows:

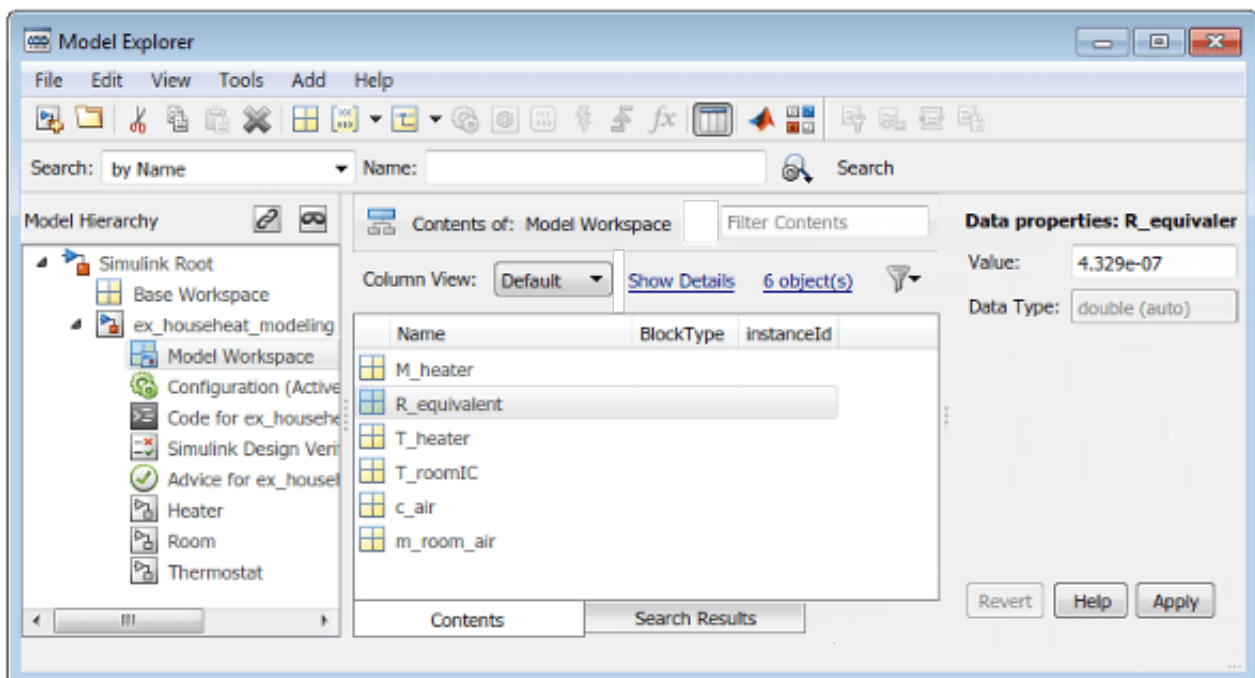
$$\frac{dQ_{loss}}{dt} = \frac{(T_{room} - T_{outside})}{R}$$

1. In the Room subsystem, add a Sum block. Set the **List of signs** parameter to $|+-$. Select the block. In the toolstrip, on the **Format** tab, click **Flip left-right** .
2. Connect T_{room} to the Sum block. Click the signal line for T_{room} and the + input on the Sum block.
3. Add another Inport block and connect it to the – input of the Sum block. Rename it to Outside Temperature.
4. Add another Gain block. Set the **Gain** parameter to $1/R_{equivalent}$. Select the block. In the toolstrip, on the **Format** tab, click **Flip left-right** .
5. Connect the blocks as shown in the figure.



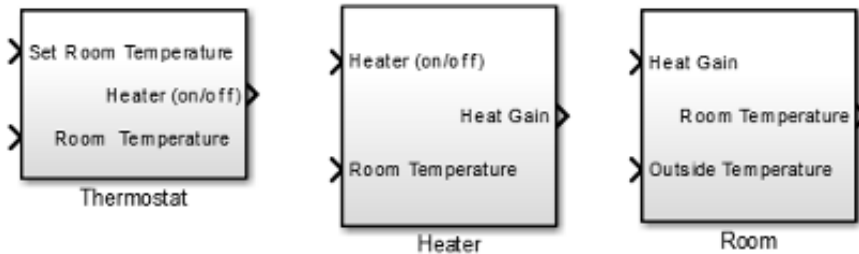
Define Room Model Parameters. You can define parameters in the MATLAB Workspace and then enter their names in the block parameter dialog boxes. However, a more robust method is to use the Simulink Model Workspace, which saves parameter values with the model.

1. In the Simulink Editor, on the **Modeling** tab, under **Design**, click **Model Workspace**.
2. In the Model Explorer, select **Add > MATLAB Variable**.
3. In the middle pane, click the new variable Var and enter the name `m_room_air`. In the right pane, enter the value 1470 (kilograms).
4. Add the variables `T_roomIC = 20` (degrees Celsius) and `R_equivalent = 4.329e-7` (hour· degree/joule).



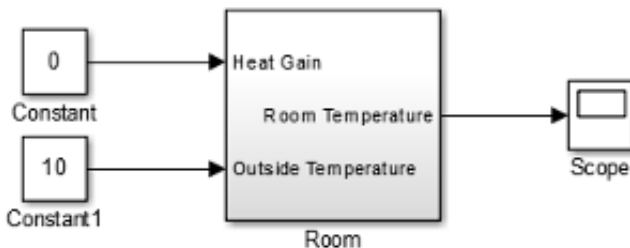
Prepare Room Model for Simulation. Prepare the Room subsystem for simulation. Think about the expected behavior and how you can test that behavior with a simulation. When the heater is off (Heat Gain = 0) and the initial temperature of the room (20) is above the outside temperature (10), heat loss should continue until the room temperature is equal to the outside temperature.

1. From the Room subsystem, click the **Navigate Up To Parent** button  to navigate to the top level of your model. Resize the Room block as shown in the figure.




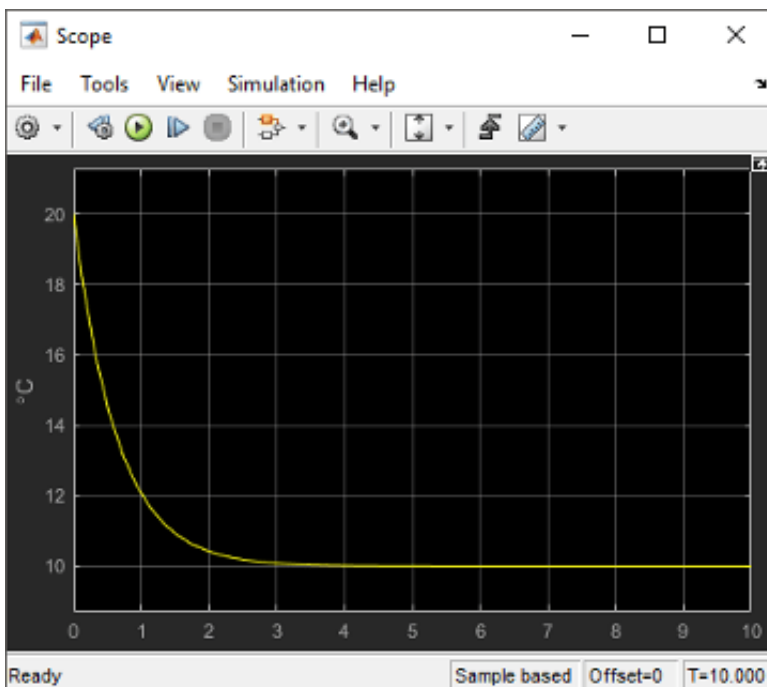
The Room block now has a second input port. Each input port corresponds to an Inport block in the subsystem.

2. Add a Constant block and connect it to the Heat Gain input. Set the **Constant value** parameter to 0 (degrees Celsius) to mean that the heater is turned off.
3. Add another Constant block and connect it to the Outside Temperature input. Set the **Constant value** parameter to 10 (degrees Celsius).
4. Add and connect a Scope block to view the changing room temperature.



Simulate Room Model and Evaluate Results

1. In the toolstrip, set the **Stop Time** to 20.
2. Simulate the model.
3. Open the Scope and click the **Autoscale** button  to view the scope trace.

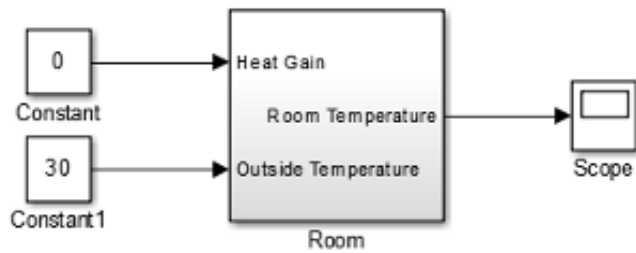


4. Determine if this result is what you expected.

The room temperature starts at the initial room temperature set in the Integrator block. Because the heat gain is 0, the signal decays to the outside temperature (10). The simulation validates the expected behavior.

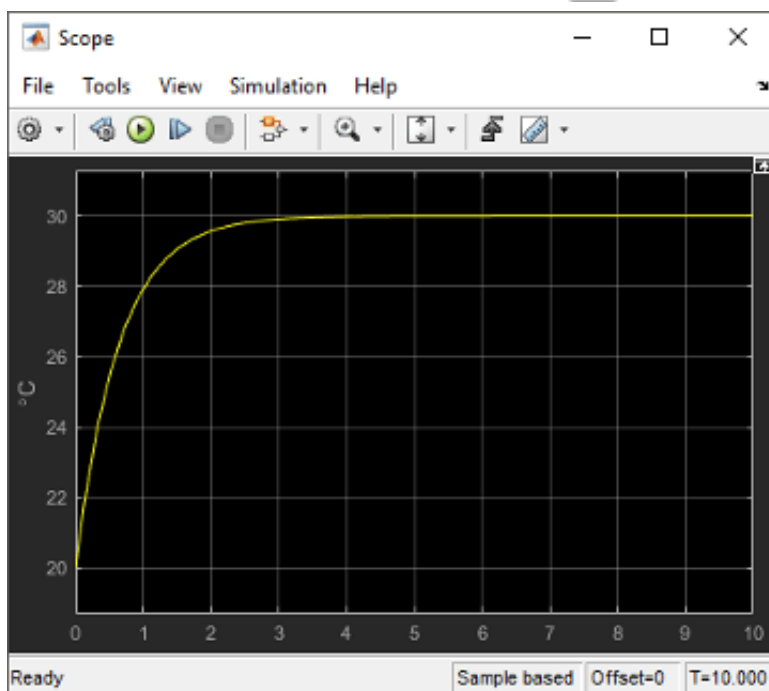
Prepare Room Model for Second Simulation. Set the constant outside temperature to a value above the initial room temperature (20).

In the Constant block that is connected to the Outside Temperature input, set **Constant value** to 30 (degrees Celsius).



Simulate Model and Evaluate Results

1. Simulate the model.
2. Open the Scope and click the **Autoscale** button  to view the scope trace.



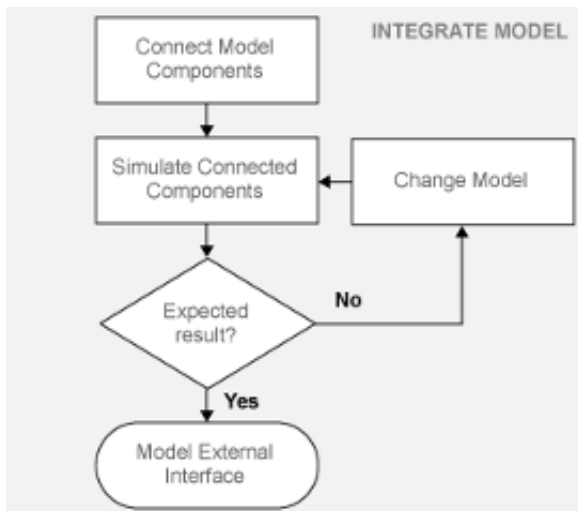
3. Determine if this result is what you expected.

Room temperature starts at the initially set temperature of 20, but with the heater off (heat gain = 0) the room temperature rises to the outside temperature — a behavior that the model did not explicitly specify, and might be considered unexpected.

The equation that models the heat loss also models the heat gain when the outside temperature is above the inside room temperature. While the model did not explicitly specify this behavior when the heater is turned off, the result makes sense physically.

Integrate a House Heating Model

Connect model components, add realistic input, and then simulate the model behavior over time to validate the model design.

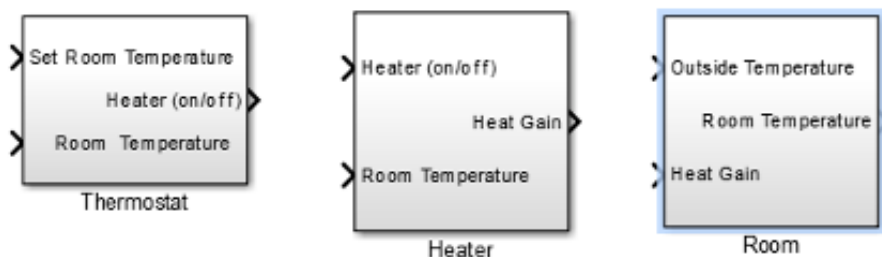


Integrate Heater and Thermostat Components

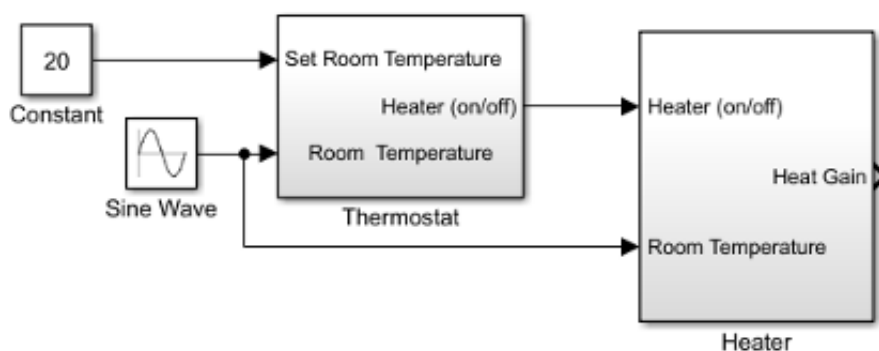
To simulate the heater and thermostat subsystems without the Room subsystem, you need a signal for the changing room temperature. Use a Constant block to set the thermostat temperature and a Sine Wave block for a realistic outside temperature signal.



Prepare Model for Simulation

1. Open your model with the completed subsystems. Remove any blocks you added to test the separate components.
2. Open the Room subsystem. Double-click the Inport block labeled Heat Gain. In the Inport block dialog box, set **Port number** to 2. The Heat Gain port moves to the bottom of the Room subsystem.



3. Connect the Heater (on/off) signal from the Thermostat subsystem output to the Heater subsystem input.
4. Add a Constant block to set the thermostat room temperature. Set **Constant** value to 20 (degrees Celsius).
5. Add a Sine Wave block to represent the changing room temperature. Set the parameters **Amplitude** to 10 (degrees Celsius), **Bias** to 15, and **Frequency** to 0.5.
6. Connect the blocks as shown in the figure.

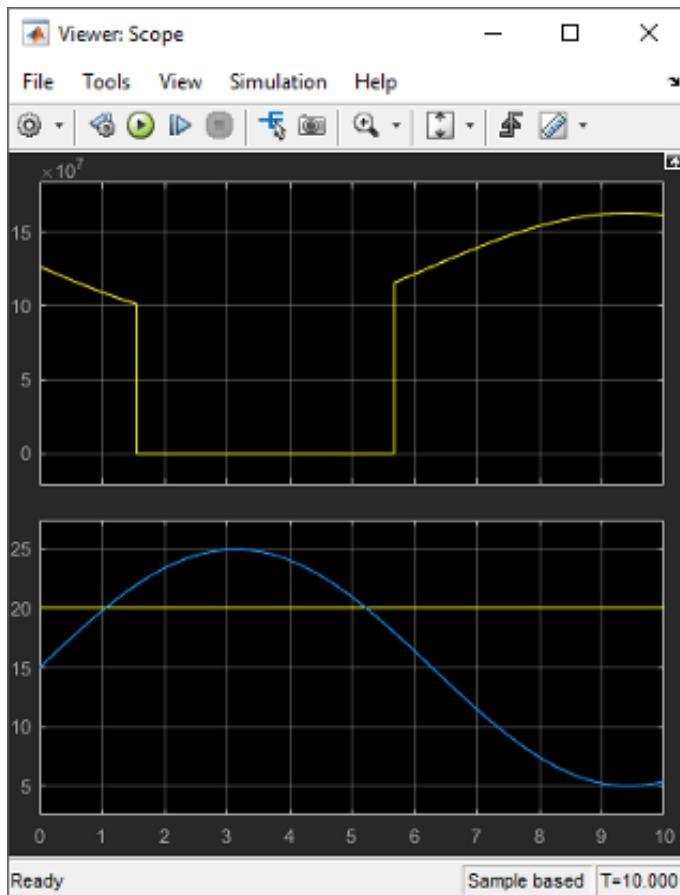


7. Add a Scope viewer and add the output signals from Heater, Constant, and Sine Wave blocks. See [Add Signal Viewer](#).
8. On the Scope viewer window, in the **Configuration Properties** button  click the arrow and then click **Layout** icon . Select two boxes. A second empty graph appears below the first.
9. In the **Simulation** tab under **Prepare**, select **Viewers Manager**.
10. From the Viewers and Generators Manager, select the **Scope**. Under Connected Signals, in the rows for **Constant** and **Sine Wave**, select the Display entry and change the Display to **2**.



Simulate Model and Evaluate Results. Simulate the model using the default stop time of 10.

1. Simulate the model.
2. Open the Scope Viewer and view the simulation results. The top graph is the heater gain while the lower graph shows the changing room temperature modeled with a sine wave.



3. Determine if this result is what you expected.

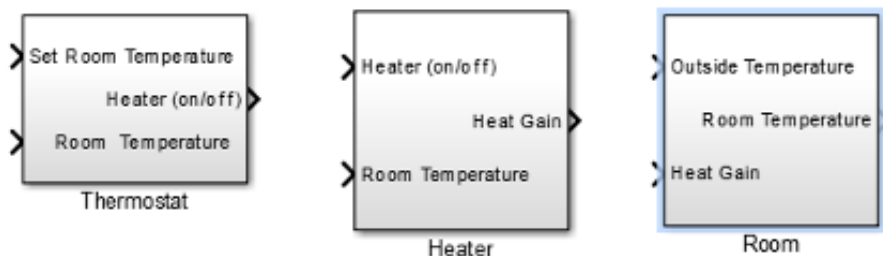
From about 0 to 1.5 hours, the heater is turned on. Heat gain is not constant but changes because heat gain is a function of the difference between the heater air temperature and the room air temperature. From 1.5 to 5.6 hours, the heater is turned off and the heat gain (top graph) is zero. The simulation confirms the expected behavior.

Integrate Room Component

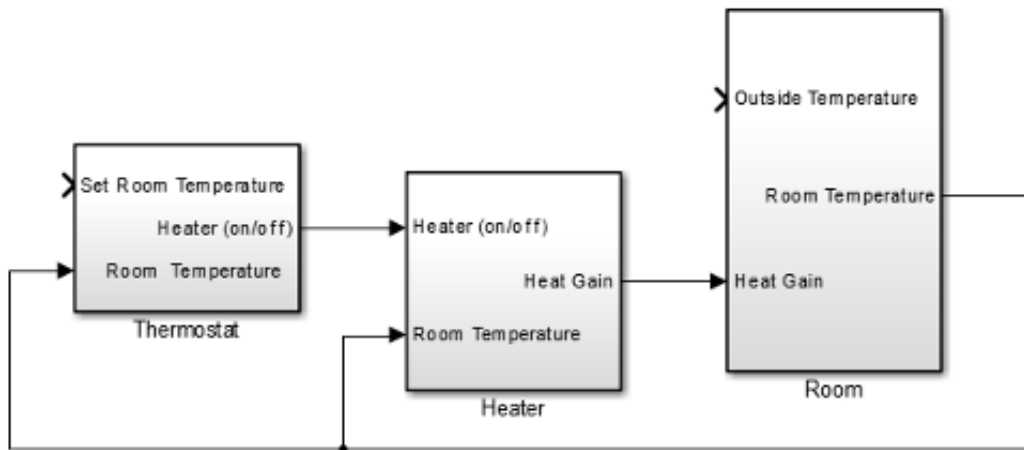
To simulate the Heater and Thermostat subsystems with the Room subsystem, you need a signal for the changing outside temperature. Simulating the model allows you to observe how the thermostat setting and outdoor temperature affect the indoor temperature.

Prepare Model for Simulation

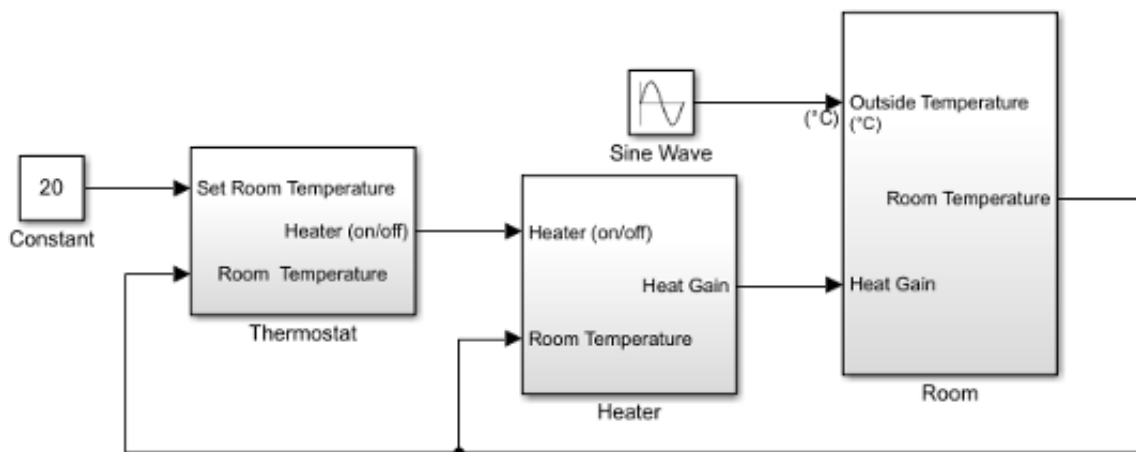
1. Open your model with completed subsystems. Remove any blocks you added to test the separate components.




2. Connect the subsystems as shown.



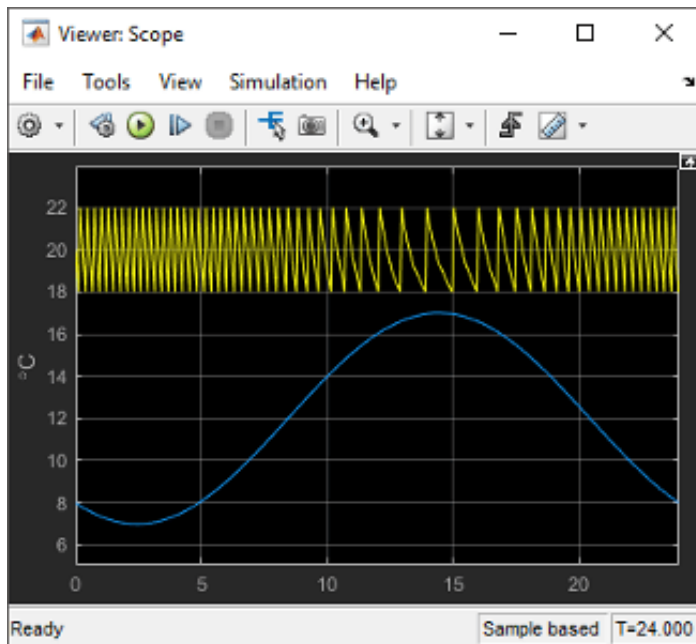
3. Add a Constant block for setting the room temperature. Set **Constant value** parameter to 20 (degrees Celsius).
4. Add a Sine Wave block to represent the changing outside temperature. Set **Amplitude** to 5, **Bias** to 12, **Frequency** to $2\pi/24$, and **Phase** to 180.



5. Add a Scope Viewer block to view simulation results.
6. In the Signal Viewer, click the **Signal Selector** button . In the Signal Selector dialog box and in the left pane, select the top model hierarchy. In the right pane, select the Room and Sine Wave signals.

Simulate Model and Evaluate Results

1. Set the simulation stop time to 24 (hours) to represent a day.
2. Simulate the model.
3. Open the Scope Viewer and view results.



4. Determine if the simulation result matches your expectation.

When the outside temperature is below the set room temperature, the room temperature fluctuates 2 degrees above and below the set temperature. Since the thermostat subsystem includes a 2 degree hysteresis, this simulation result is expected.

5. You can compare your results with an example model. In the MATLAB Command Window, enter

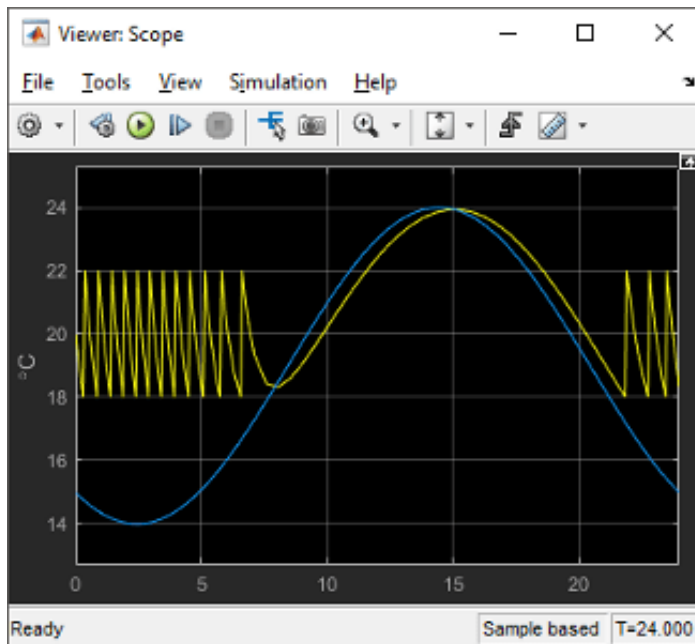
```
open_system(fullfile(matlabroot,...
    'help', 'toolbox', 'simulink', 'examples', 'ex_househeat_modeling_prepared'))
```

or click [ex_househeat_modeling_prepared.slx](#).

Refine Model Parameters. With Simulink models, you can interactively change model parameters and then observe changes in the behavior of your model. This approach allows you to evaluate your model quickly and validate your design.

Change the outside temperature in the Sine Wave block so that upper values are above the set temperature.

1. In the Sine Wave dialog box, set **Amplitude** to 5 and **Bias** to 19. These settings show what happens when outside temperature is higher than inside temperature.
2. Simulate the model and view the results.



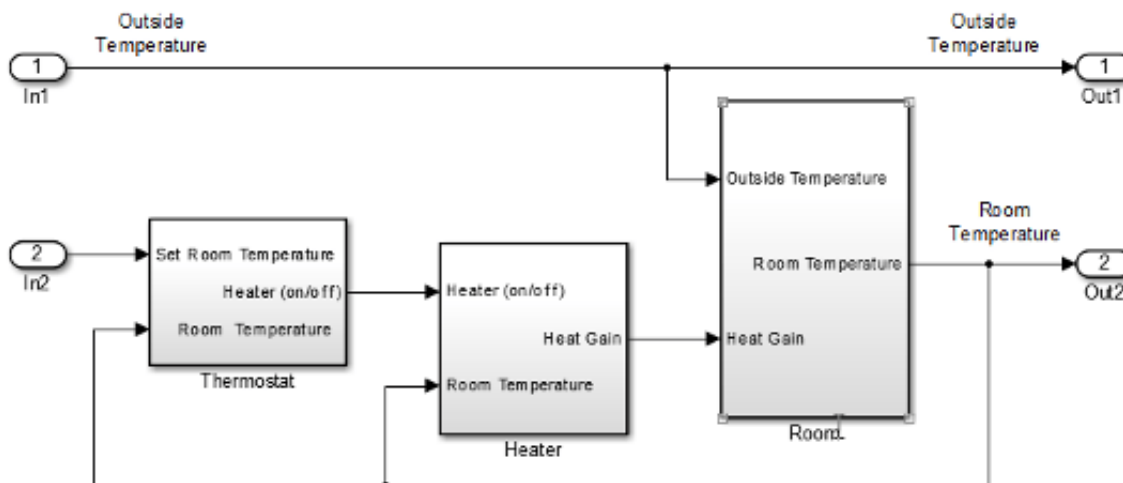
3. Determine if the results match your expectations.

When the outside temperature is above the set temperature, the room temperature follows the outside temperature with a slight delay. In this case, heat loss works in the reverse direction - and represents the loss of heat from the outside environment into the room.

Model External Interface

Model the external interface for further testing and possible use in a larger model. In Simulink, you model the external interface using Inport and Output blocks.

1. Add Inport blocks to read data from the outside temperature and thermostat set temperature into your model.
2. Add Output blocks to connect the outside temperature and room temperature to a larger model or to visualize results.

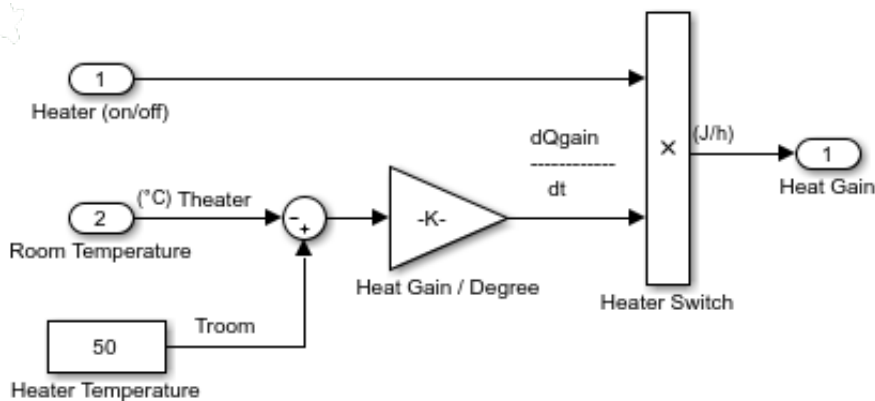


Specify Physical Units

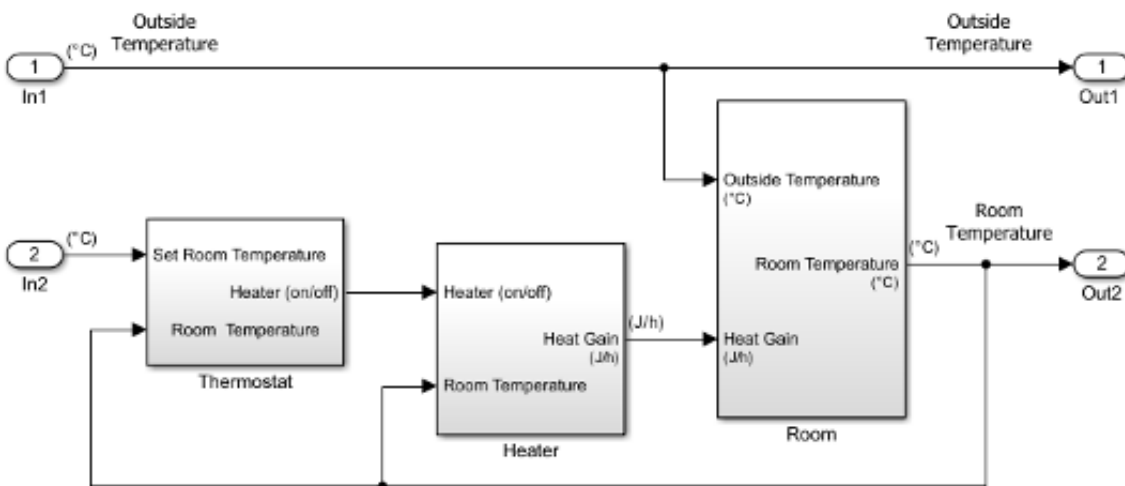
By specifying physical units for model signals, you ensure the consistency of calculations across model components. In Simulink, you specify signal units through Inport and Output blocks.

1. Double-click the In1 block to open the Block Parameters dialog box. Select the **Signal Attributes** tab.
 2. In the **Unit** box, start typing degree. From the list of symbols and names, select °C degree_Celsius.
- For the remaining temperature Inport and Outport blocks, set the **Unit** parameter to °C degree_Celsius.

3. Display units on block ports. On the **Debug** tab, select **Information Overlays > Units**.
4. Double-click the Heater Subsystem block. Double-click the Heat Gain Output block to open the Block Parameters dialog box. Select the **Signal Attributes** tab.



5. In the **Unit** box, start typing joule/hour. From the list of symbols and names, select joule/h joule/hour.
6. Update the model. Press **Ctrl+D**.



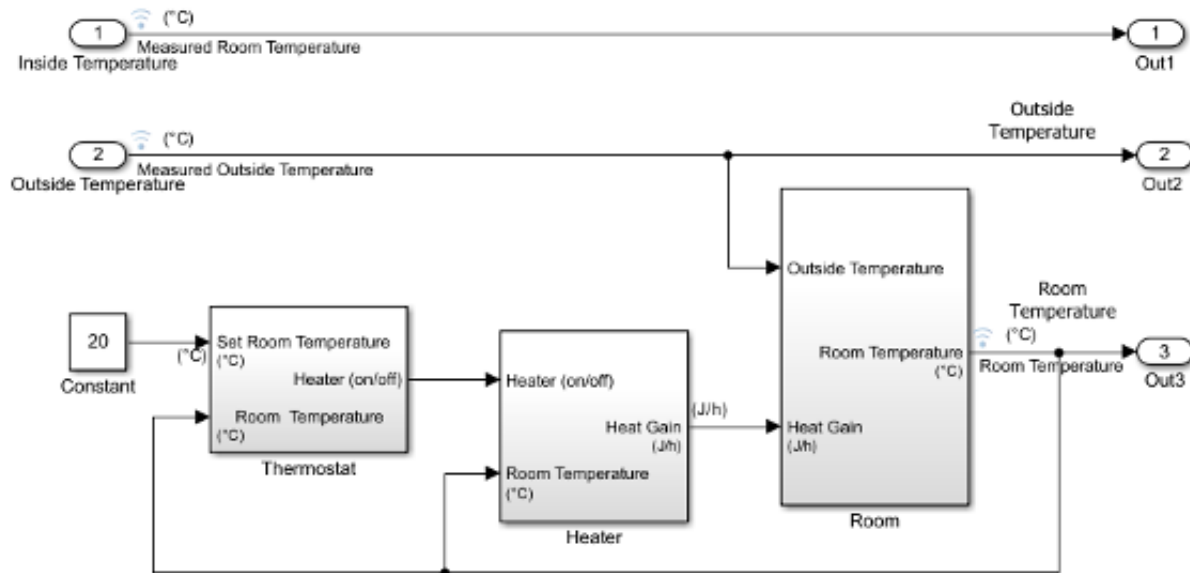
Your next step is to verify the correctness of the model by comparing simulations with real system data.

Prepare for Simulation

After initial simulations, you can use the results to improve the model to match model behavior to measured data. After you prepare the model for simulation, you can use an interface to input measured system data and set room temperature.

To load the finished example model, in the MATLAB Command Window, enter

```
copyfile(fullfile(matlabroot,...
    'help', 'toolbox', 'simulink', 'examples', 'ex_househeat_measured_data.mat'))
open_system(fullfile(matlabroot,...
    'help', 'toolbox', 'simulink', 'examples', 'ex_househeat_simulation_prepared'))
```



Verify that a simulation represents the behavior of the system you modeled. Begin by experimentally measuring physical characteristics of the system that have comparable signals in your model:

- Collect data from physical system
- Prepare model for simulation

Collect and Plot System Data

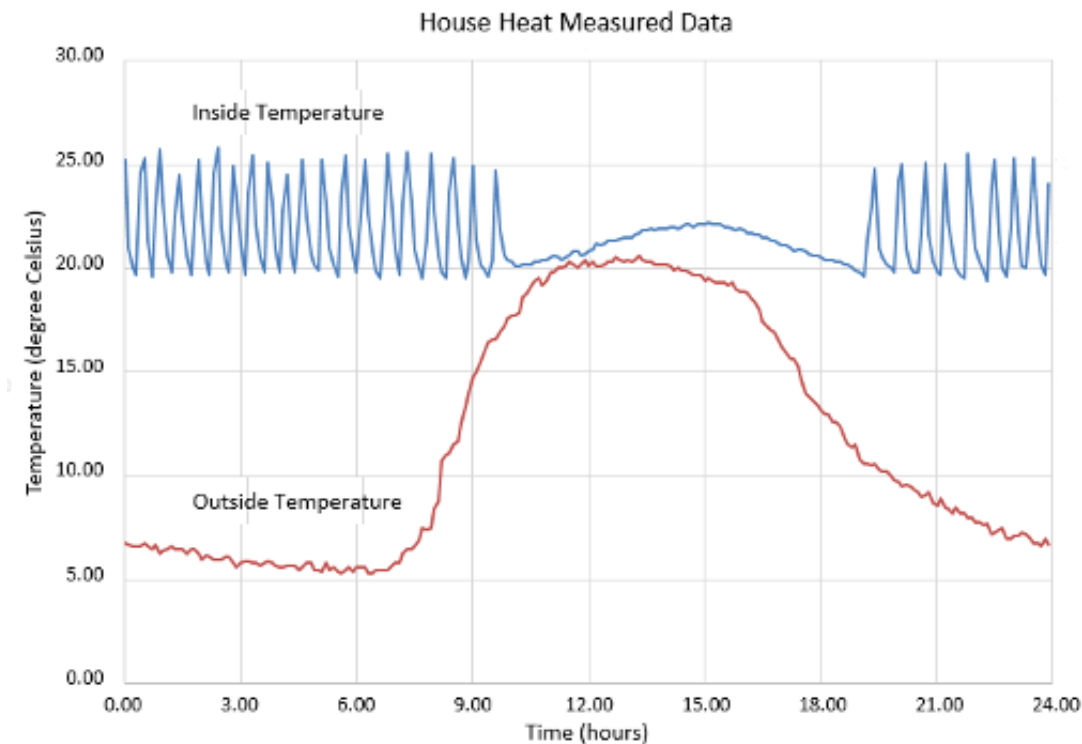
Measure the dynamic characteristics from an actual house heating system. You will use the measured data with model simulations to verify the behavior and accuracy of your model.

1. Measure the outside and inside temperatures of a house every 6 minutes for 24 hours.
2. Enter the measured data into a Microsoft® Excel® worksheet or open an example spreadsheet. In the MATLAB Command Window, enter

```
winopen(fullfile(matlabroot,...
    'help', 'toolbox', 'simulink', 'examples', 'ex_househeat_measured_data.xls'))
```

	A	B	C
1	Time (hours)	Inside Temperature	Outside Temperature
2	0.00	25.20	6.60
3	0.10	21.00	6.50
4	0.20	20.00	6.60
5	0.30	19.70	6.60
6	0.40	24.60	6.60
7	0.50	25.30	6.40
8	0.60	21.30	6.60
9	0.70	19.80	7.00
10	0.80	22.80	6.70

3. Review a plot of the measured data. The inside temperature data shows temperature spikes when the hot air heater turns on. This pattern is typical for a hot air heating system.

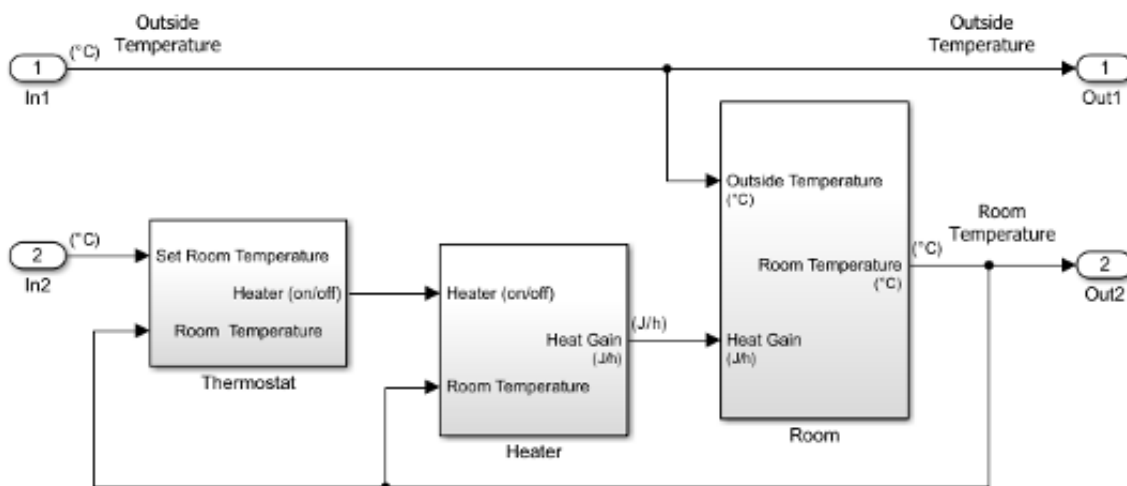


Prepare Model for Simulation

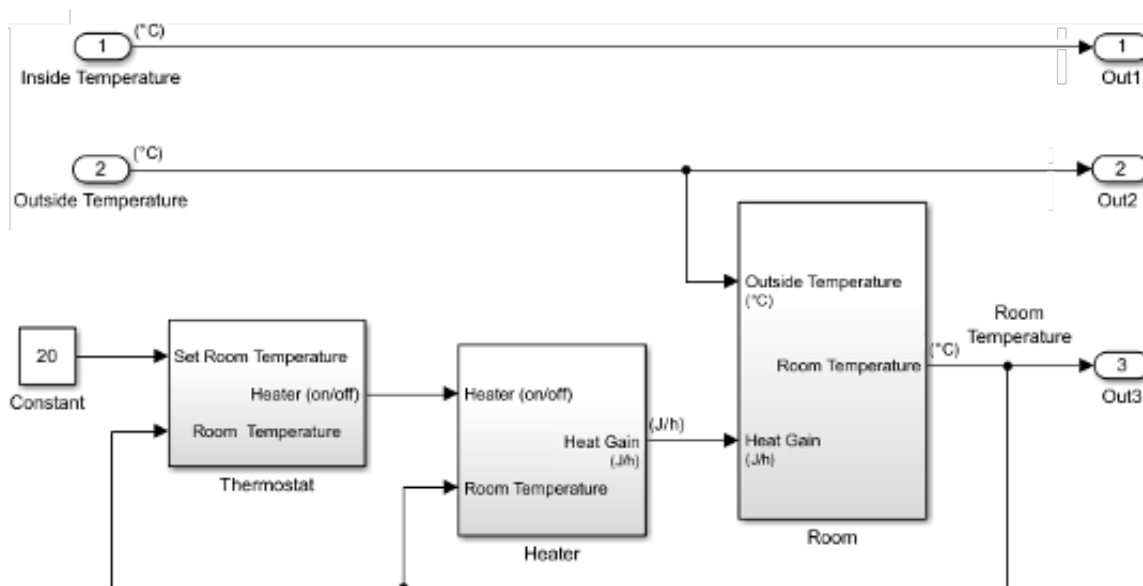
Prepare a model for simulation by adding an external interface for data input and input control signals.

1. Use the model you created in the tutorial Model House Heating System or open the example model. In the MATLAB Command Window, enter

```
open_system(fullfile(matlabroot,...
    'help', 'toolbox', 'simulink', 'examples', 'ex_househeat_modeling'))
```



2. Replace the Inport block In2 with a Constant block and set the **Constant** parameter to 20. The Constant block sets the thermostat temperature.
3. Add an Inport block. Set **Port number** to 1. This action also sets the **Port number** of the outside temperature signal to 2.
4. Rename the first Inport block to Inside Temperature. Rename the second Inport block to Outside Temperature.
5. Add an Outport block and connect it to the first Inport block (Inside Temperature). The Outport blocks are needed for saving (logging) the signals. Set **Port number** to 1.



Run and Evaluate Simulation

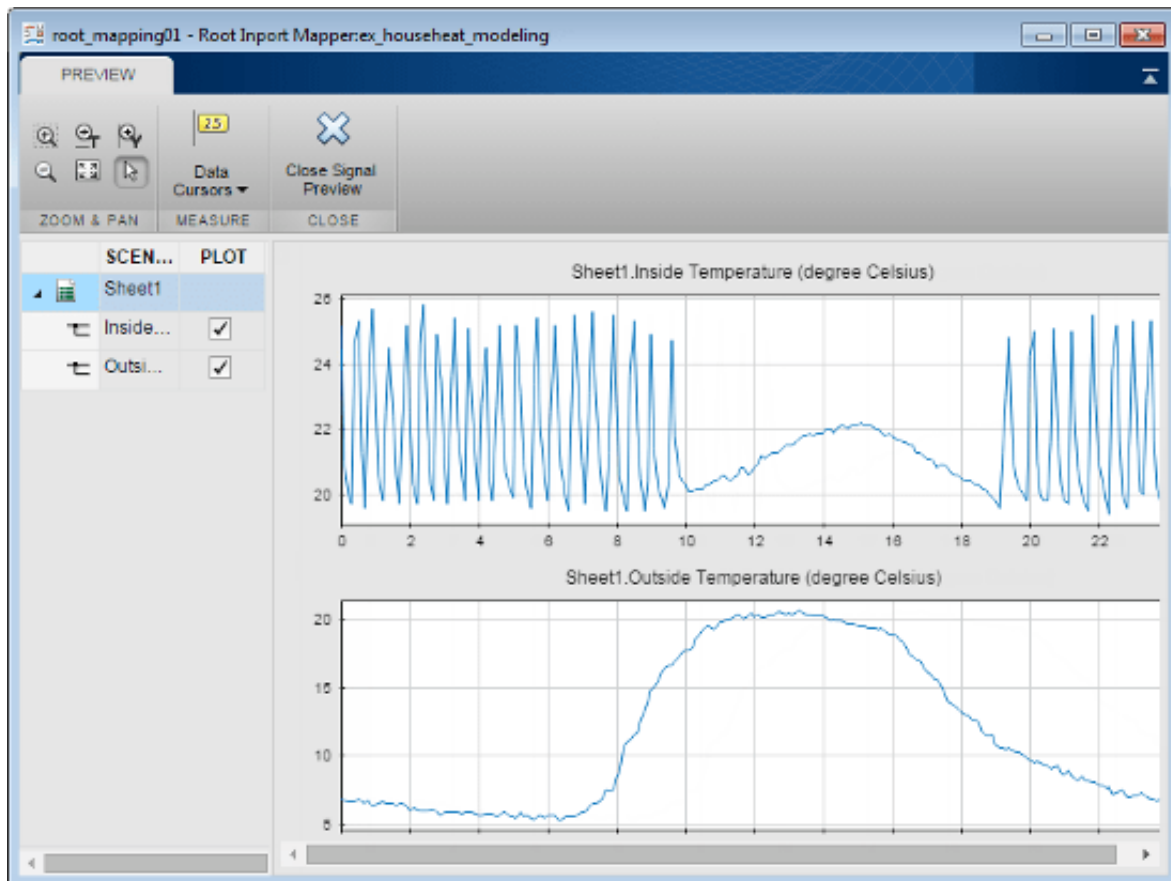
Verify the accuracy of the model and optimize parameters. Some parameters to consider for optimization are heater hysteresis, temperature offset, and the resistance of the house to heat loss. Follow these steps to verify your model:


- Import data
- Run simulation
- Evaluate simulation result
- Change model parameters
- Rerun simulation

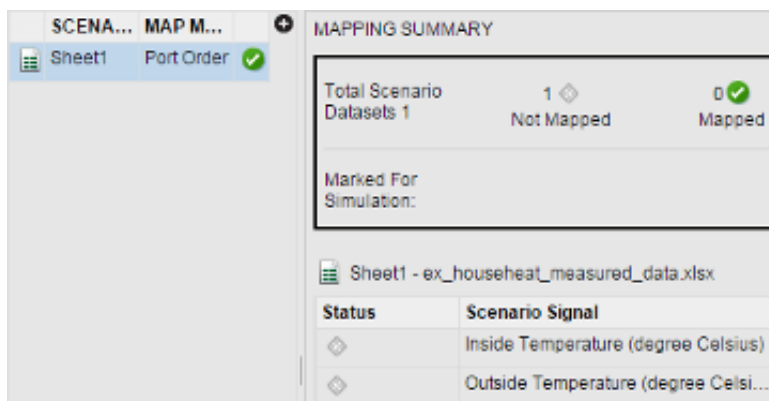
Import Data with Root Input Mapping

You can use the Root Input Mapper tool to bring measured signal data from an Excel spreadsheet into a Simulink model.

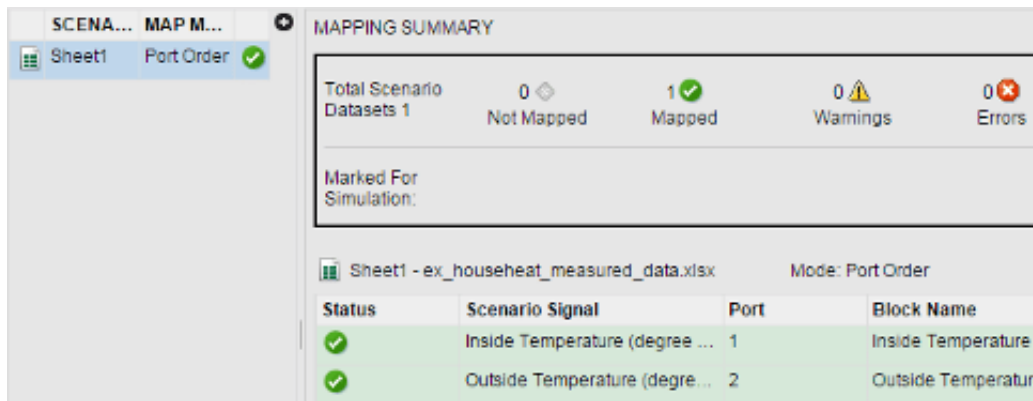
1. Open any Inport block. Click the **Connect Input** button to open the Root Input Mapper.
2. On the toolbar, click **From Spreadsheet**.
3. In the From Spreadsheet dialog box, click the browse button. Browse to and select the file `matlabroot\help\toolbox\simulink\examples\ex_househeat_measured_data.xls`. Click **Open**. Click **OK** to import the spreadsheet.
4. From the **Signals** drop-down list, select **Preview Signals**.
5. On the left side, expand the tree view of Sheet1. Select the Inside Temperature and Outside Temperature check boxes.



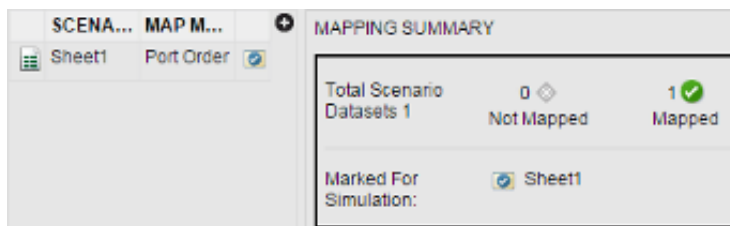
6. Click **Close Signal Preview**.
7. On the left side, select Sheet1. The **Scenario Signal** column shows the two signals from the Excel spreadsheet and an icon  indicating the signals are unmapped.



8. On the toolbar, select the **Port Order** option. From the **Options** drop-down list, select the **Update Model** check box.
9. From the **Map to Model** dropdown list, select Map Unconnected. The mapping summary shows the signals from the Excel spreadsheet mapped to the Input port blocks.



10. Click **Mark for Simulation**. The mapping summary shows Sheet1 is marked for simulation and a Dataset object is created in the MATLAB Workspace.



11. Save the signal data in a MAT-file. In the MATLAB Command Window, type
`save('ex_househeat_measured_data.mat', 'Sheet1')`

Configure Model to Load Signal Data

Signal data mapped to input ports is located in a MATLAB workspace variable. With each new MATLAB session, you have to manually reload the data or let the model preload function do it for you.

- From the Simulink Editor, on the **Modeling** tab, select **Model Settings** > **Model Properties** to open the Model Properties dialog box.
- Select the **Callbacks** tab.
- In the **Model callbacks** section, select PreLoadFcn.
- In the **Model pre-load function** box, enter
`load('ex_househeat_measured_data.mat')`
- Click **OK**.


Configure Model to Save Simulation Results

Configure your model to save (log) signal data during a simulation. You can then view logged signals from a simulation using the Simulation Data Inspector.

- In the model, on the **Modeling** tab, click **Model Settings**.
- In the left pane, select **Data Import/Export**.
- In the right pane, clear the **Time** and **Output** check boxes.
- Select the **Signal logging** check box.
- Select the **Record logged workspace data in Simulation Data Inspector** check box.
- Click **OK**.

Select Signals to Save

Identify signals to display in the Simulation Data Inspector, name the signals if they are unnamed, and set the logging parameters.

1. Right-click the Inside Temperature signal line and select **Properties**.
2. In the **Signal name** box, enter Measured Room Temperature. Select the **Log signal data** check box. A logging badge  appears above the signal line.
3. Name and select logging for these signals.

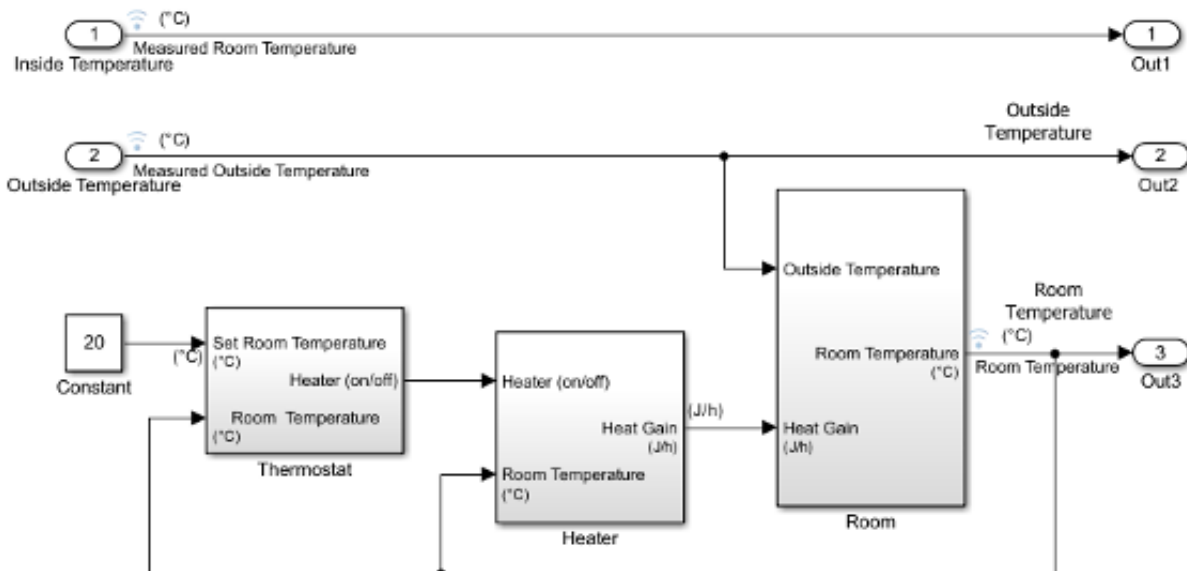
Location of signal	Signal name
Outside Temperature from output port 2.	Measured Outside Temperature
Room Temperature from Room subsystem output port	Room Temperature

Run Simulation

After importing data and enabling logging of data for the signals, you can run a simulation.

1. Use the model you prepared for simulation or open the example model. In the MATLAB Command Window, enter

```
open_system(fullfile(matlabroot,...
    'help', 'toolbox', 'simulink', 'examples', 'ex_househeat_simulation_prepared'))
```



2. On the Simulink Toolstrip, set **Stop Time** to 24 (hours).

3. Click the **Run** button .

The model simulation runs from 0.0 to 24.0 hours using the outside temperature data from the root import block as input.

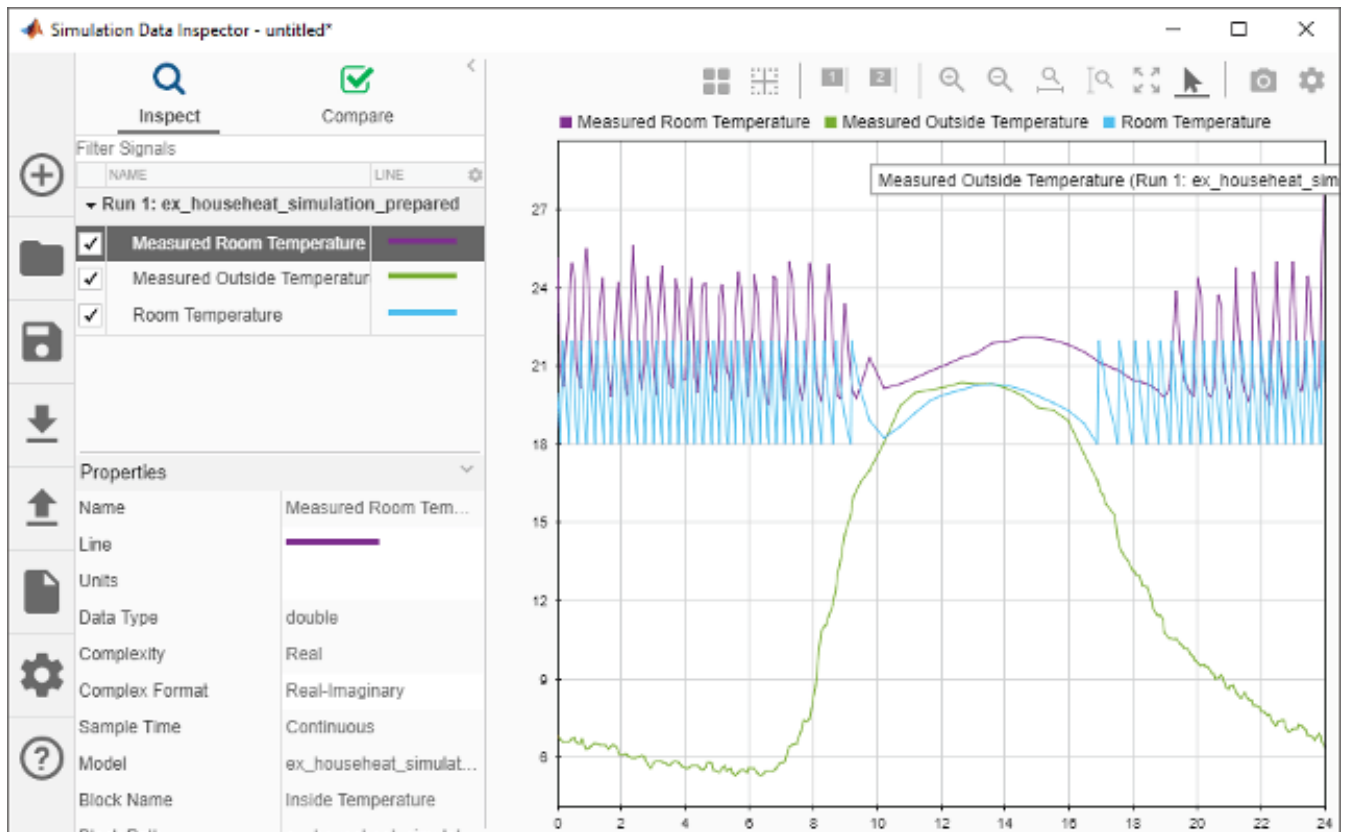
Compare Simulation Results with Measured System Data

Use the Simulation Data Inspector to compare the simulated output signals with measured data.

1. On the Simulink Toolstrip, click the **Simulation Data Inspector** button .

A separate run appears in the **Runs** pane each time you simulate the model.

2. Select all the signal check boxes. The graph show the plot of each signal you select.



The top signal is the Measured Room Temperature. The middle signal is the Measured Outside Temperature. The bottom signal is the simulated Room Temperature.

Determine Changes to Model

One obvious change to the model is the hysteresis of the thermostat. The simulated room temperature oscillates 18–22 degrees around the temperature set point of 20 degrees. The measured room temperature oscillates 20–25 degrees with the same set point.

1. Open the Relay block in the Thermostat subsystem.
2. Change **Switch on point** from 2 to 0 because the difference between the room temperature and set point is 0.
3. Change **Switch off point** from –2 to –5. When the room temperature is 5 degrees above the set point, you want to turn off the heater. The set point is 5 degrees below the room temperature.

Compare Results Between Simulations

Use the Simulation Data Inspector to compare differences between two simulations that use different model parameters. This comparison shows how changes improve the accuracy of your model.

1. Simulate the model.
2. Open the Simulation Data Inspector.
3. Expand the list of logged signals by selecting the arrow to the left of the run. For **Run1**, select the Measured Outside Temperature and Measured Room Temperature check boxes. For **Run2**, select the Room Temperature check box.
4. Review the signals. The minimum and maximum values for the simulated room temperature now match the measured room temperature values.

