Contents

Index

- Contents
- CyclePad License
- Overview
- How CyclePad Views Thermodynamic Cycles
- A Guided Tour of CyclePad
- Build Mode
- Creating a New Design
- Adding Components to the Design
- Connecting Components
- Changing the Labels of Components and Stuffs
- Manipulating Component Icons
- Deleting Stuffs
- Analyze Mode
- Meters
- Meter Color Coding
- Interacting with Meters - Choosing a Substance
- Interacting with Meters - Making Modeling Assumptions
- Interacting with Meters - Assuming and Changing Numeric Values
- Interacting with Meters - Derived Values
- Contradiction Mode
- How Does CyclePad Detect Contradictions?
- How Do I Get Out of Contradiction Mode?
- Investigating a Cycle via the Explanation System
- A Summary of Explanation System Question Types
- Investigating a Cycle via the Sensitivity Analysis Tool
- Economic Model
- Design and Analysis Coaching
- CyclePad Command Reference
- File
- Edit
- Instructor
- Insert
- Library
- Mode
- Tools

Cycle

Windows

Preferences

Preferences: Units

Preferences: Miscellaneous

Preferences: Personal

Preferences: Advanced

Preferences: Internet

Help

CyclePad's Knowledge Base

Components

Processes

Stuff

Subcycles

Modeling Assumptions

Parameters

C

C

cop-r

Carnot cop-r

cop-hp

Carnot cop-hp

delta-H

delta-H-isentropic

delta-P

delta-spec-s

gamma (k)

H

Hout-isentropic

mass-flow

max-T

min-T

molar-mass

net-Q

net-work

eta-isentropic

eta-thermal

- P
- PR
- Psat
- Q
- Q-in
- Q-out
- R
- S
- shaft-work
- spec-h
- spec-hf
- spec-hg
- spec-q
- spec-s
- spec-sf
- spec-sg
- spec-shaft-work
- spec-shaft-work-isentropic
- spec-u
- spec-v
- T
- Tout-isentropic
- Tsat
- U
- V
- work-in
- work-out
- Substances
- Ideal Gas Law
- Thermodynamic Property Tables
- Turbine
- Compressor
- Pump
- Heater
- Cooler
- Heat Exchanger
- Throttle

- Splitter
- Mixer
- Source
- Sink
- Combustion Heating
- Cooling
- Polytropic Expansion
- Polytropic Compression
- Diagrams of Library Examples of Cycles
- Simple Steam Rankine Cycle
- Simple Refrigerator Cycle
- Basic Gas Turbine Cycle
- Steam Cycle with Reheat
- Combined Gas Turbine & Rankine Cycle
- Basic Engineering Thermodynamics
- Thermodynamic systems and properties
- Processes and reversibility
- Thermodynamic cycles
- Index
- The Assignment Builder
- Creating a New Assignment

Contents

Note also that the thermodynamic property data in _CyclePad_ was assembled from a variety of sources, generally noted in the files containing the data. _CyclePad_ was developed as noncommercial, research software. Any implementation of the data in _CyclePad_ may require authorization of the thermodynamic property data copyright owners.

This is the CyclePad Help file, compiled in February, 2002. Every effort is made to ensure that this file is current, but due to the pace of CyclePad's development it is possible that new features are not yet documented in this file. In addition, the graphics may look somewhat different on your computer than they do in this file, either due to changes in CyclePad or differences in Windows settings. The graphics in this file are from Windows 2000, so they may appear different from those in other environments.

A note on units and terminology: CyclePad was first developed at Oxford university, and so some of the terms used in this file reflect British usage. In these cases, the equivalent American term is given in parentheses.

CyclePad License

Overview

How CyclePad Views Thermodynamic Cycles

A Guided Tour of CyclePad

Build Mode

Analyze Mode

Contradiction Mode

The Explanation System

Sensitivity Analysis

Design and Analysis Coaching

The Assignment Builder

CyclePad Command Reference

CyclePad's Knowledge Base

Diagrams of Library Examples

Index

For Help on Help, Press F1

Top

CyclePad License

USE OF CYCLEPAD (HEREINAFTER REFERRED TO AS "THE SOFTWARE"), CONSTITUTES ACCEPTANCE OF THIS LICENSE AGREEMENT. IF YOU DO NOT AGREE TO ALL OF THE TERMS SET FORTH BELOW, YOU MUST IMMEDIATELY UNINSTALL THIS SOFTWARE FROM YOUR SYSTEM AND DELETE ALL COPIES THEREOF.

CYCLEPAD END USER LICENSE AGREEMENT

You are granted a non-exclusive license to use CyclePad free of charge.

TERMS APPLICABLE TO ALL LICENSE GRANTS

You may:

- Use the Software on any computer;
- Use the Software on a network;

- Copy the Software, provided any copy must contain this license agreement;
- Or distribute the Software, provided any copy must contain this license agreement.

You may not:

- Modify, translate, reverse engineer, decompile, disassemble, or create derivative works based on the Software;
- Rent, lease, grant a security interest in, or otherwise transfer rights to the Software, although you may charge a nominal distribution fee for the physical act of transferring a copy of the program;
- Or remove any proprietary notices or labels on the Software.

DISCLAIMER OF WARRANTY AND LIMITATION OF LIABILITY

This software is made available AS IS, and Kenneth D. Forbus and Peter B. Whalley disclaim all warranties, express or implied, including without limitation the implied warranties of merchantability and fitness for a particular purpose, and notwithstanding any other provision contained herein, any liability for damages resulting from the software or its use is expressly disclaimed, whether arising in contract, tort (including negligence) or strict liability, even if Kenneth D. Forbus and/or Peter B. Whalley is advised of the possibility of such damages. Should the Software prove defective, you and not Kenneth D. Forbus or Peter B. Whalley assume the entire cost of any service and repair. This disclaimer of warranty constitutes an essential part of the agreement.

TITLE

Title, ownership rights, and intellectual property rights in the Software shall remain in Kenneth D. Forbus and Peter B. Whalley. The Software is protected by copyright laws and treaties. Title and related rights in the content accessed through the Software is the property of the applicable content owner and may be protected by applicable law. This License gives you no rights to such content.

TERMINATION

The license will terminate automatically if you fail to comply with the limitations described herein. On termination, you must destroy all copies of the Software.

MISCELLANEOUS

If any provision of this Agreement is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This Agreement shall be governed by Illinois law (except for conflict of law provisions).

CITATION

In publications that refer to CyclePad, the appropriate citation is
CyclePad, 1995-2002 Kenneth D. Forbus, Peter B. Whalley,
http://www.qrg.northwestern.edu/software.htm

Top

Overview

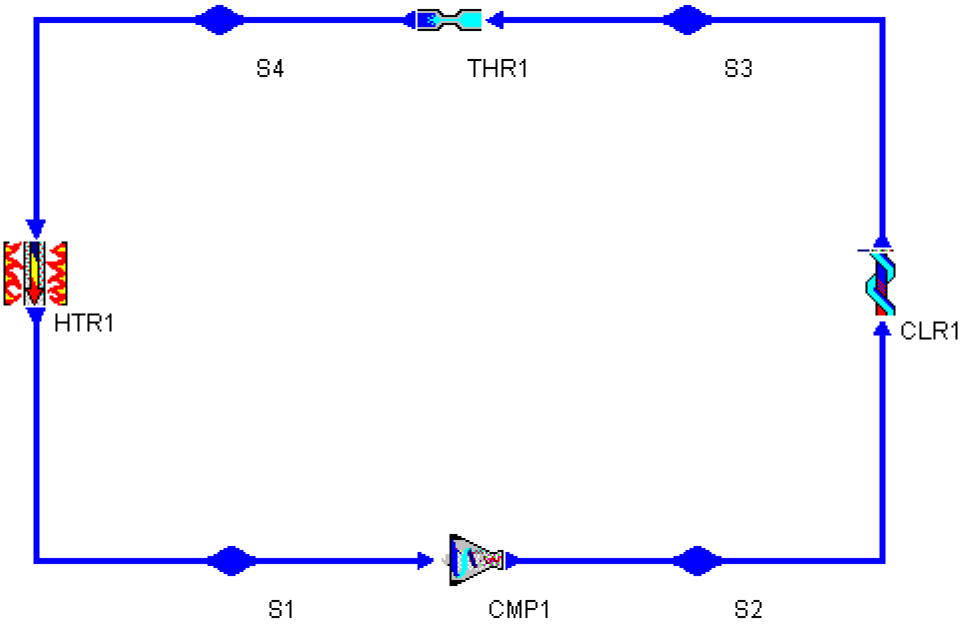
A thermodynamic system a collection of components which either takes in heat from the environment and produces energy, or takes in work and produces some transfer of heat from one region of the environment to another, perhaps as a refrigerator or as a heat pump. Examples of thermodynamic systems include power plants, refrigerators, propulsion plants, and engines. CyclePad helps you:

- Specify the structure of your design , in terms of the physical parts and processes of the cycle and how they are connected to one another.
- Analyze your design, by making assumptions about it and figuring out the consequences of those assumptions. Such assumptions include numerical values, e.g. operating temperatures and pressures, and modeling assumptions, e.g., whether or not to consider a turbine as isentropic.
- Plot the T-s and P-v diagrams of a cycle to understand where the working fluid is in its property space (e.g. how hot is it, how high is its pressure, et cetera).
- Perform sensitivity analyses to understand how different choices of your design contribute to its performance. For example, CyclePad can figure out how the efficiency of a system changes as a function of other parameters, such as a turbine inlet temperature.

CyclePad performs steady-state analyses of both open and closed cycles. In an open cycle, the working fluid passes through various "open" components, while in a closed cycle different processes act upon a closed volume or mass of working fluid (sometimes called the control volume). Gas turbines are therefore open cycles, and piston engines are closed cycles. Note that a closed-loop steam cycle containing a boiler, turbine, condenser, and pump is still considered to be an open cycle.

Steady-state analyses provide the kind of initial guidance needed in conceptual design, because in the conceptual design of thermodynamic cycles the important questions concern the operating conditions and estimates of efficiency and cooling/heating/power produced by the cycle. (Later stages of design concern issues such as the response of the system to transients, developing procedures for safe startup and shutdown, and ensuring that the system is easy to monitor and maintain. CyclePad works with steady-state and equilibrium analysis; it does not deal with transient analysis.)

CyclePad works in two phases, build mode and analyze mode. In the build mode, you use a graphical editor to place components and connect them with stuffs. Such a structure might look like this:



(To read more about how CyclePad construes this cycle, click here)

While you can always quit CyclePad at any time, you can only proceed to the next phase (analysis) when CyclePad is satisfied that your design is fully laid out, that is, when every component is connected via some other component via stuffs, and every stuff has been used as both an input and an output for components in the design. Once your design is laid out, the real fun begins--the analysis phase

In the analysis phase, you specify:

- What working fluid you are using
- What modeling assumptions you wish to make in analyzing your design.
- Numerical values for the properties of components and stuffs

As soon as you give CyclePad some information, it draws as many conclusions as it can about your design, based on everything you have told it so far. When you specify a working fluid, for instance, it knows whether to use property tables or an ideal gas approximation. When you specify numerical values, CyclePad sees if it can then calculate other numerical values. It displays the results of its calculations, and you are free to inquire about how values were derived and how one might proceed at any time, using a hypertext query system.

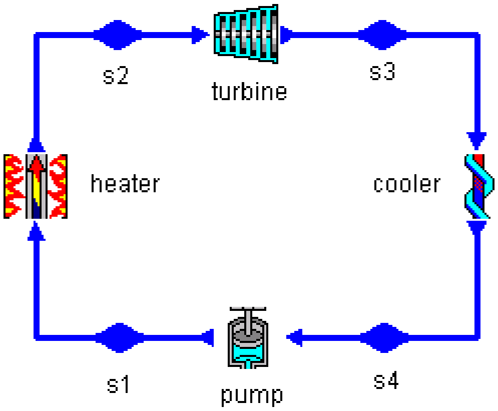
As you provide more information, CyclePad deduces more about the physical system. Eventually, you may have filled in all of the relevant information about the cycle, so that you have numerical values for properties such as the coefficient of performance (if you are designing a refrigerator), the thermal efficiency (if you are designing a heat engine), or other properties of interest such as the total amount of work produced or consumed by the cycle. How far you go is up to you. At any time you can save your design to a file so that you can continue working on it later, and generate reports describing the state of your analysis of the design.

CyclePad also supports sensitivity analyses. For instance, suppose you wanted to understand how the thermal efficiency of the cycle varies as a function of the efficiency of a compressor or some other component. Such analyses are quite tedious to do by hand, but CyclePad makes them quite easy and will generate such information for you in graphical form.

Top

How CyclePad Views Thermodynamic Cycles

A thermodynamic cycle consists of a collection of components connected together in some appropriate fashion. The components CyclePad knows about include compressors, turbines, heaters, coolers, pumps, mixers, splitters, throttles, and heat exchangers. CyclePad describes connections in terms of the properties of the material at the connection, that is, the properties of the stuff that is flowing between the components.



Simple Rankine Cycle

In the above diagram, for example, you will see that there are four components: a heater, a turbine, a cooler, and a pump. These components are connected via four stuffs, S1, S2, S3, and S4.

The major source of information about the cycle is the set of parameters associated with each stuff and each component. CyclePad knows what parameters are associated with each component and with each stuff. It knows that the set of what parameters are relevant can vary; when a stuff is saturated, for example, its dryness (quality) becomes relevant, and a turbine which is not approximated as isentropic requires some specification of its presumed efficiency. Part of your job as a designer is selecting numerical values and modeling assumptions to see if a particular design can satisfy your performance criteria (e.g., desired work output, efficiency, etc.)

Top

A Guided Tour of CyclePad

Let's step through the design of a simple Rankine cycle, from beginning to end. You interact with CyclePad in three modes, Build, Analyze, and Contradiction, generally in that order. You may take a tour of any of these modes by clicking on them below, but we suggest that, at least for your first time, you proceed in the order in which they are listed.

Build Mode

Analyze Mode

Contradiction Mode

Top

Build Mode

Creating a New Design

Adding Components to the Design

Connecting Components

Changing the Labels of Components and Stuffs

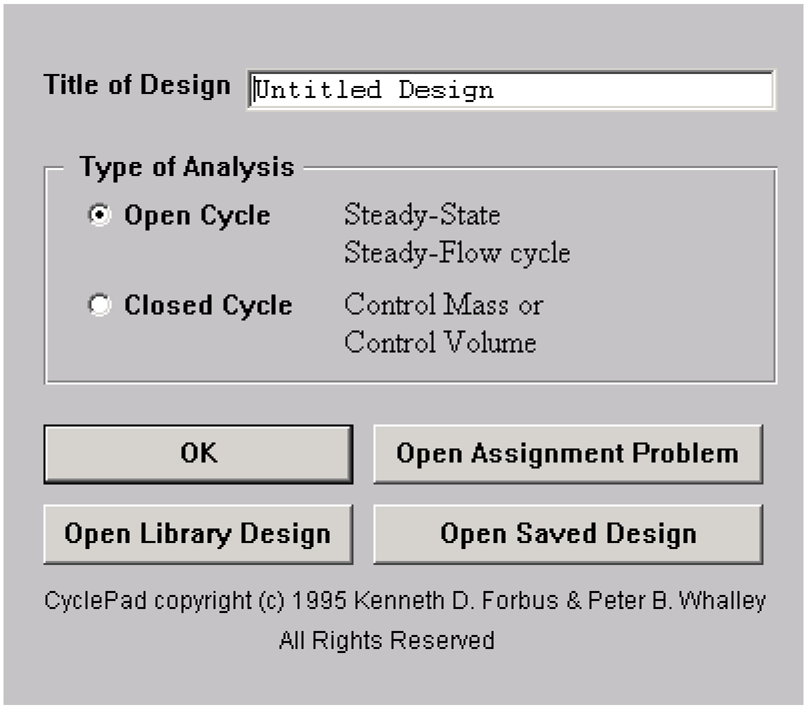
Manipulating Component Icons

Deleting Stuffs

Top

Creating a New Design

When you first start CyclePad, you are confronted with the New Design dialog box:



Title of Design: Untitled Design

Type of Analysis

● Open Cycle — Steady-State Steady-Flow cycle

○ Closed Cycle — Control Mass or Control Volume

CyclePad copyright (c) 1995 Kenneth D. Forbus & Peter B. Whalley
All Rights Reserved

First, enter a title for your design. (You can change this title later, via the Edit menu). Next, choose the type of analysis you want to perform. In the current version of CyclePad you may choose either an open-cycle (steady-state, steady-flow designs) or closed-cycle (control-mass or control-volume) analysis. Once made, you cannot change this choice, but you can of course start a new design of another type at any time (use the File|New command).

If you want to open one of the Library examples, click on Open Library Design. Note that you can modify and then save these designs freely.

If you would rather open an existing design, perhaps one you had saved to disk during a prior session, click on Open Saved Design. Otherwise, choose OK.

If you wan to open an assigned CyclePad problem file (.PRB), choose Open Assignment Problem. Note that building the problem files is done with the Assignment Builder option from the Instructor menu.

The New Design dialog box disappears and CyclePad presents a blank blue screen on which you will now design your cycle. You are by default in Build mode, and CyclePad will not let you change to Analyze mode until you have completed the design of your cycle.

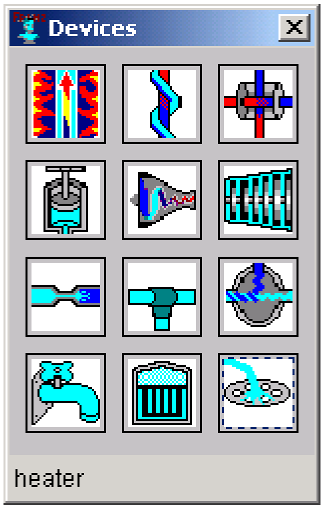
See also:

Adding Components to the Design
Connecting Components
Changing the Labels of Components and Stuffs
Manipulating Component Icons
Deleting Stuffs

Top

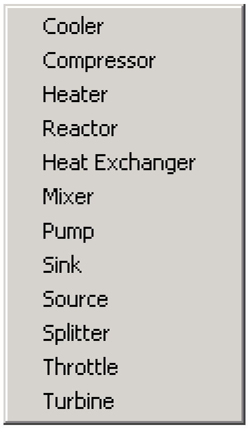
Adding Components to the Design

When you enter Build Mode, the Device Palette is open. To start adding components, simply click on the background, and the device currently selected in the Device Palette will appear on the screen. To select a different device, just click on it on the Device Palette.



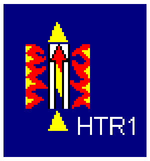
If you prefer, you may close the Device Palette and add devices to your design by clicking on the screen. An Add Component pop-up menu will appear. Select a component to add by clicking on the appropriate name.

Clicking on any component name in the menu below will display more information about that component, including its icon.



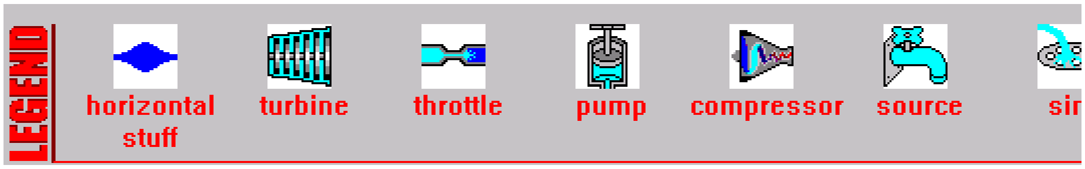
If you wish to re-open the Device Palette after closing it, it is available under Windows in the menu bar.

Once you have selected a component, you add it to the design by clicking on the design screen where you want it to be. The selected component will appear on the screen. For example, had you selected a heater, the icon for a heater would have appeared:



The two yellow triangles are the heater's *ports*. They indicate the direction of flow through the heater (in this case, bottom to top), and they enable you to connect the heater to other components.

To help familiarize you with the icons CyclePad uses for components, a legend runs along the bottom of the screen and each icon has its name underneath it. You can close the legend if you like. Part of the legend is shown below.



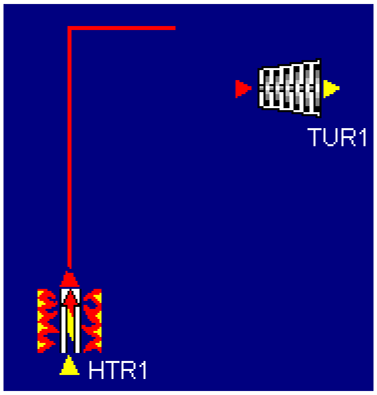
See also:

Creating a New Design
Connecting Components
Changing the Labels of Components and Stuffs
Manipulating Component Icons
Deleting Stuffs

Top

Connecting Components

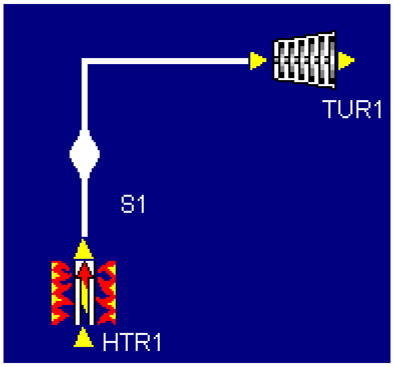
To connect two icons, first click on a port, such as the top port of the heater:



Note that the port you clicked on turns red, and a red line appears, attached to the port and to the mouse pointer. This line can only be horizontal or vertical (to simplify the layout of diagrams); to change direction, click to create a right angle joint. Also note that all ports to which you could connect this line have turned red. In the above figure, the inlet to the turbine has turned red, but the turbine's outlet has not, because you cannot connect two outlets to one another.

Now that you have clicked on a port, *you must first drag the red line some distance away from the icon in the direction of flow (i.e., the direction in which the port arrowhead points)*. Above all, resist the temptation to start the connection by immediately moving at right angles to the direction of flow, because the interface won't let you do that in this version of CyclePad.

Having dragged the red line for some distance, click when you want to make a right angle change in direction. Don't worry about lining up exactly with the port to which you want to connect. CyclePad will automatically "snap" the connecting line to the second icon. In the above example, the joint is clearly above the inlet to the turbine, but if you simply click on the red inlet port of the turbine, the connection will be correctly made:



If you decide that you want to abandon the connection operation, a right click will restore the screen to its state prior to initiating the connection.

When the connection is completed (by clicking on the second port), a stuff appears, indicated by the bulge in the "pipe" connecting the two components. Like each device, each stuff has a unique label, which you can drag to make your diagram easier to read and edit in Build mode.

See also::

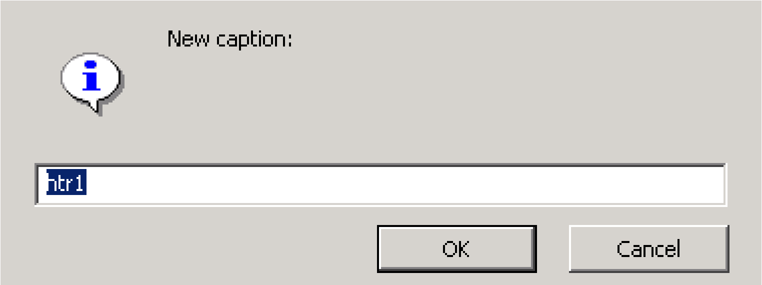
Creating a New Design
Adding Components to the Design
Changing the Labels of Components and Stuffs
Manipulating Component Icons

Deleting Stuffs

Top

Changing the Labels of Components and Stuffs

To edit a stuff or component label, right click on the label in build mode. .A dialog box allowing you to change the item's label will appear.



New caption:

ntr1

You can type your desired name for the component or stuff in this box. The only constraint is that each label must be unique. CyclePad will not allow you to enter a duplicate caption label.

See also:

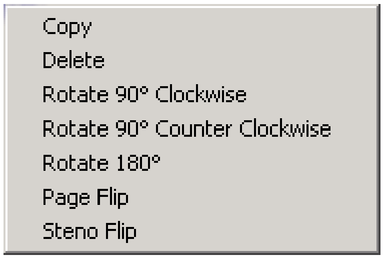
Creating a New Design
Adding Components to the Design
Connecting Components
Manipulating Component Icons
Deleting Stuffs

Top

Manipulating Component Icons

In building larger cycles, you may want to move, copy, delete, rotate, or flip a device icon. Moving is easy; just left-click on the icon and drag it to where you want it. (It takes CyclePad a second to decide that you are trying to move a component, so make sure that the color of the component changes before you move it.) Although you can move connected icons, be aware that the connections can restrict how much you can move a given icon, so try to get devices in roughly the right place before you connect them.

As for the other things you can do to device icons, a right click on a component icon in build mode will pop-up the edit-icon menu:



Copy — Copies the selected icon in its current orientation. This can be faster than adding and rotating several icons of same type.

Delete — Deletes the selected icon

Rotate 90 Clockwise — Rotates the icon ninety degrees clockwise.

Rotate 90 Counter-Clockwise — Rotates the icon ninety degrees counter-clockwise.

Page flip — Reflects the icon about its vertical axis, as though you were turning the page of a book. Flips are especially important for asymmetrical icons such as heat exchangers.

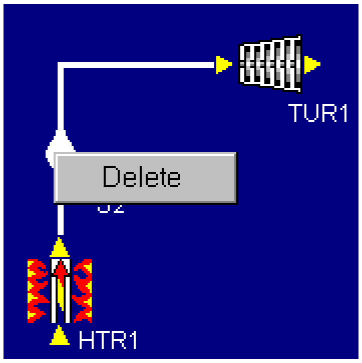
Steno flip — Reflects the icon about its horizontal axis, as though you were turning the page of a steno pad or reporter's notebook.

See also:

Creating a New Design
Adding Components to the Design
Connecting Components
Changing the Labels of Components and Stuffs
Deleting Stuffs

Top

Deleting Stuffs



The edit-stuff menu is considerably simpler, since the only thing you can do with stuffs is delete them. You pop-up this menu by right-clicking on the stuff's bulge.

When you have completed the design of your cycle, by connecting all components to one another, you can now move to Analyze Mode by choosing Analyze from the Mode menu in the menubar.

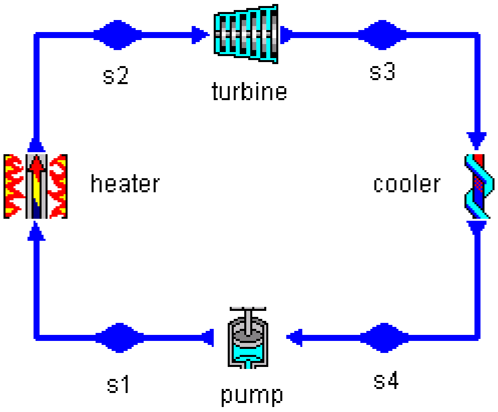
See also:

Creating a New Design
Adding Components to the Design
Connecting Components
Changing the Labels of Components and Stuffs
Manipulating Component Icons

Top

Analyze Mode

In Analyze Mode the screens background color changes to white. Your primary means of interaction with CyclePad in analyze mode is via meters, which are pop-up windows that display information about components, stuffs, and the cycle as a whole. Clicking on a component of stuff brings up its meter. Other meters, such as those which display global system properties (like thermal efficiency) are accessable through the Cycle item in the menu bar.



Meters

Interacting with Meters - Choosing a Substance

Interacting with Meters - Making Modeling Assumptions

Interacting with Meters - Assuming Numeric Values

Interacting with Meters - Derived Values

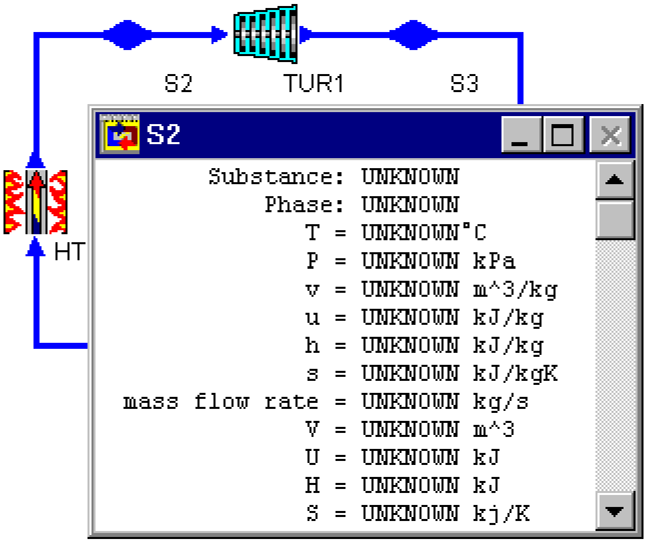
Investigating a Cycle via the Explanation System

Investigating a Cycle via the Sensitivity Analysis Tool

Top

Meters

A left click on any icon (including the cycle icon) will pop-up a meter that displays the modeling assumptions and parametric values associated with that entity. For example, a left click on the S1 stuff icon pops up the meter for that stuff:



S2

```
       Substance: UNKNOWN
           Phase: UNKNOWN
               T = UNKNOWN°C
               P = UNKNOWN kPa
               v = UNKNOWN m^3/kg
               u = UNKNOWN kJ/kg
               h = UNKNOWN kJ/kg
               s = UNKNOWN kJ/kgK
  mass flow rate = UNKNOWN kg/s
               V = UNKNOWN m^3
               U = UNKNOWN kJ
               H = UNKNOWN kJ
               S = UNKNOWN kj/K
```

Initially everything in the meter is black, since the cycle's parameters are as yet unspecified, but as you make assumptions, you will find that the meter displays them in color.

Top

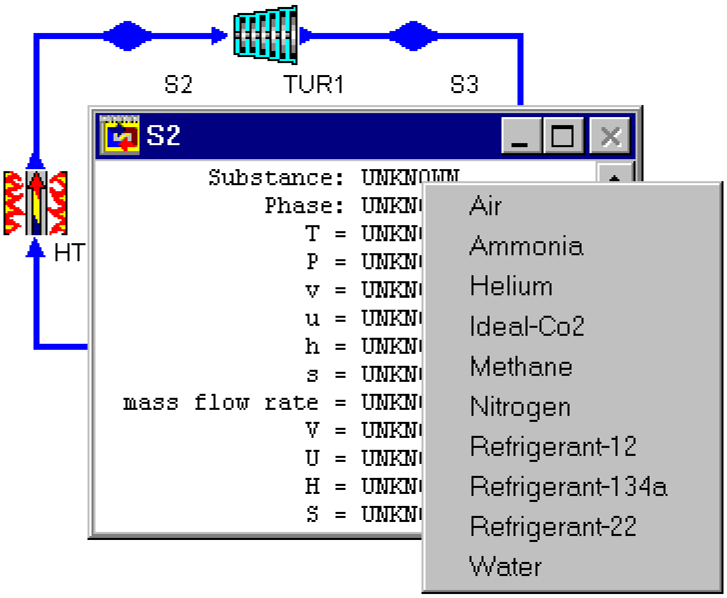
Meter Color Coding

Green	Your assumptions
Blue	Information derived from your assumptions
Black	Information that is unknown or cannot change (e.g., the names of components)
Red	Notification of a problem

Top

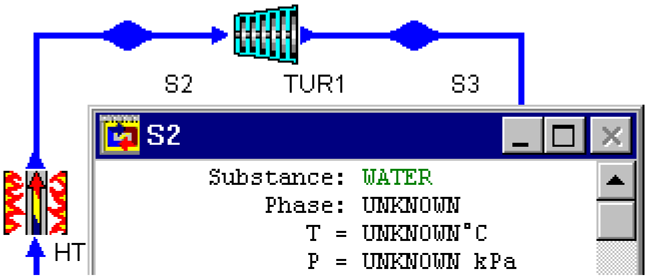
Interacting with Meters - Choosing a Substance

Each of the `UNKNOWN` values is mouse-sensitive. Clicking on one pops up a menu of things you can either ask about that value, via the hypertext system, or things you can do to that value, such as changing or retracting it. For example, clicking on "Select a substance for S2" pops-up the substance menu:



```
S2
      Substance: UNKNOWN
          Phase: UNKN
              T = UNKN
              P = UNKN
              v = UNKN
              u = UNKN
              h = UNKN
              s = UNKN
 mass flow rate = UNKN
              V = UNKN
              U = UNKN
              H = UNKN
              S = UNKN
```

Let's suppose that we choose Water from the substance list. The meter now changes to reflect this choice. Note that water is in green, to indicate it is an assumption that we have made.

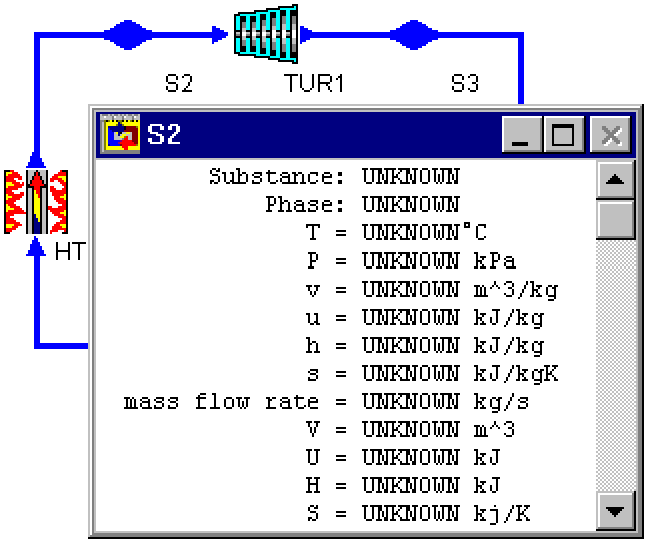


```
S2
      Substance: WATER
          Phase: UNKNOWN
              T = UNKNOWN°C
              P = UNKNOWN kPa
```

Top

Interacting with Meters - Making Modeling Assumptions

The proper use of modeling assumptions is critical to successful modeling. They indicate that a device is adiabatic (contributes no heat flow to the working fluid), isothermal (works without changing the temperature of the working fluid), et cetera. You can make and retract modeling assumptions by clicking on the first line of a component's meter, which will initially read "`Modeled as: Make Assumption.`"



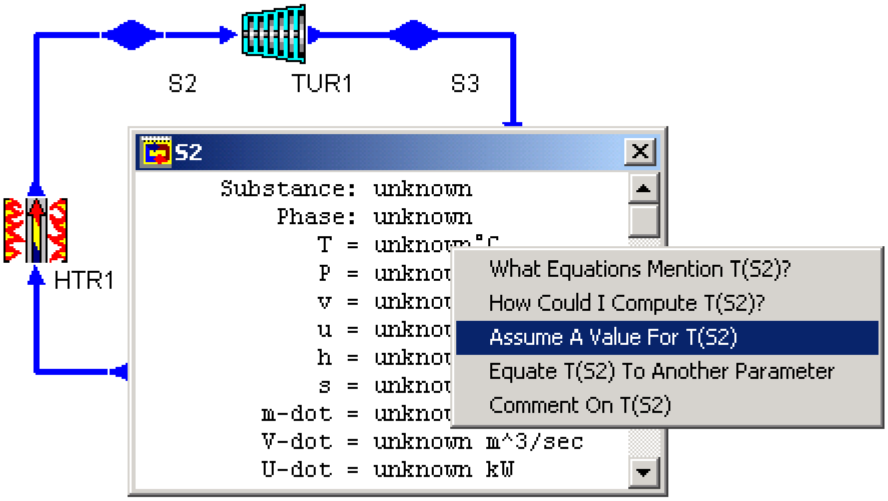
```
S2
          Substance:  UNKNOWN
              Phase:  UNKNOWN
                  T = UNKNOWN°C
                  P = UNKNOWN kPa
                  v = UNKNOWN m^3/kg
                  u = UNKNOWN kJ/kg
                  h = UNKNOWN kJ/kg
                  s = UNKNOWN kJ/kgK
    mass flow rate = UNKNOWN kg/s
                  V = UNKNOWN m^3
                  U = UNKNOWN kJ
                  H = UNKNOWN kJ
                  S = UNKNOWN kj/K
```

The assumptions that CyclePad allows are detailed here.

Top

Interacting with Meters - Assuming and Changing Numeric Values

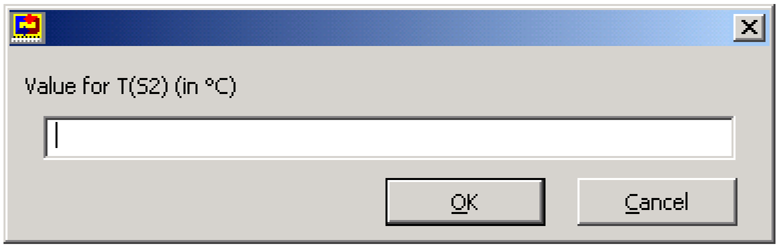
Now let's choose a temperature (by clicking on the `UNKNOWN` to the right of temperature):


S2
TUR1
S3
HTR1
What Equations Mention T(S2)?
How Could I Compute T(S2)?
Assume A Value For T(S2)
Equate T(S2) To Another Parameter
Comment On T(S2)

```
S2
     Substance:  unknown
         Phase:  unknown
             T = unknown°C
             P = unknow
             v = unknow
             u = unknow
             h = unknow
             s = unknow
         m-dot = unknow
         V-dot = unknown m^3/sec
         U-dot = unknown kW
```

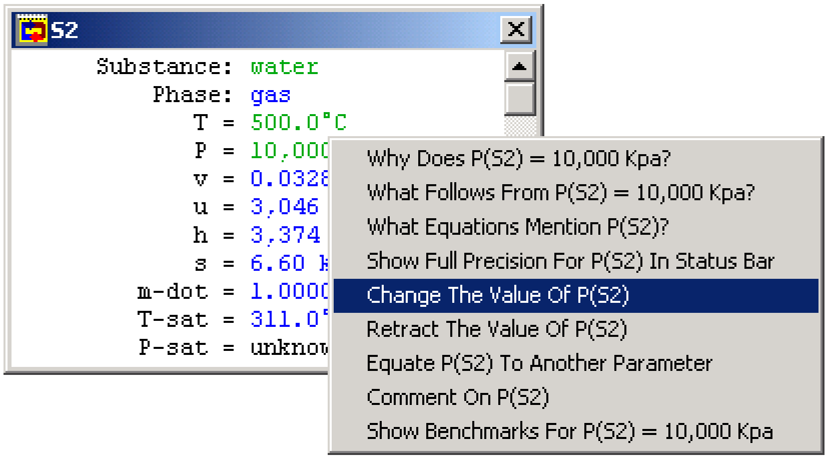
A pop-up menu appears, displaying some questions and actions, including the Assume a value... command for this parameter. The questions provide access to the Hypertext system, which will become more useful as we proceed in the analysis.

Clicking on Assume a value for temperature enables us to set a value for this parameter:



Value for T(S2) (in °C)

Note that the units are indicated in the prompt. If you would prefer to use different units, choose Change Units from the Preferences page under the Edit pull-down menu.
Once a numeric value has been assumed, it may also be changed or retracted by left-clicking on the value.



S2

```
Substance: water
    Phase: gas
        T = 500.0°C
        P = 10,000
        v = 0.0328
        u = 3,046
        h = 3,374
        s = 6.60 k
    m-dot = 1.0000
    T-sat = 311.0°
    P-sat = unknow
```

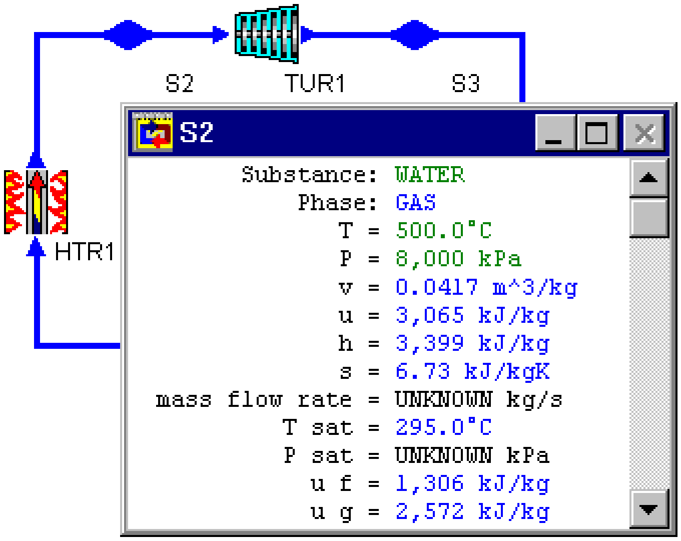
Why Does P(S2) = 10,000 Kpa?
What Follows From P(S2) = 10,000 Kpa?
What Equations Mention P(S2)?
Show Full Precision For P(S2) In Status Bar
Change The Value Of P(S2)
Retract The Value Of P(S2)
Equate P(S2) To Another Parameter
Comment On P(S2)
Show Benchmarks For P(S2) = 10,000 Kpa

❖❖❖

Top

Interacting with Meters - Derived Values

Let's assume that you have set the pressure to be 8000 kPa. Note that CyclePad has now used the information you have provided to calculate several values, such as the specific volume of the working fluid and several of its saturation property values. These values appear in blue in the meter window.



```
      Substance: WATER
          Phase: GAS
              T = 500.0°C
              P = 8,000 kPa
              v = 0.0417 m^3/kg
              u = 3,065 kJ/kg
              h = 3,399 kJ/kg
              s = 6.73 kJ/kgK
 mass flow rate = UNKNOWN kg/s
          T sat = 295.0°C
          P sat = UNKNOWN kPa
            u f = 1,306 kJ/kg
            u g = 2,572 kJ/kg
```

Your analysis proceeds by adding information to CyclePad until you have all the answers you want. You can make assumptions and ask questions in any order you like. *As you go, and when you are finished, don't forget to save your work!*

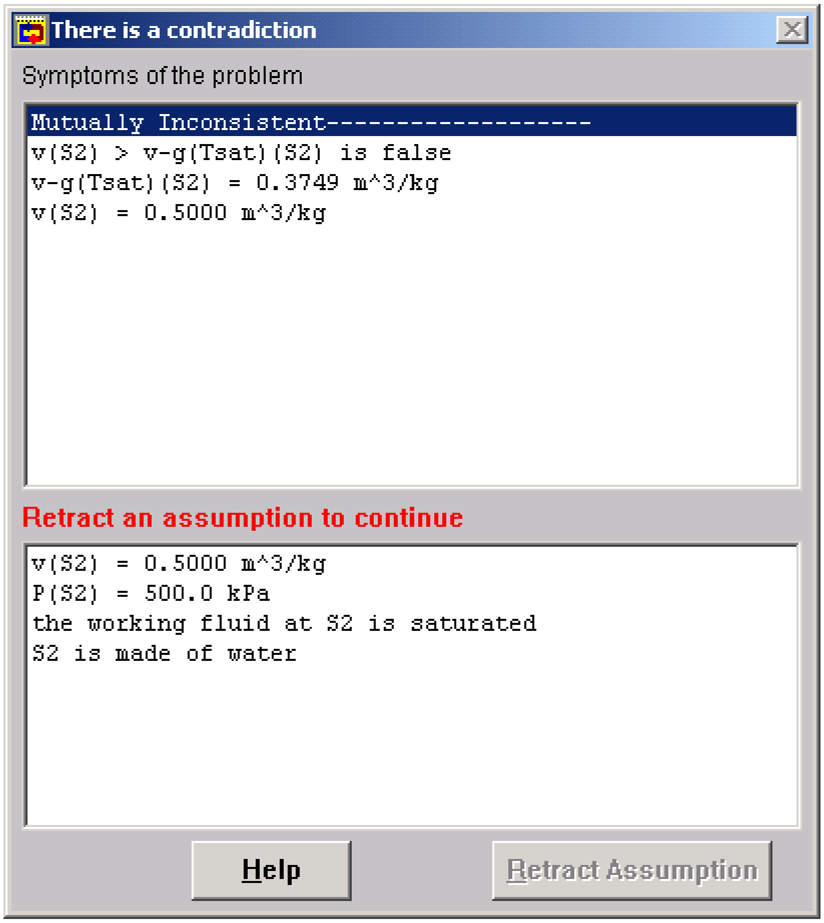
As you assume parameter values, you are in effect constraining the values that the other parameters can take on. You may inadvertently overconstrain the value of a parameter, in which case it can either not have a value at all, or CyclePad finds that it is calculating two or more different values for that parameter. When this happens, the contradiction dialog will appear. It displays the assumptions implicated in the contradiction, and you must retract one or more of them to resolve the contradiction before you can continue with your analysis of the cycle.

When you have derived the values you are looking for, you may want to investigate the relationship between a value that you have assumed and one that CyclePad has calculated. You use the Sensitivity Analysis tool, from the Tools pulldown menu, to do this.

Top

Contradiction Mode

It is possible to make assumptions that conflict with each other. In such a case, CyclePad cannot continue to analyze your design until one or more of the conflicting assumptions has been retracted. When CyclePad detects a conflict, it enters Contradiction Mode. The Contradiction Resolution window appears on the screen to inform you that there is a contradiction:



There is a contradiction

Symptoms of the problem

```
Mutually Inconsistent-------------------
v(S2) > v-g(Tsat)(S2) is false
v-g(Tsat)(S2) = 0.3749 m^3/kg
v(S2) = 0.5000 m^3/kg
```

Retract an assumption to continue

```
v(S2) = 0.5000 m^3/kg
P(S2) = 500.0 kPa
the working fluid at S2 is saturated
S2 is made of water
```

How are contradictions caused? Common examples include assuming a particular phase for a working fluid at some point in the cycle that is inconsistent with the assumptions about its state variables, or setting the pressures or temperatures across a component in such a way that it cannot function.

Often, several assumptions lead to a contradiction. In the example shown above, we have specified that the pressure is 500 kPa and that we are working with water. This allows CyclePad to determine the saturation properties for the working fluid, including that its specific volume for vapor is 0.3749 m3/kg. We have also assumed that the working fluid is, in fact saturated. Finally, we assumed the specific volume to be 0.5 m3/kg. The contradiction occurs because there is no way for water to be saturated yet have a specific volume greater than its saturation vapor specific volume. (If it did, it would be a gas.) This example allows us to retract any one of the four assumptions that lead to this contradiction.

How Does CyclePad Detect Contradictions?

How Do I Get Out of Contradiction Mode?

Top

How Does CyclePad Detect Contradictions?

CyclePad knows the baseline rules and equations for thermodynamic systems. It uses these rules to determine when, for example, a number is outside the range it is allowed or when an equation will result in a property having a different value from the value derived another way, such as by assumption or table lookup.

Another important aspect of carrying out any form of mathematical analysis is to check whether or not your results are physically meaningful. CyclePad helps in this respect by detecting when the assumptions you have made correspond to a physically impossible situation. CyclePad detects such situations by using an approximation of the kinds of intuitive knowledge that human engineers use to catch such mistakes. For example, CyclePad knows that

- To make compression occur you must perform work.
- An expansion process generates work.
- If you put stuff through a heater then it cannot come out cooler than it came in.
- If you put stuff through a cooler then it cannot come out hotter than it came in.

When these or other simple physical intuitions are violated, CyclePad switches to contradiction mode. In this mode, the report window shows you the particular intuitions that were violated and the list of assumptions which caused the contradiction.

This style of computation is called constraint propagation. It is a very simple yet powerful way to calculate answers given numerical assumptions. Because it uses constraint propagation, CyclePad does more work than it has to, strictly speaking. When analyzing designs by hand, one generally thinks carefully about the solution process, in order to do the minimal work necessary to deduce the information we need for our goals. The advantage of CyclePad's method is that it is relentless in checking all the consequences of your assumptions, not just the ones you might have thought about checking. This means that CyclePad is reasonably good at finding out when the numbers you have proposed are physically not possible.

See also:

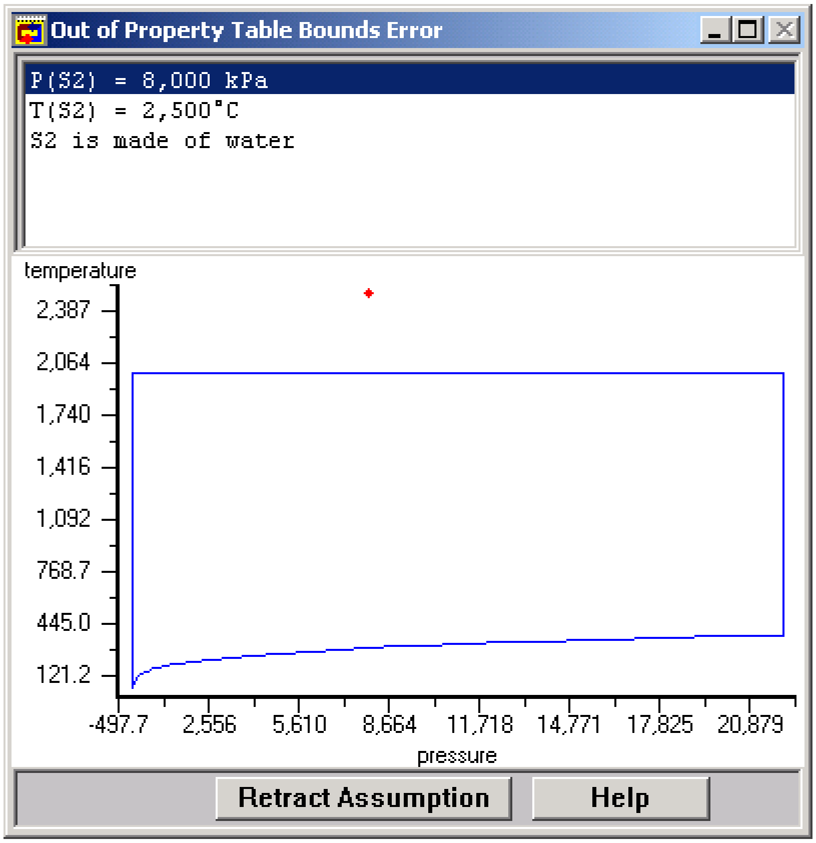
How Do I Get Out of Contradiction Mode?

Top

How Do I Get Out of Contradiction Mode?

Once you enter contradiction mode, CyclePad will not let you proceed until you clear one of the assumptions underlaying the contradiction. To clear the assumption you must retract one of the assumptions in the report window. Once you have done so, click on OK and CyclePad will check to see that all contradictions (there may be more than one) have been resolved. You can use the facilities of the explanation system to figure out which of your assumptions to retract.

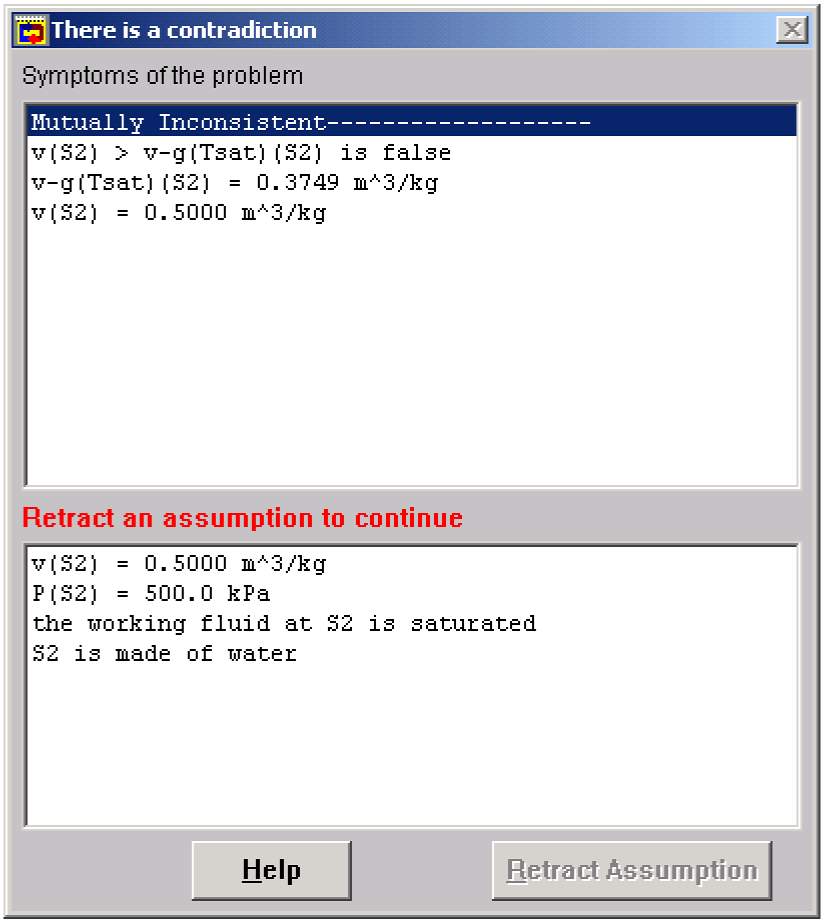
If you have assumed a numerical value for a substance that is outside the range of CyclePad’s tables, the contradiction dialog will show you where your value is with respect to the approximate range of the table:



```
P(S2) = 8,000 kPa
T(S2) = 2,500°C
S2 is made of water
```

This is probably the easiest contradiction to recover from, because it was very likely provoked by the most recent value you entered. In the example shown, a temperature of 2500 °C is above our table range, whose boundaries are shown on the plot. In any case, it will be relatively easy to determine which value is out of range, and retract that value.

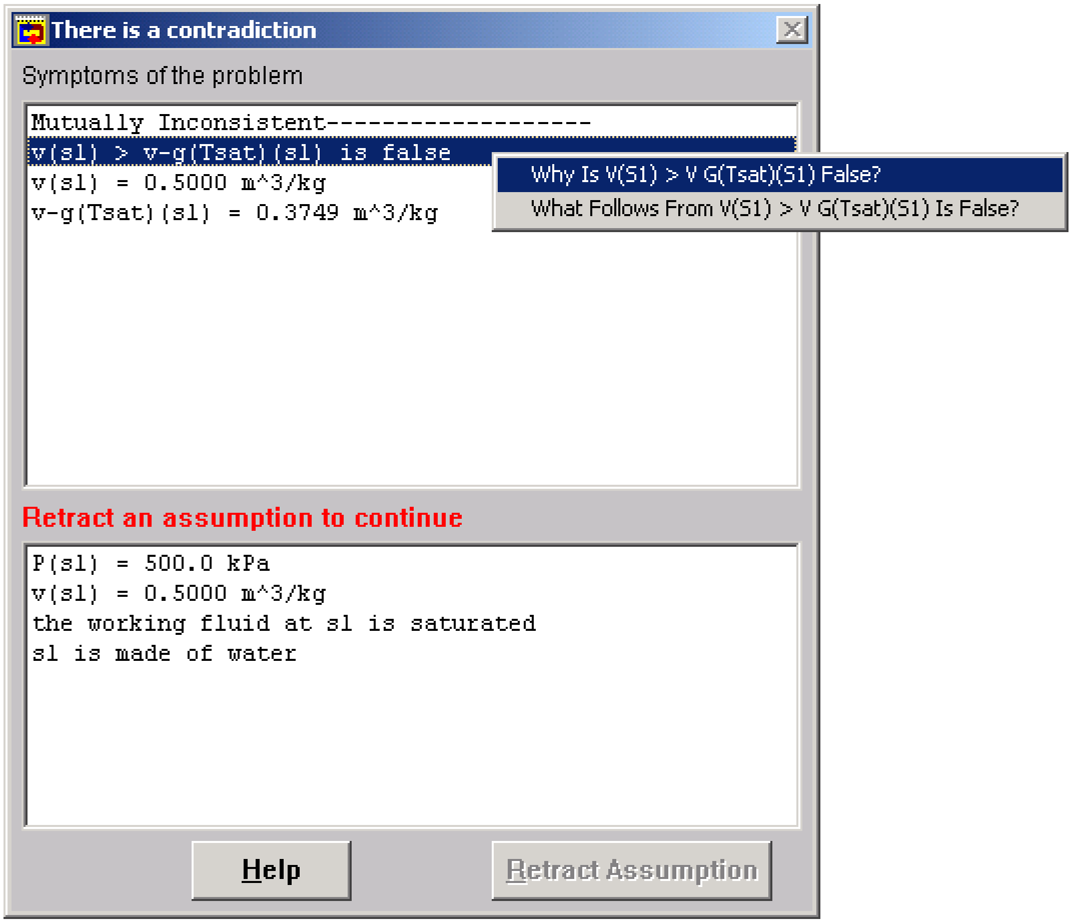
If you do not see this dialog, then the contradiction is not due to a value falling outside of CyclePad's tables.



```
Mutually Inconsistent-------------------
v(S2) > v-g(Tsat)(S2) is false
v-g(Tsat)(S2) = 0.3749 m^3/kg
v(S2) = 0.5000 m^3/kg
```

```
v(S2) = 0.5000 m^3/kg
P(S2) = 500.0 kPa
the working fluid at S2 is saturated
S2 is made of water
```

Depending on the particular problem, there are several ways to proceed. You may find it helpful to work backwards, asking why the contradictory facts were believed and working back to the assumptions. Or you may find it helpful to work forward from your assumptions, moving through their consequences to see where things went astray. Often a combination of working forwards and backwards provides the best results.



There is a contradiction

Symptoms of the problem

```
Mutually Inconsistent-------------------
v(s1) > v-g(Tsat)(s1) is false
v(s1) = 0.5000 m^3/kg
v-g(Tsat)(s1) = 0.3749 m^3/kg
```

Why Is V(S1) > V G(Tsat)(S1) False?
What Follows From V(S1) > V G(Tsat)(S1) Is False?

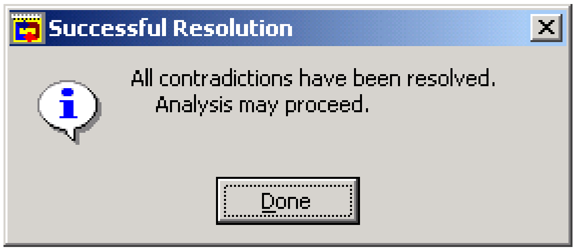
Retract an assumption to continue

```
P(s1) = 500.0 kPa
v(s1) = 0.5000 m^3/kg
the working fluid at s1 is saturated
s1 is made of water
```

In the above example, asking "Why is V(S1) > V G(Tsat)(S1) False?" will pop up an explanation window telling you that S1 is not a gas. If we continue along this line of questions, we will find that S1 is not a gas because we have assumed that S1 is saturated. Exploring in this manner, eventually we will learn how one of our assumptions has led to a contradiction.

All contradictory statements and assumptions are hypertext items that can be explored and manipulated by clicking on them with the mouse to get a menu of appropriate commands.

Once an assumption underlaying a contradiction has been retracted, CyclePad removes the contradiction window and returns you to continue your analysis.



Successful Resolution

All contradictions have been resolved.
Analysis may proceed.

See also:

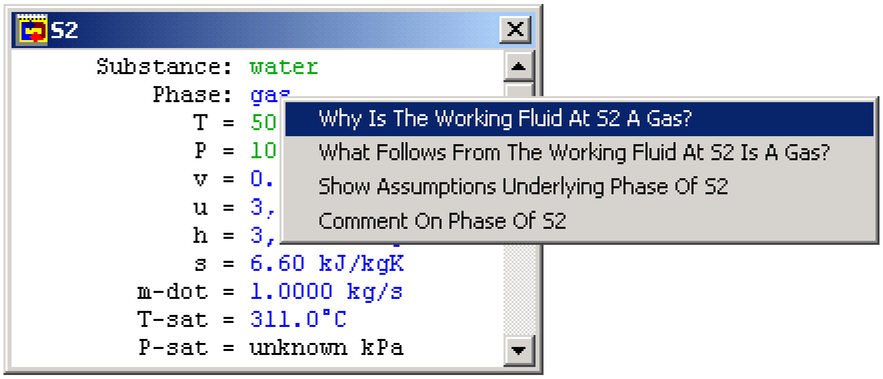
Determining how CyclePad derives values

Top

Investigating a Cycle via the Explanation System

CyclePad automates the numerical analysis of cycles so that you can spend more time thinking about the implications of your design. To assist you in your investigations, CyclePad provides a hypertext-based query system. Using this system, you can obtain explanations of any parametric value, substance phase, or modeling assumption that CyclePad has derived. (These values appear in blue text on your screen).

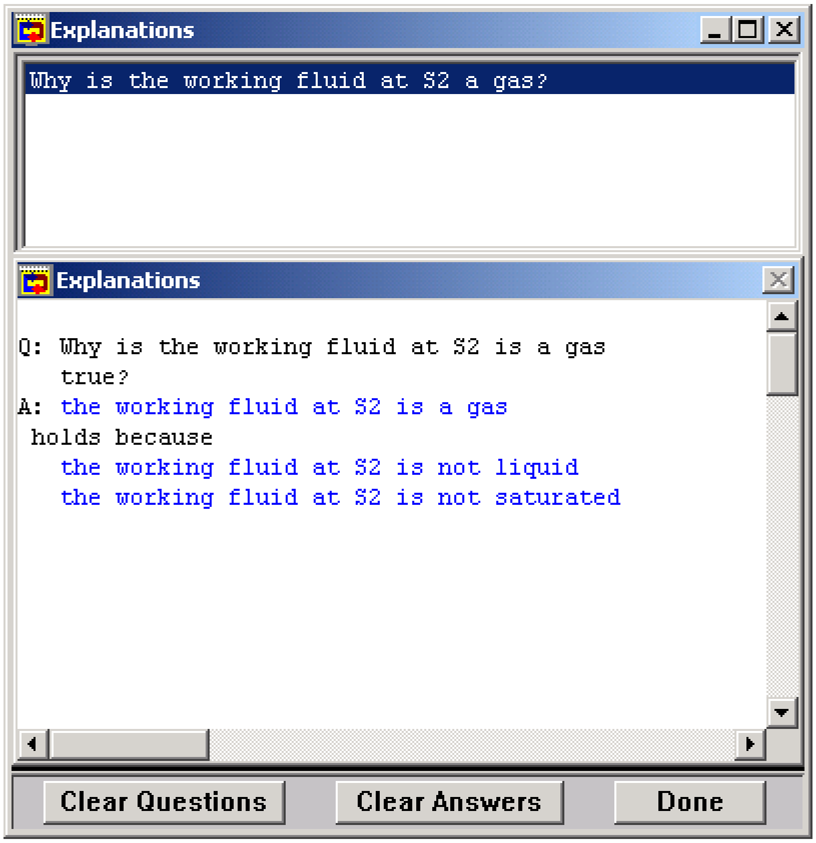
Let's suppose that you are analyzing a simple Rankine cycle, and you have just filled in values for the mass-flow, pressure, and temperature of the stuff leaving the boiler. You notice that CyclePad has deduced that the stuff is a gas, and you want to know why that is so. Clicking on that statement pops up a menu of questions you can ask:



```
S2
  Substance: water
      Phase: gas
          T = 50
          P = 10
          v = 0.
          u = 3,
          h = 3,
          s = 6.60 kJ/kgK
      m-dot = 1.0000 kg/s
      T-sat = 311.0°C
      P-sat = unknown kPa
```

Why Is The Working Fluid At S2 A Gas?
What Follows From The Working Fluid At S2 Is A Gas?
Show Assumptions Underlying Phase Of S2
Comment On Phase Of S2

Clicking on "Why is the working fluid at S2 a gas true?" causes the Explanations Window to appear:

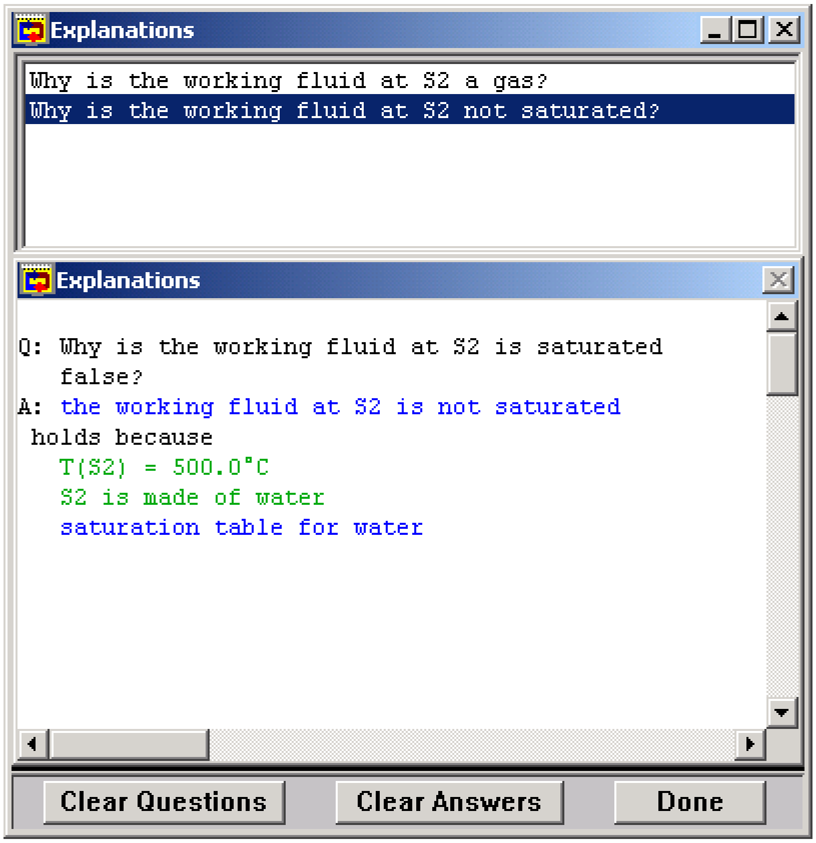


Why is the working fluid at S2 a gas?

```
Q: Why is the working fluid at S2 is a gas
   true?
A: the working fluid at S2 is a gas
 holds because
   the working fluid at S2 is not liquid
   the working fluid at S2 is not saturated
```

A record of your queries appears in the gray box at the top of this window. You can go back to any question you have asked by clicking on it, or you can clear the queries record by clicking on the Clear Questions button.

The white box at the bottom of the window displays the current question and its answer. In the above example, note that some of the text is blue, indicating CyclePad-derived values. These pieces of text are themselves sensitive to mouse clicks, so if you want to know why CyclePad believes that the phase of S2 is not saturated, you would click on `the working fluid at S2 is not saturated` and ask why that is true.



```
Why is the working fluid at S2 a gas?
Why is the working fluid at S2 not saturated?
```

```
Q: Why is the working fluid at S2 is saturated
   false?
A: the working fluid at S2 is not saturated
 holds because
   T(S2) = 500.0°C
   S2 is made of water
   saturation table for water
```

Notice that clicking on the blue text pops up a list of questions that you can ask, just as your original click on the statement `Phase: Gas` popped up a list of questions. Choosing "Why is the working fluid at S2 a gas true?" places that question in the record of queries and displays the answer in the lower box.

Notice that `T(S2) = 500.0°C` is in green, indicating that it is a value you assumed. You can click on it, but CyclePad will only inform you that it's your value! However, you can generally click on most blue values and eventually determine either which of your assumptions have allowed CyclePad to find the value it is showing or which of the underlaying assumptions is one of CyclePad's background facts. For example, clicking on `saturation table for water` will show that this is just a background fact, something that CyclePad has in its knowledge base.

Top

A Summary of Explanation System Question Types

Each time you query a CyclePad-derived statement, CyclePad pops up a list of questions. Here is a list of all the questions you could ask, along with the context(s) in which they appear:

If the item is a parameter *parameter* whose value is unknown:
    What equations mention *parameter*?
    How could I compute *parameter*?
    Assume a value for *parameter*.

If the item is a parameter *parameter* whose derived value is *numeric-value*:
- Why does *parameter* = *numeric-value*?
- What follows from *parameter* = *numeric-value*?
- Show benchmarks for *parameter*.

If the item is a parameter *parameter* whose assumed value is *numeric-value*:
- Why does *parameter* = *numeric-value*?
- What follows from *parameter* = *numeric-value*?
- Retract the value of *parameter*?
- Change the value of *parameter*?
- Show benchmarks for *parameter*.

If the item is an equation *equation*,
- Why does *equation* hold?
- How has *equation* been used?

If the item is a component *component*, then for each modeling assumption *true modeling assumption* which holds about *component* and each modeling assumption *unknown-modeling-assumption* that is not known to be either true or false about *component*:
- How has *true-modeling-assumption* been used?
- When is *true-modeling-assumption* appropriate?
- Retract the assumption *true-modeling-assumption*.
- Assume that *unknown-modeling-assumption*.

If the item is a statement about the substance *substance* of stuff *stuff*:
- If the substance is currently unknown:
  - Select a substance for *stuff*.
- If the substance is currently known via derivation:
  - Why is *stuff* made of *substance*?
  - What follows from *stuff* made of *substance*?
  - Show assumptions underlaying *stuff* made of *substance*.
- If the substance is currently known via assumption:
  - Retract *parameter* as the substance of *stuff*.

If the item is a statement about the phase *phase-parameter* of stuff *stuff*:
- If the phase is currently unknown:
  - Select a phase for *stuff*
  - What phase can't *stuff* be?
- If the phase is currently known:
  - Why is *stuff phase-parameter*?
  - What phase can't *stuff* be?
- If the phase is currently known via assumption:
  - Retract *parameter* as the phase of *stuff*.

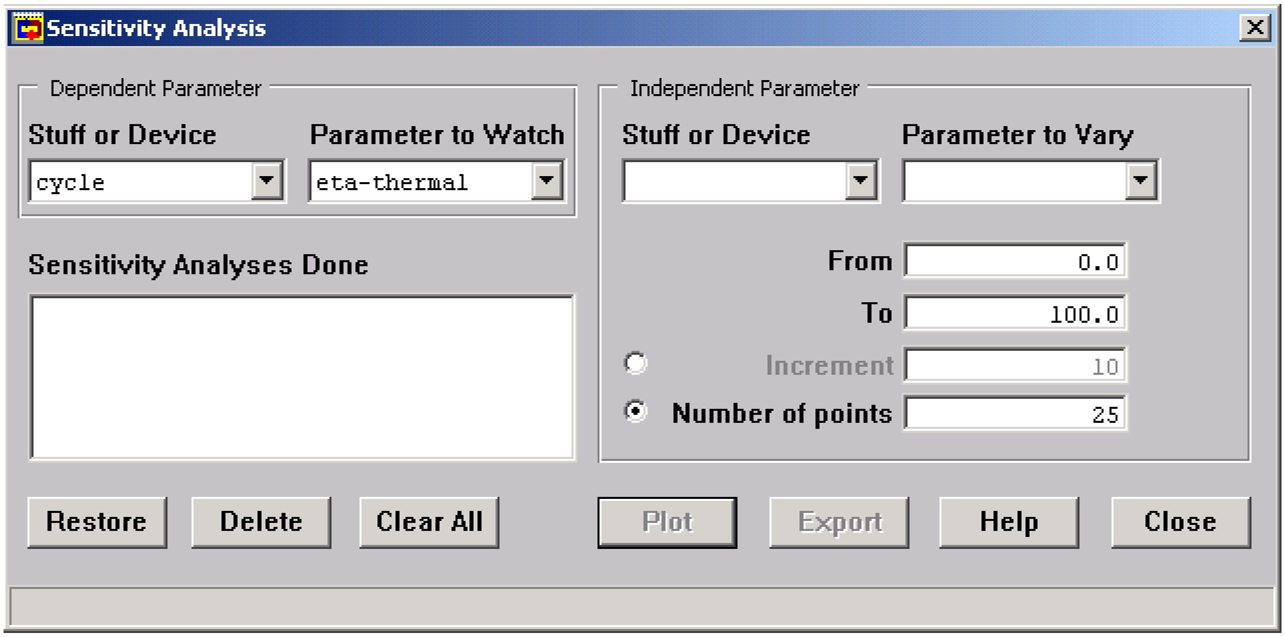
Top

Investigating a Cycle via the Sensitivity Analysis Tool

To figure out how to improve your design or to more generally explore how changing one parameter affects the value of another, use the sensitivity analysis tool. The form of sensitivity analysis implemented in CyclePad allows you to plot how the value of a dependent parameter changes as the value of an independent parameter varies.

From a design standpoint, the dependent parameter is generally what you would like to improve, such as the thermal efficiency, work produced, or reducing the amount of waste heat. The independent parameter is one of your numerical assumptions whose value affects the dependent parameter, such as the outlet temperature of a boiler or the pressure ratio of a compressor, that you wish to change in order to affect the dependent parameter.

Selecting Sensitivity from the Tools pull-down menu pops-up the Sensitivity Analysis dialog box:



Dependent Parameter

Stuff or Device: cycle

Parameter to Watch: eta-thermal

Independent Parameter

Stuff or Device:

Parameter to Vary:

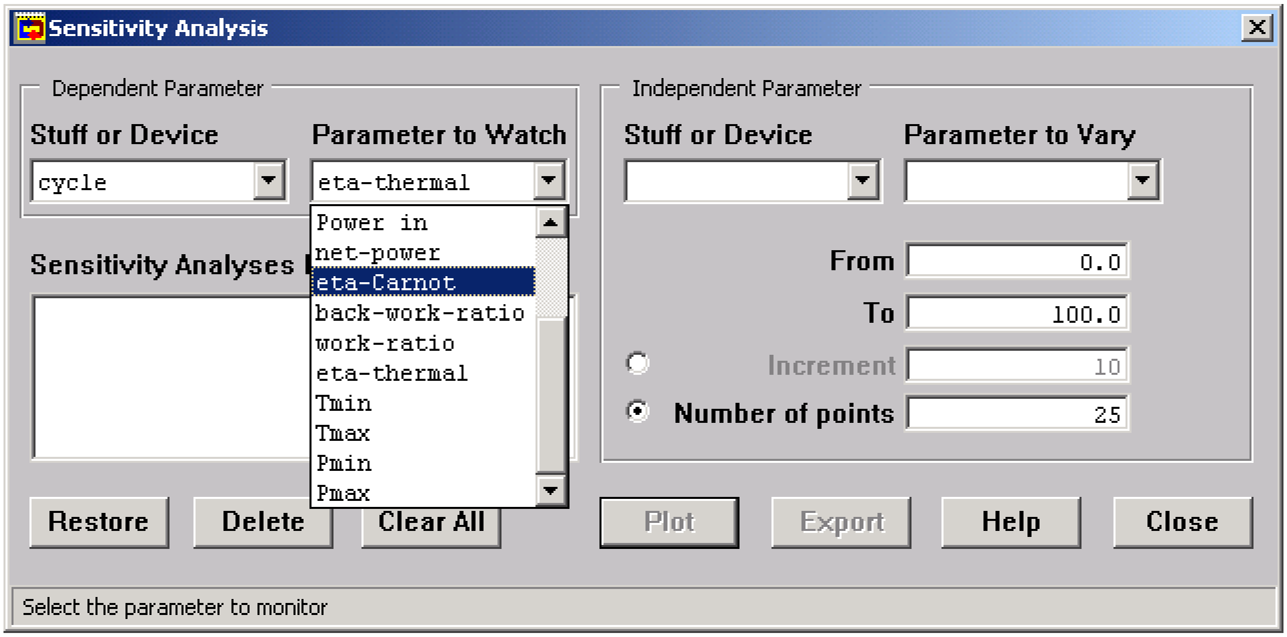
From: 0.0

To: 100.0

○ Increment: 10

● Number of points: 25

In the upper left of this dialog, you select the parameter of the stuff or device that you would like to watch. By default, this dialog box shows the Cycle as the device and Eta-thermal (the cycle efficiency) as the parameter to affect. Notice that each of the four boxes across the top half of the screen have buttons to their right with arrows pointing down. Clicking on these buttons displays a drop-down menu of the possible choices for that box. In the figure below, you can see that Work Ratio, Eta-Thermal, Min-T, and Max-T (among others) are all possible dependent parameters of the cycle.



Stuff or Device: cycle

Parameter to Watch: eta-thermal

Stuff or Device:

Parameter to Vary:

From 0.0

To 100.0

○ Increment 10

● Number of points 25

Select the parameter to monitor

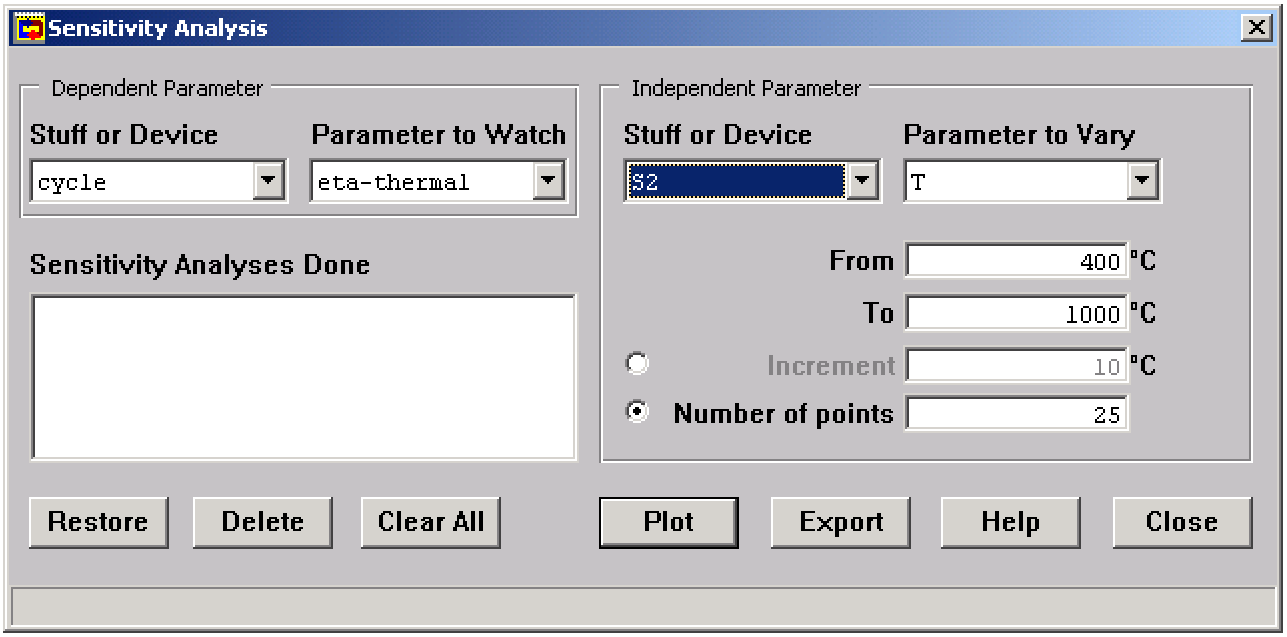
In the upper right half of the dialog, you choose the parameter of the stuff or device that you want to vary. These two boxes display only those parameters for which you assumed a value during the analysis phase, because these are the only parameters you can directly vary.

Before you can plot the pressure, you must provide a reasonable range of values for your independent parameter. By default, this range is 0 to 100 in the units of the dependent parameter. You can change it in the lower left part of the screen. The number of points to plot, which defaults to 25, determines the appearance of the resulting graph; more points will make the plot smoother but take more time, particularly if we are using slow but safe sensitivity (see below).

Clicking on the Plot Data button pops up a graph of the behavior of the dependent parameter over the range of the independent parameter:

Moving the mouse pointer over the red points causes the exact x and y values to be displayed in the status bar at the bottom of the screen.

Suppose, for instance, that we wish to examine how the temperature of steam entering the turbine changes the thermal efficiency of a Rankine cycle. We set up the following sensitivity analysis.



Sensitivity Analysis

Dependent Parameter

Stuff or Device: cycle

Parameter to Watch: eta-thermal

Sensitivity Analyses Done

Independent Parameter

Stuff or Device: S2

Parameter to Vary: T

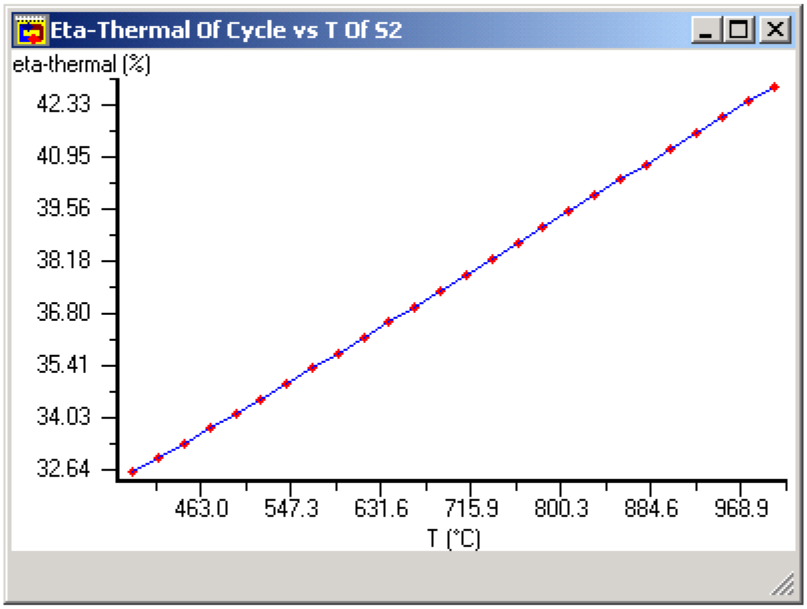
From 400 °C

To 1000 °C

○ Increment 10 °C

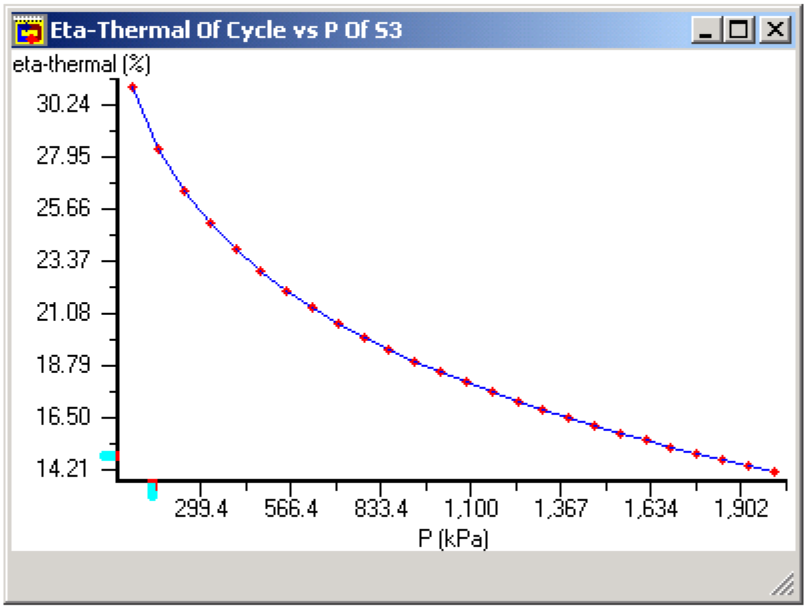
● Number of points 25

Which produces a plot showing the relation between turbine inlet temperature and the cycle's thermal efficiency.

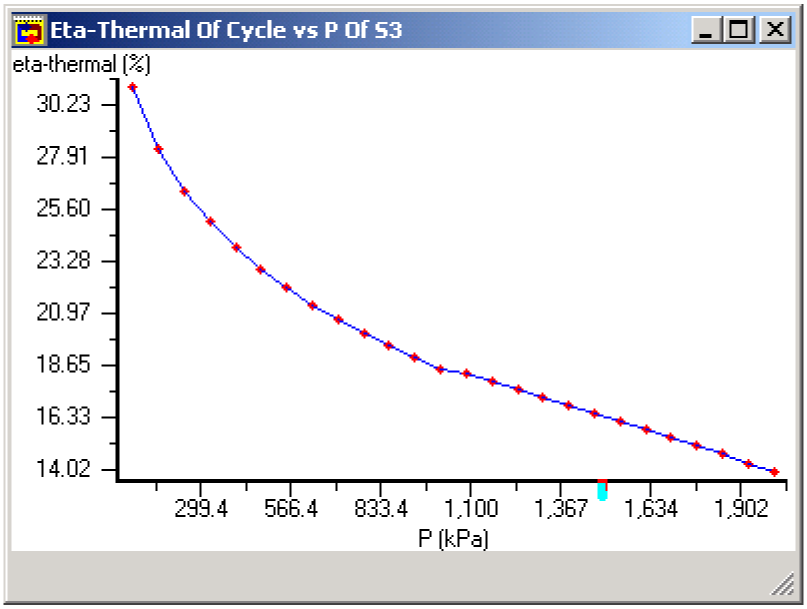


A concern in performing sensitivity analyses arises in some cycles. CyclePad ordinarily uses an analytical shortcut to speed up the process of generating points for sensitivity analysis plots. The shortcut assumes things like the phase of all the stuffs will stay the same. For certain cycles, this assumption isn't appropriate because, for instance, a state will change from saturated to superheated as we vary the dependent parameter. For such cycles, we must turn on the "Slow But Safe Sensitivity" option under the Preferences | Advanced menu item.

For example, consider the same Rankine cycle for which we plotted Thermal Efficiency versus Turbine Inlet Temperature above. Suppose that we were interested in how the thermal efficiency depended on the turbine outlet pressure, which is 100 kPa in the default design and which results in the turbine outlet state being saturated. If we ask CyclePad to plot eta-thermal as P at S3 varies from 50 kPa to 2 MPa, we get the following plot



However, since as we raise the outlet pressure the turbine outlet phase eventually becomes a gas. If we turn use slow but safe sensitivity, we get a slightly different outcome.



The difference, though not huge, exists because the slow but safe plot accounts for the transition of the turbine outlet state from saturated to superheated at around 1000 kPa.

Top

Economic Model

CyclePad has a fairly simple economic model that you can activate by choosing Preferences|Advanced and checking Consider Economics. We describe the underlying model in this section.

The punchline of an economic analysis is the Net Present Value, or NPV, of the cycle you have created. This is how much the cycle is worth to us today, given the interest rate at which we can invest our money and the amount of money we expect to make once we construct the cycle, or in other words, our revenue. In some applications, such as building a power-plant for a ship, there won't be any directly attributable revenue, and so the focus of the economic analysis will shift to the present value (PV) of the cost of cycle.

The concept of present value is based on the fact that a dollar today is worth more than a dollar a year from now. While this might seem intuitive, we can actually figure out exactly how much more a dollar today is worth. To do this, suppose that you have a savings account that pays 7% interest. (For the sake of simplicity in this model, we assume that interest compounds annually). If you deposit a dollar today in this account, it will be worth $1.07 a year from now. Likewise, if we deposit about 93 cents today, you'll have $1.00 a year from now. So a dollar a year from now is worth 93 cents today.

CyclePad calculates a PV of revenue and of cost for heat engines, and just the PV of cost for refrigerators and heat-pumps. Conceptually, the PV of revenue is the sum of all expected future revenue payments, each one discounted back to its equivalent value today. To do this calculation, we need to know several things:

- The interest rate
- The useful lifetime of the cycle
- The expected utilization of the cycle
- The price of a kWh

Each of these is a parameter of the cycle, which you can access via the Cycle|Whole Cycle menu choice. Note that these parameters are only visible when the Consider Economics box is checked on the Preferences|Advanced dialog box. The lifetime of the cycle lets us know how far into the future to project revenues. The utilization tells us how much of a given year we can expect the cycle to be operating, and hence producing revenue. The formula that CyclePad uses to calculate the PV of revenues is:

$$PV_{\text{revenue}} = \sum_{n=1}^{\text{lifetime}} \frac{8760 \times U \times P}{(1+i)^n}$$

where

8760 is the number of hours in a year
U is the utilization rate

P is the price per kilowatt·hour
i is the interest rate
n is the *n*th year.

CyclePad calculates the PV of the cost of the cycle in the same manner. However, figuring the cost of the cycle is a little more involved. In general, there are two types of cost, fixed and variable. A fixed cost is only incurred once over the lifetime of the cycle, and is typically incurred in the construction of the cycle. For example, a turbine might cost $400,000, and if we are an electric utility constructing a new plant, we have to pay General Electric that much money today in order to obtain the turbine. We might also have to pay a contractor another $75,000 to come in with a crane, lift the turbine into place, and make the necessary connections to enable it to operate. In the CyclePad economic model, we consider this $75,000 to be a part of the total cost of the turbine, because we incur it once in the course of constructing the power plant, and it is a direct result of our desire to obtain an operating turbine.

These fixed costs are also called capital costs because they are associated with capital assets, such as turbines. A capital asset is something that persists over time and is used in the process of either making money or obtaining something we value, such as national security.

In contrast, variable costs include the cost of the fuel and maintenance necessary to operate the cycle once it is constructed. Unlike capital costs, which are independent of whether or not we ever use the cycle, variable costs vary directly with the utilization of the cycle. For example, utilities construct two types of plants, baseload and peaking. A baseload plant is expected to run all the time, whereas a peaking plant is only expected to run occasionally, on the hottest and coldest days when the demand for electricity spikes. Because a baseload plant will incur high variable costs, a key factor in its design will be making it as efficient and robust as possible. In contrast, since a peaking plant is used infrequently, designers will tend to choose designs that are less expensive to build but more expensive to operate.

The PV of the cycle cost is calculated by assuming that all capital costs are incurred immediately, but variable costs are incurred over the life of the cycle. To do this calculation, we need to know two things, in addition to the utilization rate, interest rate, and lifetime of the cycle that we needed for the revenue calculation:

- The total capital cost of the cycle
- The total annual cost of fuel
- The total annual maintenance cost

The PV of costs is calculated in a similar manner to the PV for revenue:

$$PV_{\text{cost}} = \sum_{n=1}^{\text{lifetime}} \frac{M + F}{(1 + i)^{n}}$$

where

M is the maintenance cost
F is the annual fuel cost
i is the interest rate
n is the *n*th year.

Each of these costs requires a calculation, so we will discuss them each in turn.

Total Capital Cost

Total capital cost is the sum of the capital cost for each component of the cycle. However, only the components listed in the following table have a cost model. Those components lacking a cost model include mixers, splitters, throttles, sources and sinks. The rationale for this is that these components are little more than conduits (although they can be quite large conduits), and as such are so inexpensive compared to their more complex cousins that their costs would be lost in the noise.

Device	Cost Function Based on	Prerequisite Assumptions	Device is Considered a Stage of a Composite
Turbine	Shaft-power	--	Yes
Compressor	Shaft-power Pressure	--	Yes

	Mass-flow		
Pump	Shaft-power	--	No
Heater	Mass-flow	Boiler	No
	Pressure	Reheater	
Cooler	Area	Condenser	No
Heat-exchanger	Area	Co-current	No
	Pressure	Counter-current	

Although one might think that CyclePad could simply apply a cost function to each device and add up the results, the generality of CyclePad's device representation requires additional assumptions in the case of heaters and coolers. In the case of compressors and turbines, it turns out that many designs call for using a set of these devices in series, so each device actually represents a stage of larger device; to obtain accurate costs, we must be sure to apply the cost function only to this larger device, and not to each individual stage.

A heater can (and must) be used to model both the burners in a jet engine, which are little more than cans with holes punched in them to admit atomized fuel, and the boiler of a baseload plant, which in reality might be seven stories tall. Clearly we cannot capture the cost of each device with one function, so we have added three modeling assumptions for heaters that show up when the control-flag for the economic model is checked. These assumptions are boiler, burner, and reheater. A boiler has its own cost function that includes both the capital cost to build and install the device, and the fuel and maintenance costs required to operate it. To model a reheating vapor cycle, one would use a second heater device to represent the reheater, placing it between two turbines. In actuality, this reheater is not a separate boiler, but in effect a series of pipes that route the working fluid back through the boiler to reheat it. The cost function for such a reheater is modeled by applying a reheater-factor to the cost of the boiler in the system. Consequently, the current CyclePad model is limited to systems containing one boiler, but as such systems comprise the vast majority of vapor power cycles, this limitation should not be too onerous. Finally, the burner assumption sets the cost of the heater equal to zero, on the basis that the cost is minimal in comparison to the rest of the cycle. A reheater in a gas cycle (eg, an afterburner for a jet engine) should therefore be modeled as a burner and not a reheater.

In an similar vein, coolers may be used to model both condensers and the atmosphere that, for example, a gas-jet engine exhausts to. The economic model therefore adds the assumptions condenser and atmosphere to each cooler. The cost model for a condenser assumes that it is a heat-exchanger whose cold side is a "standard" environment of water from a natural source at 20°C. No explicit model of additional cooling capital equipment, such as a cooling tower, is included.

There is a related problem one encounters when dealing with turbines and coolers. In many cases one wants to consider a series of such devices as one aggregate device, as, for example, in a regenerative cycle that bleeds steam from the turbine at several points. Compressors interleaved with coolers or heat-exchangers represent an intercooled compressor. CyclePad automatically aggregates compressors and turbines into what we refer to as stage devices, in that they are comprised of a series of stages. The aggregation is based on an analysis of the structure of the cycle, and only aggregates those devices that do in fact operate in series. For example, the combined cycle in the Library is comprised of two subcycles communicating via a heat-exchanger. The top subcycle contains two turbines, which CyclePad identifies as a stage-turbine, and the bottom subcycle contains a lone turbine, which CyclePad identifies as a separate stage turbine of one stage.

The reason for this aggregation is that the cost functions for devices include a factor for estimating the installation cost, and one would not incur a separate installation cost for each stage of a turbine. The primary complication that stage devices introduce, from the point of view of you, the eternally patient CyclePad user, is that you have to look in the Global Properties menu for the meters that show their properties. When the economic model is active, however, a message to this effect is posted in each compressor and turbine meter.

The basic equation that CyclePad uses to calculate cost is:

$$Cost = \left(\frac{CPB}{CPB_{ref}}\right)^{S} \times Cost_{ref} \times f_{module} \times f_{other}$$

where

CBP is the cost-basis-parameter
*CBP*ref is the reference cost-basis-parameter
*Cost*ref is the reference cost
s is the size-exponent
*f*module is the module factor
*f*other is the combination of other factors

For example, the cost of a turbine is based on its shaft-power, and the cost of a cooler is based on its area. CyclePad's knowledge base contains approximate cost information for each device type, in the form of three parameters, the value of the cost-basis-parameter, the cost of a device operating at that value, and a size-exponent. For example, a $50,000 pump might require 100kW of input shaft-power to operate. These two values are plugged into the above equation in order to determine the cost of a different-size pump, say, one that requires 48kW. The size-exponent governs this relationship. The data in CyclePad is from *Chemical Engineering Economics*, (1989) by Donald E. Garrett.

The module factor accounts for the cost of installing the device, and typically falls in a range of 1.2 to 2.0. The other-factors may include pressure or mass-flow, depending on the device. For example, the cost of a boiler depends on the mass-flow through the device (this is the cost-basis-parameter), but it also depends on the pressure, since it costs more to build a boiler to withstand higher pressures. In the table above, the first parameter is the cost-basis-parameter, and any others listed below it are represented as other factors.

Other factors, such as the pressure factor, are calculated as linear functions of the maximum or minimum parameter value the working fluid attains at the inlet or outlet of the device. These functions were derived from data in *Chemical Engineering Economics.*

In addition to these factors, a material factor is applied to each device, so one must select a material for each device. CyclePad contains information on seven materials, including carbon steel, stainless steel, nickel-alloy steel, aluminum, titanium, molybdenum, and unobtainium. The last is very expensive, but it can withstand extremely high temperatures. The other six materials have limits on the maximum temperature they can withstand, and CyclePad will signal a contradiction if the temperature of the working fluid flowing through the device is higher than the material's limit.

Carbon steel is the least expensive material, and is generally well-suited for components that don't endure particularly high or low temperatures or operate in excessively corrosive environments. The current CyclePad model does not take corrosive effects into account.

	Material	Density	Tmin	Tmax
Material	Factor	kg/m^3	Kelvin	Kelvin
Carbon steel	1.0	7800	125	750
Aluminum	3.0	2700	75	650
Molybdenum	10.0	10300	20	2000
Stainless steel	2.5	8000	100	850
Nickel alloy	5.0	8500	50	1100
Titanium	7.0	10300	200	700
Unobtainium	10^1000	1	0	10^1000

The other materials are superior to carbon steel either in corrosion-resistance, ability to withstand high or low temperatures (e.g., not becoming brittle in a cryogenic cycle), or in density. Tensile strength is not a concern, as all the materials listed are capable of withstanding the highest pressures attained in thermodynamic cycles. The material factors for the materials in CyclePad's knowledge base are relative to carbon steel, which has a factor of 1.0.

Annual Fuel Cost

Heaters that are assumed to be boilers or burners also acquire a fuel property. CyclePad contains information about the cost per unit-weight and the heating (or calorific) values for five fuels: coal, oil, natural-gas, coal-gas, and hydrogen. The annual fuel cost for each heater is calculated by the following formula

$$\text{Annual Fuel Cost} = \frac{Q_{\text{heater}}}{HV_{\text{fuel}}} \times U_{\text{cycle}} \times C_{\text{fuel}}$$

where

*Q*heater is the heat output of the heater
*HV*fuel is the heating value of the fuel
*U*cycle is the cycle's utilization rate
*C*fuel is the unit costof the fuel

This determines how much fuel is necessary to attain the steady-state heat-flow from the heater to the working fluid and multiplies that amount by the utilizatoin rate of the cycle and the cost per unit weight of the chosen fuel.

Maintenance Cost

Maintenance costs are calculated simply as a fraction of the total capital cost. Maintenance for a thermodynamic cycle generally falls into the range of 10-20% of the initial capital cost. If maintenance costs much more than this, say 33%, one would in effect be buying a new cycle every three years or so, and a more robust design that requires less maintenance would be more cost-effective.

Materials Used

CyclePad also contains knowledge about the density of these materials, which it uses to calculate the total weight of the cycle. CyclePad requires that the control flag Consider Velocity be checked in order to do this calculation, because adding velocity information enables the calculation of cross-sectional area. The weight calculation is quite abstract, and should not be relied upon to provide accurate estimates of a cycle's actual weight. However, it is internally consistent, so it will provide accurate *relative* measures of the weight of a cycle constructed with two different materials.

The weight function for each device depends on a geometric abstraction of the shape of that device. For example, the weight function for a turbine models the turbine as a conic section with inlet and outlet diameters derived from the inlet and outlet cross-sectional areas. The thickness of the conic section is assumed to be 12% of the average outer diameter, and a solid inner cylindrical shaft of 20% of the inlet diameter approximates the weight of the turbine blades. The weight models for all components are given in the following table

Device	Model
Compressor	Conic section with solid inner shaft
Turbine	Conic section with solid inner shaft
Heater	Cylinder of length three times diameter
Cooler	Cylinder of length equal to total tube length plus 20% of inner weight for outer casing.
Heat-exchanger	Cylinder of length equal to total tube length plus 20% of inner weight for outer casing.
Mixer	Sphere with diameter derived from cross-sectional area
Splitter	Sphere with diameter derived from cross-sectional area

ThrottleConic section with no shaft
Source--
Sink--

Top

Design and Analysis Coaching

CyclePad incorporates an integrated coaching facility to aid the user in designing and analyzing cycles. To provide this help, CyclePad first makes inferences about the role that each device is playing. A role is defined as the function that a particular device is intended to perform. For example, some gas liquefaction plants use turbines rather than throttles to expand the working fluid because a resisted expansion produces a larger drop in temperature than an unresisted expansion. In this case, even though the cycle may make use of the power derived from the expansion, the intended function of the turbine is to cool the working fluid, not to produce power.

You can view the roles that CyclePad has inferred for your cycle by choosing Tools|Annotations. Left-clicking on the displayed roles will provide an explanation of why CyclePad inferred that role.

CyclePad uses these role inferences to generate advice for improving the analysis of your cycle. To see this advice, choose Tools|Analysis Coach , and any advice that CyclePad has will be displayed in the Explanation Window .

The roles that CyclePad assigns to devices are described in component roles:

Component Roles

Cooler A cooler ejects heat energy from the working fluid to the environment. When acting as a heat-ejector, its purpose is to reduce the energy of the working fluid flowing through it. When acting as a heat-provider, its purpose is to provide heat to the environment. A hot-water radiator is actually a cooler operating as a heat-provider. A heat-ejector role is further partitioned into a fluid-cooler, which acts to cool its working fluid, and an intercooler, which is interleaved with compressor stages and is intended to reduce the work required of later stages by cooling the gas that is being compressed. The intercooler role subsumes the fluid-cooler role, yet we make this distinction because intercoolers are constituents of common thermodynamic structural "idioms" that provide valuable functional information. For example, an intercooler is an indication that the attendant rise in the temperature of the working fluid is an undesirable side-effect.

Heater A heater injects heat energy from the environment to the working fluid. When acting as a heat-injector, its purpose is to increase the energy of the working fluid flowing through it. When acting as a heat-absorber its purpose is to cool the environment by absorbing heat from it. The coils in a domestic refrigerator comprise a heater that absorbs heat energy from the food in the refrigerating compartment. A heat-injector role is partitioned into fluid-heater, preheater, and reheater roles. A preheater adds heat to the working fluid upstream of the main heater (which acts as a fluid-heater) and a reheater adds heat between the stages of a turbine. Both are strategies for increasing the efficiency of the system (the second rational-designer goal), and hence are important roles to distinguish.

Reactor A reactor, like a heater, injects heat-energy into the working fluid. Unlike a heater, however, it does not have a heat-path that crosses the system boundary to connect it to a heat-source in the environment. Reactors, therefore, are never heat-absorbers.

Heat-exchanger The representation of heat-exchangers is slightly more involved. From a structural perspective, a heat-exchanger consists of a device with two inlets and two outlets, but from a functional perspective it is more useful to consider it to be a heater and a cooler connected via a heat-conducting path. We therefore represent heat-exchangers as an hx-heater and an hx-cooler connected via a heat-path, although in CyclePad you'll always see heat-exchangers as a single device, with the cooler indicated by a red-blue pipe and the heater by a blue-red pipe.

Hx-cooler An hx-cooler is a cooler that ejects thermal energy to a heater, to which it is coupled. Hx-coolers take on the same roles as coolers, but additional testing is necessary to ensure that their corresponding heater halves take on appropriate roles. For example, if a hx-cooler is acting as a heat-provider, then its heater half must be acting as a heat-injector. Otherwise the design would fail to achieve any of the design goals; but would instead be arranging to provide heat to a device that in effect throws it away.

Hx-heater An hx-heater is a heater that receives thermal energy from a coupled hx-cooler, but in other respects behaves like a heater. As in the case of the hx-cooler, additional testing is necessary to ensure that its corresponding half is playing an appropriate role.

Turbine A turbine consists of a series of fan blades arranged along a shaft; high-energy working fluid flowing through the turbine causes the shaft to rotate, converting thermal energy into mechanical energy, or work. The most common role of the turbine is to produce work, but in this process the temperature of the working fluid falls appreciably, and so turbines can also act as fluid coolers. They are used in this capacity most often in gas-liquefaction systems.

Compressor A compressor is very similar to a turbine, but instead of allowing the working fluid flowing through it to expand, mechanical energy applied to its shaft causes the working fluid to be compressed. Compressors can only operate on gaseous working fluids, because condensation causes significant erosion of their fan blades. Compressors are among the least functionally ambiguous components, most often acting as pressure-increasers. There are circumstances, however, in which the attendant increase in thermal energy is the desired effect, and hence they can also act as fluid-heaters.

Pump A pump, like a compressor, generally acts as a pressure-increaser, although it can only operate on liquids. If the working fluid is a mixture of liquid and gas (i.e., in the saturated phase) then the pump will cavitate, which is likely to cause mechanical failure. For this reason, pumps can also act as flash-preventers; by increasing the pressure of the working fluid, they prevent it from flashing into a saturated vapor mixture. This is often necessary when preheaters are used to improve the efficiency of the cycle, and is an example of the third rational-designer goal, preserving the integrity of the system.

Throttle A throttle, like a turbine, causes the working fluid to expand, but unlike a turbine a throttle generates no work. Throttles have two roles, saturator and pressure-decreaser. A saturator (which subsumes pressure-decreaser) is intended to cause the working fluid to change from either a gas or a liquid to a saturated mixture. A domestic refrigerator uses a throttle to cause the refrigerant that has been cooled in the coils on the back or bottom of the refrigerator to partially vaporize. This vaporization takes place in the cooling coils of the refrigerator, and, because vaporization requires heat, the refrigerant literally sucks the heat out of the contents of the refrigerator. Throttles are also used to step down the pressure of a working fluid to allow it to mix with another, lower-pressure stream. This is a common throttling application in heat-engines, where steam bled from the turbine and used to preheat the working fluid flowing to the boiler must be fed back into the system at a lower pressure.

Mixer Mixers are among the most flexible of components. They may act as simple flow-joins, as open heat-exchangers, jet-ejectors, or, when coupled with a splitter, as the inlet half of a steam-drum. A flow-join simply joins two flows, and is subsumed by the other three roles. An open heat-exchanger acts to increase the heat of one stream by mixing it with a hotter stream. This direct-contact mixing is more thermodynamically efficient, although it requires the streams to be at the same pressure, which may require more pumps in the system. Direct-contact heat-exchange is also used to deaerate the working fluid in power-plants, as oxygen dissolved in the working fluid can cause corrosion. A jet-ejector uses a high-velocity stream of working fluid to entrain and compress another stream of working fluid, in effect acting as a pump. Whereas pumps receive an input of mechanical energy, jet-ejectors utilize thermal energy. This can be more efficient when there is a source of heat that would otherwise go to waste, and so industrial plants often take advantage of process heat to generate electricity in so-called cogeneration systems. A steam-drum acts as a reservoir of steam, typically to be fed into a turbine series. A saturated mixture is typically fed into a steam drum, and separated into gas, which is fed to the turbines, and liquid, which is recirculated. Saturated mixtures that are more than 7% liquid can do great damage to turbine blades, as the liquid forms into droplets that impinge at high velocities on the turbine blades, so steam drums are often used to provide a pure gas stream to the turbines. They are most common in nuclear plants, which tend to operate at lower temperatures than fossil-fueled plants. Steam drums also act as a buffer against changing loads on the system.

Splitter Splitters, like mixers, can play several different roles. They may act as simple flow-forks, as steam-drum outlets, as flash-chambers, or as bleed-valves. A flow-fork splits a flow into two flows, and is subsumed by the other three roles. The function of a steam drum is described above, in the chapter on mixers. A flash-chamber is a container in which the working fluid, having just undergone a decrease in pressure, either by flowing through a turbine or through a throttle (which would be acting as a saturator), suddenly changes phase to a saturated mixture. The liquid part of the flow is then separated from the gas part. Flash-chambers are most commonly used in gas liquefaction plants, in which a cooled and compressed working fluid is allowed to precipitate in the flash-chamber. Occasionally industrial refrigerators or air-conditioners will make use of a flash-chamber, and in some nuclear plants the steam flowing into the final turbine stage first passes through a flash chamber to remove the liquid that would otherwise damage the turbine blades. Bleed valves are flow-forks between the stages of a turbine that are intended to bleed off a portion of the working fluid, typically for use in preheating the working-fluid flowing to the boiler. Although bleed-valves are tantamount to flow-forks, they tend to be used in recurring design "idioms" and so we have found distinguishing this role to be quite useful in inferring the role of other components connected to them.

Source A source simply supplies a working fluid, and hence plays only this one role.

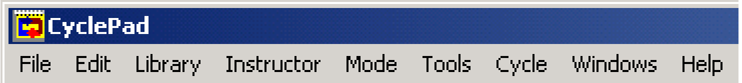
Sink A sink simply receives a working fluid, and hence plays only this one role.

Top

CyclePad Command Reference

This section describes the commands available from the menu bar.

Click on any of the commands in the menubar below to see help for that command.



Top

File

New
Pops up the New Design dialog box for entering the initial information pertaining to the new design.

Open Design
Opens an existing .dsn file, perhaps one that you had saved from a prior session. Note that CyclePad attempts to case convert all files in case they were from an earlier version. If case covnersion is barred because the file is read-only, CyclePad will prompt to unset the read-only flag and proceed with the case conversion.

Open Assignment Problem
Opens an assignment (.prb) file, perhaps one that your instructor assigned. Assignment files contain information about problems to be done in CyclePad.

Close
Closes the current design

Close All
Closes all open designs

Save
Saves the current design as a .dsn file. .dsn files are ASCII files that record, in a CyclePad readable manner, information about the design, including the schematic layout and any assumptions that you may have made.

Save As...
Pops up the Save dialog and enables you to change the name or location of the current design file.

Save All
Saves all open designs.

Revert to Saved
Re-load the current design as it was last saved, discarding changes since that save.

Save Bitmap of design
Save a bitmap image of the current design window to a file. The shape and size of the bitmap are chosen by the user.

Save separate bitmaps of layout & meters
Save a separate bitmap of the design layout and each of the open meter windows to a directory. This is useful when there are many open meters and it is difficult to see all of them at once.

Case Convert File
Case convert a saved design/problem/assignment file saved from an earlier version of CyclePad.

Export Cycle Data
Exports cycle data in tab-delineated rows and columns, which most spreadsheets will import directly.

Export Comments
Exports comments that you made on particular devices, stuffs, and parameters to a text file. You may find this helpful in writing up a report on your cycle.

Printer Setup...
Check your current printer settings.

Print
Prints the current cycle diagram, subject to the constraints of your printer.

Exit
First confirms that you want to exit CyclePad, saves any unsaved designs.

Top

Edit

Design Notes
Opens a text-editing window for inserting comments about the design. Use this facility for documenting designs to be turned in as homework or for recording notes for your own purposes. Note that you can also attach comments to particular devices or stuffs by right clicking on them in analyze mode, and also to particular parameters by choosing the Comment on <parameter name> option from the pop-up menu that appears when you left-click on a parameter.

Design Title
Pops up a dialog that enables you to change the design’s title.

Retract All Assumptions	Retract all assumptions made since entering design mode.
Recalculate	Force a recalculation of numerical values. CyclePad does this automatically.
Preferences	Allows you to change the program preferences, units, et cetera.

❖❖❖

Top

Instructor

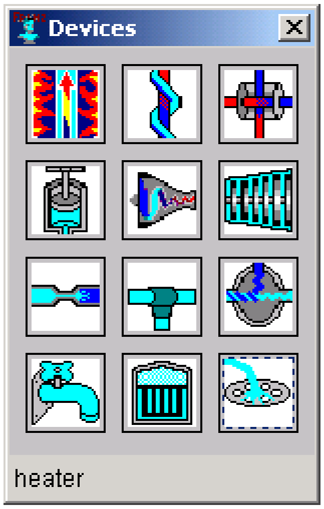
These are functions available to instructors for creating assignment problems for students and grading existing assignments.

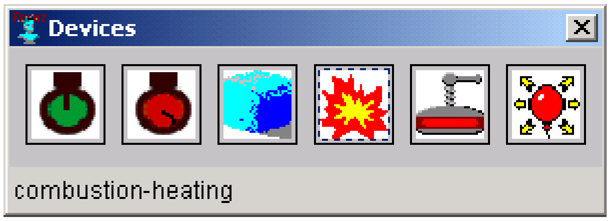
Assignment Builder	Build a design problem and assignment file. This allows creation of CyclePad problem files (.prb) and assignment files (.asn) which may be distributed to students as homework problems. Such problem files (.prb) allow for a description of the problem to be solved and specific questions about a thermodynamic cycle that must be answered. In addition, the students may be given a partially completed design from which to start the problem.
Design Grader	Grade assignment problem (.prb) files.

Top

Insert

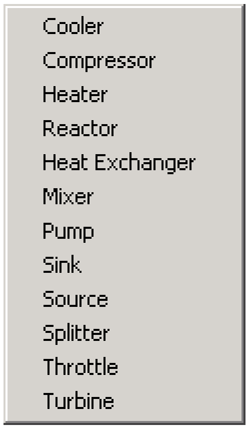
There are two methods available to you for adding components to the blueprint. By default, the component palette is displayed when you start a new design:





The device or process in the top-left corner is selected by default. Clicking anywhere on the background of the blueprint will place a component of this type at that spot. This method is convenient for building cycles that contain several components of the same type, such as five turbines.

The alternative method, *direct-background-clicking*, may be activated by either choosing Click on Background from the Add Component menu or by closing the component/process palette. Clicking on the blueprint will cause a menu of components or processes to appear. Select a particular one from this menu (or click elsewhere on the background to cancel this operation) and a component or process will be placed on the blueprint where you first clicked.



❖❖❖

Top

Library

This menu contains a list of built-in example cycles that you can load, analyze, alter, and save as you wish.

❖❖❖

Top

Mode

Note: This menu enables you to toggle between Build and Analyze mode; the checkmark indicates which mode you are currently in. Although you can do some things in either mode (such as saving the cycle or printing out the schematic or cycle data), there are certain actions that can only be accomplished in one or the other mode. You will typically start in Build mode, then switch to Analyze mode after completing the structure of the design. If you try to switch to Analyze mode prior to completing the cycle (i.e., one or more components is not connected to another component via all of its inlet and outlet ports), CyclePad will display an error message to this effect.

Build Enable Build mode, in which you can add components, connect components, move connected components, and edit device and stuff names.

Analyze Enable Analyze mode, in which you can see component, stuff, and cycle parameter values and modeling assumptions, change these values, obtain explanations for them, and do sensitivity analysis

Top

Tools

Lenses	Show information about the cycle stuffs and devices.
Analytic Progress	Displays a checkmark for each stuff or device that is completely analyzed, and a meter window with a red bar indicating the relative number of known parameters for those stuffs and devices that aren't completely analyzed.
Energy Flows	View the flows of heat and power or work in your cycle. When you choose this option, the icons representing each device through which there is a flow of either heat or work between the cycle and the environment will be replaced with little square readouts that contain a single vertical bar in their centers. This bar indicates the relative amount of flux taking place at each device. The color of the bar indicates the type of flux. Red indicates a heat flow in, blue a heat flow out, yellow a work flow in and green a work flow out.
Temperatures	This lens enables you to view the temperature at each stuff relative to the maximum temperature of the cycle. Note that these readouts have rounded corners to distinguish them from Energy Flow readouts. Temperature bars are displayed in red, to distinguish them from Pressure bars, which are displayed in blue.
Pressures	This lens enables you to view the pressure at each stuff relative to the maximum pressure of the cycle. Note that these readouts have rounded corners to distinguish them from Energy Flow readouts. . Pressure bars are displayed in blue, to distinguish them from Pressure bars, which are displayed in red.
Restore Icons	Restore the default icons if you have been using the lens or analytic progress views (see below).
Annotations	Show information about the cycle stuffs and devices.
Device Role Annotations	Displays the role of each device.
Temperature Annotations	Displays the temperature at each point.
Pressure Annotations	Displays the pressure at each point.
Remove Annotations	Removes pressure or temperature annotations (see below) from screen.
Sensitivity	Analyze the relationship between an assumed parametric value and a derived one and present the results as a line graph. For example, you might want to understand how the efficiency of your cycle varies with changes in the outlet pressure of the boiler.
Edit Equations	Enables you to equate any two parameters, via the Edit Equation dialog. For example, you might be analyzing a cycle with five non-ideal turbines in it; rather than set each to a given efficiency, you can set the efficiency of the first one and then set the other four efficiencies equal to the first.
Look up Properties	Pops up the Property Table Interpolator, which enables you to directly access CyclePad's property tables.
Property Table Bounds	Displays a list of all property tables contained in CyclePad. Choosing a table from this list will display a graph indicating the range of the table. You may find this helpful when working with unfamiliar working fluids.

Top

Cycle

Assumptions Made
Displays all assumptions that you have made in the Explanations Window

Cycle T-s Diagram
Displays a Temperature-Entropy diagram for the cycle. It is possible that this command will not work for complex cycles.

Cycle P-v Diagram
Displays a Pressure-(specific) volume diagram for the cycle. It is possible that this command will not work for complex cycles.

Whole Cycle
Pops up a meter window displaying parameters pertaining the cycle as a whole

Subcycle-*x*
If your cycle is comprised of two or more subcycles, as in the combined cycle library example, a menu item for each subcycle will appear. The letter in the label will correspond to the letters on the stuffs of the subcycle.

Stage-<device>
If you are using the economic model, CyclePad aggregates turbines and compressors into stage devices, because otherwise it would assume that each turbine and/or compressor is a separate device, and apply the cost of installation to each, which would grossly overestimate the capital cost. These menu items enable you to view parameters particular to these stage devices.

Memory
Refreshes the display of available memory at the lower left corner of the screen. When available memory drops below a safe amount, this message will flash to warn you to save your work and restart CyclePad.

Recalculate
Causes CyclePad to attempt to calculate more information about the current cycle. You probably won’t have any use for this command, as CyclePad automatically calculates every value possible when you input information. If you think that CyclePad should have calculated more information than it is currently displaying, try this command—it is possible that CyclePad failed to propagate a particular value. If no additional information appears, then you should re-examine your assumptions.

❖❖❖

Top

Windows

This menu lists the title of each open design. Click on the title of the design you want to use to make that design visible (it may be hidden under other design windows).

Here also one may close all meter windows and all graph windows.

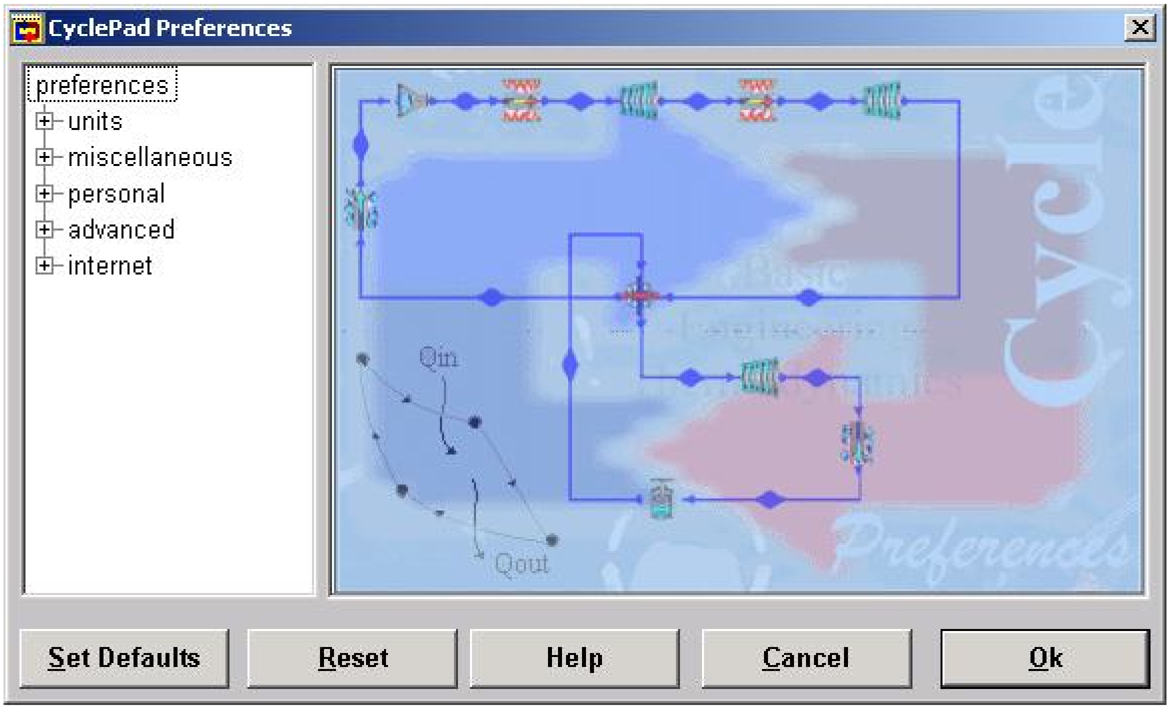
❖❖❖

Top

Preferences

Pops up a tabbed menu that enables you to customize CyclePad to your liking. When preferences are saved using "Set Defaults", CyclePad records these preferences in the cyclepad.ini file that is loaded at startup, so, unless this is a lab machine, your preferences will be restored when you come back to CyclePad. Otherwise, preferences apply only to the active design.

"Reset" sets the preferences for a particular design back to those stored by the last "Set Defaults" setting, or to CyclePad's initial defaults if "Set Defaults" has never been used.

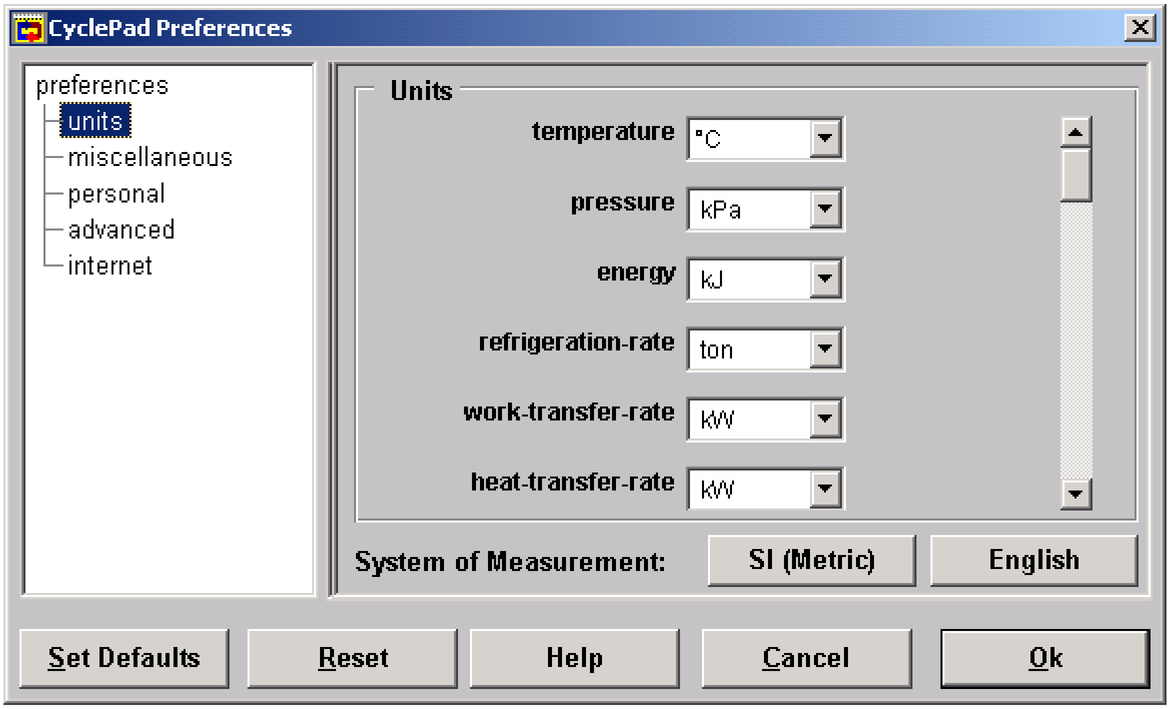


The items in the Preferences notebook are:

Units - how CyclePad displays numerical data
Miscellaneous - various settings, including popup help level
Personal - name and email address used for email help queries and assignment submission
Advanced - options affecting how CyclePad analyzes cycles
Internet - how CyclePad sends help requests and routes email.

Top

Preferences: Units



temperature °C

pressure kPa

energy kJ

refrigeration-rate ton

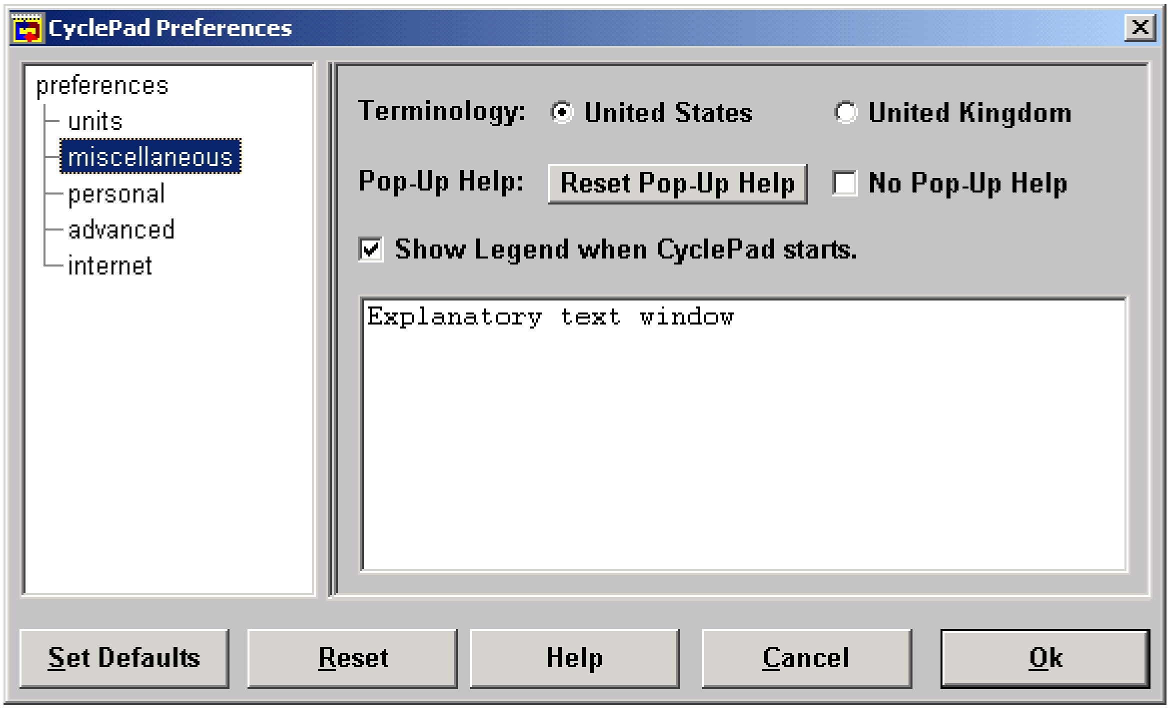
work-transfer-rate kW

heat-transfer-rate kW

Units The Units tab enables you to set the default units you would like CyclePad to start with. Library examples will come up in the units you set here. However, if you save a particular design with different units, those units will override your global defaults. This enables you to do a problem that calls for English units and later reload it without the units being reset to SI, if those are your global defaults.

Top

Preferences: Miscellaneous



Terminology: ● United States ○ United Kingdom

Pop-Up Help: □ No Pop-Up Help

☑ Show Legend when CyclePad starts.

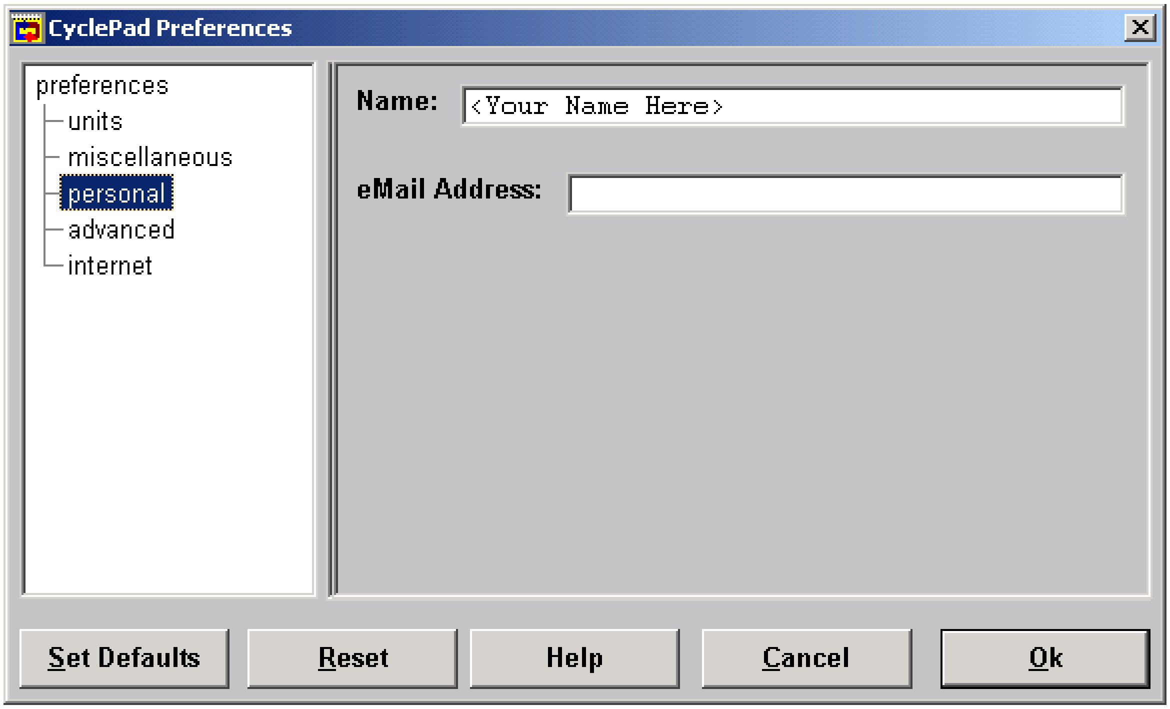
```
Explanatory text window
```

Miscellaneous

Under this tab you can choose either UK or US usage, which will dictate which terms CyclePad uses (eg, "dryness" vs "quality") and the default unit for certain parameters, such as pressure. You can also selectively enable or disable the pop-up help dialogs.

Top

Preferences: Personal



Name: <Your Name Here>

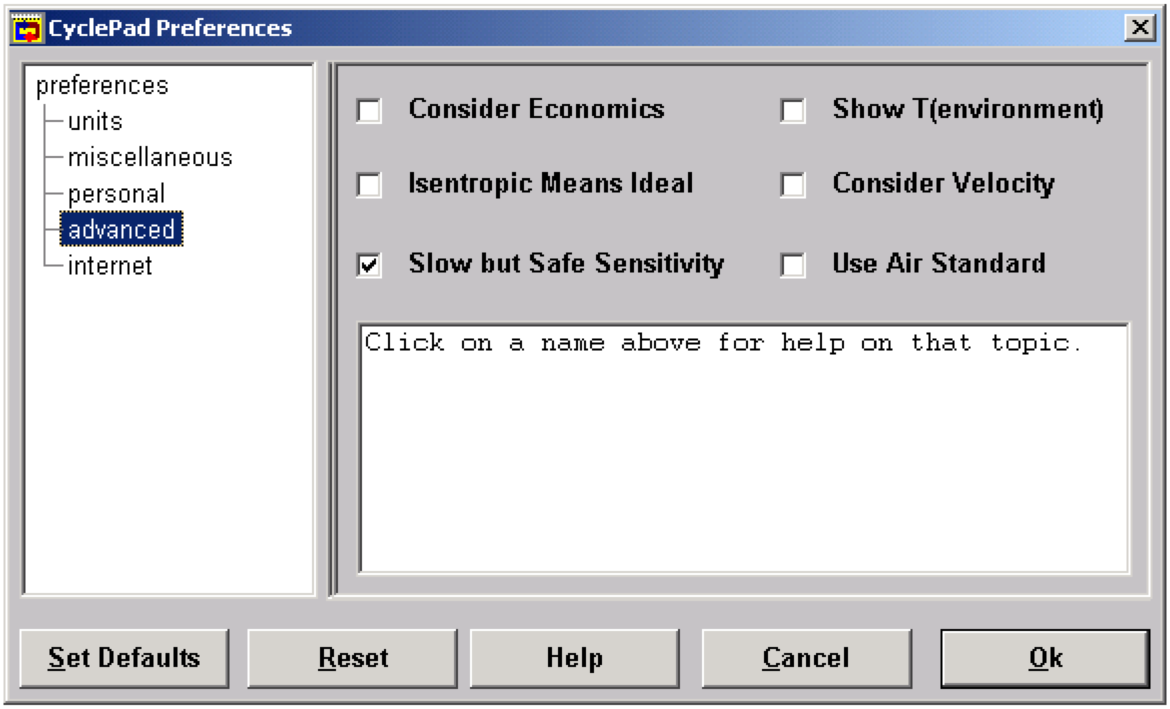
eMail Address:

Personal

Under the personal tab you can enter your name and e-mail address, which will prevent CyclePad from prompting for your name when you save a design. The information under this tab will automatically fill in the appropriate fields when you request email help.

Top

Preferences: Advanced



☐ Consider Economics ☐ Show T(environment)

☐ Isentropic Means Ideal ☐ Consider Velocity

☑ Slow but Safe Sensitivity ☐ Use Air Standard

Click on a name above for help on that topic.

Advanced Allows setting advanced options for design calculations. None of these options are on by default. The options are:

Consider Economics Consider economic factors in the design, such as component costs, material selection, et cetera.

Show T(environment) Allows computation of certain factors where high and low temperatures must be known, such as the Carnot efficiency.

Isentropic Means Ideal Causes CyclePad to combine modeling assumptions such that all isentropic components to also be adiabatic, which is the ideal assumption for work-producing components.

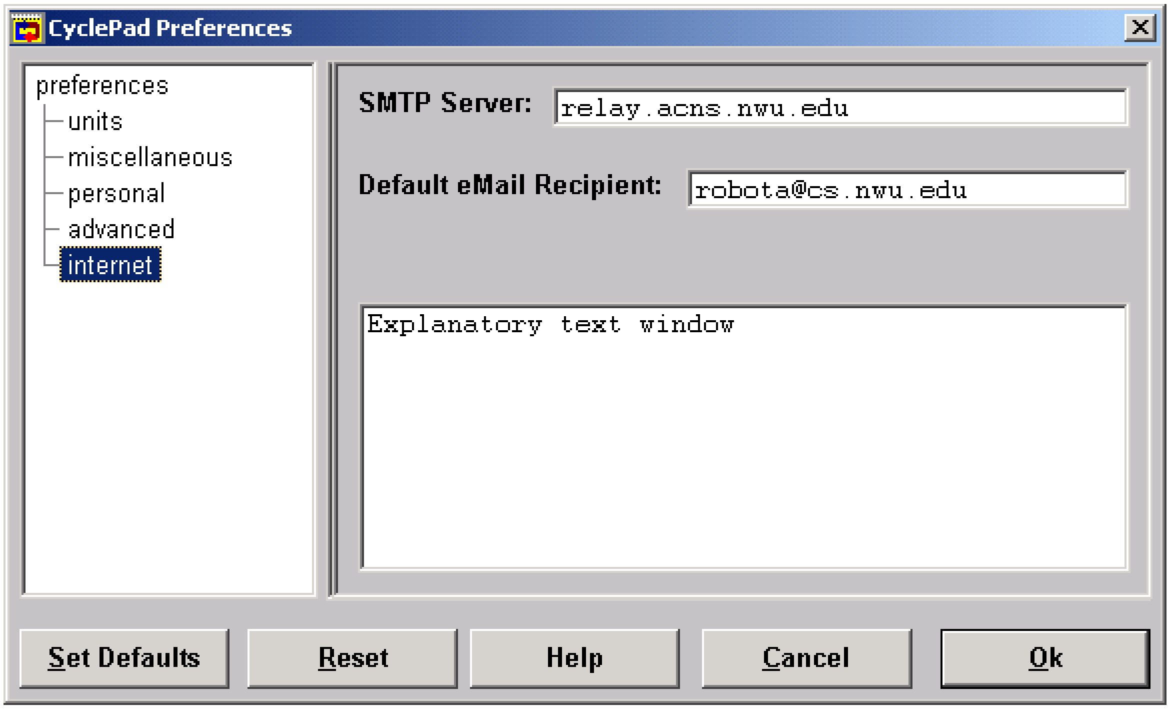
Consider Velocity Use the velocity of the working fluid to calculate kinetic energy when possible and consider it in First Law calculations. This adds extra parameters to certain meter windows, such as cross-sectional area to turbines and throttles so that fluid velocity may be computed.

Slow but Safe Sensitivity When doing sensitivity analyses, do not use mathematical shortcuts to speed up calculations. Instead, individually assume and retract each numerical value when generating plot points. This ensures that changes in phase as the independent parameter is varied will be observed.

Use Air Standard Forces CyclePad to use air standard tables for cycles where the working fluid is air. Air is still considered an ideal gas, but the tables allow consideration of specific heats that vary with temperature. Ordinarily, the specific heats for air are taken to be constant at their 300 K values. This is a somewhat more realistic model, but problems with the air standard tables can cause contradictions in CyclePad. If this occurs, we recommend turning this option off.

Top

Preferences: Internet



SMTP Server: relay.acns.nwu.edu

Default eMail Recipient: robota@cs.nwu.edu

Explanatory text window

Internet — Specifies where CyclePad should email help requests and submitted assignment problems. Also allows specification of which mail server CyclePad should use for sending mail.

Top

Help

Email — Enables you to email questions about your work to your TA or to CyclePad support. Your design is automatically appended to the email. **NOTE: CyclePad is research software, and we do not have the resources to respond to all inquiries. At various times we may have either a person or a case-based AI tutoring system available to respond to inquiries, but we cannot guarantee such services will**

be available at any given time. You should receive return email notifying you of the support you can expect if you send email.

Analysis Coach — Provides information that may help you improve your cycle’s parametric values. CyclePad first figures out what role each device in your cycle is playing and then compares the parameters of each device to norms for the role of that device. If your values are out of the normal range, CyclePad reports the deviation along with an explanation of the range. Note that CyclePad may not find anything abnormal about your cycle. For more information, see Design and Analytical Coaching .

Constraint Check — See if the constraints of an assignment problem have been met.

Legend — Turn on or off the device legend at the bottom of the main CyclePad window. The legend is described in the section on adding components to a design.

Contents — Provides access to this very file.

About CyclePad — Pops up a menu that displays the version number, creation date and time of the CyclePad you are using, and the web address for the CyclePad Project. CyclePad, being research software, is in a continual state of improvement, so although we cannot guarantee immediate responses to all email, you can have an impact if you let us know what you think! Please see the website (www.qrg.ils.northwestern.edu/cyclepad) for current support options and email addresses.

Top

CyclePad's Knowledge Base

This section outlines what CyclePad knows about engineering thermodynamics, to provide you with a good grasp of what you have to work with in designing cycles.

Components

Stuff

Subcycles

Substances

Modeling Assumptions

Parameters

Top

Components

CyclePad currently has models for the following ten components. CyclePad models components as thresholds that cause changes in the stuffs crossing them. For example, CyclePad considers a heater to be a threshold across which a stuff gains thermal energy. Components, therefore, do not contain stuffs, and there is no representation of the time a process takes, nor of its internal mechanism. However, CyclePad does make some inferences about the behavior of processes based on the states of the inlet and outlet stuffs. For example, if the stuff at a pump's inlet is saturated or gas, then CyclePad infers that the pump is cavitating, which is a failure mode.

Turbine

Compressor

Pump

Heater

Cooler

Heat Exchanger (two types, counter-current and co-current)

Throttle

Splitter

Mixer

Sink

Source

If you do not see the component you need for your design here, think about how you might construct it from other components (e.g., a multi-stage turbine can be built from several turbines plus splitters and mixers). You will also find a list of the modeling assumptions available for each component. (When analyzing your design, the modeling assumptions available for any component can be found via that component's meter). Each modeling assumption can be either true, false, or unknown. CyclePad automatically keeps track of the relationships between modeling assumptions.

Top

Processes

CyclePad can also model closed cycles, in which different processes occur in the same location, which is temporarily closed off from the cycle by valves or other mechanical parts. For example, processes of heating, compression, and expansion all occur in the cylinder of an automobile engine.

Combustion heating

Cooling

Polytropic expansion

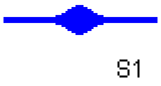
Polytropic compression

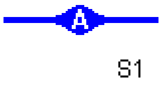
Start

Stop

Top

Stuff

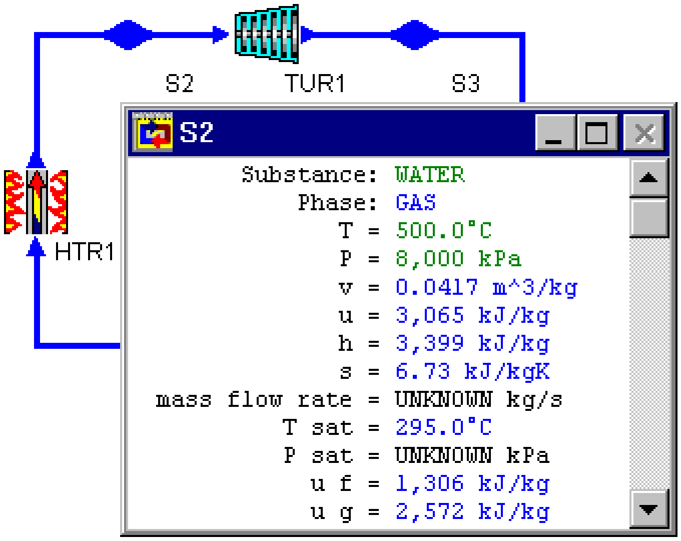




A stuff represents the state properties of the working fluid at that particular point. For example, the pressure of the stuff connecting the outlet of a heater to the inlet of a turbine represents the pressure at that point.

The blue line represents the pipe connecting two components, and the bulge in the line represents the working fluid in that pipe. Each stuff has a label, S1 in the above figure. If your design has more than one subcycle, then each stuff will be labeled with a letter indicating which subcycle it is part of, as in the stuff on the right above.

In build mode, you can drag this label to reposition it, and right clicking on the label will pop-up a dialog box for changing the displayed name of the stuff. Right clicking on the bulge in the pipe will bring up the edit menu, which enables you to delete the stuff. In analyze mode, clicking on either the stuff bulge or the stuff label displays the stuff-meter window:



```
        Substance: WATER
            Phase: GAS
                T = 500.0°C
                P = 8,000 kPa
                v = 0.0417 m^3/kg
                u = 3,065 kJ/kg
                h = 3,399 kJ/kg
                s = 6.73 kJ/kgK
   mass flow rate = UNKNOWN kg/s
            T sat = 295.0°C
            P sat = UNKNOWN kPa
              u f = 1,306 kJ/kg
              u g = 2,572 kJ/kg
```

Top

Subcycles

A subcycle consists of all those parts of your design to which a particular piece of working fluid could travel. Most designs will have only one subcycle. However, a heat exchanger can connect two otherwise separate cycles, as in the combined cycle in the Library. In this situation the stuff in subcycle A will never mix with the stuff in subcycle B; in fact, the stuff in A is air whereas the stuff in B is water. Note that, while it is necessary to use a heat-exchanger to create a cycle with more than one subcycle, one can use a heat-exchanger in a system that has only one subcycle, for example in a Rankine cycle with regeneration. CyclePad will always label subcycles with letters (A, B, C, etc) if there is more than one, and you can open meters displaying the properties of each subycle from the Global Properties menu.

Top

Modeling Assumptions

The cycle designs you construct in CyclePad are only models of real artifacts, and the results that CyclePad produces are therefore only approximations of the actual results one would observe with a physical cycle operating under real-world conditions. Models intentionally simplify reality, in order to make it tractable. What is critical is that you understand how your model simplifies reality.

Good modeling assumptions make life easier and don't adversely affect the results, while bad ones will produce results that diverge wildly from reality. For example, because liquids are incompressible, there is little change in the temperature of a working fluid as it passes through a pump. There is in fact a little, as some of the energy the pump imparts to the fluid is converted into heat, but for the most part an assumption that temperature across the pump remains constant is a sound one. In contrast, compressing an ideal gas generally causes a large increase in its temperature, so assuming that the temperature of the working fluid remains constant across a compressor would be a bad idea.

The advantage that you gain from making modeling assumptions is that you license CyclePad to propagate values. For example, assuming that a pump is isothermal enables CyclePad to derive the outlet temperature from its inlet temperature or vice versa.

Here is a table of the modeling assumptions you may make in CyclePad. Only a subset of these assumptions will apply to any given component.

Isobaric	Pressure remains constant. This is a sound assumption to make for components such as heaters and coolers, because in reality the pressure won't change drastically across them.
Isochoric	Volume remains constant. This assumption is most commonly made for heating and cooling processes in closed-cycle systems. As a gas is heated or cooled, it will expand or contract. In an isochoric heating process, the gas will attempt to expand, but the constant volume of the process will result in an increase in the pressure of the gas. Therefore, isobaric and isochoric are in general mutually exclusive assumptions.
Isothermal	Temperature remains constant.
Isentropic	Entropy remains constant.
Polytropic	For ideal gases, the term pv^k remains constant. Polytropic processes approximate actual expansion and compression curves for pressure in the range of hundreds of psi. The specific heats of the working fluid are assumed to remain constant.
Non-polytropic	The term pv^k is free to vary.
Adiabatic	The process allows no heat transfer to occur between the working fluid and the environment. Under this assumption, for example, the casing of a turbine radiates no heat from the steam passing through it, which is a very reasonable assumption for many turbines, where the heat lost to the environment is quite tiny compared to the heat converted to work.
Non-isoparametric	When applied to splitters and saturated working fluids, the parameters of the two outlet stuffs are not constrained to be the identical. That is, for a saturated working fluid entering the splitter, one outlet leg could contain fluid of a different quality than the other and, thereby, different specific volumes, et cetera.
Isoparametric	When applied to splitters, the two outlet stuffs are assumed to be parametrically identical and will have the same quality and other intensive values.

Top

Parameters

Parameters provide numerical information about aspects of the cycle's behavior. A parameter may be applicable to a stuff, a substance, the cycle as a whole, or a subset of components. The following table lists all parameters present in CyclePad. Click on particular parameters to obtain more information.

cp	cv	cop-r
Carnot-cop-r	cop-hp	Carnot-cop-hp
delta-H	delta-H-isentropic	delta-P
delta-spec-s	gamma (k)	H
Hout-isentropic	mass-flow	max-T
min-T	molar-mass	net-Q
net-work	eta-isentropic	eta-thermal
P	PR	Psat
Q	Q-in	Q-out
R	S	shaft-work
spec-h	spec-hf	spec-hg
spec-q	spec-s	spec-sf
spec-sg	spec-shaft-work	spec-shaft-work-isentropic
spec-u	spec-v	T
Tout-isentropic	Tsat	U
V	work-in	work-out

Top

C*p*

Specific heat at constant pressure. This is the heat transfer that must occur per unit mass of a substance to raise its temperature by one degree, given that the pressure remains constant.

Units: heat flux per mass unit per degree temperature

Applies to: substances

Top

C*v*

Specific heat at constant volume. This is the heat transfer that must occur per unit mass of a substance to raise its temperature by one degree, given that the pressure remains constant.

Units: heat flux per mass unit per degree temperature

Applies to: substances

Top

cop-r

Coefficient of performance for refrigerators

Units: none (a ratio)

Applies to: refrigerator cycles

Top

Carnot cop-r

Ideal coefficient of performance for refrigerators

Units: none (a ratio)

Applies to: refrigerator cycles

Top

cop-hp

Coefficient of performance for heat pumps

Units: none (a ratio)

Applies to: heat pump cycles

Top

Carnot cop-hp

Ideal coefficient of performance for heat pumps

Units: none (a ratio)

Applies to: heat pump cycles

Top

delta-H

Change in enthalpy

Units: Joules. Note that both work and heat flows are measured in the same units.

Applies to: compressor, pump, turbine, cooler, heater

Top

delta-H-isentropic

Isentropic change in enthalpy

Units: Joules Note that both work and heat flows are measured in the same units.

Applies to: compressor, turbine

Top

delta-P

Change in pressure

Units: force per unit area

Applies to: compressor, pump, turbine, cooler, heater

Top

delta-spec-s

Change in specific entropy

Units: work per degree temperature per unit mass

Applies to: compressor, pump, turbine, cooler, heater

Top

gamma (k)

The ratio of the specific heats (cp/cv) of a substance

Units: none

Applies to: substance

Top

H

Enthalpy

Units: typically in Joules or kiloJoules. Note that both work and heat flows are measured in the same units.

Applies to: stuff

Top

Hout-isentropic

Isentropic outlet enthalpy

Units: typically in Joules or kiloJoules. Note that both work and heat flows are measured in the same units.

Applies to: compressor, turbine

Top

mass-flow

Rate of flow of working fluid

Units: mass per unit time

Applies to: compressor, pump, turbine, cooler, heater, stuff

Top

max-T

Maximum temperature

Units: Degrees of temperature. Be aware that many thermodynamic calculations must be done in absolute temperature units, that is, in either the Kelvin or Rankine temperature scales. These scales, in contrast to Centigrade and Fahrenheit, set zero at absolute zero, the temperature at which no thermal energy is present.

Applies to: cycle

Top

min-T

Minimum temperature

Units: Degrees of temperature. Be aware that many thermodynamic calculations must be done in absolute temperature units, that is, in either the Kelvin or Rankine temperature scales. These scales, in contrast to Centigrade and Fahrenheit, set zero at absolute zero, the temperature at which no thermal energy is present.

Applies to: cycle

Top

molar-mass

Amount of mass in terms of number of molecules. Specifically, a mole is the amount of substance within a given envelope that contains the same number of elementary entities as may be found in 0.012 kg of carbon 12.

Units: mole

Applies to: substance

Top

net-Q

Net heat flow

Units: Joules. Note that both work and heat flows are measured in the same units.

Applies to: cycle

Top

net-work

Net work flow

Units: force times distance. Joules. Note that both work and heat flows are measured in the same units.

Applies to: cycle

Top

eta-isentropic

Isentropic thermal efficiency

Units: none (ratio)

Applies to: compressor, turbine

Top

eta-thermal

Thermal efficiency

Units: none (ratio)

Applies to: cycle

Top

P

Pressure

Units: force per unit area

Applies to: stuff

Top

PR

Pressure ratio across a component.

Units: none (ratio)

Applies to: compressor, pump, turbine, cooler, heater

Top

Psat

Saturation pressure, the pressure at which the stuff becomes saturated. Useful as a benchmark for comparing to actual pressure.

Units: force per unit area

Applies to: stuff

Top

Q

Heat flow

Units: Joules Note that both work and heat flows are measured in the same units.

Applies to: compressor, turbine, heat-exchanger, cooler, heater

Top

Q-in

Aggregate heat flow into cycle

Units: Joules Note that both work and heat flows are measured in the same units.

Applies to: cycle

Top

Q-out

Aggregate heat flow out of cycle

Units: Joules Note that both work and heat flows are measured in the same units.

Applies to: cycle

Top

R

Universal gas constant

Units: distance times force per unit molar mass times degree of temperature

Applies to: substance

Top

S

Entropy

Units: work per degree temperature

Applies to: stuff

Top

shaft-work

Work done by rotating machinery

Units: work

Applies to: compressor, pump, turbine

Top

spec-h

Enthalpy per unit mass

Units: Joules Note that both work and heat flows are measured in the same units.

Applies to: stuff

Top

spec-hf

Enthalpy per unit mass for that fraction of a saturated working fluid that is liquid

Units: Joules Note that both work and heat flows are measured in the same units.

Applies to: saturated stuff

Top

spec-hg

Enthalpy per unit mass for that fraction of a saturated working fluid that is gas

Units: Joules Note that both work and heat flows are measured in the same units.

Applies to: saturated stuff

Top

spec-q

Heat flow per unit mass

Units: Joules Note that both work and heat flows are measured in the same units.

Applies to: cooler, heater

Top

spec-s

Entropy per unit mass

Units: work per degree temperature per unit mass

Applies to: stuff

Top

spec-sf

Entropy per unit mass for that fraction of a saturated working fluid that is liquid

Units: work per degree temperature per unit mass

Applies to: saturated stuff

Top

spec-sg

Entropy per unit mass for that fraction of a saturated working fluid that is liquid

Units: work per degree temperature per unit mass

Applies to: saturated stuff

Top

spec-shaft-work

Shaft work rate per unit mass

Units: Joules Note that both work and heat flows are measured in the same units.

Applies to: compressor, pump, turbine

Top

spec-shaft-work-isentropic

Isentropic shaft work per unit mass

Units: Joules Note that both work and heat flows are measured in the same units.

Applies to: compressor, turbine

Top

spec-u

Internal energy per unit mass

Units: work (force times distance)

Applies to: stuff

Top

spec-v

Volume per unit mass

Units: distance cubed per unit of mass

Applies to: stuff

Top

T

Temperature

Units: Degrees of temperature. Be aware that many thermodynamic calculations must be done in absolute temperature units, that is, in either the Kelvin or Rankine temperature scales. These scales, in contrast to Centigrade and Fahrenheit, set zero at absolute zero, the temperature at which no thermal energy is present.

Applies to: stuff

Top

Tout-isentropic

Isentropic outlet temperature of a turbine. This is the temperature that would prevail at the outlet of the turbine if the turbine were isentropic (that is, ideal).

Units: Degrees of temperature.

Be aware that many thermodynamic calculations must be done in absolute temperature units, that is, in either the Kelvin or Rankine temperature scales. These scales, in contrast to Centigrade and Fahrenheit, set zero at absolute zero, the temperature at which no thermal energy is present.

Applies to: turbine

Top

Tsat

Saturation temperature

Units: Degrees of temperature. Be aware that many thermodynamic calculations must be done in absolute temperature units, that is, in either the Kelvin or Rankine temperature scales. These scales, in contrast to Centigrade and Fahrenheit, set zero at absolute zero, the temperature at which no thermal energy is present.

Applies to: stuff

Top

U

Internal energy

Units: Joules

Applies to: stuff

Top

V

Volume

Units: distance cubed

Applies to: stuff

Top

work-in

Aggregate inflow of work to cycle

Units: Joules (force times distance)

Applies to: cycle

Top

work-out

Aggregate outflow of work from cycle

Units: Joules (force times distance)

Applies to: cycle

Top

Substances

CyclePad currently has models for the following substances:

Air	modeled as an ideal gas
C02	(carbon dioxide) modeled as an ideal gas
Helium	modeled as an ideal gas
Ammonia	modeled via superheat and saturation property tables

Methane	modeled via superheat and saturation property tables
Nitrogen	modeled via superheat and saturation property tables
Refrigerant-12	modeled via superheat and saturation property tables
Refrigerant-22	modeled via superheat and saturation property tables
Refrigerant-134a	modeled via superheat and saturation property tables
Water	modeled via superheat and saturation property tables

Top

Ideal Gas Law

PV = mRT

Pressure times volume is equal to the mass of the gas times the gas constant times the temperature. This relation holds for superheated gasses, that is, gasses that are at a significantly greater temperature than their saturation temperature. As the temperature approaches t-saturation, the behavior of the gas will increasingly diverge from this relationship. For substances that undergo phase transitions, therefore, one must use property tables to determine the thermodynamic parameters of the substance. [tbf]

Top

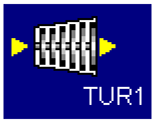
Thermodynamic Property Tables

Empirically-derived values for the thermodynamic properties of commonly used working fluids, such as water, ammonia, and freon (refrigerant-12). One must know the values of at least two parameters in order to use the tables to derive the values of other parameters, except that for saturated substances, one must also know the relative proportions of gas and liquid in the mixture (i.e., the quality or dryness of the substance) to use the tables.

In most cases, the properties of subcooled liquids are approximated via the saturation approximation, which makes use of the saturation table. Because this is an approximation, it introduces slight errors which do not in fact affect the accuracy of the calculations. However, due to the way in which CyclePad propagates numerical values, such inaccuracies often lead CyclePad to signal contradictions. These contradictions most often arise when the working fluid is water, as the pressure and temperature ranges tend to be larger than for other working fluids, so for water CyclePad derives the relevant values from an extensive subcooled table. It is quite possible that these values will differ slightly from values you may calculate by hand, but the differences will most likely be small. The important thing to remember is that the saturation approximation is widely applicable and should be your first resort when solving cycles without the benefit of CyclePad.

Top

Turbine

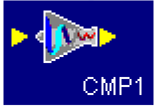


Turbines consist of a set of rotor blades interleaved with a set of stationary blades, or stators. The working fluid enters at high velocity and expands as it flows through the turbine. This expansion, resisted by the rotor blades, causes the rotor shaft to turn, creating shaft work. The temperature of the working fluid also drops during this process. The fluid entering the turbine must be either dry saturated vapor or gas; wet saturated vapors or liquids will seriously erode the turbine blades due to the high velocity of the fluid stream.

Possible modeling assumptions for turbines include isentropic and adiabatic.

Top

Compressor

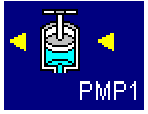


Compressors pressurize gases. In general, this requires much more work than pumping liquids, and the working fluid experiences a significant increase in temperature, in accordance with the ideal gas law. Compressors cannot handle wet saturated vapors or liquids, as such fluids cause excessive wear.

Possible modeling assumptions for compressors include adiabatic, isentropic, isothermal, polytropic, and non-polytropic.

Top

Pump

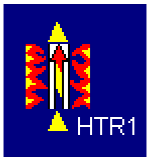


Pumps cause fluid flows in liquids. Pumping a liquid is more efficient than compressing a gas, because one can pump a great deal more liquid per unit volume, and liquids, being basically incompressible, do not gain an appreciable amount of heat during the pumping process. Pumps cannot handle gasses or saturated vapors, because such fluids tend to cavitate, or boil. Cavitation causes excessive shocks within the pump and can rapidly lead to pump failure.

Possible modeling assumptions for pumps include: isothermal.

Top

Heater

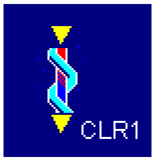


Heaters add thermal energy to the working fluid flowing through them. This may cause the fluid to change phase. A heater may be used to represent a boiler, a superheater, a reheater, a combustion chamber, or a nuclear reactor.

Possible modeling assumptions for heaters include: isobaric.

Top

Cooler

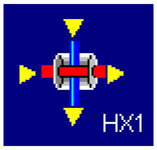


Coolers cool the working fluid flowing through them by allowing heat to flow to the environment. This process may cause the fluid to change phase.

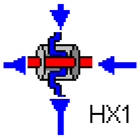
Possible modeling assumptions: isobaric.

Top

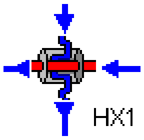
Heat Exchanger



Simple Model



Counter-Current



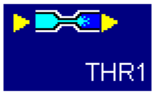
Co-Current

Heat exchangers enable a transfer of thermal energy from a high-energy working fluid to a low-energy fluid. In the counter-current heat exchanger, the hot and cold fluids flow in opposite directions, which minimizes the temperature differential between the fluids at any given point and thus the resulting irreversibility. In the co-current heat exchanger, the fluids flow in the same direction. You can use the co-current heat exchanger to investigate the gain in efficiency achieved via the counter-current design, but you need not choose one or the other; the simple heat-exchanger model will calculate the exchange of energy via First Law constraints.

Possible assumptions: Co-current, counter-current, isobaric hot-side, isobaric cold-side

Top

Throttle

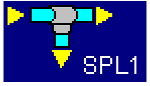


Throttles enable an unresisted expansion of the working fluid flowing through them. Because there is no resistance, there is no work done, and hence no change in enthalpy. The behavior of a fluid flowing across a throttle depends on its Joule-Thompson coefficient and whether it is above or below its inversion point. A fluid above the inversion point will experience an increase in temperature, while one below the inversion point will drop in temperature. A gas passing through a throttle may precipitate into liquid, while a liquid may suddenly vaporize, creating a refrigerating effect as it absorbs heat from its environment to sustain the evaporation process.

Possible modeling assumptions for throttles: none

Top

Splitter

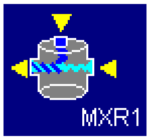


Splitters split the flow of working fluid into two streams. A splitter may be used to combine turbines into a multi-stage turbine with bleeds of steam used for regeneration.

Possible modeling assumptions for splitters: saturated, non-saturated

Top

Mixer

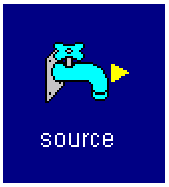


Mixers join two streams. If the two streams differ markedly in temperature, the mixer may function as an open heat-exchanger. Power plants often employ open heat exchangers as a means for deaerating the working fluid. If the two streams differ significantly in pressure, the mixer may function as a jet-ejector, a pump in which the high pressure stream entrains and compresses the low-pressure stream. Such pumps may be used in air conditioning applications, in which they are "driven" by a cheap fuel source, such as natural gas.

Possible modeling assumptions for mixers: none

Top

Source

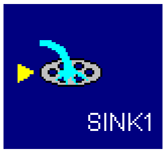


Sources start streams. There are two types of streams: those that are part of loops and those that start at sources and end at sinks. Using splitters and mixers, it is possible to combine the two types of streams, but for every source, there must be a sink and vice-versa.

Possible modeling assumptions for sources: none

Top

Sink



Sinks end streams. There are two types of streams: those that are part of loops and those that start at sources and end at sinks. Using splitters and mixers, it is possible to combine the two types of streams, but for every sink, there must be a source and vice-versa.

Possible modeling assumptions for sources: none

Top

Combustion Heating

Combustion heating processes cause an increase in the thermal energy of the working fluid in closed-cycle systems. In effect, they model the combustion that occurs in the cylinder of an automobile engine as if heat is directly injected into the working fluid, without the presence of a fuel vapor.

Possible modeling assumptions: isobaric, isochoric, isothermal

Top

Cooling

The cooling process removes thermal energy from the working fluid. In reality, this process is often accomplished by simply ejecting the spent working fluid (ie, the exhaust gasses) to the environment. You might want to consider a cooler as modeling the environment.

Possible modeling assumptions: isobaric, isochoric, isothermal

Top

Polytropic Expansion

The expansion process in a closed-cycle system generates the work that the cycle produces.

Possible modeling assumptions: adiabatic, isentropic, isothermal, polytropic

Top

Polytropic Compression

A compression process in a closed-cycle system is necessary in order to create the conditions for heating (i.e., detonation of the fuel-air mixture).

Possible modeling assumptions: adiabatic, isentropic, isothermal, polytropic

Top

Diagrams of Library Examples of Cycles

These examples, taken from the textbook *Basic Engineering Thermodynamics* by Peter Whalley, are provided to help you get started on building your own designs. Please feel free to experiment with them and modify them in any way you like.

Simple Steam Rankine Cycle

Simple Refrigerator Cycle

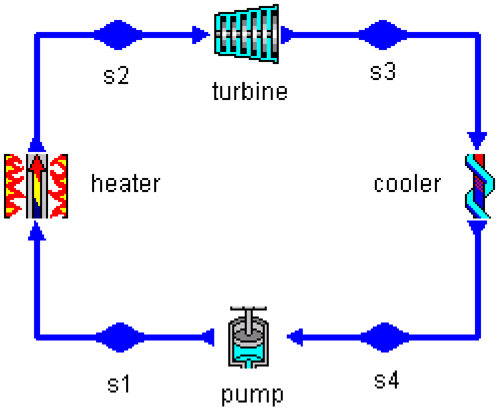
Basic Gas Turbine Cycle

Steam Cycle with Reheat

Combined Gas Turbine and Rankine Cycle

Top

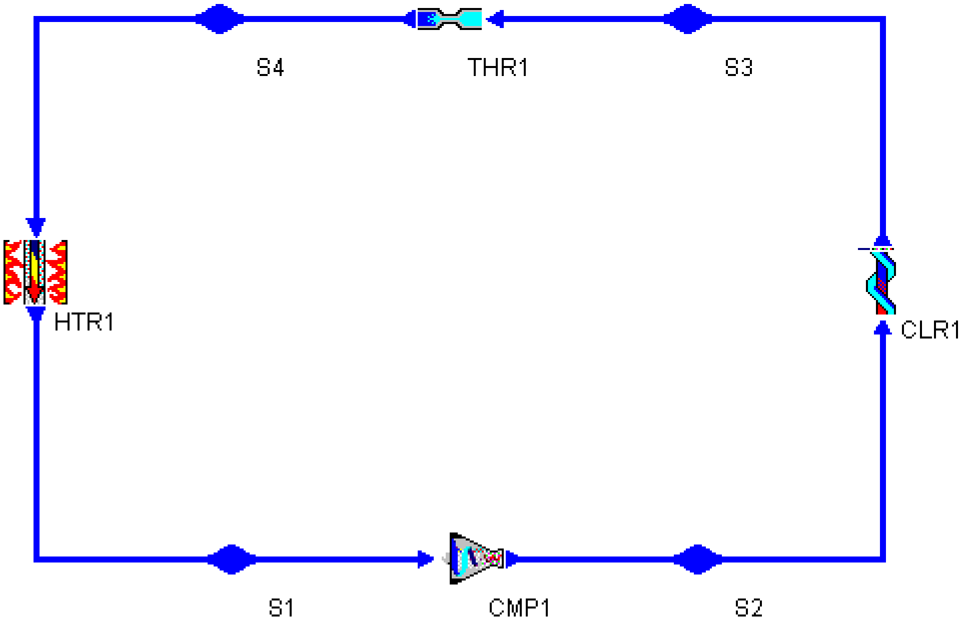
Simple Steam Rankine Cycle



This example is drawn from Figure 14.1, page 164 of *Basic Engineering Thermodynamics*. The working fluid is water.

Top

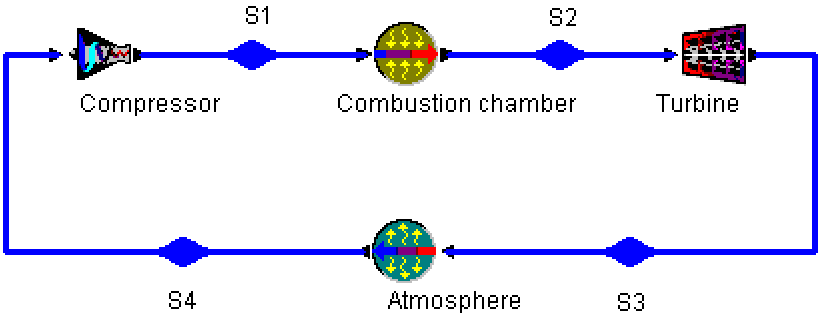
Simple Refrigerator Cycle



This example is drawn from Figure 16.3, page 190 of *Basic Engineering Thermodynamics*. The working fluid is Refrigerant-12.

Top

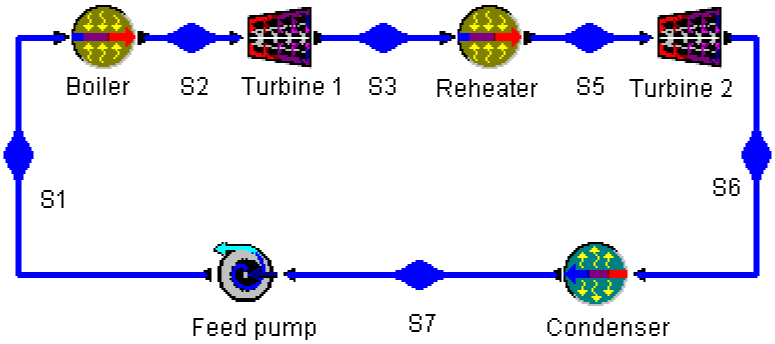
Basic Gas Turbine Cycle



This example is based on Example 15.1, page 175 of *Basic Engineering Thermodynamics*. The working fluid is air.

Top

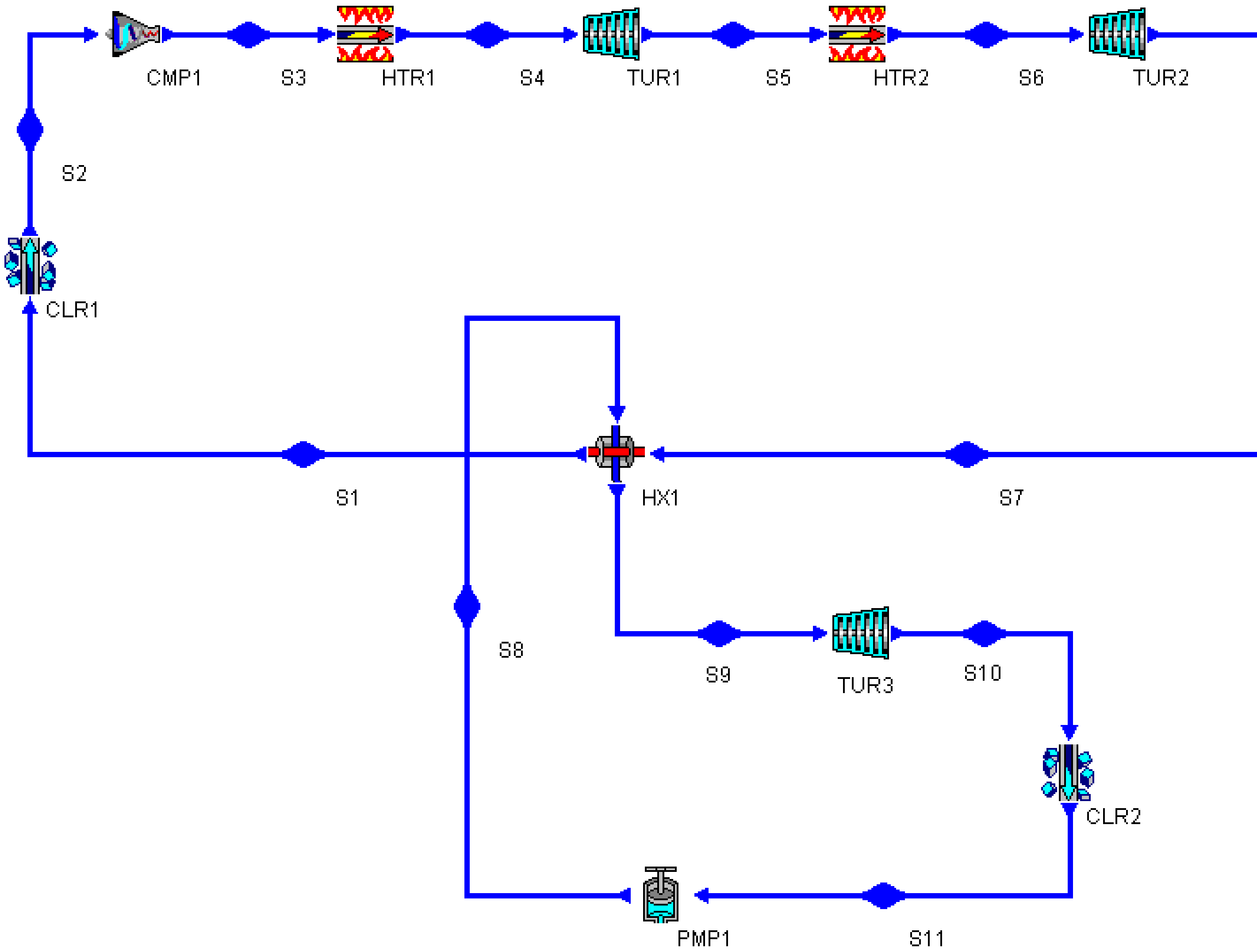
Steam Cycle with Reheat



This example is based on Figure 14.6, page 168 of *Basic Engineering Thermodynamics*. The working fluid is water.

Top

Combined Gas Turbine & Rankine Cycle

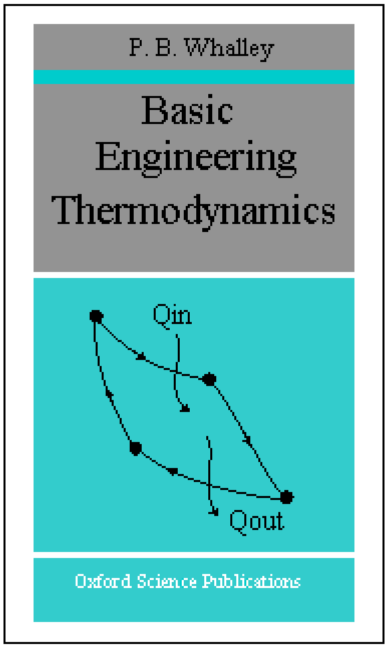


This example is based on Figure 15.11, page 185 of *Basic Engineering Thermodynamics*. The working fluid in the top loop is air and the working fluid in the lower loop is water.

❖❖❖

Top

Basic Engineering Thermodynamics



by Peter B. Whalley, Department of Engineering Science, University of Oxford

Oxford University Press 1992
ISBN: 0-19-856255-1

from the back cover:

This is an introduction to thermodynamics for engineering students. No previous knowledge is assumed. The book covers the first and second laws of thermodynamics and their consequences for engineers. Each topic is illustrated with worked examples and subjects are introduced in a logical order, allowing the student to tackle increasingly complex problems as he reads. Problems and selected answers are included.

The heart of engineering thermodynamics is the conversion of heat into work. Increasing demands for more efficient conversion, for example to reduce carbon dioxide emissions, are leading to the adoption of new thermodynamic cycles. However, the principles of these new cycles are very simple and are subject to the standard laws of thermodynamics as explained in this book.

Dr Whalley teaches thermodynamics to students in the Department of Engineering Science, University of Oxford. He is the author of *Boiling, condensation, and gas-liquid flow* (1987, paperback 1990)

Dr Whalley is also the co-author of CyclePad's knowledge base and the principal thermodynamics collaborator on the CyclePad project.

Top

Thermodynamic systems and properties

In thermodynamics the totality of the world is usually termed the universe, and is divided into two parts.

1. **The system** is the part of the universe in which we are particularly interested.

2. The rest of the universe is **the surroundings**.

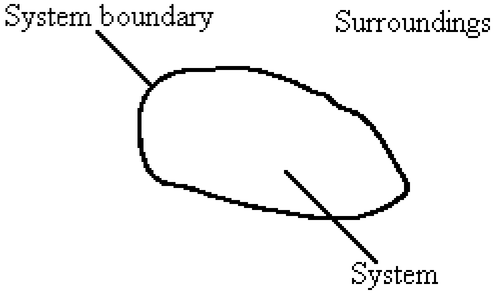


Figure 1 The system, the surroundings, and the system boundary

Separating the system from the surroundings is **the system boundary,** see Figure 1. Systems can be of three types depending on what can pass through the system boundary.

1. If matter, heat and work can pass through the system boundary, then the system is **an open system**.

2. If heat and work can pass through the boundary, but matter cannot, then the system is **a closed system**.

3. If neither matter, not heat, nor work can pass through the boundary, then the system is **an isolated system**.

The various types of system are summarised in Table 1.

Table 1 The various types of system

	Boundary permeable to:		
System type	matter	heat	work
isolated	no	no	no
closed	no	yes	yes
open	yes	yes	yes

The system might typically be a certain mass of gas, or a mixture of liquid and gas (such as water and steam). The condition of the system is identified by a number of properties of state such as the temperature, pressure, and volume. It will be noted later that heat and work are not properties of state like these other properties.

The phase rule specifies how many properties of state are necessary to define the system, see Table 2. It is often stated that two properties are necessary to specify a system, and this will be used frequently later, but it must be remembered that this is only true for a single phase system.

Table 2 Number of properties of state required to specify a system with a single pure component

Number of Phases	Number of properties of state required
1	2
2	1
3	0

Properties are sometimes divided into two types:

1. **Extensive properties** which depend on the mass of substance present, for example volume. The greater the mass of air, the greater will be the volume.

2. **Intensive properties** which do not depend on the mass of substance present, for example pressure and temperature. Specific volume is defined as the volume per unit mass, and so must be an intensive property.

Extensive properties are usually given an upper case symbol, for example V for volume. Intensive properties are usually given a lower case symbol, so p for pressure, and v for specific volume. However the use of T for the intensive property temperature is an exception to this rule.

from Basic Engineering Thermodynamics, *by Peter B. Whalley. Reprinted with permission of Oxford University Press*

Top

Processes and reversibility

A process is a change in the system from one state to another. Hence the change in the pressure and temperature of a mass of ideal gas from (*p1*, *v1*) to (*p2*, *v2*) is a process. The state defined by (*p1*, *v1*) is the **initial state** of the system, and the state defined by (*p2*, *v2*) is the **final state** of the system.

Processes can be divided into two types as follows.

1. **Reversible processes** these can be reversed so that both the system and the surroundings are returned to their original condition after the process and the reverse process have been carried out.

2. **Irreversible processes** these are ones in which this reversal cannot be carried out without leaving some change in the system or the surroundings.

The concept of reversibility is a very important one and is best illustrated by a number of examples.

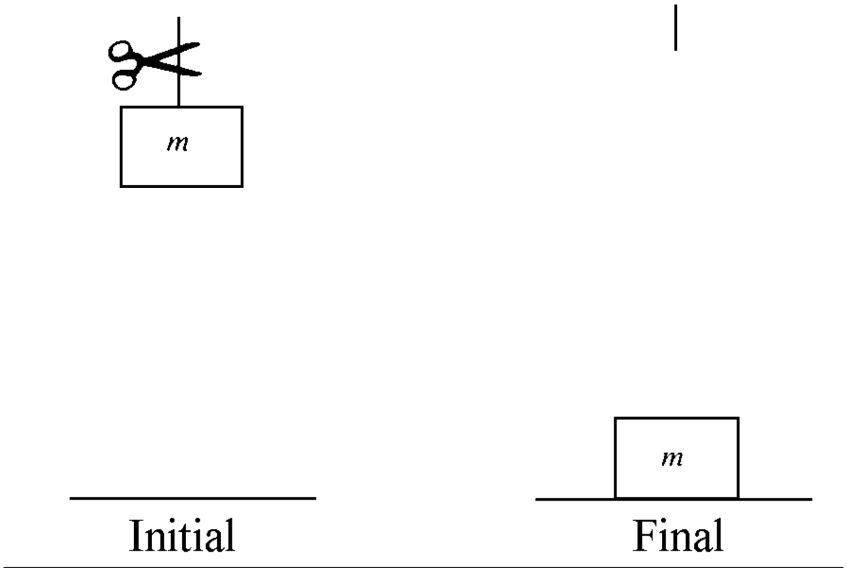


Figure 1 A mass is dropped onto a surface and comes to rest - an example of an irreversible process

1. If a mass is dropped onto a surface (see Figure 1), the mass loses potential energy–a form of mechanical energy. This mechanical energy is lost as the mass comes to rest. A little of it is converted into sound energy which is absorbed by surrounding material and has the effect of heating the absorbing material very slightly. More is converted directly into heat by the action of friction in the material onto which the mass is dropped. This conversion of mechanical energy into heat energy is a sure sign of an irreversible process because as will be proved later, all this heat cannot be converted back into work.

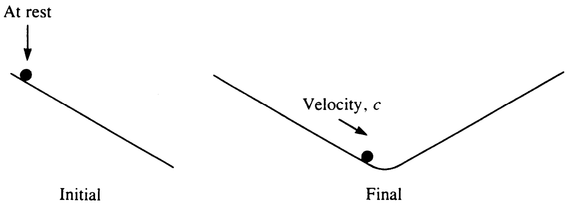


Figure 2 A ball-bearing rolls down a slope–an example of a reversible process

2. If a ball-bearing rolls down a slope without friction the ball loses potential energy, as in the last example. However this time the energy is converted into kinetic energy. This time it is possible to imagine that the process can be reversed, by causing the ball to roll up a similar slope, see Figure 2. In theory and in the absence of friction the kinetic energy will be reconverted into potential energy, and the ball will come to rest at the original height. It is easy to see that any friction will cause the final height to be less than the original height. In general, it is true that friction is a cause of irreversibility in real processes. The potential reversibility of this process occurs because one form of mechanical energy (potential energy) is converted into another form of mechanical energy (kinetic energy).

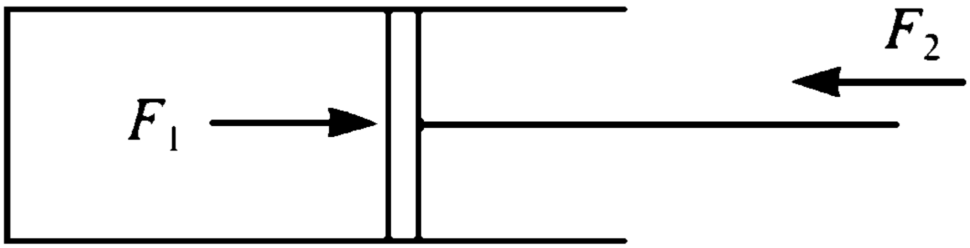


Figure 3 A cylinder containing compressed gas is confined by a movable piston

3. Figure 3 shows compressed gas in a cylinder. A force has to be applied to the piston to keep the gas compressed. Now if the force exerted by the gas on the piston is *F*1 and the external force applied to the piston is *F*2, and if *F*1 is very slightly greater than *F*2 then the piston will move slowly upwards and the gas will gradually expand. The pressure energy of the gas in the cylinder (a form of mechanical energy) is being converted into another form of mechanical energy as the force *F*2 is pushed back - perhaps the piston is pushing against a spring. This slow expansion and inter-conversion of mechanical energy is another example of a reversible process. If now *F*2 is very slightly greater than *F*1 the gas in the cylinder will be slowly recompressed. Again friction, this time between the piston and the cylinder will tend to cause irreversibility.

4. Alternatively the restraining force on the piston in Figure 3 could be suddenly removed, and the gas in the cylinder allowed to expand quickly. The pressure energy of the gas has decreased, but what has happened to the energy? It has been dissipated in friction at the walls of the cylinder, and in friction between rapidly moving adjacent streams of gas by means of viscous dissipation. The mechanical pressure energy has been turned into heat energy, and so the process is an irreversible one. Expansion of a gas in a cylinder, as in Figure 4 can only be reversible if the expansion is **fully resisted**, in other words if the forces acting on the cylinder are opposite and almost equal.

Processes can be described in other ways: if a process is carried out at constant pressure it is said to be **isobaric**, if it at constant temperature it is said to be **isothermal**, and at constant volume **isochoric**. **Diabatic** processes may exchange heat with the surroundings, but **adiabatic processes** are thermally insulated from the surroundings.

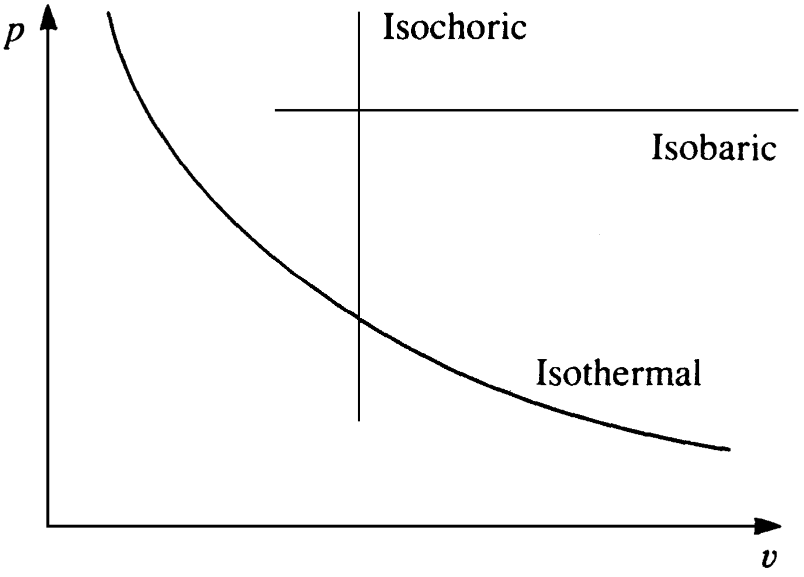


Figure 4 Various types of process on a *p*-*v* diagram

Processes can be illustrated as a series of points on a plot of two properties of state. A plot of pressure against specific volume: a *p*-*v* diagram is often used for this purpose, see Figure 5 where isobaric, isochoric, and (for an ideal gas) isothermal processes are marked. This latter process has the equation for an ideal gas:

$$pv = C$$

from Basic Engineering Thermodynamics, *by Peter B. Whalley. Reprinted with permission of Oxford University Press*

Top

Thermodynamic cycles

A **thermodynamic cycle** is a linked series of processes such that the outlet of the final process is the same state as the input of the first process. It will be seen later that when a thermodynamic cycle is used to convert heat into work, heat must necessarily be rejected from the cycle. Figure 1 shows the block diagram of a generalised cycle. At various points heat and work are input and output. Two important quantities are the **thermal efficiency** of the cycle, ηth, and the **work ratio**, *r*w defined by:

The thermal efficiency is a measure of how much of the input heat is converted into useful work, and makes the assumption that the heat rejected in the cycle is valueless. This is often, but not always, true. In an internal combustion engine the input heat is the heat liberated by the burning of the fuel. Of course a good cycle has a high thermal efficiency. A consequence of the first law of thermodynamics is that the maximum value possible for the thermal efficiency is 1.

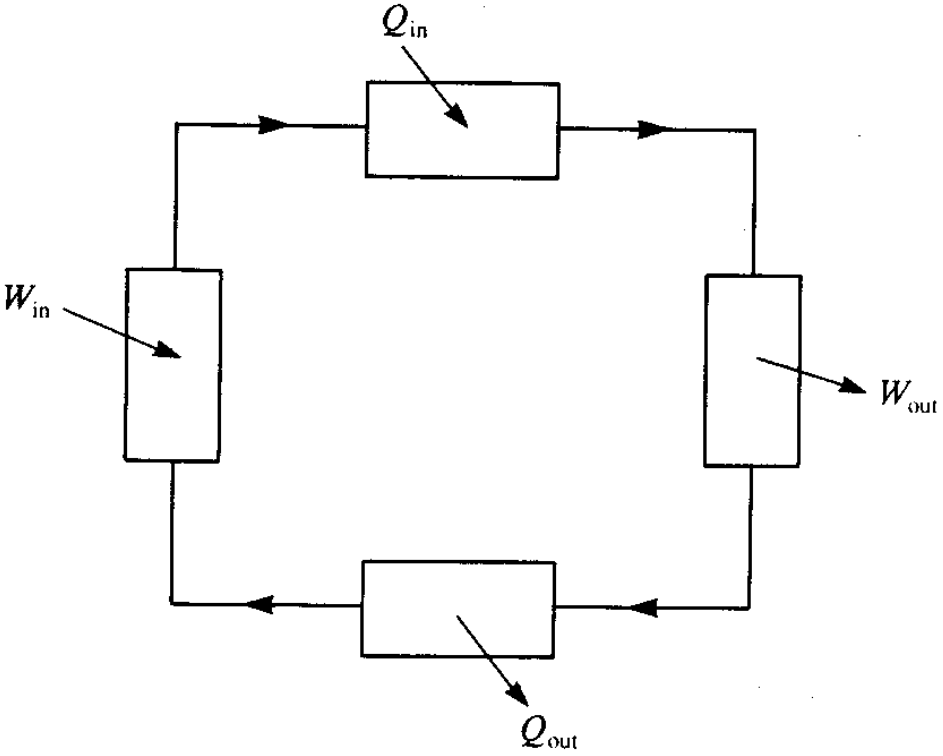


Figure 1 A block diagram of a generalised thermodynamic cycle showing transfers of heat and work in and out of the cycle}

The work ratio measures the relative sizes of the work out and the work in terms in Figure 1. If they are of comparable size, giving a low work ratio, then there are two undesirable consequences:

For a given value of the net work out, individual terms Wout and Win will be large. Thus large machines will be required to handle these large work transfers.

If the machines forming the cycle are not as mechanically efficient as expected, then Wout will be smaller than expected, and Win will be smaller than expected. This has the effect of drastically reducing the net work output below its expected value.

Hence a high work ratio is a desirable aim. The maximum value of the work ratio is 1.

from Basic Engineering Thermodynamics, *by Peter B. Whalley. Reprinted with permission of Oxford University Press*

Top

Index

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

A

Analyze Mode

B

Basic Engineering Thermodynamics
Basic Gas Turbine Cycle
Build Mode

C

Carnot cop-hp
Carnot cop-r
Combined Gas Turbine & Rankine Cycle
components
compressor
contradiction mode
cooler
cop-hp
cop-r
cp
cv

D

delta-H-isentropic
delta-H
delta-P
delta-spec-s

E

F

File menu

G

gamma

H

Heat Exchanger
Heater
Help menu
H
Hout-isentropic

I

Ideal Gas Law
J

K

L

Library

M

mass-flow
max-T
meter color coding
meters
min-T
mixer
mode menu
modeling assumptions
molar-mass

N

net-Q
net-work
eta-isentropic
eta-thermal

O

overview

P

parameters
P
preferences menu
PR
Psat
pump

Q

Q-in
Q-out
Q

R

R

S

shaft-work
Simple Refrigerator Cycle
Simple Steam Rankine Cycle
spec-hf
spec-hg

spec-h
spec-q
spec-sf
spec-sg
spec-shaft-work-isentropic
spec-shaft-work
spec-s
spec-u
spec-v
splitter
S
Steam Cycle with Reheat
stuff
substances

T

thermodynamic property tables
throttle
tools menu
Tout-isentropic
Tsat
T
turbine

U

U

V

V

W

windows menu
work-in
work-out

X

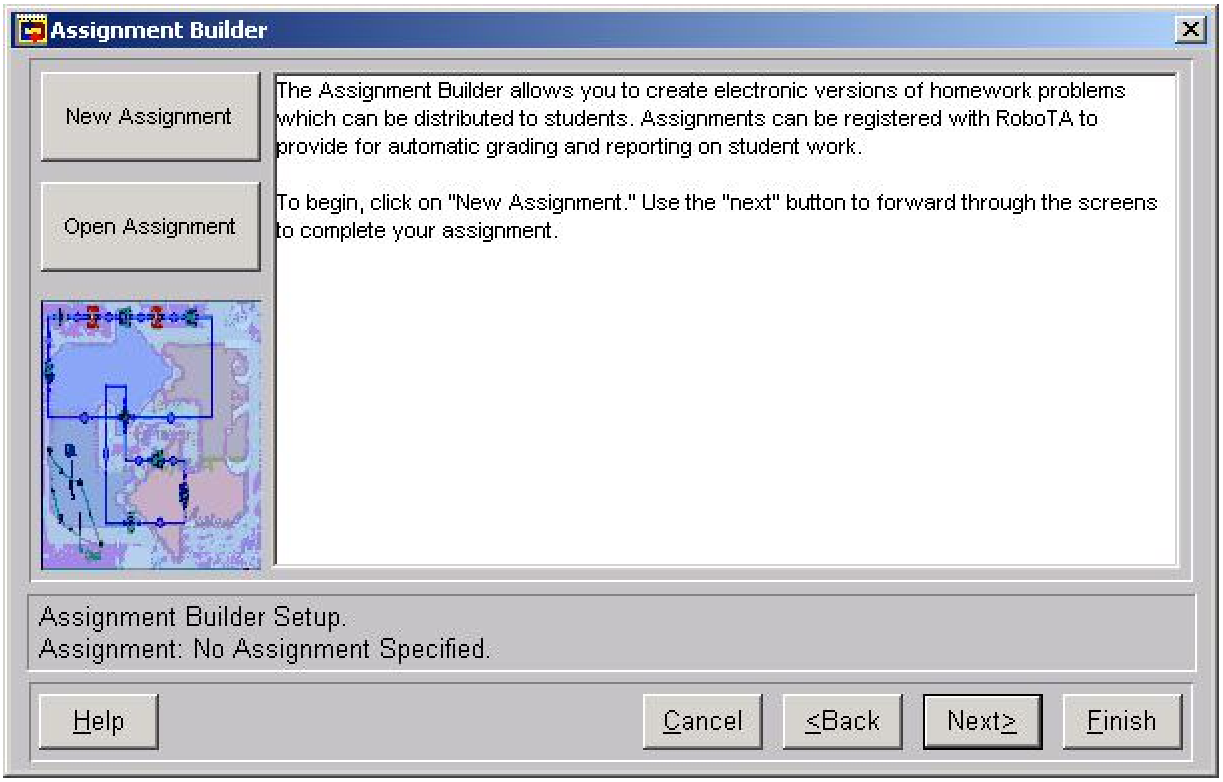
Y

z

Top

The Assignment Builder

Under the Instructor menu item, we can open the Assignment Builder. The Assignment Builder is a "wizard" that allows the creation and editing of CyclePad assignment problems.



The Assignment Builder allows you to create electronic versions of homework problems which can be distributed to students. Assignments can be registered with RoboTA to provide for automatic grading and reporting on student work.

To begin, click on "New Assignment." Use the "next" button to forward through the screens to complete your assignment.

Assignment Builder Setup.
Assignment: No Assignment Specified.

The Assignment Builder allows an instructor to create assigment problems for students to complete in CyclePad. Assignments created using the Assignment Builder reside in *assignment files* (with an extension of .ASN) which contain all the information the instructor has for the assignment and *problem files* (whose extension is .PRB) which are for distribution to the students and which contain a subset of the assignment information. The Assignment Builder creates both types of files.

These assignments have a number of features that instructors may find helpful.

- The problem files may be made available on a web site or distributed electronically.
- Assignments contain a text description of the thermodynamic system to be designed and the questions to be answered about the design.
- The instructor may specify *constraints* which the design must meet.

- Assignments may be submitted via CyclePad's built-in email submission facility and constraints and numerical answers can be graded electronically, with a report being sent to the instructor.
- In addition to the text description of the design, the instructor may provide as much of the design as he chooses to help the students start the assignment.

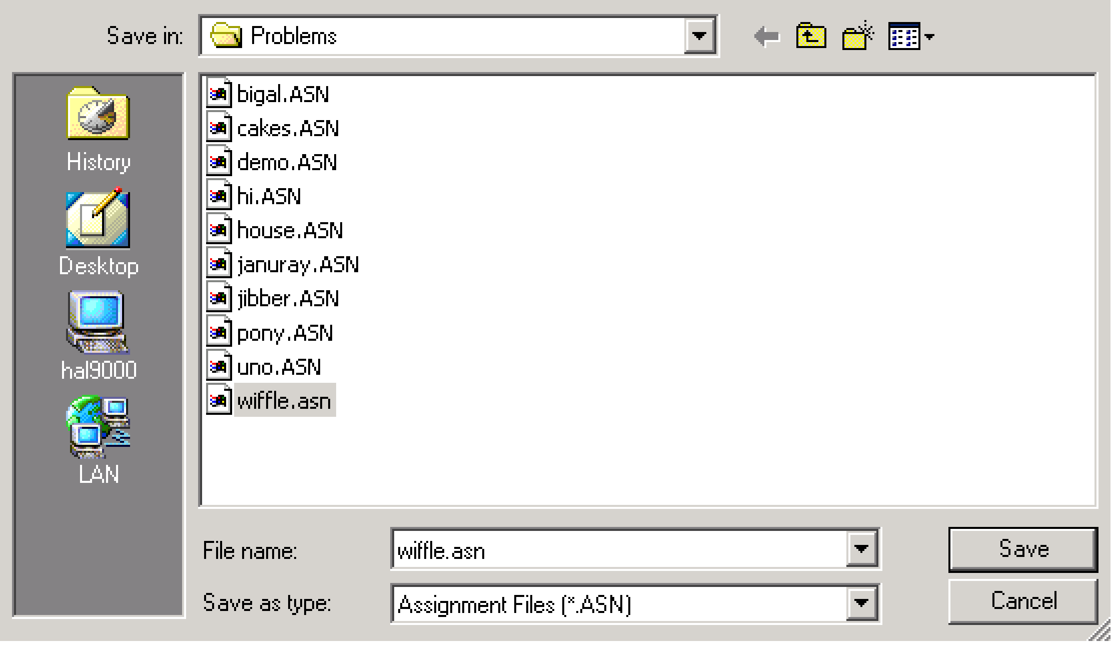
Creating an Assignment

Top

Creating a New Assignment

When creating an assignment using the Assignment Builder, the wizard takes us through several dialog boxes to guide us through the process.

Naming the Assignment File



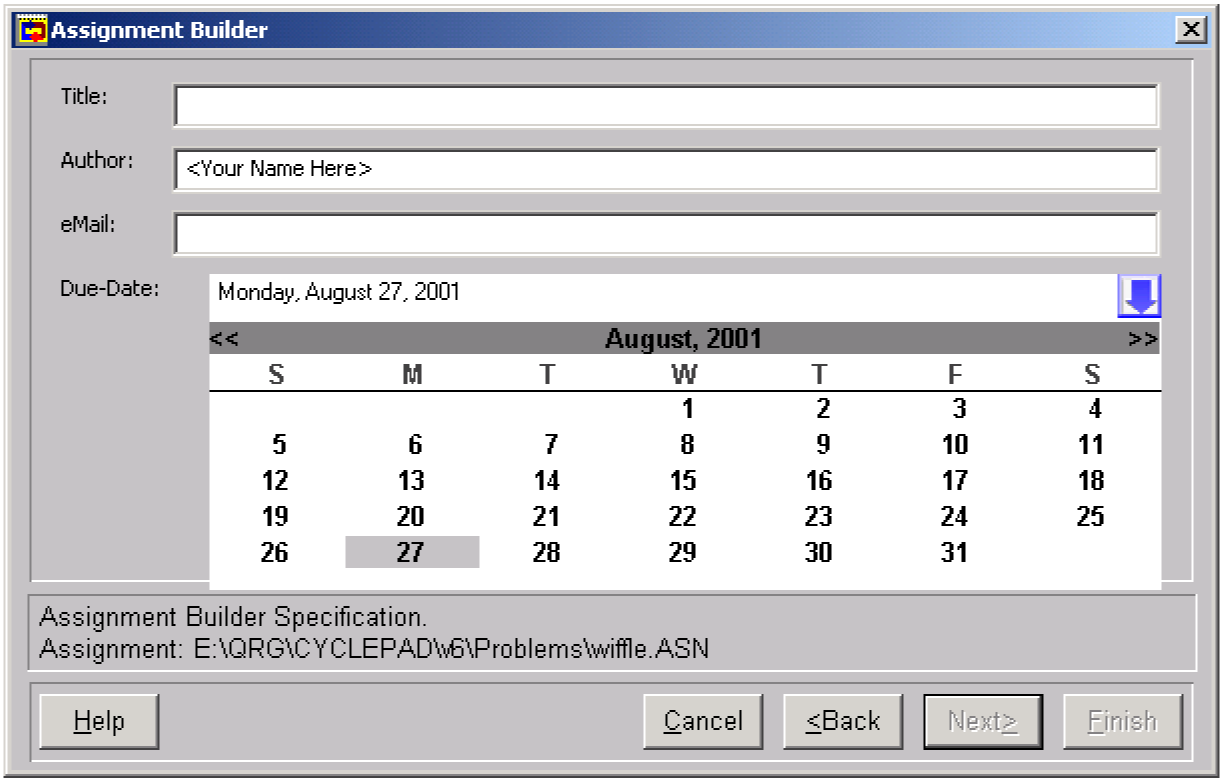
Save in: Problems

File name: wiffle.asn

Save as type: Assignment Files (*.ASN)

When we create a new assignment, the first step is to give a file name under which the assignment will be saved.

Basic Assignment Information



Title:

Author: <Your Name Here>

eMail:

Due-Date: Monday, August 27, 2001

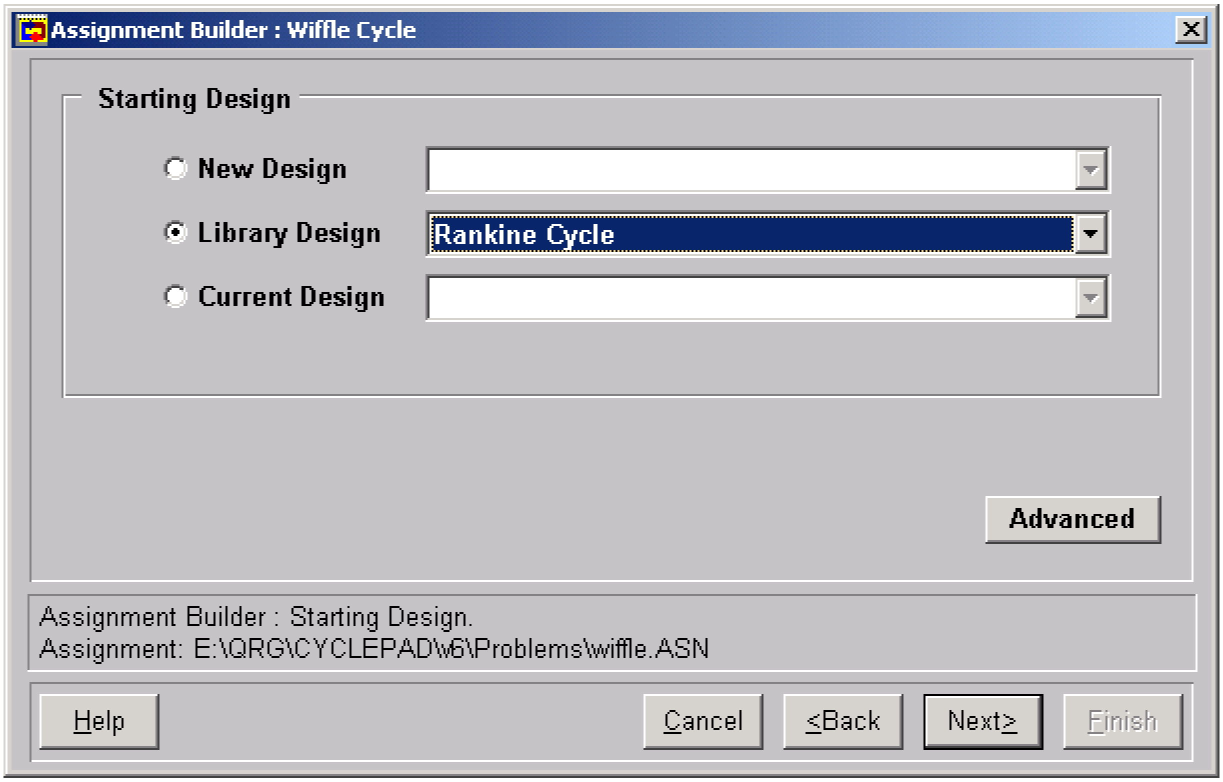
<< August, 2001 >>

S	M	T	W	T	F	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Assignment Builder Specification.
Assignment: E:\QRG\CYCLEPAD\v6\Problems\wiffle.ASN

Once a file name has been chosen, we click "Next" which takes us to a dialog allowing us to specify the assignment's title, it's author (the instructor), the author's email address (to which assignment grade reports will be sent for assignments graded electronically), and the assignment's due date. The due date is specified using the pull-down calender at the right of the due date field.

Setting Up the Problem



Starting Design

○ New Design

● Library Design: Rankine Cycle

○ Current Design

Assignment Builder : Starting Design.
Assignment: E:\QRG\CYCLEPAD\v6\Problems\wiffle.ASN

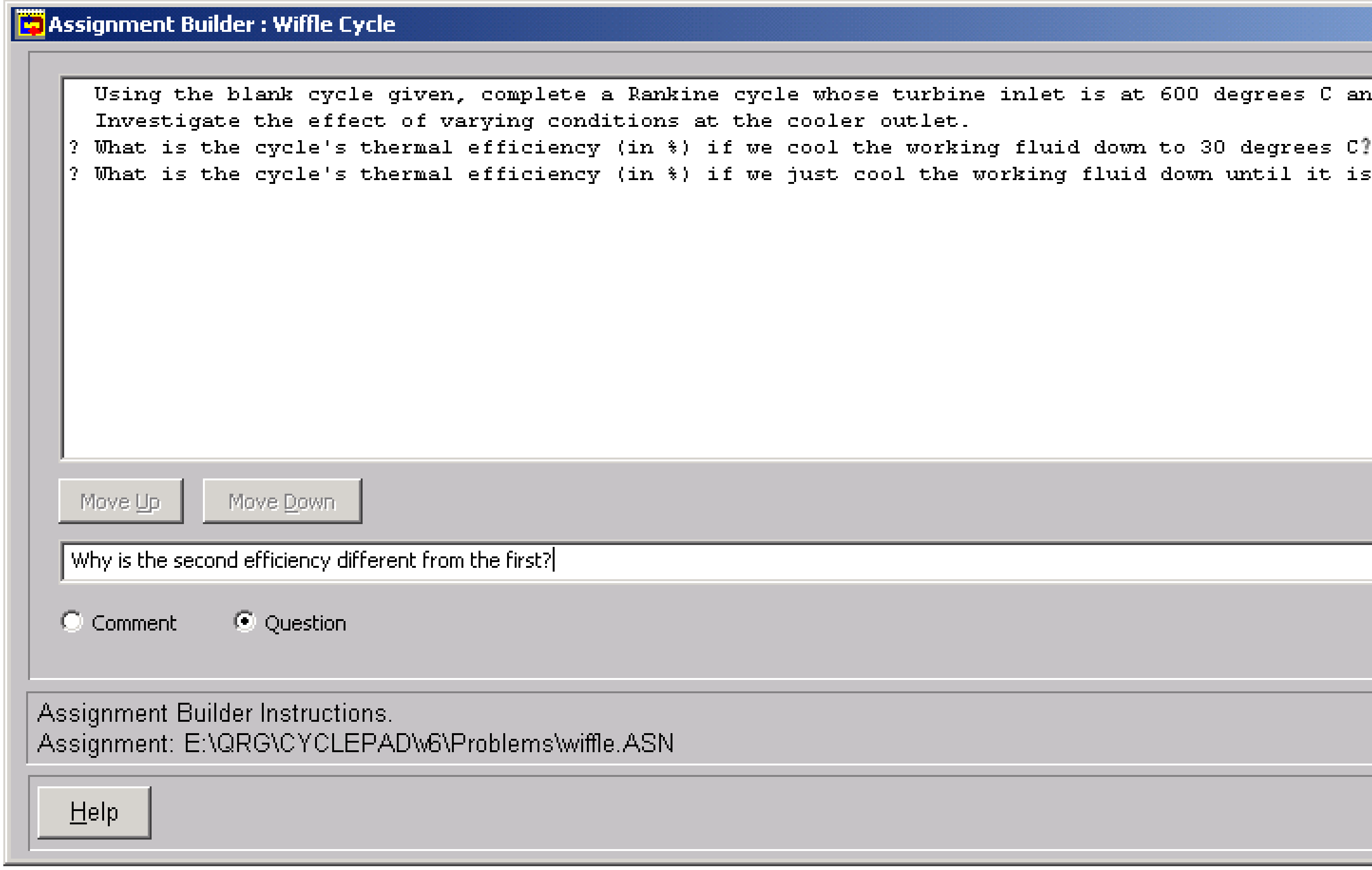
Next, we specify what design the student sees as he opens the assignment. We can choose several options.

- A new design. The student must build the assignment from scratch from the text description of the system.
- A CyclePad library design. The student starts with one of the designs in CyclePad's built-in library of examples. The student may modify these to complete the assignment.
- The current open design. If the instructor has a particular starting point in mind for the students, he can create that design on his own and have it open when entering the Assignment builder. Such designs may be at whatever level of completeness the instructor chooses.

In the assignment shown above, we have chosen a Rankine cycle as the students' starting point. Note that this is just the Rankine Cycle, and not the Rankine Cycle Solved, so no numerical data has been entered. The student will have to do that. Of course, had we chosen one of the solved library cycles (or provided one ourself), the student would have all of that design's data already entered. In this way, we can design introductory assignments where students have a chance to learn from the ground up and also more advanced assignments where the student need not spend unnecessary time on well-understood material.

Similar to those available under the Edit|Preferences menu item, this dialog contains a button for more advanced options. From here the instructor might choose to specify certain advanced behavior, such as that CyclePad use the slow-but-safe method when doing sensitivity analyses. Here also can be chosen units, though the units of the starting design are used by default.

Adding Assignment Instructions and Questions



```
  Using the blank cycle given, complete a Rankine cycle whose turbine inlet is at 600 degrees C an
  Investigate the effect of varying conditions at the cooler outlet.
? What is the cycle's thermal efficiency (in %) if we cool the working fluid down to 30 degrees C?
? What is the cycle's thermal efficiency (in %) if we just cool the working fluid down until it is
```

Why is the second efficiency different from the first?

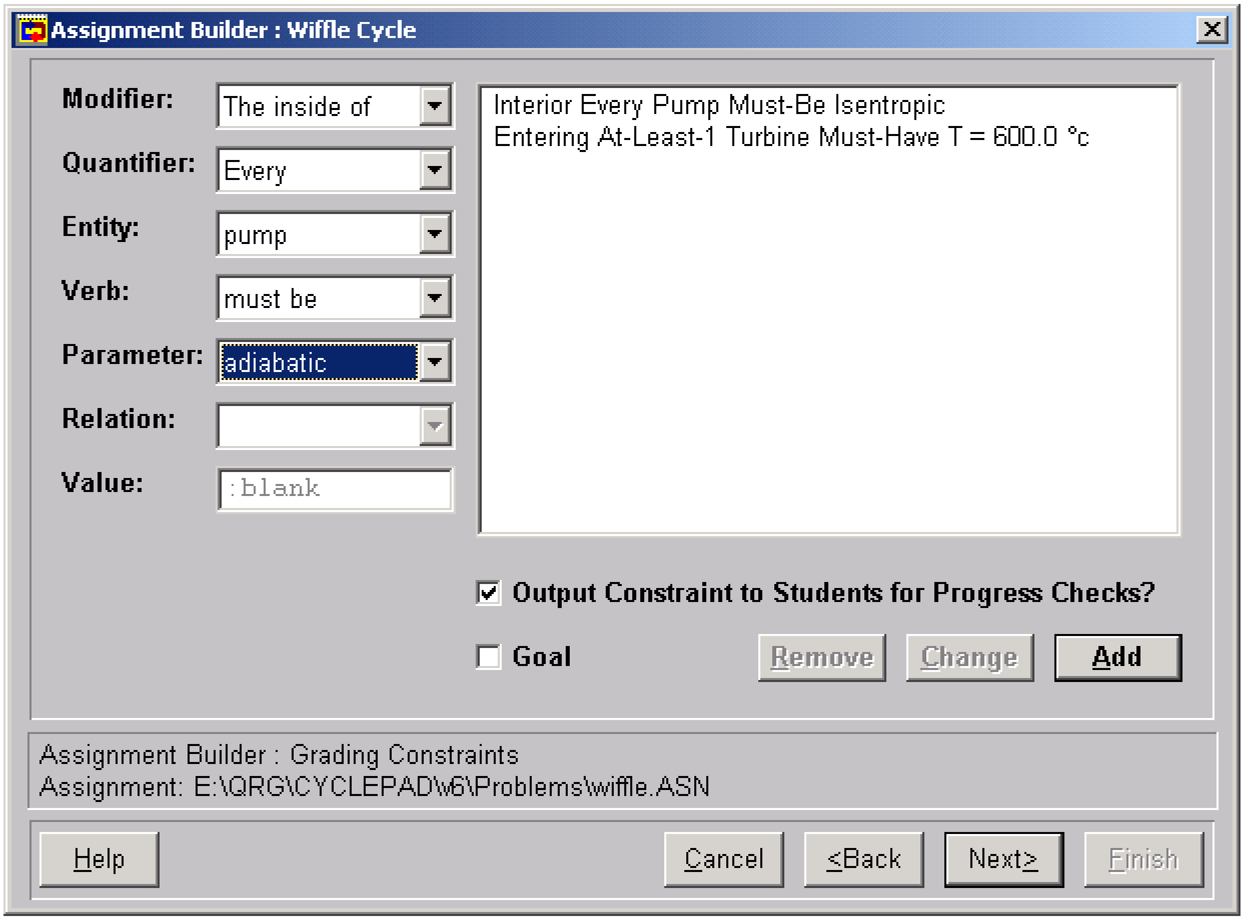
○ Comment ● Question

Assignment Builder Instructions.
Assignment: E:\QRG\CYCLEPAD\v6\Problems\wiffle.ASN

In the next dialog in the Assignment Builder Wizard, we provide the text description of the assignment, along with any instructions and the questions to be answered by the students.

Any description of the system and any instructions are "comments" and any questions for which the students must provide an answer are "questions". For each question specified here, the student will see a blank text box in which to provide answers when working on the assignment. Once a question or comment is written, we click "Add" to add it to the list that students will see when they work on the assignment. To change a question or comment, click on it then change it and then click "Change" to update the list.

Adding Problem Constraints



Assignment Builder : Wiffle Cycle

Modifier: The inside of

Quantifier: Every

Entity: pump

Verb: must be

Parameter: adiabatic

Relation:

Value: :blank

Interior Every Pump Must-Be Isentropic
Entering At-Least-1 Turbine Must-Have T = 600.0 °c

☑ Output Constraint to Students for Progress Checks?

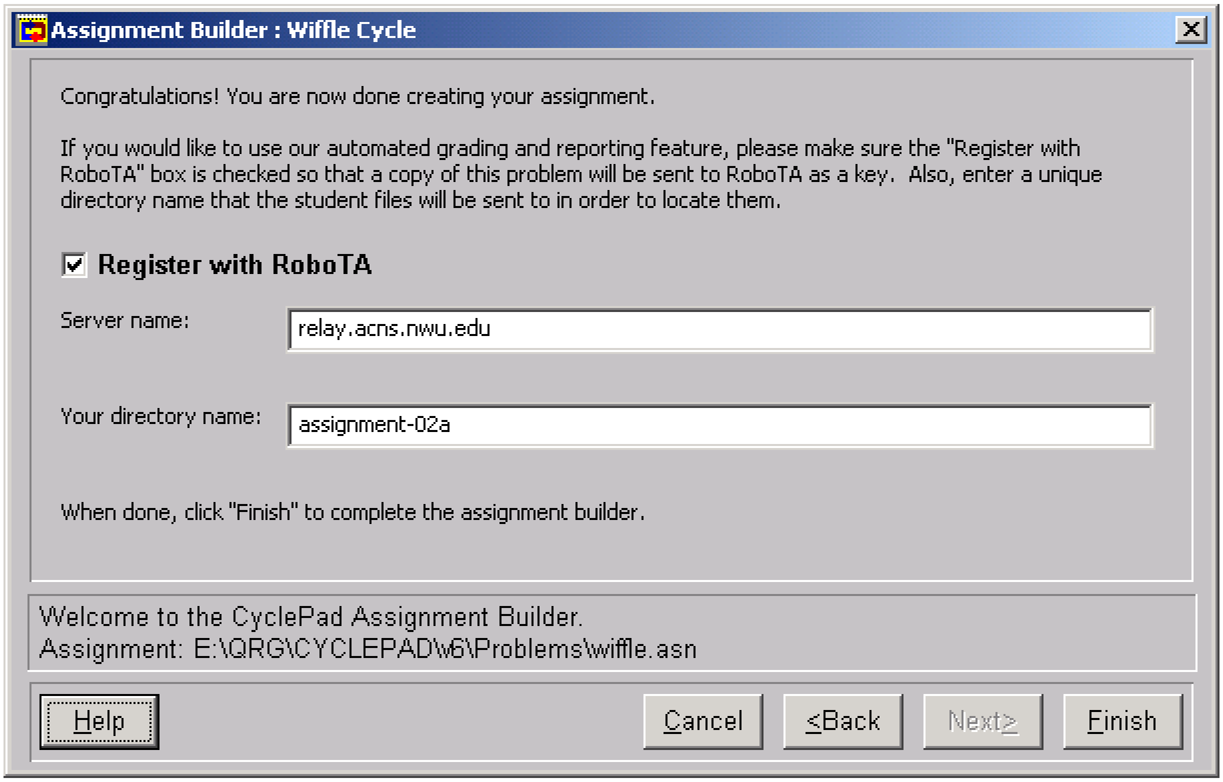
□ Goal

Assignment Builder : Grading Constraints
Assignment: E:\QRG\CYCLEPAD\v6\Problems\wiffle.ASN

The next dialog allows specifications of constraints that the students should follow when completing their design. The pull-down menus at the left allow each constraint to be specified in a way that CyclePad understands. Once each constraint is specified "Add" it to the list at the right. We can have as many constraints as we want, being careful not to overconstrain a problem.

In the example shown above, we have already specified that all the pumps are isentropic and that the fluid entering the turbine is at 600 degrees C. We are about to add a constraint that all pumps are adiabatic. Note that we can choose to allow the students to see some or all of the constraints so that they can check their own progress. We can also specify a goal constraint which will be tallied when the submitted assignment is graded.

Registering the Assignment File



Congratulations! You are now done creating your assignment.

If you would like to use our automated grading and reporting feature, please make sure the "Register with RoboTA" box is checked so that a copy of this problem will be sent to RoboTA as a key. Also, enter a unique directory name that the student files will be sent to in order to locate them.

☑ Register with RoboTA

Server name: relay.acns.nwu.edu

Your directory name: assignment-02a

When done, click "Finish" to complete the assignment builder.

Welcome to the CyclePad Assignment Builder.
Assignment: E:\QRG\CYCLEPAD\v6\Problems\wiffle.asn

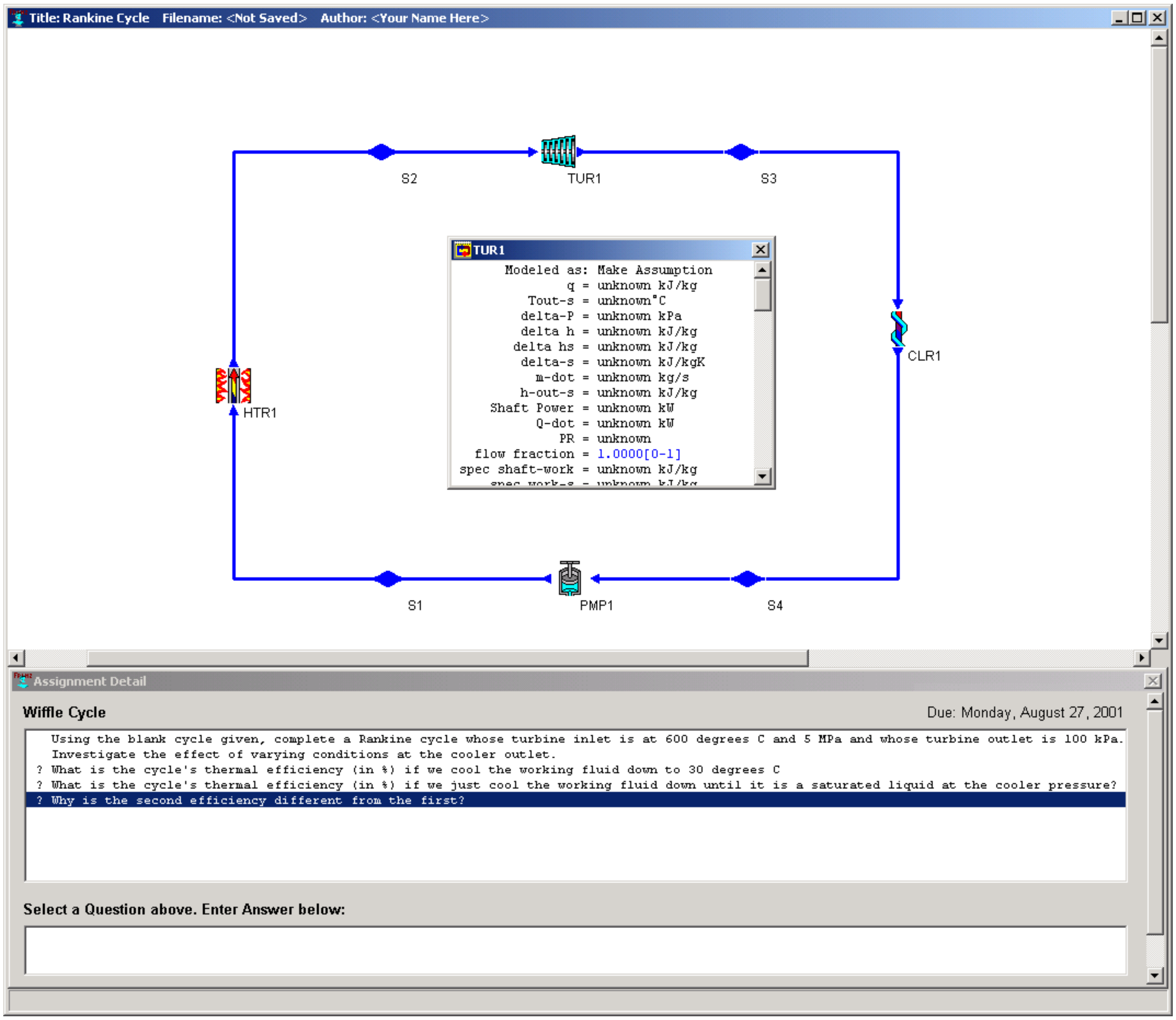
The final dialog in the Assignment Builder allows us to save the assignment file and creates the problem file, which the students receive. In addition it allows us to register the assignment with a RoboTA email post office and grading agent and specify in which directory on the RoboTA server the submitted assignments will be stored.

Of course, Registration with a RoboTA is not mandatory. Students can be instructed to submit their assignments directly to the instructor or his proxy for grading. Completed problem files can be opened and examined in CyclePad just as any design file can.

Whether or not it is desired to register the assignment with RoboTA, it is necessary to hit "Finish" to ensure that the problem file is created and the assignment file is saved.

What the Student Will See

When a student opens up the problem file (wiffle.prb in this example), he will see something like the following.



Title: Rankine Cycle Filename: <Not Saved> Author: <Your Name Here>
S2
TUR1
S3
HTR1
CLR1
S1
PMP1
S4
TUR1
Assignment Detail

```
      Modeled as: Make Assumption
               q = unknown kJ/kg
          Tout-s = unknown°C
         delta-P = unknown kPa
         delta h = unknown kJ/kg
        delta hs = unknown kJ/kg
         delta-s = unknown kJ/kgK
           m-dot = unknown kg/s
         h-out-s = unknown kJ/kg
     Shaft Power = unknown kW
           Q-dot = unknown kW
              PR = unknown
   flow fraction = 1.0000[0-1]
 spec shaft-work = unknown kJ/kg
     spec work-s = unknown kJ/kg
```

Wiffle Cycle

Due: Monday, August 27, 2001

```
  Using the blank cycle given, complete a Rankine cycle whose turbine inlet is at 600 degrees C and 5 MPa and whose turbine outlet is 100 kPa.
  Investigate the effect of varying conditions at the cooler outlet.
? What is the cycle's thermal efficiency (in %) if we cool the working fluid down to 30 degrees C
? What is the cycle's thermal efficiency (in %) if we just cool the working fluid down until it is a saturated liquid at the cooler pressure?
? Why is the second efficiency different from the first?
```

Select a Question above. Enter Answer below:

Top