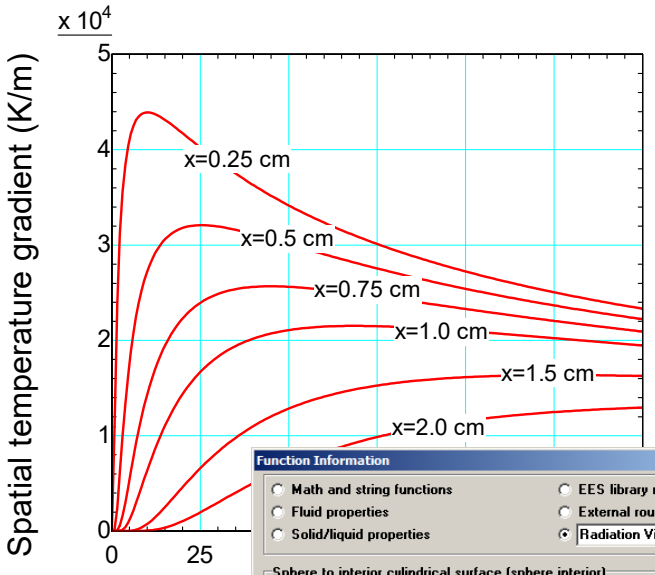
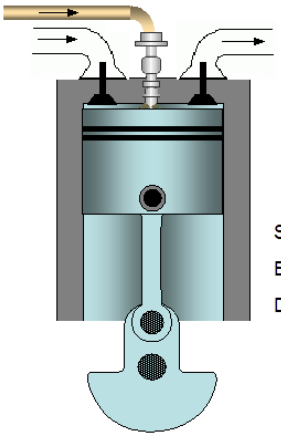
F-Chart Software

Mastering EES

Sanford Klein and Gregory Nellis

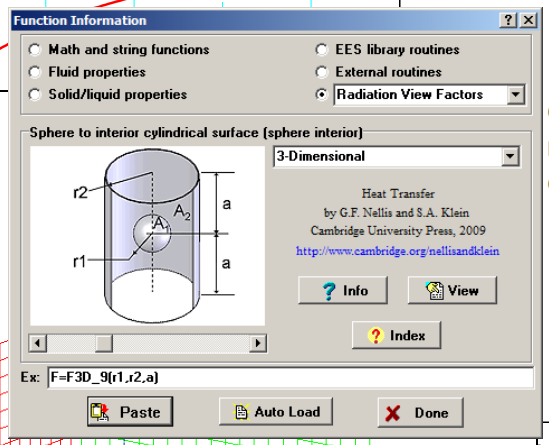


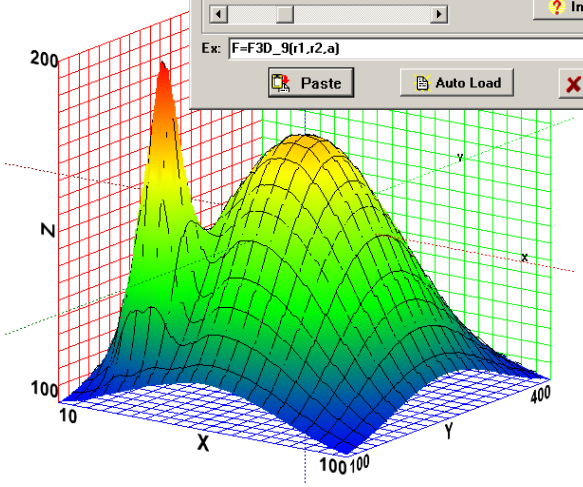


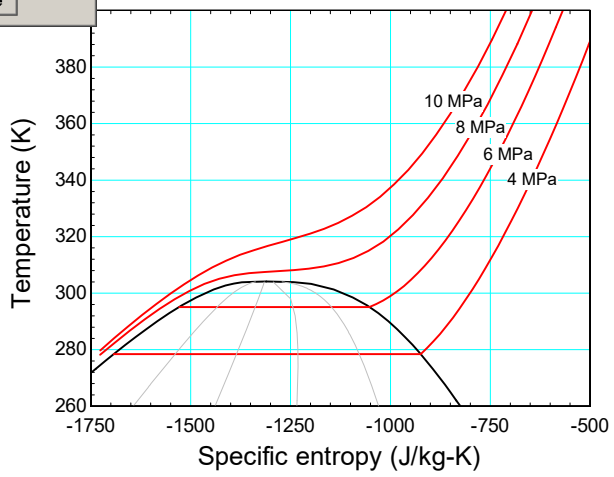
Stroke = 75 [mm]
Bore = 130 [mm]
Displacement = 0.9955 [l]

CompressionRatio = 18 Efficiency = 0
RPM = 3000 [1/min] Power = 0 [kW]
CutoffRatio = 2 T_{max} = 0 [K]

#10 Run 14







© S.A. Klein and G.F. Nellis, 2012-22

ALL RIGHTS RESERVED. This book contains material protected under International and Federal Copyright Laws and Treaties. Any unauthorized reprint or distribution of this book is prohibited. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system without express written permission from the author / publisher.

F-Chart Software

Box 44042

Madison, WI 53744

http://fchartsoftware.com

(116) Dec. 9, 2022

Starting with version 9.370, EES provides a Mastering EES menu item in the Help menu. If the Mastering-EES.pdf file is placed in the same directory as the EES application, selecting the Mastering EES menu item will start Adobe Reader, open the Mastering-EES.pdf file and automatically enter the password.

TABLE OF CONTENTS

1 INTRODUCTION TO EES 1
1.1 Acquiring and Installing EES 1
Acquiring EES 1
Installing EES 2
Silent Installation 3
Online Help 3
1.2 Entering and Solving Equations 6
Entering Equations 6
The Solution Window 8
Decimal vs Comma Separator 9
The Variable Palette and Information dialog 9
Guess Values 11
Limits 11
Display Format 12
Rules for Entering Equations 14
Comments 14
Hyperlinks in the Equations Window 15
Visible Hyperlinks 15
Hidden Hyperlinks 16
Built-In Math and String Functions 17
External Functions and Procedures 17
String Variables 18
The $TabStops Directive 19
Showing Values and Setting Variable Units in the Equations Window 19
The Status Bar and Screen Background Color 19
Equations and Display Preferences 20
The Formatted Equations Window 24
Password Protection 26
Key Variables 26
1.3 Parametric Tables 28
Creating a Parametric Table 30
Alter Values 31
Solving a Parametric Table 32
Formatting Columns 35
The $If, $IfNot, $Else, and $EndIf Directives 35
Naming and Documenting Parametric Tables 36
Saving and Loading Parametric Tables 37
1.4 Basic Plotting 39
Generating a Plot 39
Modifying Axes 40
Overlaying Plots 42
Modifying Plots 43
Automatic Update 44

Error Bars *45*
The Plot Tool Bar *46*
The Cross-Hairs, Move, and Zoom Tools *49*
1.5 Units *51*
Unit System *51*
Entering Units *51*
Dual Units *53*
Units Recognized by EES *54*
Checking Units *55*
Automatically Setting Units *58*
The Convert Function *58*
The Units List *59*
Suggested Method for Working with Units *61*
The ConvertTemp Function *63*
Adding Units *63*
Using String Variables for Units *65*
EXAMPLE 1.1 Oxygen Dewar *66*
1.6 Printing *66*
Printing a Hard Copy *66*
Copying Equations *67*
LaTeX Report *68*
1.7 Arrays *70*
Assigning Array Variables *70*
Array Range Notation *71*
Two-Dimensional Arrays *71*
The Array Editor *72*
The Duplicate Statement *74*
Arrays in the Variable Information Dialog *75*
Using the Variable Information Dialog with Large Arrays *76*
Purge Unused Variables *76*
EXAMPLE 1.2 Extended Surface Problem *78*
1.8 Lookup Tables and Lookup Files *78*
Creating a Lookup Table *78*
Entering Data *78*
Using Strings in Lookup Tables *80*
Using Equations in a Lookup Table *80*
Saving and Loading Lookup Files *82*
Deleting Lookup Tables from the Lookup Table Window *82*
Lookup Table/File Commands *83*
Lookup File Formats *84*
EXAMPLE 1.3 Pump Performance Model *86*
1.9 Other Features *86*
Built-In Constants *86*
Adding Constants *87*
The Calculator Window *88*
Preferences and Syntax Highlighting *90*

References *93*

2 CURVE FITTING & INTERPOLATION **94**
2.1 Curve Fitting *94*
Curve Fitting Plotted Data *94*
EXAMPLE 2.1 Fitting Compact Heat Exchanger Data *99*
Curve Fitting Data in an Array Table with the CurveFit1DT Procedure *99*
Curve Fitting Data in a Lookup Table using the CurveFit1DT Procedure *101*
EXAMPLE 2.2 Fitting the Saturation Pressure of Water *102*
2.2 Linear Regression *103*
The Linear Regression Menu command *103*
The LinearRegression Procedure *107*
EXAMPLE 2.3 Fitting Compressor Data *107*
EXAMPLE 2.4 Fitting Compressor Data (revisited) *111*
2.3 One Dimensional Interpolation *111*
2.4 Two Dimensional Interpolation *112*
The Interpolate2D Function *113*
The Interpolate2DM Function *114*

3 FUNCTIONS AND PROCEDURES **115**
3.1 Equations and Assignment Statements *115*
The Assignment Operator *116*
3.2 Functions *116*
Format of Functions *116*
First Example of a Function *117*
Second Example of a Function *119*
Setting and Checking Units for Function Variables *119*
Variable Information Page for a Function *120*
3.3 Procedures *121*
Format of Procedures *121*
Example of a Procedure *122*
3.4 Logic Statements *124*
If-Then-Else Statements *124*
EXAMPLE 3.1 Wind Turbine *125*
Return Statements *125*
GoTo Statements and Statement Labels *127*
Repeat-Until Construct *128*
Case Statements *128*
EXAMPLE 3.2 Compressor Refrigeration Capacity *129*
3.5 Units in Functions and Procedures *131*
3.6 Arrays in Functions and Procedures *134*
Arrays as Arguments *134*
Using the $Common Directive to Access Arrays *136*
Using the $Constant Directive to Set Array Limits *136*
Arrays Table Window for Functions and Procedures *137*
EXAMPLE 3.3 Lumped Capacitance Problem *137*
3.7 Using Lookup Tables *140*
3.8 The Beep and Print Commands *141*

3.9 The Warning and Error Procedures ... *143*
The Warning Procedure ... *143*
The Error Procedure ... *145*
References ... *146*

4 PROPERTY DATA ... **147**
4.1 Unit System ... *147*
Unit System Dialog ... *147*
Specific Properties on a Molar vs Mass Basis ... *147*
The $UnitSystem Directive ... *148*
Unit Settings in the Status Bar ... *149*
Importance of Unit System Selection ... *149*
4.2 Function Information ... *149*
4.3 Property Functions for Real Fluids and Mixtures ... *150*
Calling Protocol for Property Function ... *150*
List of Property Functions ... *151*
The RealThermoProps Procedure ... *153*
Built-in Real Fluids ... *155*
Property Indicators ... *157*
Fixing the State ... *157*
Two-Phase States ... *158*
The Example Box ... *159*
EXAMPLE 4.1 Vapor Compression Cycle ... *160*
Equations of State ... *163*
Properties of Water ... *163*
The Reference State ... *165*
The $Reference Directive ... *166*
4.4 Property Functions for Ideal Gases ... *167*
The Ideal Gas Model ... *167*
List of Ideal Gas Fluids ... *168*
Ideal Gas vs Real Gas Fluids ... *168*
The IdealGasThermoProps Procedure ... *169*
EXAMPLE 4.2 Gas Turbine Engine ... *171*
The NASA Ideal Gas Database ... *174*
The FluidType$ Function ... *176*
EXAMPLE 4.3 Turbine Function ... *176*
The Ideal Gas Reference State & Chemical Reactions ... *177*
EXAMPLE 4.4 Chemical Reaction between Methane and Oxygen ... *177*
EXAMPLE 4.5 Adiabatic Combustion with Chemical Equilibrium ... *179*
4.5 Psychrometric Properties ... *184*
The Fluid AirH2O ... *184*
Properties Specific to Psychrometrics ... *184*
List of Psychrometric Properties ... *185*
List of Indicators for Psychrometric Properties ... *186*
Psychrometrics Example ... *186*
The PsychProps Procedure ... *187*
EXAMPLE 4.6 Humidification System ... *188*

4.6 Brine Properties ... *188*
Brine Property Functions ... *188*
Brine Fluid Mixtures ... *190*
Using the Brine Property Functions ... *191*
The BrineProp2 External Procedure ... *192*
4.7 The Incompressible Substance and TPSX Libraries ... *193*
Incompressible Property and TPSX Property Library Functions ... *194*
Using the Incompressible and TPSX Library Property Functions ... *196*
Solid-Liquid_Props Library File (deprecated) ... *197*
Incompressible and TPSX Property Tables ... *197*
Adding Incompressible Property Data ... *201*
4.8 More Libraries ... *205*
4.9 Property Plots ... *206*
The Property Plot Dialog ... *206*
Property Plots for Real Fluids ... *208*
Property Plots for Ideal Gases ... *211*
Psychrometric Plots ... *212*
Property Plots for Brines ... *214*
Property Plots Using the EES_REFPROP Interface ... *215*
Property Plots for Substances in the Incompressible and TPSX Libraries ... *215*
Overlaying States onto Property Plots ... *217*
4.10 NH3H2O (Ammonia-Water) ... *220*
EXAMPLE 4.7 Absorption Refrigeration Cycle ... *222*
4.11 Property Data in External Procedures ... *225*
The IdealGasMixtureProps Library Procedure ... *226*
The NASA Library ... *228*
Ammonia-Water Properties ... *230*
Sea Water Properties ... *232*
Lithium Bromide-Water and Lithium Chloride-Water Properties ... *232*
The GENEOS Library ... *234*
The Peng-Robinson Library ... *234*
The EES_REFPROP Interface ... *235*
EXAMPLE 4.8 Using EES Property Functions to Access REFPROP data ... *235*
EXAMPLE 4.9 Using the EES_REFPROP Procedure ... *244*
Metal (Solid, Liquid, Vapor) Properties ... *245*
The Rocket Propellants Library ... *246*
4.12 Adding Property Information ... *248*
Providing Data in Lookup Tables ... *249*
Providing Ideal Gas Property Data ... *249*
Providing Real Fluid Property Data represented by the Martin-Hou Equation of State ... *252*
Using REFPROP .fld Files ... *257*
References ... *259*

5 CONVERGENCE AND DEBUGGING ... 260
5.1 Solution Methodology Used in EES ... *260*
Numerical Solution of One Non-Linear Equation ... *260*
Stopping Criteria ... *263*

EXAMPLE 5.1 Setting Guess Values and Limits 264
Numerical Solution of Simultaneous Non-Linear Equations 265
Blocking and Reordering Equation Sets 269
5.2 The Residuals Window 271
Blocks 271
Residuals 272
Units 272
Calls and Time 273
Procedures in the Residuals Window 274
EXAMPLE 5.2 Blocking Equations 275
5.3 The Computational Flow Window 275
5.4 Setting Guess Values and Limits 282
The Variable Information dialog 282
Setting Guess Values and Limits using Variables 283
Arrays in the Variable Information dialog 284
Changing Variable Names 285
Variable Information Files 285
EXAMPLE 5.3 Using Arrays as Guess Values and Limits 286
Default Variable Information 289
5.5 Using the Debug Window 289
Incorrect Degrees of Freedom 289
Constrained Solution 291
5.6 Debugging Techniques 291
Effective Use of EES 292
The Update Guesses Command and Directive 292
EXAMPLE 5.4 Using the Update Guesses Command 293
The Residuals or Computational Flow Window as a Debugging Tool 296
Common Problems 298
The $Trace Directive 299
References 301

6 OPTIMIZATION 302
6.1 One-Dimensional Optimization 302
Implementing an Optimization Problem 303
Objective Function 305
Degrees of Freedom 305
The Min/Max Dialog 305
Stopping Criteria 306
Guess Value and Bounds 307
The Golden Section Search 307
The Quadratic Approximations Optimization Method 309
Min/Max Table 311
EXAMPLE 6.1 Maximum Power of a Heat Transfer-Limited Power Cycle 313
6.2 Multi-Dimensional Optimization 316
The Conjugate Directions Method 319
The Variable Metric Method 320
The Nelder-Mead Simplex Method 320

The DIRECT Algorithm *321*
The Genetic Method *322*
EXAMPLE 6.2 Non-linear Regression using Optimization *325*
6.3 Constrained Optimization *328*
Implementing Constraints using Bounds *328*
Parameterizing Variables *329*
Penalty Function *331*

7 INTEGRATION **334**
7.1 Numerical Integration of Ordinary Differential Equation *334*
Example ODE *334*
Analytical Solution *335*
Discretizing Time *336*
Euler's Technique (First Order Explicit) *336*
Heun's Method (Second Order Explicit) *340*
The Fully Implicit Method (First Order Implicit) *343*
The Crank-Nicolson Method (Second Order Implicit) *344*
7.2 Equation-Based Integral Function *345*
Calling Protocol for the Integral Command *345*
Entering the State Equations *345*
Carrying out the Integration *346*
The Integral Table *347*
Adaptive Step Size *348*
The $IntegralStop Directive *350*
Integrating Simultaneous ODEs *351*
EXAMPLE 7-1 Simultaneous ODEs *351*
The IntegralValue Function *354*
EXAMPLE 7-2 The IntegralValue Function *354*
Double Integration *358*
Multiple Integral Tables *360*
7.3 Table-Based Integral Function *362*
EXAMPLE 7.3 Use of the Table-Based Integral Function *362*
7.4 Solving Partial Differential Equations *363*
EXAMPLE 7.4 Solving Partial Differential Equations *363*

8 UNCERTAINTY PROPAGATION **369**
8.1 Uncertainty Propagation using the RSS Method *369*
The Root Sum Square Method *369*
Example of the RSS Method *370*
8.2 Uncertainty Propagation in EES *372*
Assigning Uncertainties to Measured Variables *372*
Determining the Uncertainties of Calculated Variables *374*
UncertaintyOf Function *375*
Uncer.Pardif and Uncert.Percent Functions *376*
8.3 Uncertainty Propagation in Tables *376*
Uncertainty Propagation Table *376*
Plotting Data with Uncertainties *377*
Uncertainty Propagation with Lookup Tables *378*

EXAMPLE 8-1 Uncertainty Calculations using Data in a Lookup Table *378*
8.4 Uncertainty Propagation in Integrals *382*
EXAMPLE 8-2 Uncertainty in the Calculation of Work to Compress Ammonia *382*

9 ADVANCED PLOTTING **386**
9.1 General Plot Window Options *386*
Naming and Documenting Plots *386*
Duplicating a Plot *387*
Saving a Plot to a File *387*
Copying a Plot to a Different Application *387*
Storing and Retrieving Plots *389*
Deleting Plot Windows *389*
Setting the Default Plot Appearance *390*
Plot Templates *390*
Plot Thumbnails Window *390*
9.2 Advanced Capabilities of X-Y Plots *392*
The Cross-Hairs Tool *393*
Point Labels *394*
EXAMPLE 9-1 Analysis of a Refrigeration Cycle *395*
Dual Axis Scales for Two-Dimensional Plots *398*
EXAMPLE 9-2 Plots with Dual Axes *398*
Plotting Date and DateTime Information *401*
Using Variables as Axis Limits *404*
9.3 Other Types of 2-D Plots *406*
Bar Plots *406*
Overlays on Bar Plots *409*
Bar Plots using String Values *412*
Side-by-Side Bar Plots *412*
Area Plots *414*
Polar Plots *417*
Time Sequence Plots *419*
9.4 Three-Dimensional Plots *427*
Three Column Data *428*
Contour-Lines Contour Plot *429*
Contour-Bands Plot *430*
Contour-Gradient Plot *432*
3D Surface Plots Using 3-Column Data *432*
3D Surface Plots Using 2-D Tabular Data *436*
3D Polygon/Points Plots *438*
Symbol Color Plots *440*
References *441*

10 SUBPROGRAMS and MODULES **442**
10.1 Subprograms *442*
Format of a Subprogram *442*
Solution and Residuals Windows *444*
Guess Values and Limits of Inputs and Outputs *446*
Subprograms with the Integral Command *447*

Subprograms to Determine the Limit of an Integral ... *452*
10.2 Modules ... *454*
Module Execution ... *454*
The Solution Window ... *455*
The Residuals Window ... *456*
Guess Values and Limits of Inputs and Outputs ... *457*
10.3 Should you use Subprograms or Modules? ... *457*
References ... *458*

11 INTERNAL LIBRARY FILES ... 459
11.1 Writing and Saving Library (.lib) Files ... *459*
An Example Library File Function ... *460*
Units in Library File Functions ... *460*
Testing a Library File Function ... *463*
Saving Functions as a Library File ... *463*
11.2 Loading and Using Library Files ... *464*
EES Library Routines in the Function Information Dialog ... *464*
Libraries in the Userlib (Userlib64) Folder ... *465*
Manually Loading Library Files ... *466*
11.3 Help for Library Files ... *467*
Using a Dedicated Help File ... *468*
11.4 Application Library Files ... *470*
11.5 The Textbook Menu ... *470*
11.6 The Library Manager ... *473*
References ... *474*

12 THE HEAT TRANSFER LIBRARY ... 475
12.1 Boiling and Condensation ... *475*
EXAMPLE 12.1 Steam flow in a pipe ... *479*
12.2 Compact Heat Exchangers ... *483*
Compact Heat Exchanger Data ... *483*
Organization of the Compact Heat Exchanger Library ... *484*
Non-Dimensional Functions ... *485*
Geometry Functions ... *485*
Coefficient of Heat Transfer Functions ... *486*
Pressure Drop Functions ... *486*
12.3 Steady State Conduction ... *487*
Conduction Shape Factors ... *487*
Spreading Resistances ... *489*
EXAMPLE 12.2 Spreading Resistance ... *490*
12.4 Convection ... *491*
Internal Forced Convection ... *491*
EXAMPLE 12.3 Designing a Water Heater ... *494*
External Forced Convection ... *496*
Free Convection ... *498*
Impinging Jets ... *499*
Regenerator Packings ... *500*
12.5 Fin Efficiency ... *501*

12.6 Flow Maps ... *503*
EXAMPLE 12.4 Flow Maps ... *503*
12.7 Fouling Factors ... *505*
12.8 Heat Exchangers ... *506*
Log Mean Temperature Difference Solutions ... *506*
Effectiveness-NTU Solutions ... *508*
Axial Conduction and Regenerator Models ... *510*
12.9 Radiation View Factors ... *510*
12.10 Total Hemispherical Emissivity ... *511*
12.11 Transient Conduction ... *512*
12.12 Pipe Dimensions ... *515*
12.13 Minor Losses and Flow Measurement ... *516*
References ... *518*

13 COMPLEX ALGEBRA ... **519**
13.1 Introduction to Complex Algebra in EES ... *519*
Assigning Complex Variables in Rectangular Form ... *520*
Assigning Complex Variables in Polar Form ... *521*
Units of Complex Variables ... *522*
The Variable Information dialog ... *523*
Complex Algebra ... *523*
Functions Specific to Complex Variables ... *524*
Built-In Functions and Complex Variables ... *525*
13.2 EXAMPLE 13.1 Analysis of an AC Circuit ... *525*
Parametric Table ... *527*
The $Real Directive ... *529*
Subprograms ... *530*

14 DIRECTIVES ... **531**
14.1 Directives related to Display & Formatting Options ... *531*
$Allow '//' ... *531*
$Arrays On/Off ... *531*
$Bookmark ... *532*
$CheckUnits On/Off/AutoOn/AutoOff ... *533*
$DiagramUpdate On/Off ... *534*
$Font ... *534*
$HideWindow ... *534*
$Private ... *535*
$ShowWindow ... *536*
$SumRow On/Off ... *536*
$Syntax ... *537*
$TabStops ... *538*
$Warnings On/Off ... *538*
14.2 Directives related to Units ... *538*
$AutoSetArrayUnits On/Off ... *538*
$CheckUnits On/Off/AutoOn/AutoOff ... *539*
$ConvertEESREFPROPUnits On/Off /mass ... *539*
$UnitSystem ... *540*

14.3 Directives for Code Segments ... *540*
$Common ... *540*
$Constant ... *541*
14.4 Directives related to Complex Algebra ... *542*
$Complex On/Off ... *542*
$Real ... *542*
14.5 Directives related to Saving & Copying Data ... *543*
$CopyToLookup ... *543*
$Export ... *544*
$ExportPlot ... *546*
$ExportText ... *546*
$Import ... *547*
$ImportText ... *549*
$OpenLookup ... *550*
$SaveLookup ... *551*
$SaveTable ... *551*
$SaveVarInfo /A ... *552*
14.6 Directives related to Program Execution & Debugging ... *553*
$ClearLookup 'LookupTableName' /C1..3 /R2..5 ... *553*
$ClearParametric 'ParametricTableName' /A /R220..500 ... *553*
$DefaultArraySize=100 ... *553*
$DefaultDirectory DirName ... *553*
$DoLast & $EndDoLast ... *553*
$If, $IfNot, $Else & $EndIf ... *555*
EXAMPLE 14.1 Use of $IF and other Directives ... *556*
$Include FileName /R ... *559*
$Input V ... *560*
$IntegralAutoStep ... *561*
$IntegralStop ... *561*
$IntegralTable ... *562*
$Keyboard US/EU ... *562*
$Load ... *562*
$MaxCalls N ... *562*
$Python Python_Directory ... *563*
$Recompile ... *563*
$Reference ... *563*
$RunMacroAfter ... *564*
$RunMacroBefore ... *565*
StopCriteria ... *565*
$Trace ... *566*
$Unload LibName ... *566*
$UpdateGuesses /A ... *566*
$VarInfo ... *567*
References ... *568*
15 THE DIAGRAM WINDOW ... **569**
15.1 Placing Graphic Objects in the Diagram Window ... *569*

Development and Application Modes *569*
Graphical Objects from a Drawing Program *570*
Graphical Objects Drawn in the Diagram Window *571*
Drawing and Modifying Lines and Arrows *572*
Drawing and Modifying Rectangles *574*
Drawing and Modifying Ellipses or Circles *576*
Drawing and Modifying Polylines and Polygons *577*
Accessing the Diagram Window Palette *579*
The Add Picture from File Capability *582*
15.2 Placing Text Items in the Diagram Window *583*
Entering Plain Text *583*
Entering Formatted Text *585*
EXAMPLE 15.1 Performance of an IC Engine *588*
15.3 Entering Input and Output Variables *590*
EXAMPLE 15.2 Setting up a Graphical User Interface *590*
Entering Input Numerical Variables from the Diagram Window *591*
Displaying Output Variables on the Diagram Window *592*
Limits on Input Variables *594*
Entering Input String Variables from the Diagram Window *595*
Mapping String Variables in a Drop-Down List to Equations *597*
The Find Command *599*
15.4 Copying and Resizing the Diagram Window *600*
The Copy Command in Application Mode *600*
The Copy Command in Development Mode *600*
Change Window Size Dialog *601*
Clearing the Diagram Window *601*
15.5 Adding Calculate, Print, Plot Access, and Help buttons *602*
Adding a Calculate Button *602*
Moving and Resizing the Calculate Button *602*
Calculate Button Characteristics Dialog *602*
Adding a Print Button *604*
Adding a Plot Access Button *605*
Showing a Plot on the Diagram Window *607*
EXAMPLE 15.3 Using a Macro to Create the Plot *609*
Adding a Help Button *609*
15.6 Child Diagram Windows *610*
Creating a Child Diagram Window *610*
How to Modify Child Diagram Property Information *611*
How to Access a Diagram Window from another Diagram Window *612*
15.7 Other Professional License Enhancements *613*
Group and Ungroup Selected Items Buttons *613*
Align Selected Items Button *615*
Showing a Grid on the Diagram Window *615*
Creating Links *615*
Adding an Audio-Visual Item *618*
Saving and Loading User Inputs *619*

Add Check Box Item *621*
Creating and Using Radio Button Groups *623*
Using Sliders for Inputting Values *626*
EXAMPLE 15.4 Creating a Quiz Using Equations in Diagram Input fields *628*
Entering a Filename into a String Variable *632*
EXAMPLE 15.5 Converting Units for Inputs and Outputs in the Diagram Window *632*

16 DIAGRAM WINDOW ANIMATION **638**
16.1 Controlling the Attributes of Diagram Window Objects *638*
Attributes of Rectangle and Ellipse/Circle Objects *638*
The GetAttribute function *641*
Attributes of Line Objects *642*
Attributes of Polyline and Polygon Objects *642*
Attributes of Text Objects *643*
Attributes of Diagram Window Plot Objects *644*
EXAMPLE 16.1 Selecting a Plot to Display in the Diagram Window *645*
Attributes of Button, Radio Group and Check box Objects *648*
Showing and Hiding Diagram Window Objects *648*
16.2 Setting Attributes within the Diagram Window *649*
Setting Attributes with Text Drop Down Lists *649*
Setting Attributes with a Check Box Object *653*
Setting Attributes with a Radio Button Group *654*
Controlling the Equations Window with Check Boxes and Radio Button Groups *655*
Using Attributes to Select a Picture in the Diagram Window *657*
EXAMPLE 16.2 Selecting a Heat Exchanger Core in the Diagram Window *658*
Improving the Response Time of Diagram Window Controls *660*
16.3 Exchanging Inputs and Outputs *660*
16.4 Using Parametric and Integral Tables for Animations *665*
EXAMPLE 16.3: Setting up an Animation Using the Parametric Table *666*
Saving Animation Files for the Generation of a Movie *667*
EXAMPLE 16.4: Setting up an Animation Using the Integral Table *668*

17 DISTRIBUTABLE PROGRAMS **674**
17.1 Distributable Program Setup and Startup Dialog *674*
17.2 Setting Distributable File Information *679*
17.3 Completing the Distributable File *683*

18 MACROS **685**
18.1 The EES Macro Window *685*
The Macro Window Toolbar *687*
Creating a Macro in the Macro Window *689*
EXAMPLE 18.1 Repeated Solving Using a Macro *691*
The EES Log File *693*
18.2 EES Macro Commands *695*
18.3 Interacting with External Programs using Macros *735*
EXAMPLE 18.2 Interacting with Microsoft Excel *735*
EXAMPLE 18.3 Interacting with MATLAB *738*
18.4 Using a Macro for Serial Port Communications *742*

Basic Communication with the Arduino UNO *742*
Logging Data to a Lookup Table and Plotting *744*
18.5 Executing EES Macros from External Programs *750*
Executing an EES Macro File from the Windows Run dialog *750*
Executing an EES Macro File from MATLAB *751*
Executing an EES Macro File from Excel *753*
Executing an EES Macro File from LabView *756*
18.6 Interacting with EES using Dynamic Data Exchange *760*
Dynamic Data Exchange with Excel *760*
Dynamic Data Exchange with MATLAB *762*
18.7 Alternative Ways to Run a Macro from EES *763*
The CallMacro Statement *763*
The $RunMacroAfter and $RunMacroBefore Directives *766*
18.8 Useful Macro Examples *767*
EXAMPLE 18.4 Set up a Lookup Table for a Factorial Design *767*
EXAMPLE 18.5 Select all Files of a Particular Type from a Directory *768*
EXAMPLE 18.6 Create a Property Plot and Superimpose State Data *769*
EXAMPLE 18.7 Open all .EES files in a directory and copy the equations to .docx files . *770*
EXAMPLE 18.8 Repeatedly call EES from MATLAB *771*
EXAMPLE 18.9 Use of the Input dialog with Radiation View Factors *773*
EXAMPLE 18.10 Automatically Creating 3-D Polygon Plots *776*

19 EXTERNAL FUNCTIONS AND PROCEDURES 778
19.1 EES External Functions (.dlf files) *778*
External Functions written in DELPHI Pascal *779*
External Functions written in C++ *783*
19.2 EES External Procedures - Type 1 (.dlp files) *785*
Type 1 External Procedures written in Pascal *786*
Type 2 External Procedures written in C++ *790*
19.3 EES External Procedures - Type 2 (.fdl files) *793*
Type 2 (.fdl procedure) with the MinGW Open Source GCC Fortran Compiler *797*
19.4 Multiple Files in a Single Library File (.dll or .dll64) *801*
19.5 Managing External Library Files *804*
Loading External Library Files *804*
Providing Help for External Library Files *804*
19.6 External Python Procedures *805*

20 THE COMPONENT LIBRARY 812
20.1 Overview of the Component Library *812*
20.2 Examples using the Component Library *815*
EXAMPLE 20.1 Heat Pump Design *815*
EXAMPLE 20.2 Heat Pump Simulation *819*
EXAMPLE 20.3 Analysis of a Pump and Piping System *823*
EXAMPLE 20.4 Adiabatic Combustion with Chemical Equilibrium *828*
20.3 Summary of the Component Library *831*

21 THE REPORT WINDOW 835
21.1 Creating the Report *835*

EXAMPLE 21.1 Analysys of a Refrigeration Cycle *835*
21.2 Inserting EES Variables *840*
21.3 Inserting EES Plots *841*
Appendix A: Built-In Mathematical Functions **843**
Appendix B: Built-In String Functions **853**
Appendix C: Directives **856**
Appendix D: Macro Commands **859**
Appendix E: Overriding Default Behavior **872**
Appendix F: EES File Types **875**
Appendix G: The 64-bit Professional license **880**
Appendix H: Startup Options **881**

1 INTRODUCTION TO EES

EES (pronounced 'ease') is an acronym for Engineering Equation Solver. The basic function provided by EES is the numerical solution of linear and non-linear algebraic and differential equations. It is important to recognize that EES is an *equation*-solver. EES utilizes equations rather than the assignments that are used in a formal programming language. EES solves systems of equations (i.e., relationships between variables) automatically, which frees the user from having to develop their own iterative technique for solving a set of non-linear equations. There are many additional features associated with EES; for example unit checking, optimization, numerical integration, high quality built-in thermodynamic and transport property functions, plotting and uncertainty analysis. These features make EES a powerful tool for developing mathematical models of many types of engineering systems. This chapter discusses the basic features of EES.

1.1 Acquiring and Installing EES

This section provides information about obtaining and installing the EES software. If EES is already installed on your computer, you can skip this section.

Acquiring EES

EES is licensed and distributed solely by F-Chart Software LLC. A license for EES can be purchased directly from the company website, http://fchartsoftware.com. Single and multi-user licenses are available. In addition, the company offers academic licenses for educational institutions involved in formal classroom instruction and academic research leading to a degree. Details and costs for the different license options can be found at http://fchartsoftware.com/ees/. If you do not have access to EES, you can download a demonstration copy of the program from http://fchartsoftware.com/ees/demo.php. The demonstration copy will allow you to examine most of the features and examples that are provided in this book.

Annual, Commercial and Professional EES licenses are available. The Annual and Commercial licenses provides all of the basic functionality of EES, includes the ability to solve thousands of simultaneous equations and access high accuracy property information. All EES licenses can be used to conduct parametric studies, do unit conversion and check unit consistency, provide publication quality plots, perform uncertainty analyses, and many other things.

The Professional license provides many additional features, as detailed at http://fchartsoftware.com/ees/pro-comm-versions.php. The most important of these features are the ability to solve up to 12,000 simultaneous equations and the ability to create special purpose executable EES programs that can be freely distributed to others. This book will provide information for both all EES license types. Features that are only available in the Professional license will be designated using a side bar like the one that is adjacent to this paragraph.

The Annual, Commercial and Professional licenses are 32-bit Windows executable programs that are designed to run on all modern 32 and 64-bit Windows operating systems such as Windows 10 and 11. There is also a 64-bit Professional license as described in Appendix G.

Installing EES

Installing EES is relatively straightforward, but there are a few installation options that merit discussion. To install the program, you will need both the Setup_EES.exe and EES.dft files. These files should have been provided to you when you purchased the program or by a system administrator if you obtained the program from your company or institution. There are a few institutions where the EES.dft file has already been bundled with the setup file. Execute the Setup_EES.exe program and click the Next button from the Welcome screen. Read the license agreement and, if you agree, click the 'I Agree' button and then click the Next button. Continue to click the Next button until you see the Select Destination Directory dialog, as shown in Figure 1-1.

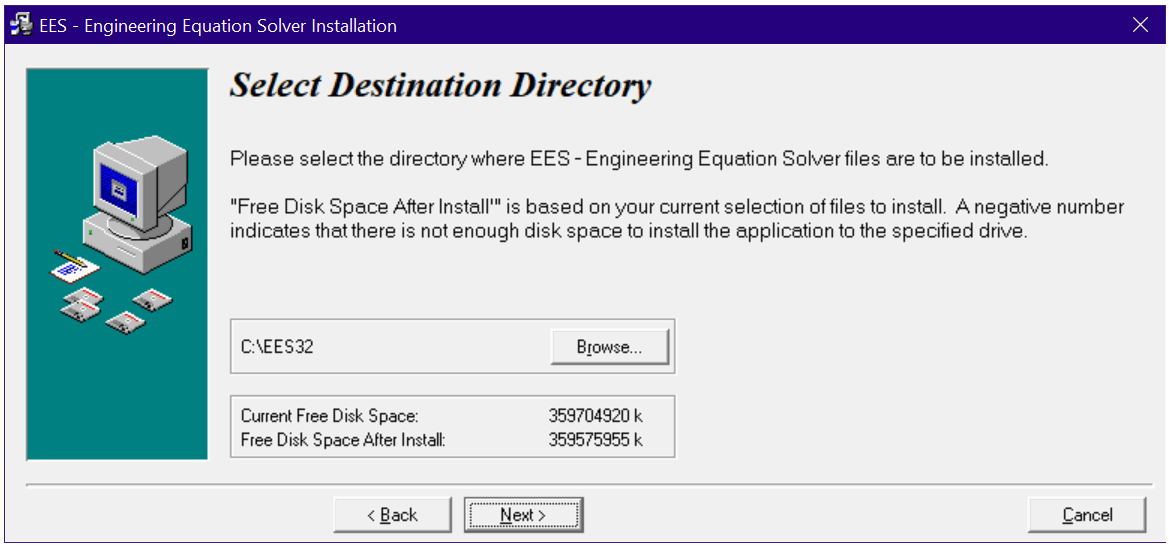


Figure 1-1: Select Destination Directory dialog.

The default directory that EES will be installed into is named EES32. EES is a 32-bit program, but it will operate on all modern 32-bit and 64-bit Windows operating systems. (See Appendix G for information on the 64-bit Professional license.) You can change the directory name, if you wish, by clicking the Browse button. Depending on the particular installation program, you will need to copy the EES.dft file to the directory that you choose for your EES installation. Click the Next button after you have selected the directory.

The next window, shown in Figure 1-2, provides several installation options. If selected, the Install Heat Transfer Library option will install an extensive set of library functions that facilitate heat transfer calculations. These functions have been developed to accompany the text books *Heat Transfer*, by G.F. Nellis and S.A. Klein (2009) and *Introduction to Engineering Heat Transfer*, by G.F. Nellis and S.A. Klein (2020). Selection of this option is recommended unless you are sure that you will never be doing any heat transfer calculations. A discussion of these library functions is provided in Chapter 12.

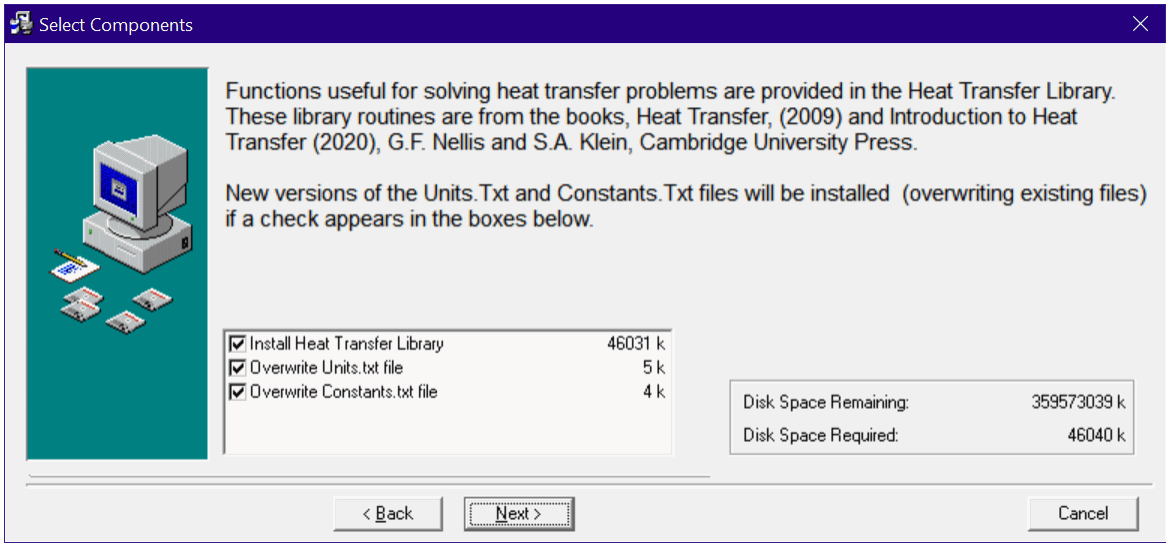


Figure 1-2: Select Components dialog.

EES allows you to save custom Units and Constants files as discussed in Sections 1.5 and 1.9. If you have customized these files then you may want to unselect the Overwrite Units files and Overwrite Constants.txt file in Figure 1-2 so that these files are not replaced by the files in the installation program.

Clicking the Next button will install the program. If EES fails to start, the most likely problem is that the EES.dft file is not contained in the directory that the program was installed in. In this case, copy the EES.dft file into the install directory and then restart EES by double-clicking on the EES.exe icon or by entering C:\EES32\EES.exe in the Windows run dialog.

Silent Installation

It is possible to install EES silently, so that it installs the program with the default options and without displaying any confirmation windows. To do a silent install, enter:

```
C:\myDir\Setup_EES.exe /s
```

into the Windows Run dialog after replacing myDir with the directory name where the Setup_EES.exe program is located. By default, EES will place the program in the C:\EES32 directory (for the 32-bit version) or in the C:\EES64 directory (for the 64-bit license).

Online Help

Extensive online help is provided with the EES program installation. The index page for this help file is shown in Figure 1-3. The online help can be made visible by selecting Help Index from the Help menu or by pressing the F1 key. The help file is heavily hyperlinked. Clicking on a word shown in blue font will move the display focus to that subject. There is also a Contents box and an Index in the left pane of the Help window, as seen in Figure 1-3. Clicking on any item in the Contents box will link directly to the help page for that item. The Index page allows you to search for a specified keyword.

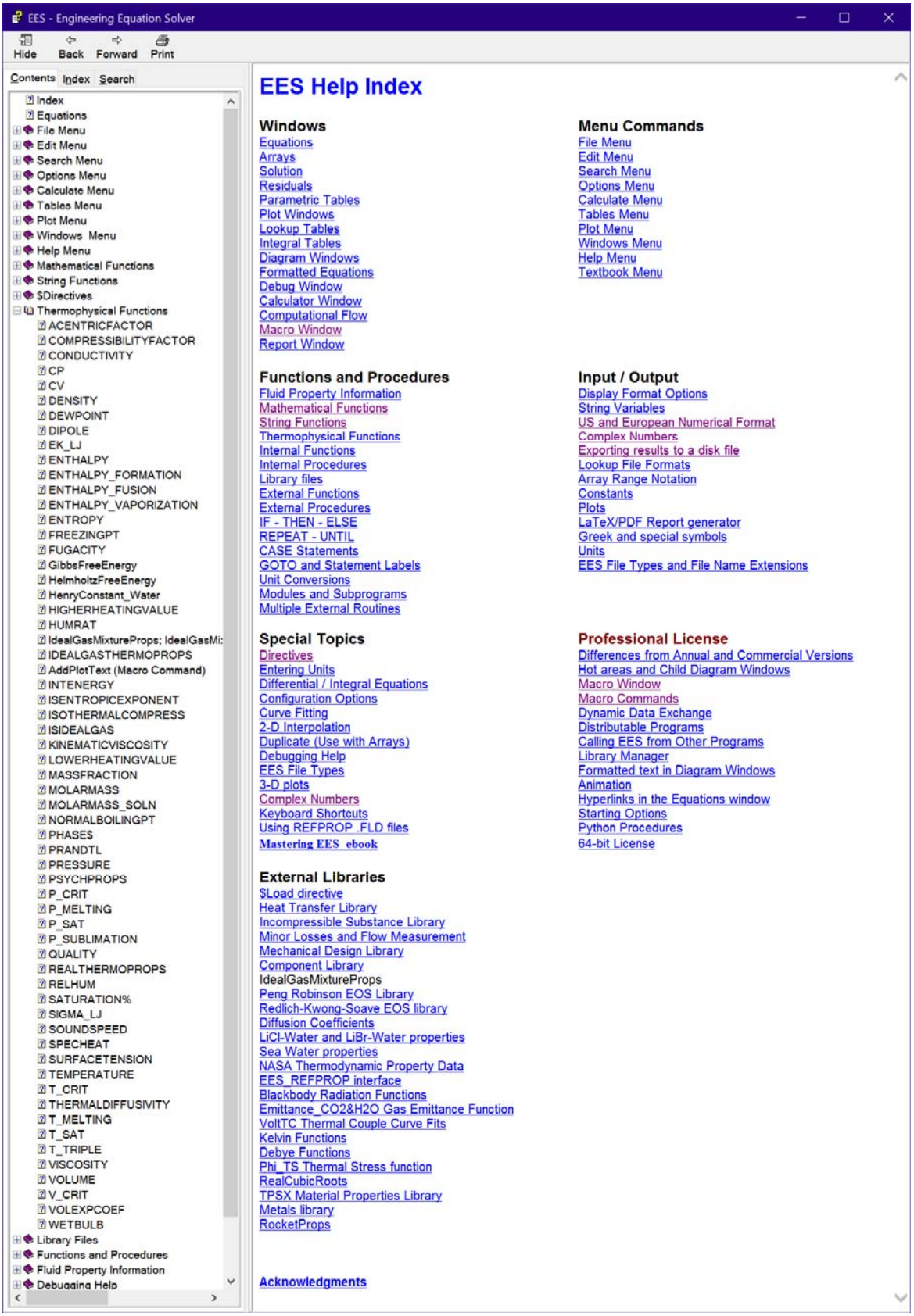


Figure 1-3: Online help provided by pressing F1

Online help information for built-in and library functions can accessed by selecting the function name in the Equations window (e.g., by double-clicking) followed by a right-click to bring up a popup menu. Help for the selected function will be one of the options.

EES help can be access from the Help Index menu command in the Help menu. This command opens file EES.chm. The .chm extension stands for compiled HTML file and it is the preferred method for distributing online help for Windows programs. However, some networks do not allow .chm files to be transferred from a server to a client computer due to possible security concerns. If you find that the online help is not working, you use access the help file on page http://fchartsoftware.com/ees/eeshelp/ees_help_index.htm using the Help Index (web) menu item in the Help menu. Note that the online help file has a Search tab that can be used to search for keywords.

Several other methods of obtaining help are available within EES; these can all be accessed from the Help menu. The EES manual is a .pdf file that is provided with EES and is also available on the F-Chart web site on page http://fchartsoftware.com/assets/downloads/ees_manual.pdf. The manual can be accessed with the EES manual command from the Help menu provided that the manual is placed in the same folder as the EES application.

The Mastering EES eBook that you are currently reading can also be accessed from the Help menu by selecting the Mastering EES menu command. Accessing Mastering EES in this manner requires that the Mastering-EES.pdf file be placed in the same folder as the EES application. EES will attempt to automatically enter the password for the .pdf file when this menu command is selected. Note that there is a Check for New Mastering EES command in the Help menu. Selecting this command will check to see if a newer version of the Mastering EES ebook is available and, if so, provide the option to download it from the website.

A number of short YouTube tutorials have been prepared that describe particular features of EES such as plotting, using Parametric tables, and checking units. These tutorials can be accessed from the YouTube Tutorials command in the Help menu.

The License Information command in the Help menu provides specific information about your EES license. If you are using an Academic license, this command will indicate the IP address that you are using and provide a list of registered IP addresses for this license. The date that Instant Update and Technical Service expires is displayed for non-Academic Commercial and Professional licenses.

1.2 Entering and Solving Equations

A mathematical model of an engineering system is based on a system of equations that result from the application of appropriate theory and simplifications. These equations describe mass, energy, and entropy balances, rate relations, properties, etc. Modern computer tools, such as EES, facilitate the solution of the large set of coupled equations that result from the analysis of a typical engineering system. When you open EES for the first time you will encounter the Equations Window where the equations that are to be solved are entered. EES allows the user to enter *equations* rather than *assignments*, as are required by most formal programming languages. This is an important distinction. In an assignment statement, the value of each variable on the right side of an expression must have been previously determined. An equation is simply a relationship between variables. Assignments are explicit and can be solved sequentially. A set of equations may be implicit and nonlinear and therefore must be solved simultaneously and iteratively.

Entering Equations

Consider the following set of equations:

$$x + y = 3 \tag{1-1}$$

$$y = z - 4 \tag{1-2}$$

$$z = x^2 - 3 \tag{1-3}$$

Equations (1-1) through (1-3) are three non-linear equations in the three unknowns *x*, *y*, and *z*. However, they are not directly solvable using most formal programming languages (e.g., MATLAB or FORTRAN) because they are equations rather than assignments. In order to solve this system of equations using a formal programming language, it would be necessary to either employ an iterative solution technique (e.g., successive substitution) or carry out sufficient algebra to convert the equations into assignments. Let's take the latter approach here. Substituting Eq. (1-3) into Eq. (1-2) provides:

$$y = x^2 - 3 - 4 \tag{1-4}$$

Substituting Eq. (1-4) into Eq. (1-1) provides:

$$x + x^2 - 7 = 3 \tag{1-5}$$

which can be rearranged:

$$x^2 + x - 10 = 0 \tag{1-6}$$

Equation (1-6) is a quadratic equation that can be solved using the quadratic formula:

$$x = \frac{-1 \pm \sqrt{1^2 - 4(-10)}}{2} \tag{1-7}$$

Equation (1-7) has two solutions:

$$x = -\frac{1}{2} + \frac{\sqrt{41}}{2} = 2.702 \tag{1-8}$$

and

$$x = -\frac{1}{2} - \frac{\sqrt{41}}{2} = -3.702 \tag{1-9}$$

Substituting Eqs. (1-8) and (1-9) into Eqs. (1-3) and (1-4) provides the two solutions to the original set of equations: x = 2.702 or -3.702, y = 0.2984 or 6.702, and z = 4.298 or 10.70.

At its most fundamental level, EES is an equation solver that solves sets of nonlinear equations directly. For example, the three equations (1-1) through (1-3) can be entered directly into the Equations Window, as shown in Figure 1-4.

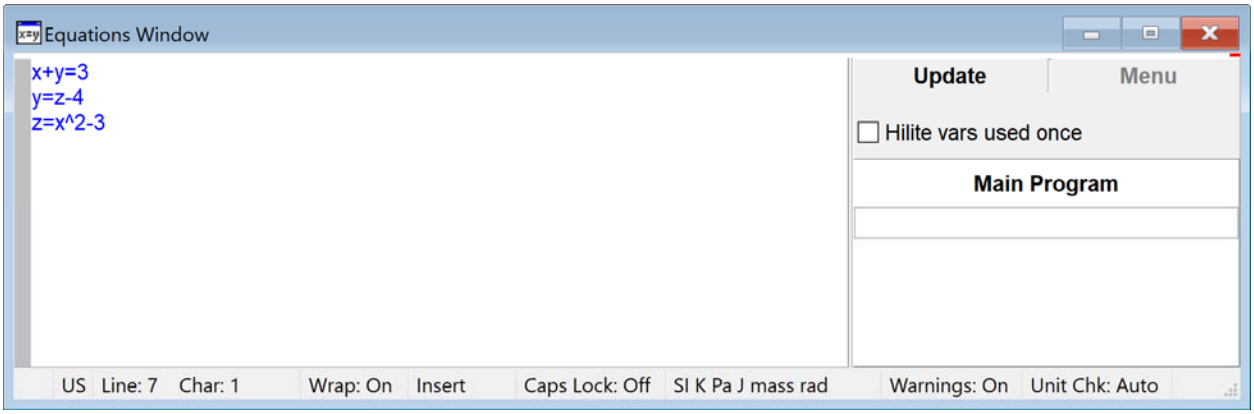


Figure 1-4: Equations Window with Eqs. (1-1) through (1-3) entered.

In this book, text that is entered in the Equations Window will be shaded, as shown below.

```
x+y=3
y=z-4
z=x^2-3
```

The Variable Information palette appears to the right of the Equations window. It shows the variables that have been defined and their units (covered in Section 1.5). Click the Update button to update the palette so that it reflects the current contents of the Equations window. Slide the bar that separates the Equations window and the palette left or right to adjust the size of the palette. Variable information for the Main Program is shown by default. Click the Menu button

to show variable information for Procedures, Functions, Subprograms and Modules (discussed in Chapters 3 and 10).

Figure 1-5: Equations Window after clicking the Update button in the Variable Information palette

The Solution Window

Select Solve from the Calculate menu (or use the shortcut key F2) in order to initiate the iterative process that EES uses internally to solve the system of equations. EES will re-order and block the equations in a logical manner and then, starting from a guessed solution, it will iteratively search for an actual solution to the equations. The result should be the dialog shown in Figure 1-6, which shows that the calculations were successfully completed and provides some of the details of the process (which will be discussed more completely in Chapter 5).

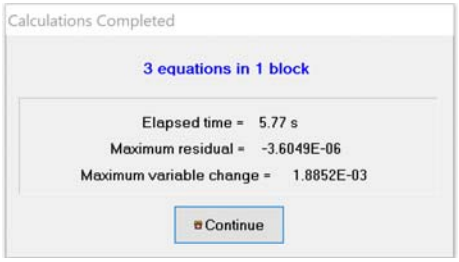


Figure 1-6: Dialog indicating that the calculations are complete.

Select Continue to proceed to the Solution Window, shown in Figure 1-7.

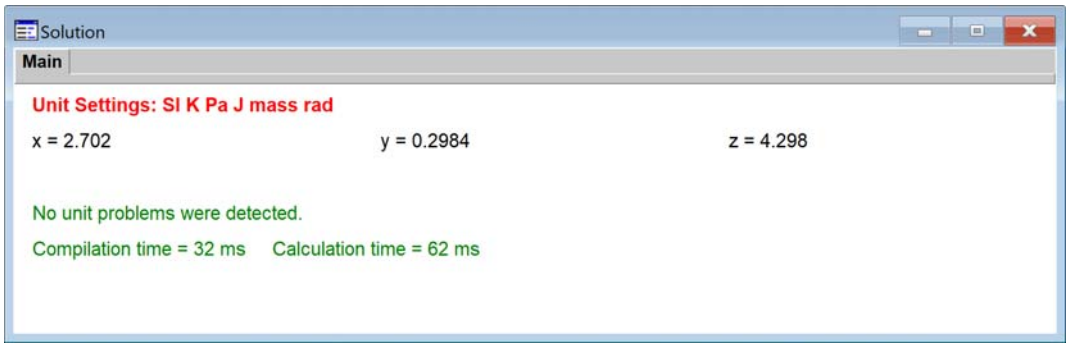


Figure 1-7: Solution Window.

EES has identified one of the two solutions to the equation set. The other solution can be found by changing the guess values for each of the variables, as explained below. The process of solving the equations was done internally; the user is not required to carry out any algebra or do any iteration.

Decimal vs Comma Separator

The solution is shown in Figure 1-7 using the decimal point as the decimal separator. However, EES operates just as well using a European numerical formatting style. Close EES and open the Regional and Language Options dialog in the Windows Control Panel. Under the Regional Options tab, select a European language (e.g., French). Alternatively, enter $Keyboard EU into the Equations window. Open EES and solve the problem again to obtain the Solutions Window shown in Figure 1-8; note that the decimal separator is now a comma rather than a decimal point.

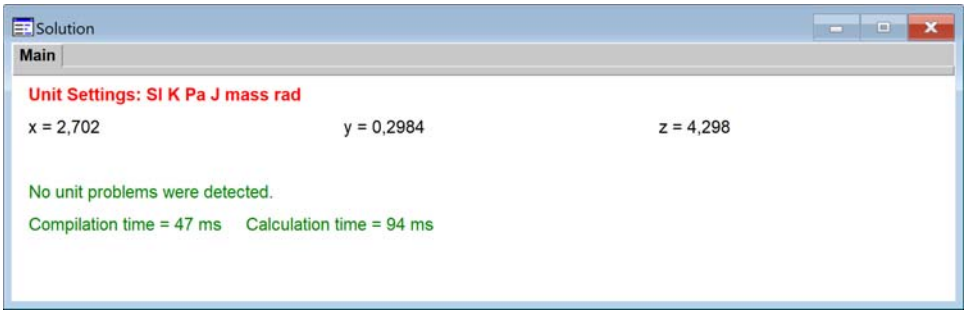


Figure 1-8: Solutions Window using the European format.

The Variable Palette and Information dialog

The equations shown in Figure 1-4 are non-linear and thus they have more than one solution. Why did EES identify one of the solutions (x = 2.702, y = 0.2984, and z = 4.298) as opposed to the other one (x = -3.702, y = 6.702, and z = 10.70)? The answer is that the result depends on the guess values and limits that have been set for each variable. EES uses a variation of Newton's method to solve systems of equations, as will be discussed in Chapter 5. This technique begins with an assumed (or guessed) value of each variable and then iteratively adjusts these values until the equations are satisfied. In situations where multiple solutions exist, EES will likely converge to the solution that is closest to the guessed values that are used to start the process.

There are several ways to view or change variable information. The simplest is to simply click the mouse button on the variable name in the Variable Information palette that appears to the right of the Equations Window. This action will bring up the information associated with this one variable, as shown in Figure 1-9. The guess value, limits, units, alternate units and other information can be viewed or changed for the selected variable in this dialog.

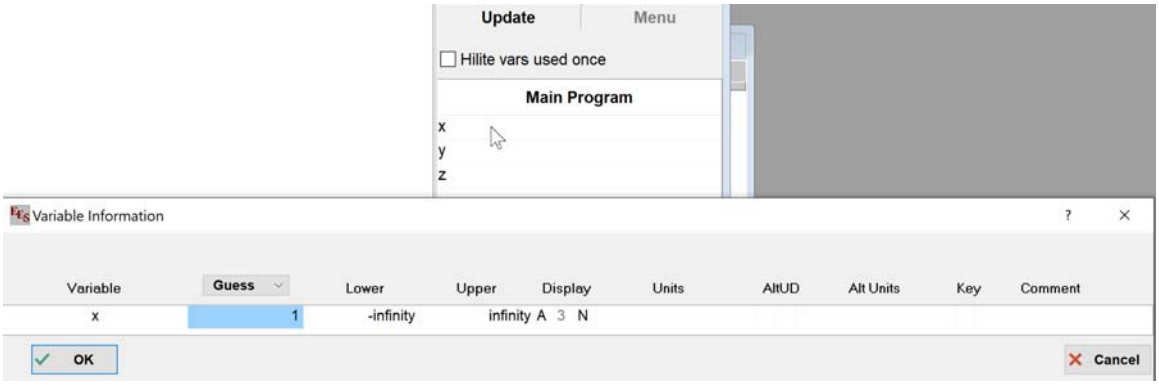


Figure 1-9: Dialog that appears after clicking on a variable in the Variable Information palette

Selecting the Hilite vars used once control button in the Variable Palette will highlight variables that only appear one time in the selected program unit with a buff yellow background. For example, add the equation, $xy = x + y$ to the equations shown in Figure 1-5. Click the Hilite vars used once and the display will appear as shown in Figure 1-10. Variables that are used only once are sometimes the result of a typing error and highlighting them may help identify such problems.

Figure 1-10: Variable Palette showing a variable used only once.

Note that the Variable Information dialog can be used as a smart Find command. Right-clicking on a variable in the palette will locate its first occurrence in the Equations window. Right-clicking on the variable a second time will find the next occurrence of the variables.

In some cases, it is more convenient to display the variable information for all variables in the program unit. This can be accomplished by selecting Variable Info from the Options menu (or by pressing F9 or by right-clicking the Update button in the Variable Information palette.) The full Variable Information dialog will appear, as shown in Figure 1-11. Use the mouse wheel to scroll through long lists of variables.

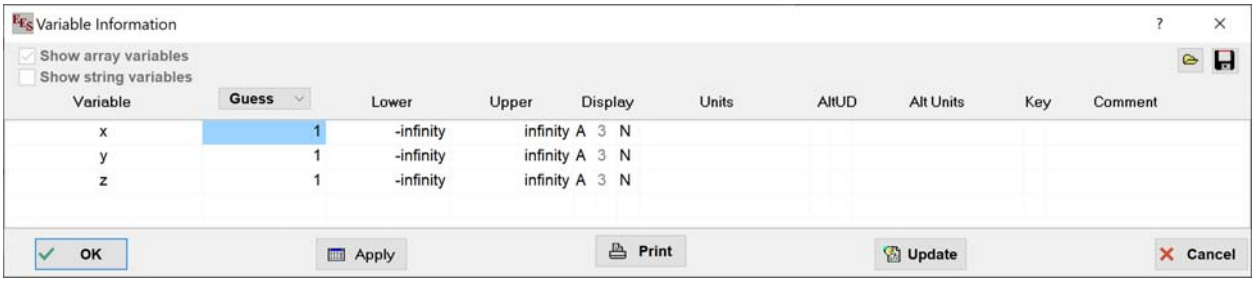


Figure 1-11: Variable Information Window.

Guess Values

There is a row in the Variable Information window corresponding to each of the three variables that make up the problem. The columns allow the user to change various characteristics of these variables. The first column corresponds to the guess values for each variable; these are the values used to start the iterative solution process. By default, the guess value for each variable is 1. Note that $x = 1$, $y = 1$, and $z = 1$ is not the correct solution, but it is closer to the solution $x = 2.702$, $y = 0.2984$, and $z = 4.298$ than it is to the alternative solution $x = -3.702$, $y = 6.702$, and $z = 10.7$. In order to get EES to converge to the other solution, it is necessary to change the guess values. For example, change the guess value of x to something closer to $x = -3.702$ (e.g., -5), select OK and press F2 (the shortcut for the Solve command). The solution identified by EES is shown in Figure 1-12. Guess values can also be provided using other EES variables or equations, as described in Section 5.3.

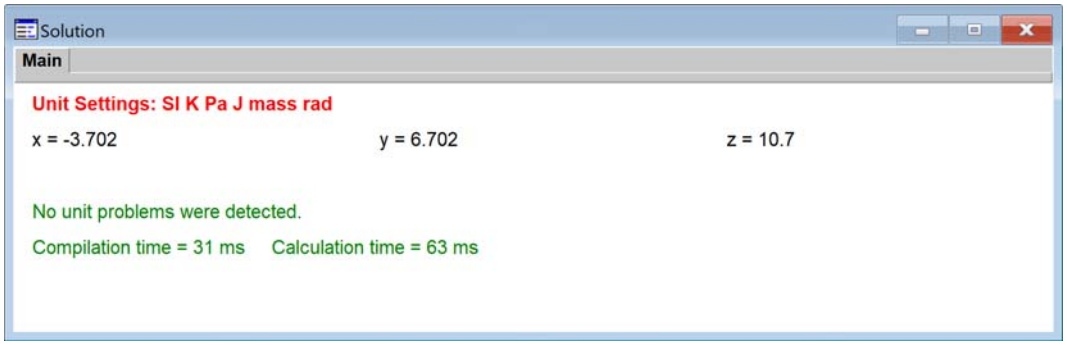


Figure 1-12: Solution Window with modified guess value.

Limits

Lower and upper limits for each variable can also be set in the Variable Information Window. For example, in order to identify only a solution for which the value of x is positive, the Variable Information Window could be set up as shown in Figure 1-13.

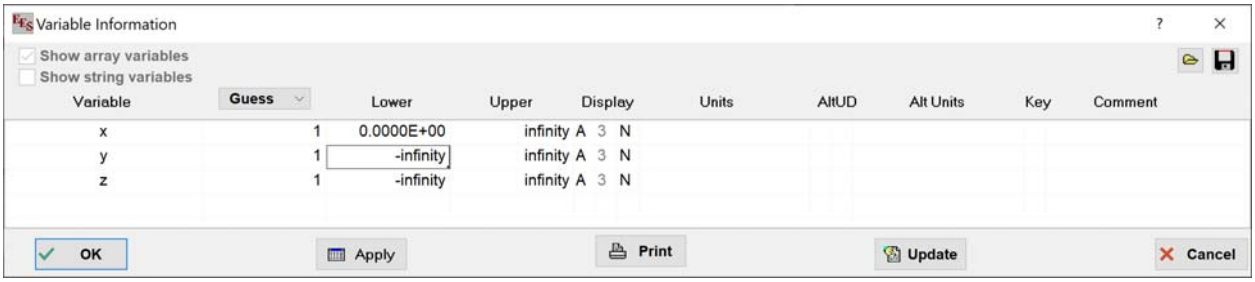


Figure 1-13: Variable Information Window with limits set for the variable x.

Display Format

All EES numerical values use double precision that internally provides 20 (for the 32-bit version) or 16 (for the 64-bit license) significant figures of precision. The display format can be separately specified for each variable. The available display formats are shown in Table 1-1.

Table 1-1: Display Formats for EES Variables

A	Auto	EES will internally choose a display format and number of significant figures
F	Fixed Decimal	Specified number of digits to the right of the decimal separator
E	Exponential	One significant figure is shown to the left of the decimal separator and an exponent following letter 'E' at the end of the display to indicate the powers of 10 that the value should be multiplied by.
G	Engineering	Similar to Exponential format, but the exponent is always a factor of 3.
N	Significant Figs.	Display numerical values with a specified number of significant figures
S	String	String format is used for string variables or for text in a Lookup table
T	Time	Time format allows time to be displayed in various ways
D	Date	Date format is appropriate for values that are to be interpreted as dates

In the Variable Information dialog, the display format is controlled by the three columns under the word Display. The first column controls the format using the letter in the first column of Table 1-1. The number of significant figures or decimal digits is controlled by the entry in the second column. This can be an integer between 0 and 9 or the letter 'M', which indicates the maximum possible number of significant figures. The third column provides additional formatting options such as underline, box, italic, etc.

For example, the A in the first of the three columns under the word Display in Figure 1-13 indicates Auto format. This is the default format. In this case, EES selects the number of significant figures and therefore the value in the second column cannot be changed. Click on the A in the first display column in the second row of the table and then select F from the drop down menu to change the display associated with the variable y from Auto to Fixed Decimal. Select 9 in the second display column and Boxed from the third column so that the dialog appears as shown in Figure 1-14.

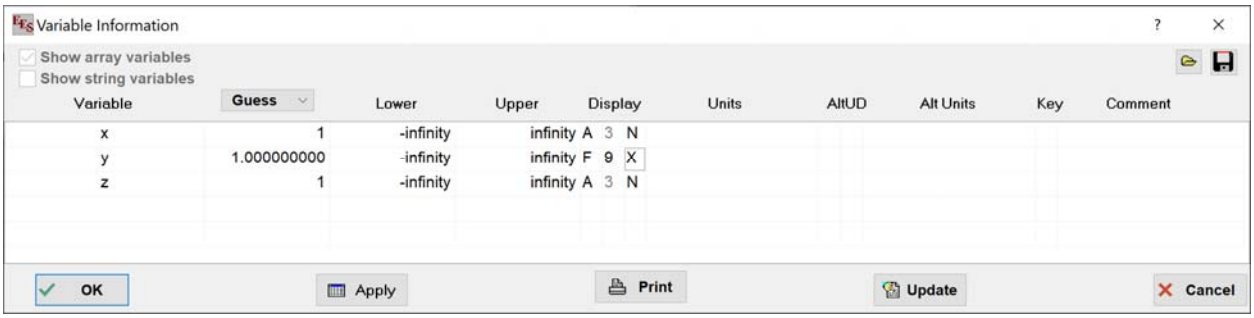


Figure 1-14: Variable Information dialog with format specified for variable y.

Click OK and then solve the equations, The Solutions window will appear as shown in Figure 1-15.

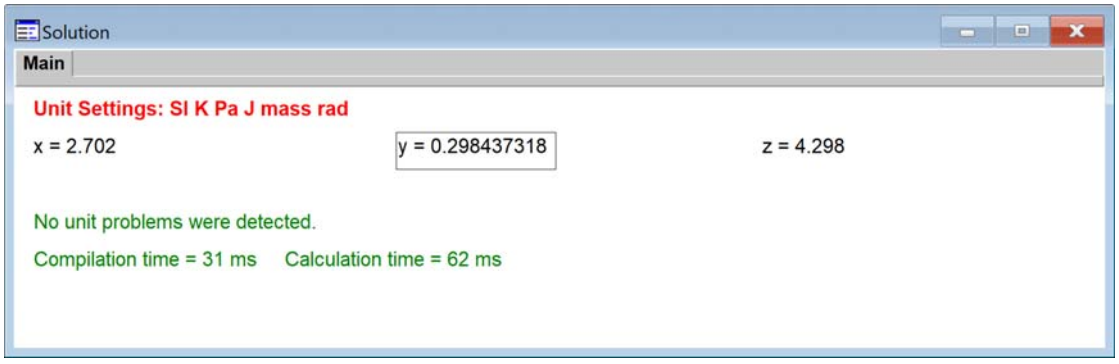


Figure 1-15: Solutions Window with an altered format for the variable y.

The display format can also be set directly in the Solution Window by right-clicking on a variable. This will bring up the Specify Format and Units dialog where the format for the selected variable(s) can be directly entered.

The units column in the Variable Information dialog shows the units that have been entered for the variables. Units can be entered directly into the column. EES also allows an alternate set of units to be entered and displayed. The alternate units are entered in the column below Alt. Units. The format for the values in the alternate unit system is, by default, Auto, but it can be reset using the format codes in the two columns under the AltUD header. The display formats for alternate unit results are also shown in Table 1.1.

Variables can be designated to be Key variables by clicking in the Key column. In this case, they are displayed in the Key Variables Solution window, along with the optional comment that can be entered in the last column.

Rules for Entering Equations

Let's return to the Equations Window. Notice that the equations can be entered in any order and rearranged algebraically in any way. For example, we can switch the order of the first two equations and switch the right and left sides of the third equation and EES will identify the same solution.

```
y=z-4
x+y=3
x^2-3=z
```

The Equations Window functions a lot like a word processor. You can use the cut (Ctrl+X), copy (Ctrl+C), paste (Ctrl+V), Find (Ctrl+F), Replace (Ctrl+R), and Undo (Ctrl-Z) commands just as you would in Microsoft Word® or other programs that manipulate text. Multiple undo/redo capability is provided for up to 32 changes. Equations are typically entered one per line; however, multiple equations can be entered on the same line provided that they are separated by a semicolon (or a colon if you are using European format):

```
y=z-4; x+y=3; x^2-3=z
```

The mathematical operators and order of operation used in the equations are consistent with those used by most any programming language. For example, the equation:

```
a=2^3+3*4+2
```

will result in $a = 22$. Variable names in EES must start with a letter but they can include any U.S. keyboard character except ( ) * / + - { } or ;. EES is case-insensitive; that is, the symbols x and X are interpreted to be the same variable in the Equations Window. Very long equations can be broken into two lines for ease of reading or improved printing by using the ampersand (&). Our example doesn't include any long equations, but we could still break the first equation into two lines according to:

```
x+y&
=3
y=z-4
z=x^2-3
```

Comments

It is good practice to annotate your EES code in order to clarify the meaning behind the equation set. Typically each equation is followed by a comment that is ignored by the equation solver itself, but is visible to the user. Comments in the Equations Window should be enclosed either in curly braces or quotes. Any line that begins with two forward slashes is also considered a comment. The comments are displayed in blue (by default) in the Equations Window to indicate that they are not part of the equation set that will be solved. Comments that begin with the ! character (referred to as comment type 2) are displayed in red by default. The default colors for comments can be changed in the Equations tab of the Preferences dialog (Options menu).

```
//this is a set of equations
```

```
x+y=3                                   {comment for line 1}
y=z-4                                   "comment for line 2"
z=x^2-3                                 "! this is a type 2 comment"
//this is a comment
```

Occasionally it will be useful to temporarily remove an equation or a group of equations from the Equations Window. Of course, you could just delete the equation(s) and then type them back in if and when they are needed again. However, a better solution is to "comment them out". Highlight the equation(s) to be removed and then right-click to bring up a pop-up menu, as shown in Figure 1-16. Select Comment {} in order to place curly brackets around the entire section of highlighted code, temporarily removing it from the equation set. To restore the equation(s), highlight the commented code, right-click and select Undo Comment {}.

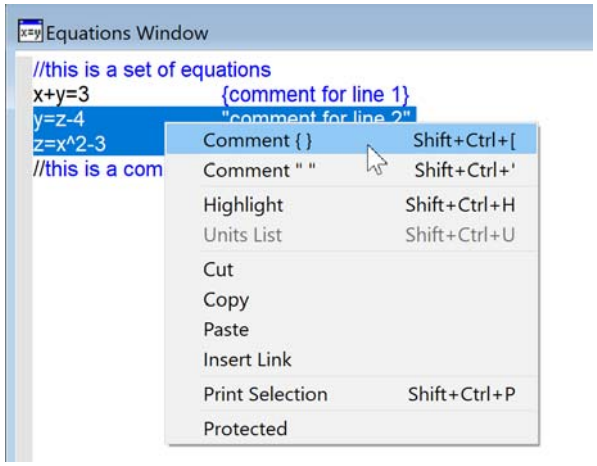


Figure 1-16: Commenting out the last two equations.

Hyperlinks in the Equations Window

Two types of hyperlinks (visible and hidden) can be entered within the comments in the Equations Window. Visible hyperlinks can be entered with all EES licenses. Hidden hyperlinks can only be entered with the Professional license.

Visible Hyperlinks

EES will automatically recognize hyperlinks that begin with http:\\, https:\\, or file: For example, enter:

```
"http:\\fchartsoftware.com"
```

and notice that it becomes a link to that web page. A summary of the recognized visible hyper link keywords is provided in Table 1-2.

Table 1-2: Summary of recognized hyperlinks.

Hyperlink	Description
http:\\fchartsoftware.com	Open the default browser and point it at the web page that follows http:
https:\\fchartsoftware.com	Open the default browser and point it at the web page that follows https:
file:c:\ees32\ees_manual.pdf	Open the file that follows file: and start the appropriate application.
\\EES_Solution	Open the Solution Window and bring it to the front.
\\EES_Format	Open the Formatted Equations Window and bring it to the front.
\\EES_Plot	Open the Plot Window and bring it to the front.
\\EES_PlotN	Open Plot Window N where N is an integer and bring it to the front.
\\EES_Parametric	Open the Parametric Table and bring it to the front
\\EES_Lookup	Open the Lookup Table and bring it to the front.
\\EES_Array	Open the Array Table and bring it to the front.
\\EES_Integral	Open the Integral Table and bring it to the front.
\\EES_Report	Open the Report Window and bring it to the front.
\\EES_Diagram	Open the Diagram Window and bring it to the front.
\\EES_Residual	Open the Residuals Window and bring it to the front.
\\EES_Calculator	Open the Calculator Window and bring it to the front.
\\EES_Solve	Solve the current set of equations.
\\EES_SolveTable	Apply the Solve Table command from the Calculate menu.
\\EES_MinMax	Apply the Min/Max command from the Calculate menu.
\\EES_MinMaxTable	Apply the Min/Max Table command from the Calculate menu.

Hidden Hyperlinks

Hidden hyperlinks do not display the link itself but instead are hidden and activated when some descriptive text is selected. The selected text that, when clicked, will execute the link is displayed with an underline font. Hidden hyperlinks can only be added with the Professional license, but once added they can be used with all EES licenses. To enter a hidden hyperlink, select the text that is to be associated with the link. Then click the right mouse button and select the Insert Link command from the popup menu (Figure 1-17). This will bring up a dialog in which you enter the hyperlink.

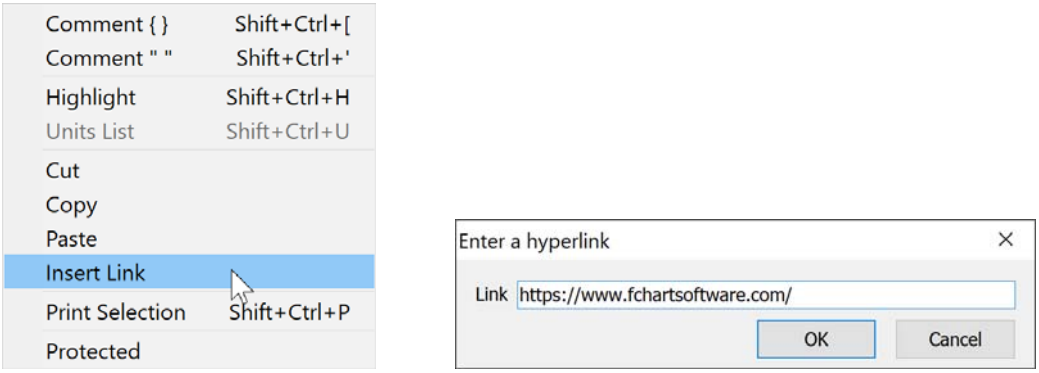


Figure 1-17: Inserting a hyperlink into the Equations Window (Professional license)

Clicking on this text in the Equations window will launch the link. The link will also be available in the Formatted Equations window. Note that syntax highlighting must be activated in order for this hidden link to be active. Syntax highlighting can be toggled on/off by clicking on the control labeled Syntax Highlight located at the bottom right of the Equations window. You can edit the text without launching the link by holding the Ctrl key down when selecting the text.

A hidden hyperlink can be removed by right-clicking the link and selecting Remove Link from the popup menu. The link can also be removed by deleting the text associated with the link, including the spaces before and after the text.

Built-In Math and String Functions

EES provides many built-in functions. The built-in math functions provide basic mathematical, trigonometric, and statistical capability as well as commands to operate on data files and tables. A complete list and description of all built-in math functions is provided in Appendix A. A list and description of built-in string functions is provided in Appendix B.

The built-in math functions can be conveniently accessed by selecting Function Information from the Options menu, which displays the Function Information dialog appearing in Figure 1-18. By default, all of the built-in math and string functions are initially shown in the list on the left side of the dialog. This list of functions can be reduced by selecting a function classification from the list on the right side. An example of the proper use of the selected function is shown in the Ex: edit control at the bottom of the dialog. The text in the Ex: edit control can be edited in the conventional manner so that the arguments of the function match your program. Clicking the Paste button will paste the contents of this control into the Equations Window at the current cursor position.

External Functions and Procedures

A function is a code segment that accepts one or more inputs and returns a single result associated with the function name. A procedure is similar to a function, but it can return one or more results and must be accessed by a Call statement. Functions and procedures that are entered directly in the Equations Window, as described in Chapter 3, are referred to as internal functions and procedures. Internal functions and procedures can also be organized and saved as library files having a .lib (for the 32-bit licenses) or .lib64 (for the 64-bit license) file name extension, as described in Chapter 11. A very powerful feature of EES is that functions and procedures can be written in any compiled language, such as Pascal, C, C++ or FORTRAN. These code segments, referred to as external functions and procedures can be used in exactly the same manner as internal EES functions and procedures that have been saved in library. External functions and procedures can be coded so that the units of the input and output variables will be checked by EES, in the same manner as for internal functions and procedures. Warning and error messages can be returned from the external routines and displayed by EES. External functions and procedures will be automatically loaded when EES starts if they are placed in the Userlib directory. Alternatively, they can be manually loaded using the Load Library command from File menu or the $Include directive. Help files can be provided for documenting external functions and procedures in the same manner for internal functions and procedures in library files. External functions and procedures offer some significant advantages. Because they are written in a compiled language they will usually execute much faster than their internal counterparts. Also, the equations that are employed in external functions or procedures are not visible to the user. External functions and procedures are written as dynamic link library routines in the Windows operating system, as explained in Chapter 19.

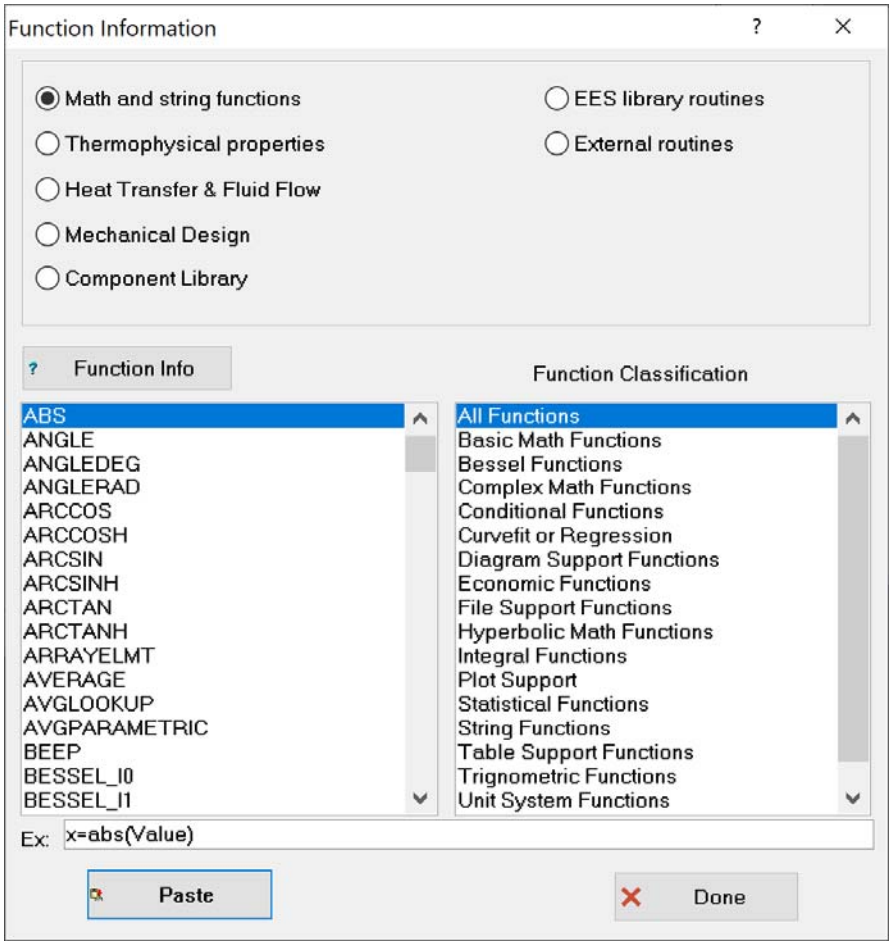


Figure 1-18: Function Information dialog showing math and string functions.

String Variables

EES provides both numerical and string variables. A string variable is identified with a variable name that ends with the $ character. The variable name must begin with a letter and consist of 30 or fewer characters, including the $ character. String variables can be set to string constants, which are strings that are enclosed with single quote marks.

```
A$ = 'carbon dioxide'
```

String variables can be used in EES equations anywhere in which character information is provided. For example the name of a fluid provided to a thermophysical property function may be a string variable, as shown in Section 4.3. A string variable may be used to supply the units of other variables, as shown in Section 1.5. String variables may be passed as arguments to internal functions, procedures, modules, and subprograms or to external functions and procedures as described in Chapters 3 and 10. String variables are very useful in the Diagram window, as shown in Chapter 15. String functions are provided to manipulate string variables and to convert them to or from numerical values. A complete list of string functions is provided in Appendix B. String functions can be accessed from the Function Information dialog, as shown in Figure 1-18.

The $TabStops Directive

Directives are instructions to EES that are usually executed before the equations are compiled and solved. Directive instructions are preceded by the dollar sign ($) and provide a powerful method for controlling an EES solution. Chapter 14 discusses the use of directives in detail and a complete list of directives is provided in Appendix C. However, we will make use of various directives throughout the text. The $TabStops directive provides a mechanism for controlling the location of the tab stops used in the Equations Window. The protocol for using the $TabStops directive is shown below:

```
$TabStops TabStop1  TabStop 2 ... TabStop5  Units
```

where TabStop1, TabStop2, etc. are numerical values corresponding to the desired positions of the tabs and Units indicates the units used for the tabs. The value of Units must either be in (for inch) or cm (for centimeter). The code below sets two tab stops at 1 inch and 2 inch; notice that the locations of the comments have shifted to 1 inch, which is consistent with the first tab stop set in the $TabStops directive.

```
$TabStops 1 2 in

//this is a set of equations

x+y=3          {comment for line 1}
y=z-4          "comment for line 2"
z=x^2-3        "! this is a type 2 comment"
```

Showing Values and Setting Variable Units in the Equations Window

Selecting a variable in the Equations Window will cause a small hint window to appear just below the variable. The hint window will show the value of the variable, if calculations have been completed, and the units assigned to the variable. The variable can be easily selected by double-clicking the left mouse button when the cursor is positioned over the variable. This capability makes it unnecessary to refer to the Solutions window to determine values of EES variables.

The Status Bar and Screen Background Color

The status bar is located at the bottom of the Equations Window, as shown in Figure 1-19.

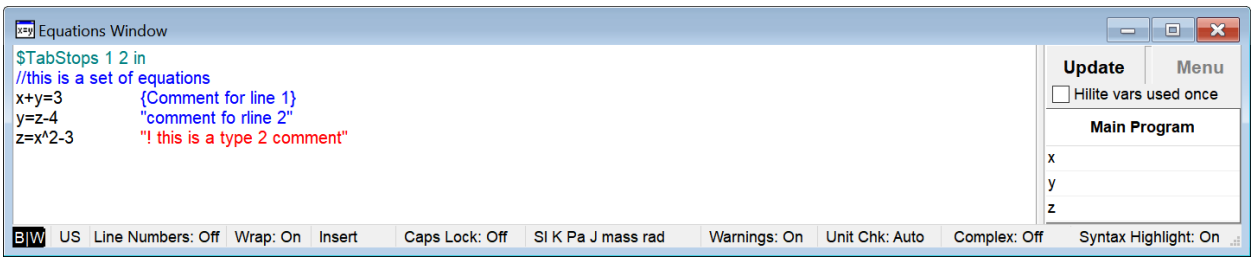


Figure 1-19: Equations window and Status bar.

Clicking the mouse button in the first panel (**B|W**) toggles the screen between a white and black background for the Equations, Formatted Equations, Solutions, and Residuals windows. The default text colors are adjusted to ensure readability (Figure 1-20) and they can be further adjusted in the Display tab of the Preferences dialog

Figure 1-20: Equations window with black background.

The second panel in the status bar will indicate whether a US or EU keyboard is set. If **US** is displayed, then the decimal separator is the decimal point; otherwise it is a comma.

Clicking in the **Line Numbers** panel will toggle the display of line numbers in the left margin of the Equations window, as show in Figure 1-21.

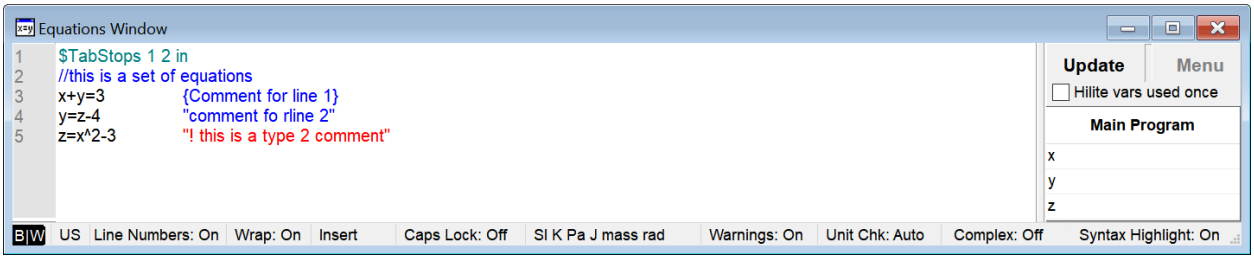


Figure 1-21: Equations window with line numbers displayed.

The various values of the current settings are displayed in subsequent panels on the status bar. These include whether word wrapping is enabled, whether overwrite or insert is on, and whether Caps Lock is on or off. Clicking the mouse button with the cursor positioned in any of these panels allows the settings to be toggled. The current unit system (discussed in Section 1.5) is displayed. The **Warnings:** panel controls whether or not a window showing any warnings messages will be automatically displayed after calculations are completed. These warnings can be manually displayed by selecting the Warnings window from the Windows menu. The next two panels indicate whether unit checking (**Unit Chk:**), and complex algebra (**Complex:**) are currently activated. The last panel (**Syntax Highlight:**) toggles the syntax highlighting setting. See Section 1.9 for information relating to syntax highlighting.

Equations and Display Preferences

Many of the characteristics of the Equations Window can be altered using the Preferences dialog accessed by selecting Preferences from the Options menu. Select the Syntax tab to bring up the dialog shown in Figure 1-22, which allows the format used to display function name, EES keywords, and EES constants in the Equations window to be specified.

If the Display Line numbers check box is checked, line numbers sill be displayed in the left margin of the Equations window. This option can also be turned on and off by clicking in the Line Numbers pane in the status bar at the bottom of the Equations window.

There are a set of dropdown menus that allow the user to set the displayed syntax for function names (i.e., the names of built-in functions in EES like Sin, Tan, etc.), keywords (i.e., reserved words that have special meaning in EES like Call, End, etc.) and variables. Some options include **as typed** in which case the format is not changed relative to how the user typed it in, **lower case** or **upper case** which causes the format of each instance to be changed to match the selected case, 1st letter UC which causes the first letter of each instance to be capitalized, and **as first entered** which causes every instance to be changed to match the case that was used the first time it was typed. The display of functions and keywords can also be modified to be bold, underline, etc.

Up to five tab stops can be set in this dialog in either centimeters or inches. (Click cm or in to toggle the unit setting.) Up to 25 tab stops can also be set with the `$TabStops` directive. See Section 1.9 for more detail on syntax settings.

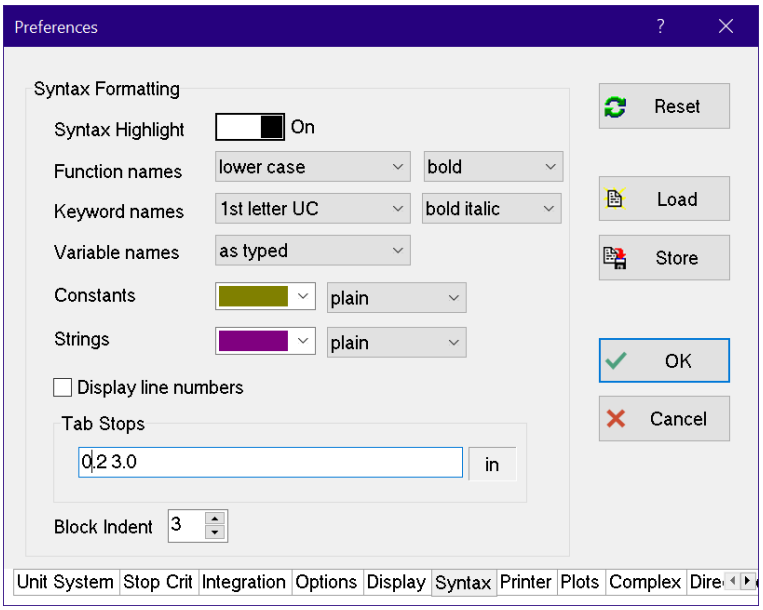


Figure 1-22: Syntax settings tab of the Preferences dialog.

Additional display setting are provided on the Display tab shown in Figure 1-23. The font type and size affects the font in all of the windows, not just the Equations window. Notice that there are two types of comments. Comments that begin with an exclamation point (!) can have a different color than normal comments. For late-night coders, the background color for the Equations, Solutions, Formatted Equations and Residuals windows can be set to black using the **Black background** toggle control at the bottom of the dialog. The background can also be set by clicking in the first pane of the status bar at the bottom of the Equations window.

If the Preferences are stored, using the Store button, then all of these settings will be set as the defaults.

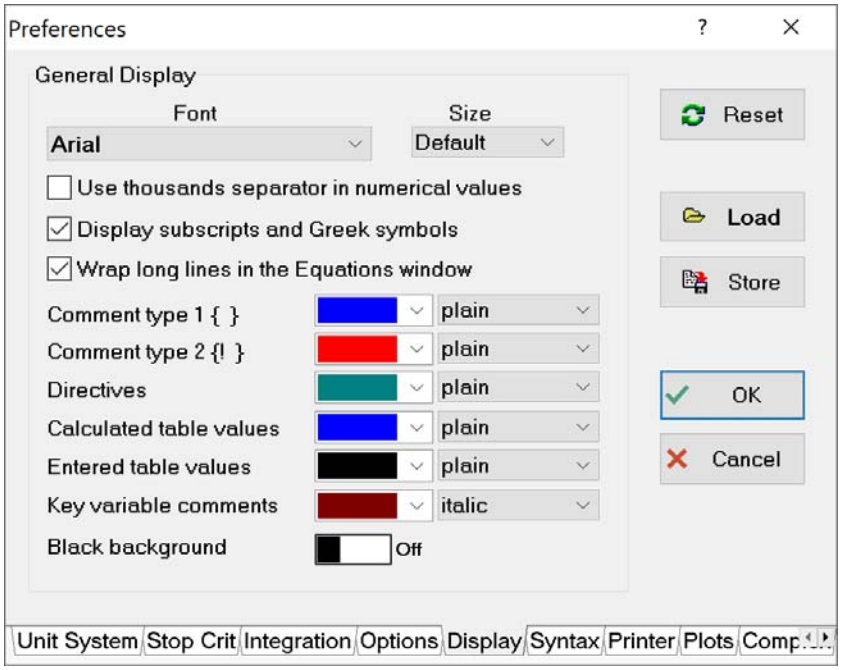


Figure 1-23: Display settings tab in the Preferences dialog.

The settings on the Options tab of the Preferences dialog are shown with their default settings in Figure 1-24. An explanation of each option follows:

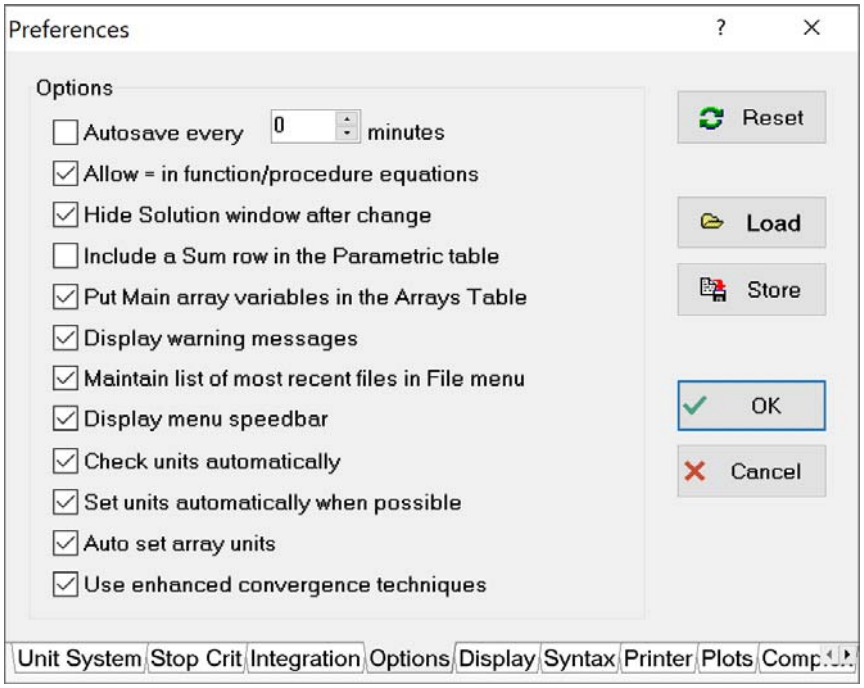


Figure 1-24: Options tab of the Preference dialog.

The **Autosave every XXX minutes** option instructs EES to save the .EES. The information is saved to a temporary .EES file that has the same name as the open EES file, but with a tilde (~) preceding the filename. Note that this autosaved file is deleted when a new file is created or opened or when EES is closed at the end of the work session, so do not depend upon this option for permanent storage. You should always save any file that you intend to use at a later time using the Save or Save As commands from the File menu. However, if EES quits unexpectedly, the last autosaved copy of the backup file should still be available in your directory.

The **Allow = in function/procedure equations** option allows either an = or := to be used in the assignment statements within functions and procedures that are written in EES. The equations used in EES functions and procedures are similar to those used in FORTRAN, C, or Pascal. In assignment statements, the variable on the left of the equation is set to equal to the value associated with the right of the equal sign (or := sign). For example, the assignment X:=X+1 is permitted in a Function or Procedure, but not in a module or in the main body of EES. To accentuate this difference, Functions and Procedures normally use the := sign (as in Pascal) rather than an = sign.

The **Hide Solution window after change** option will cause the Solution, Arrays and Residuals windows to be hidden if a change is made to the Equations window after a calculation is completed.

The **Include a Sum row in the Parametric Table** will display the sum of the values in each column of a Parametric Table in the bottom row of the table. The values in the Sum row are updated after the calculations are completed. The option can also be turned on or off with the $SumRow directive.

The **Put Main array variables in the Arrays Table** option allows array variables (i.e., variables with multiple elements indicated by brackets surrounding a subscript number) to be displayed in the Arrays window, rather than in the Solutions window. The Arrays window will be automatically created after calculations are completed, showing all array variables that are used in the Main program with the array index values in the first column. This option can also be turned on or off with the $Arrays directive. Note that the array variables in Functions, Procedures, Subprograms, and Modules can also be displayed in separate tabs of the Arrays window by using the $Arrays directive placed within each of these code elements.

The **Display warning messages** option controls the default behavior associated with whether or not warning messages are displayed after calculations are completed. Warning messages may be generated internally if, for example, thermodynamic property correlations are being applied beyond their range of applicability, or by the user with the Warning procedure. This option can also be turned on or off with the $Warnings directive or by clicking in the appropriate panel in the status bar of the Equations window.

The **Maintain a list of most recent files in File menu** option will, if selected, provide menu items for up to eight recently opened files. Selecting any of these menu items will open the corresponding file, just as if the Open command were selected.

Display menu speedbar will present a series of buttons on a palette directly below the main menu. These buttons provide quick access to the most frequently applied EES menu commands.

The **Check units automatically** will check the unit consistency of each equation after calculations are completed. This option can also be invoked from the Equations window with the $CheckUnits directive or by clicking the **Unit Chk** panel in the status bar at the bottom of the Equations window.

The **Auto set array units** will set the units (and alternate units) of every variable in an array to the unit of the first array variable. This action occurs when the Arrays table is displayed and when exiting the Variable Information dialog after clicking the OK button. This setting can also be made using the $AutoSetArrayUnits directive.

The **Set units automatically when possible** will, if enabled, cause EES to enter the units of variables that would otherwise have unspecified units. Units will be set only when a variable is by itself on the left hand side of an equal sign and a unit warning would otherwise occur. Note that the same effect can be accomplished using the $AutoSetUnits directive.

The **Use enhanced convergence techniques** will algebraically rearrange equations into equivalent forms that converge more readily, reducing the need to provide guess values.

The Formatted Equations Window

The equations can also be viewed in an easy to read, mathematical notation by selecting Formatted Equations from the Windows menu. The result is shown in Figure 1-25. Notice that, by default, the comments enclosed by quotations are shown in the Formatted Equations Window while those enclosed by curly brackets are not shown. All comments can be turned off by right clicking in the Formatted Equations Window and de-selecting the Display Comments option from the pop-up menu.

Figure 1-25: Formatted Equations Window.

The Formatted Equations window is particularly useful for viewing large equations with multiple parentheses. For example, consider the equation:

$$\overline{f} = \frac{4}{Re_D}\left[\frac{3.44}{\sqrt{L^+}} + \frac{\frac{1.25}{4L^+} + \frac{64}{4} - \frac{3.44}{\sqrt{L^+}}}{1 + \frac{0.00021}{\left(L^+\right)^2}}\right] \tag{1-10}$$

Equation (1-10) is the correlation provided by Shah and London (1978) for the average friction factor associated with developing flow in a round tube. The symbol Re_D is the Reynolds number and L^+ is the dimensionless length of the tube. This correlation can be entered in the Equations Window as shown below. Note that Chapter 12 discusses the EES heat transfer library which implements this and other useful correlations automatically.

```
Re_D=1000                                   "Reynolds number"
L|plus=10                                   "dimensionless length"
f_bar=(4/Re_D)*(3.44/sqrt(L|plus)+(1.25/(4*L|plus)+64/4-3.44/sqrt(L|plus))/(1+0.00021/L|plus^2))
    "average friction factor"
sigma=1                                     "a Greek symbol"
```

The equation for $\overline{f}$ is difficult to read or debug in the word processing format used by the Equations Window. It is much easier to view in the Formatted Equations Window, as shown in Figure 1-26.

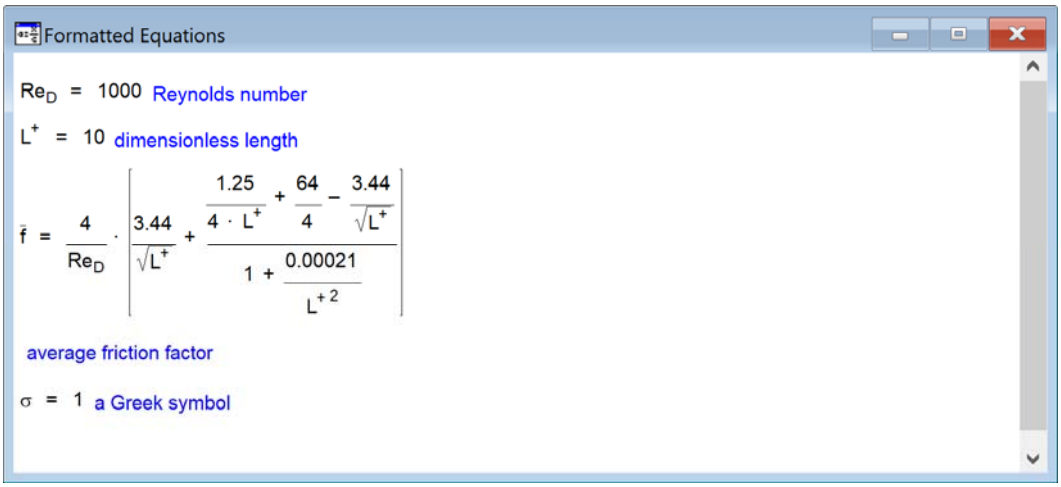


Figure 1-26: Formatted Equations Window.

Notice that the symbol f_bar is interpreted as f with an overbar in the Formatted Equations Window. Similarly Re_D is interpreted as Re with a subscript D and L|plus is interpreted as L with a superscript +. Greek symbols are also interpreted correctly; for example, the variable sigma is interpreted as σ. The formatting options that are available are summarized in Table 1-3. Section 1.6 shows how equations in the Formatted Equations Window can be copied in various formats in order to facilitate writing reports or papers.

Table 1-3: Variable display options for Formatted Equations Window.

Description	Method	Example of variable in Equations Window	Result in the Formatted Equations Window
Overbar	append _bar to the variable	X_bar	$\overline{X}$
Tilde	append _tilde to the variable	X_tilde	$\tilde{X}$
Hat	append _hat to the variable	X_hat	$\hat{X}$
Dot	append _dot to the variable	X_dot	$\dot{X}$
Double dot	append _ddot to the variable	X_ddot	$\ddot{X}$
Single prime	append _prime to the variable	X_prime	X'
Double prime	append _dprime to the variable	X_dprime	X''
Subscript	use the _ to start the subscript	X_a	X_a
Superscript	use the | to start the superscript	X|a	X^a
Infinity symbol	spell out infinity	X_infinity	X_∞
Greek symbol	spell out the name of the symbol	delta	δ
		DELTA	Δ

Password Protection

In the Professional license, it is possible to password protect the Equations Window so that only users with the password can edit it. Right click in the Equations Window and select Protected from the resulting pop-up menu. You will be prompted to enter and then re-enter a password, after which the background color will change to indicate that the text is now read only. Any subsequent attempt to edit the protected regions of the Equations Window will not be allowed until you unprotect the text. The text can be unprotected by right clicking on it and selecting Protected, at which time you must enter the password.

Key Variables

The Solution Window in a large EES program will contain many variables. Variables are listed in alphabetical order; however, it may be difficult to quickly and easily identify the small subset of variables that are of primary interest. To focus attention on a particular variable, it is possible to highlight its appearance. For example, from the Examples menu select Getting Started with EES and open the example labeled Settling the bill. This file is Dinner.ees in the Userlib\Examples folder (or Dinner.ees64 in the Userlib64\Examples folder if you have the 64 bit license). Solve the equations by selecting Solve from the Calculate menu (or by pressing the shortcut key F2) and the Solution Window should appear. Select the variable Debbie in the Solution Window and right-click on it. The Specify Format and Units dialog that appears is shown in Figure 1-27.

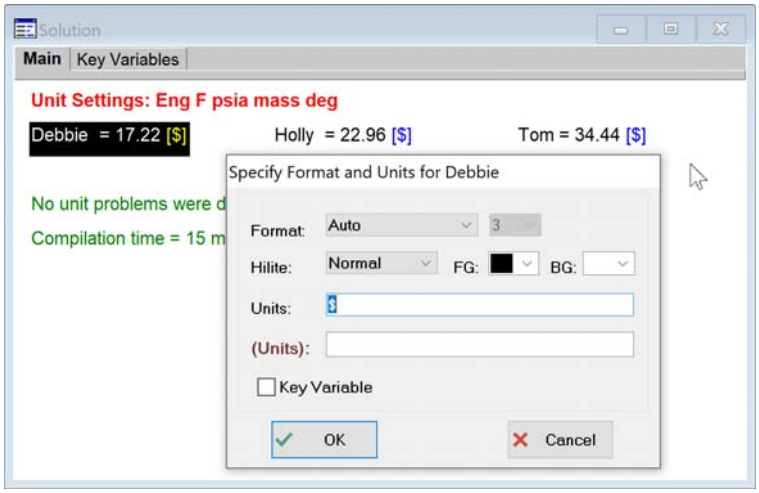


Figure 1-27: Specify Format and Units dialog.

The Specify Format and Units dialog allows you to change the formatting that is used to display the variable in the Solutions Window. For example, you can box the variable and change its background color. You can also select a subset of all of the variables in your Solution Window and make them key variables by selecting the Key Variable check box. An edit box will appear in which you can write some descriptive comments about the variable, as shown in Figure 1-28. Select OK and you will see that a new tab will appear in the Solutions Window allowing access to the Key Variables Window. All of the variables that have been identified as key variables are placed in the Key Variable Window together with their associated descriptive comments, as shown in Figure 1-29.

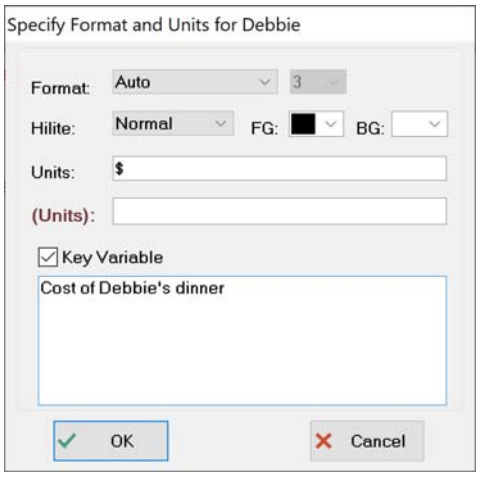


Figure 1-28: Key Variable.

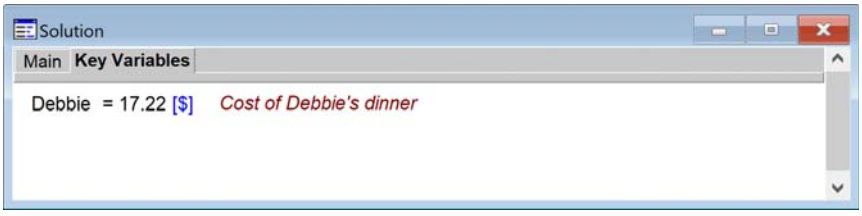


Figure 1-29: Key Variables Window.

Note that you can customize the Key Variables window in several ways. Clicking the right mouse will bring up a pop-up menu, as shown in Figure 1-30, that allows text, separators (horizontal lines) and text centered within a separator to be added, edited or deleted. Text, headers, separators and new Key Variables initially appear above the item that receives the mouse click. Clicking at the bottom of the window below all other items allows new items to be entered at the bottom of the list. The order of any of the items in the Key Variables window can be changed by 'dragging' them to a different location by pressing and holding the left mouse key down on the item that is to be moved while moving the cursor up or down.

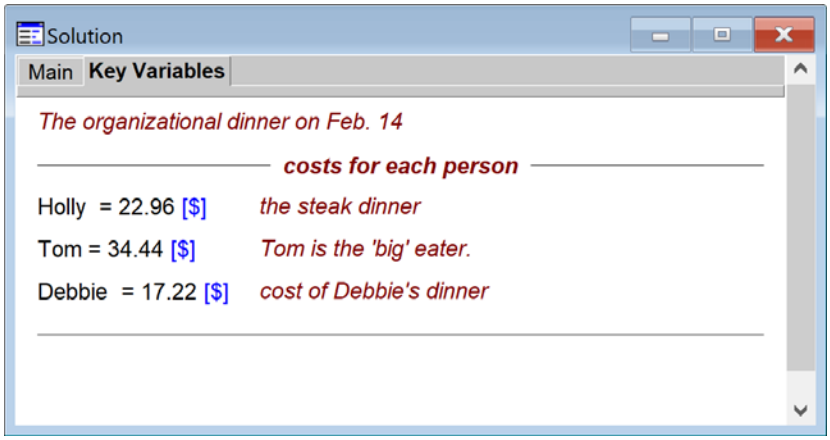


Figure 1-30. Right-click in the Key Variables window to bring up a popup menu.

1.3 Parametric Tables

Section 1.2 described how to use EES to obtain a single solution to a set of equations. It is often interesting to run a parametric study in which the effect of one variable (the independent variable) on another variable (the dependent variable) is studied. For example, let's estimate the pressure loss associated with the flow of water through a 45° elbow as a function of the flow rate. The inner diameter of the elbow is D = 2 cm and the density of water is ρ = 1000 kg/m^3. The dimensionless resistance coefficient for this type of elbow is estimated to be K = 0.3. The pressure loss can be estimated according to:

$$\Delta P = K \frac{\rho u^2}{2} \tag{1-11}$$

where u is the mean velocity of the water. The inputs are entered in the Equations Window; note that we'll start with $\dot{V}$ = 100 liter/min and subsequently vary this value in our parametric study.

```
$TabStops 2.5 in

"Inputs"
K=0.3                              "K factor for a 45 degree elbow, dimensionless"
V_dot=100 [liter/min]              "volumetric flow rate"
D=2 [cm]                           "inner diameter of elbow"
```

```
rho=1000 [kg/m^3]                    "density of water"
```

In Section 1.5, we will discuss how EES can help you deal with unit assignments and conversions. For now we'll have to keep track of units manually. Notice that the units assigned to each variable are indicated in the associated square bracket. The cross sectional area of the elbow is given by:

$$A_c = \pi \frac{D^2}{4} \tag{1-12}$$

The variable pi# is a built-in constant in EES corresponding to the value of π (other built-in constants are discussed in Section 1.9). The units of the variable A_c will be most convenient if converted from cm^2 to m^2; we will do this with the convert function in EES (See Section 1.5 for details concerning unit conversion.)

```
A_c =pi*D^2/4*convert(cm^2,m^2)      "cross-sectional area for flow"
```

The mean velocity of the water is:

$$u = \frac{\dot{V}}{A_c} \tag{1-13}$$

As written, the units for the variable u must be liter/min (the units for the variable V_dot) divided by m^2 (the units for the variable A_c). Therefore, the units for u must liter/min-m^2. Note that in this book, the hyphen sign will be used to indicate multiplication on one side of the divisor in the context of units; therefore liter/min-m^2 should be read as $\frac{\text{liter}}{\text{min m}^2}$. EES uses the same convention for unit specifications, although the hyphen can be replaced with a space or a dot (Alt-250 on a U.S. keyboard).

The units for the variable u, as calculated, are not useful; let's convert the units to m/s, which are more reasonable. Again, we'll use the convert function for this purpose.

```
u=V_dot*convert(liter/min,m^3/s)/A_c          "velocity, in m/s"
```

Finally, Eq. (1-11) is used to compute the pressure loss. Again, the units for the loss don't make a lot of sense, as calculated the units are $(kg/m^3)(m/s)^2$. Let's convert to kPa, which are more commonly used. Notice that we had to carry out several unit conversions to do this problem. The capabilities of EES discussed in Section 1.5 greatly reduces the hassle and potential errors that are associated with working with units.

```
DELTAP=K*rho*u^2/2*convert(kg/m-s^2,kPa)     "pressure loss, in kPa"
```

Solving leads to $\Delta P = 4.22$ kPa for $\dot{V} = 100$ liter/min.

Creating a Parametric Table

In order to parametrically vary the volumetric flow rate (the independent variable) and study its effect on the pressure loss (the dependent variable) we need to create a Parametric Table. Select New Parametric Table from the Tables menu. The New Parametric Table dialog will appear, as shown in Figure 1-31.

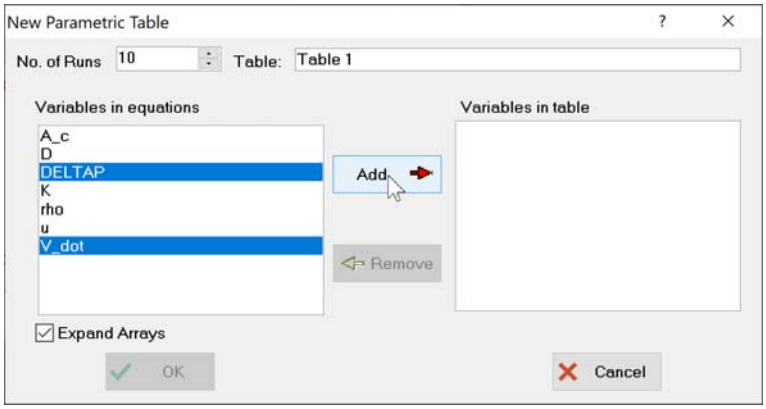


Figure 1-31: New Parametric Table dialog.

The window on the left side of the dialog provides a list of all of the variables that are included in the Equations Window. Highlight the independent and dependent variables of interest (in this case, the variables V_dot and DeltaP) by clicking on them and then select Add in order to add these variables to the list on the right hand side that shows the variables to be included in the table. Any other variables that you wish to include in the table can also be added. You can select the variables one at a time or all at once. Select OK in order to create the Parametric Table, shown in Figure 1-32. By default, there are 10 runs (rows) in the table. Runs can be added or removed by selecting Insert/Delete Runs from the Tables menu or by right-clicking on any run number and selecting Insert Runs or Delete Runs. There is a column for each of the variables included in the table. Columns can be added or deleted by selecting Insert/Delete Vars from the Tables menu or right clicking on any variable and selecting Insert Column or Delete.

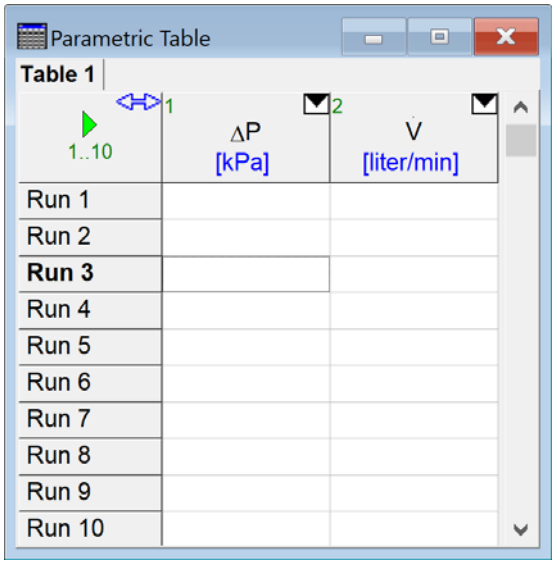


Figure 1-32: Empty Parametric Table.

Alter Values

To carry out a parametric study in which the volumetric flow rate is changed, it is necessary to fill in the column for the variable V_dot with flow rate values that are of interest. This process can be done manually, one run at a time. More typically, you will right-click on the column of interest and select Alter Values or click on the triangular icon in the column header to bring up the dialog shown in Figure 1-33.

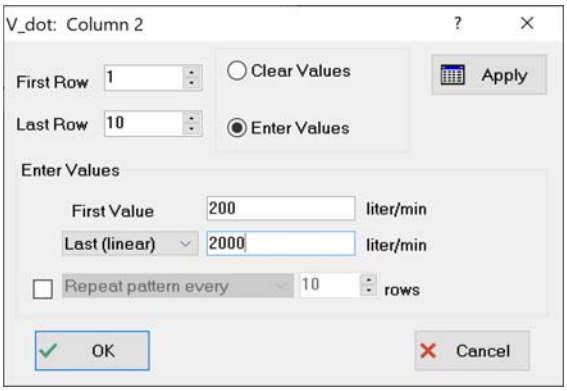


Figure 1-33: Alter values dialog.

Select the rows to be filled in (rows 1 to 10, for this example) and the pattern to be used to enter the values. In Figure 1-34, the dialog is filled in so that the volumetric flow rate varies from 200 liter/min (in row 1) to 2000 liter/min (in row 10) in equally spaced intervals, as shown in Figure 1-33. Other options can also be specified by clicking on the drop down menu under First Value. For plots that will use a logarithmic scale, the selection Last (Log) is particularly useful as it logarithmically spaces the values in the table.

Parametric Table

Table 1

1..10	ΔP [kPa]	V [liter/min]
Run 1		200
Run 2		400
Run 3		600
Run 4		800
Run 5		1000
Run 6		1200
Run 7		1400
Run 8		1600
Run 9		1800
Run 10		2000

Figure 1-34: Parametric Table with values of V_dot set.

In the Professional license, the value entered for Last Row (upper left of Figure 1-33) can be set to a value that is greater than the current number of rows in the Parametric table. After confirmation, EES will extend the number of rows in the table to this value. The Professional license also allows EES variables (in addition to numerical values) to provide the First Value and Last/Increment values.

Solving a Parametric Table

The Parametric Table is solved and filled in using the Solve Table command from the Calculate menu (or by pressing F3). EES will begin with the first run that is specified by the Solve Table command and look in the table to see which columns in the corresponding row have specified values. It will then specify the value of these variables in the Equations Window, solve the resulting system of equations, and fill in the values of the remaining columns based on the solution. Select Solve Table to bring up the Solve Table dialog shown in Figure 1-35.

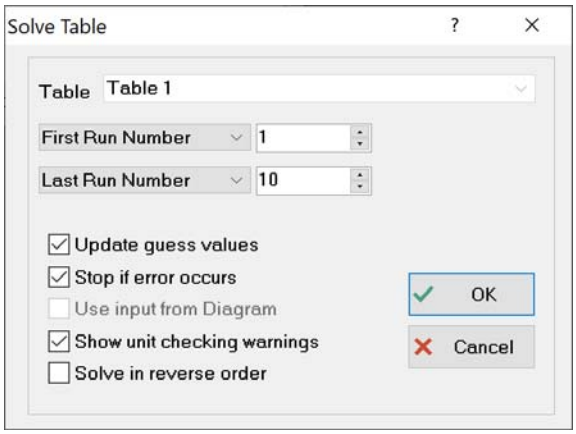


Figure 1-35: Solve Table dialog.

The Solve Table dialog is initially set so that EES will start with run 1 and end with run 10. By pressing OK, EES will go to run 1 of the Parametric Table shown in Figure 1-34 and find that the value of the variable V_dot is set to 200. Therefore, you can imagine that EES places the following equation in the Equations Window.

```
V_dot=100
```

Of course, this presents a problem because the statement:

```
V_dot=100                    "volumetric flow rate, in liter per min"
```

already exists in the Equations Window. The variable V_dot cannot have multiple values as it causes the equation set to be over-constrained (8 equations and 7 variables). Solving the table will result in the error dialog shown in 6.

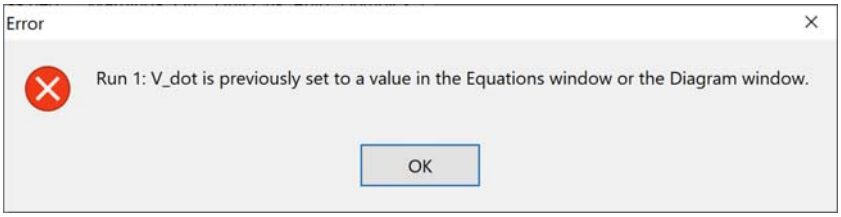


Figure 1-36: Error dialog.

This problem can be alleviated by commenting out the equation that specifies the volumetric flow rate in the Equations Window. Highlight the statement, right-click, and select Comment {} in the resulting popup menu in order to temporarily remove it from the equation set.

```
{V_dot=100                    "volumetric flow rate, in liter per min"}
```

Now solve the table and EES will fill in each column of the Parametric Table as shown in Figure 1-37.

Parametric Table

Table 1

1..10	1 ΔP [kPa]	2 V̇ [liter/min]
Run 1	16.89	200
Run 2	67.55	400
Run 3	152	600
Run 4	270.2	800
Run 5	422.2	1000
Run 6	607.9	1200
Run 7	827.5	1400
Run 8	1081	1600
Run 9	1368	1800
Run 10	1689	2000

Figure 1-37: Solved Parametric Table.

It will sometimes be useful to solve only a subset of the runs in the table. This is easily accomplished by changing the first and last run numbers in the Solve Table dialog.

All tables in the Professional license are equipped with a column-sizing control that automatically sets the required width of the columns in the table. The column-sizing control is a double-headed arrow appearing in the upper left cell of the table, as shown in Figure 1-37. When it is not enabled, the arrowheads are gray-filled as seen in Figure 1-38 (left). Clicking on the control resizes the columns so that contents of the column just fit within. The arrowheads will then be filled with a light blue color, as shown in Figure 1-38 (right). Holding the Ctrl key while clicking on the control returns the column widths to their default size.

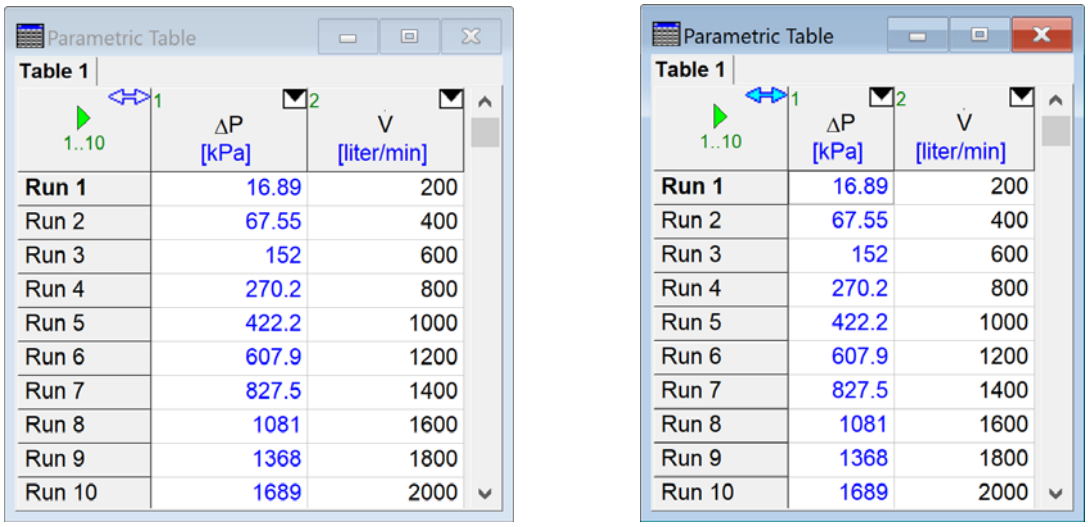


Figure 1-38: Parametric table in Professional license showing column sizing control (left) off and (right) on.

In the Professional license, the start and stop values can be set to variables. For example, add the lines below to the Equations Window.

```
startrow=3                    "first run"
endrow=7                      "last run"
```

Open the Parametric Table, right click on the DeltaP variable column and select Alter Values. Select Clear Values to clear the previous solutions from the column. Solve the table again. This time, click on the arrow next to First Run Number and change it to First Run Variable and then select startrow. Change Last Run Number to Last Run Variable and select endrow. The result is shown in Figure 1-39. Select OK and you will see that only runs 3 through 7 have been filled in.

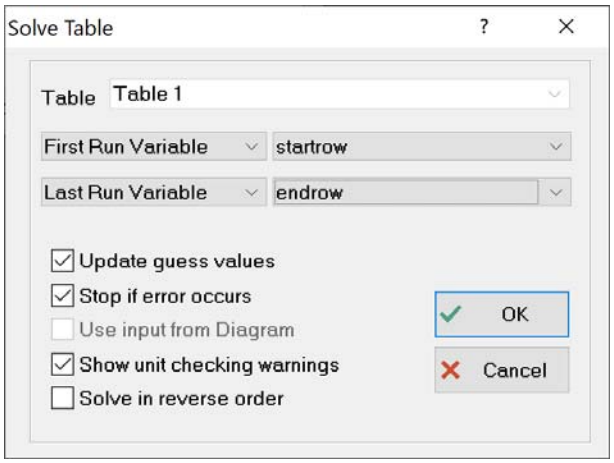


Figure 1-39: Solve Table dialog with first and last runs set using variables.

Formatting Columns

The Parametric Table shows the value of pressure drop for each corresponding value of volumetric flow rate. In Section 1.4 we will see how to use this information to prepare a plot. Before we do that, let's look at some other features of Parametric Tables. The position of the columns can be adjusted by dragging (i.e., pressing the left mouse button and then moving mouse while holding down the mouse button) the column headers. The units, alternate units , and format of the columns can be adjusted by right-clicking on the column and selecting Properties to bring up the Format Properties Table dialog shown in Figure 1-40 (for the column corresponding to the variable DELTAP). The numerical format used to display the results, the background color, and the column width can also be adjusted. The column number can also be set in this dialog. At the bottom of the dialog, statistics are provided for the entries in the column.

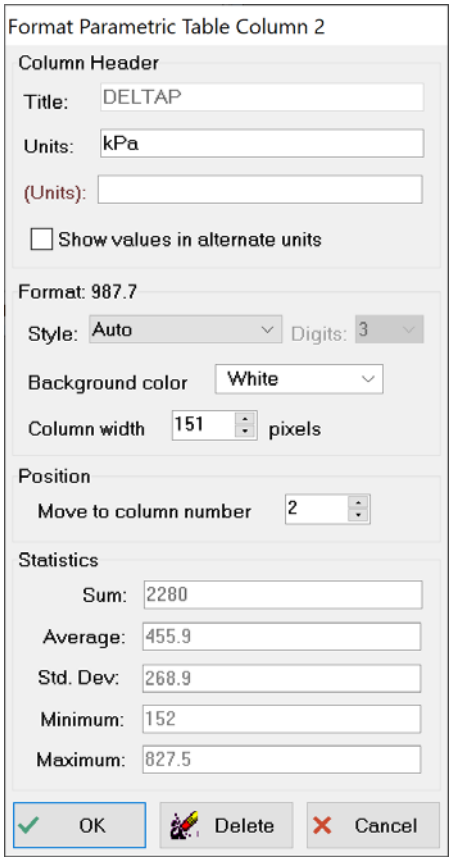


Figure 1-40: Format Properties Table dialog.

The $If, $IfNot, $Else, and $EndIf Directives

In order to run the Parametric Table we had to comment out the line in the Equations Window that specified the value of the variable V_dot. This change temporarily removes the line of code; it can be returned by highlighting the text, right clicking, and selecting Undo Comment {} from the pop-up menu.

```
V_dot=100                    "volumetric flow rate, in liter per min"
```

A more elegant method for removing one or more lines of code when a Parametric Table is being solved is to use the $If directive. The $If directive is used according to:

```
$If Condition
    line(s) of code to be executed if Condition is true
$Else
    line(s) of code to be executed if Condition is false
$EndIf
```

where Condition is a keyword that indicates an execution condition. There are many such keywords recognized by EES and these are discussed in more detail in Chapter 14. One keyword is ParametricTable, which evaluates to true when the equations are being solved from a Parametric Table. For our problem, we want to specify the value of the variable V_dot in the Equations Window only if the value of ParametricTable is false:

```
$If ParametricTable
$Else
    V_dot=100                          "volumetric flow rate, in liter per min"
$EndIf
```

Alternatively, we could use the $IfNot directive according to:

```
$IfNot ParametricTable
    V_dot=100                          "volumetric flow rate, in liter per min"
$EndIf
```

You should find that your equation set now runs if you select either Solve or Solve Table from the Calculate menu (or if you press either F2 or F3, respectively).

Naming and Documenting Parametric Tables

The Parametric Table can be renamed and annotated. Right click on the tab corresponding to the table in order to bring up the Information for Parametric Table dialog, shown in Figure 1-41. Provide a descriptive name in the Title field (DP) and record the details of the calculation in the Description panel. The description can optionally be printed with the table.

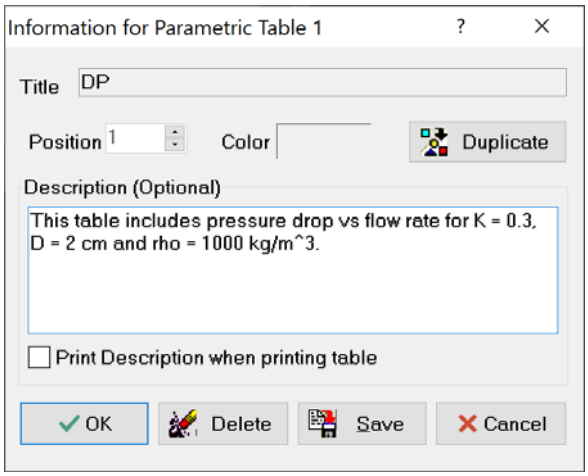


Figure 1-41: Information for Parametric Table dialog.

Saving and Loading Parametric Tables

The Parametric Table is automatically saved with the EES program and reloaded when the program is reopened. However, you can also save your Parametric Table independently by selecting Save in the Information for Parametric Table dialog (Figure 1-41). There are a number of different formats available for you to save your table, these are accessed by selecting the Save as type option as shown in Figure 1-42.

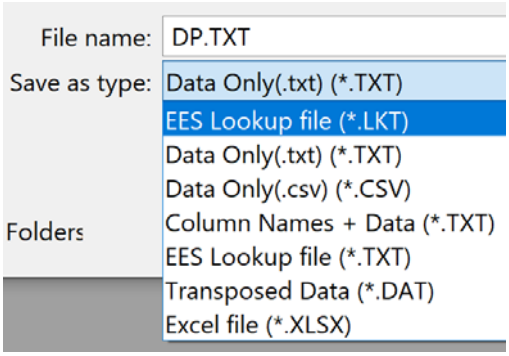


Figure 1-42: Formats available for saving Parametric Tables.

The table can be saved in a format that can be reopened by EES as a Lookup Table (.lkt). Lookup Tables are discussed in Section 1.8. By selecting either Data Only(.txt) or Data Only(.csv) the data in the table is saved as text (.txt) or comma separated values (.csv) files, respectively; either of these file formats can be opened by most other software or spreadsheets. By selecting Column Names + Data a text file is created in which the column names and units are saved as well as the data. The EES Lookup file (*.txt) option saves the data in the format of an EES Lookup table but as a text file. Transposed Data saves only the data and causes the rows and columns to be transposed. Finally, data can be saved as an Excel file (.xlsx). In this case, the first row in the Excel file will be the name of the column. The second row will be the units. The data in the table will start in the third row.

Note that a Parametric table can also be saved as a .par file using the Store Parametric Table command in the Tables menu. A .par file can only be read by EES using the Retrieve Parametric Table command from the Tables menu. This format is useful if you wish to move a Parametric table to another EES file.

Deleting Parametric Tables

Parametric tables are saved with an EES file. They can also be saved independently as explained in the previous section. A Parametric table can be deleted in two ways. One way is to right-click on the table's tab to bring up the dialog window shown in Figure 1-41 and click the Delete button, A second way to delete one or more Parametric tables is the Delete Parametric Tables command in the Tables menu. Selecting this command will bring up the dialog window shown in Figure 1-43 which lists all Parametric tables. Check off the tables that are to be deleted and then click the Delete button. Note that deleting the Parametric table means that the data that it contains will not be saved with the EES file the next time the EES file is saved. However, the

Delete command does not delete a Parametric table file that has been saved separately in an .lkt, .txt, .csv, .par or other file format.

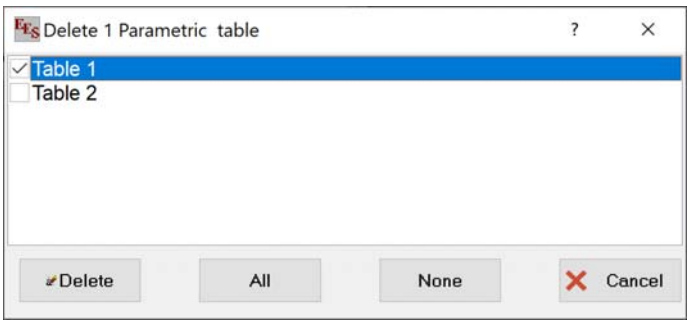


Figure 1-43: The Delete Parametric Tables dialog

Copying Information from a Parametric Table

A range of cells from the table can be copied to a temporary memory location available in Windows that is referred to as the "clipboard" and pasted into other tables in EES or to other applications, such as a word processor or a spreadsheet. For example, highlight the cells between rows 3 and 7 by clicking in upper left cell, and the pressing and holding the Shift key down while clicking in the lower right cell. Next, right click to bring up the pop-up menu shown in Figure 1-44.

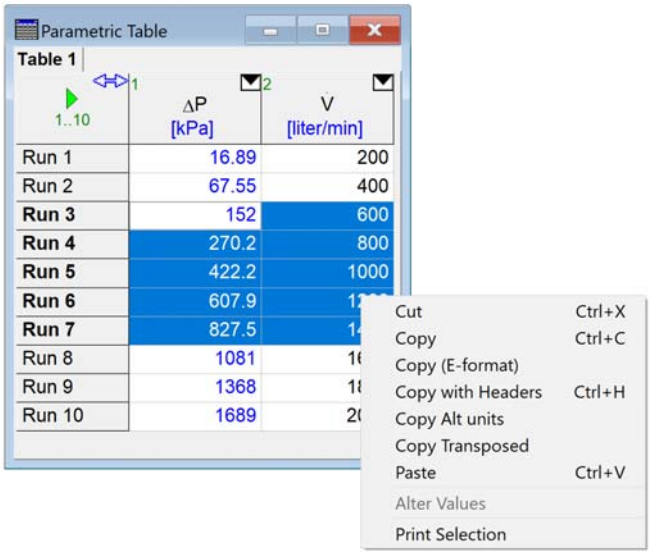


Figure 1-44: Pop-up menu for copying information from Parametric table.

Select Copy to place the information on the clipboard. Open another spreadsheet or word processor and then paste the information. The result is shown in Figure 1-45(a), using Excel. If the Copy with Headers option is selected then the header for each column is also copied; the result is shown in Figure 1-45(b). Notice in both cases that the number of significant figures associated with the data that is copied corresponds to the number of significant figures used to display the information in the Parametric Table. If the Copy (E-format) option is selected, then the data are copied with 12 significant figures so that there is no loss of accuracy associated with the copy/paste operation, as shown in Figure 1-45(c). The Copy Alt units command will copy

the numerical data in the selected section of the table in their alternate units, if they are defined. Otherwise, it will just act as the Copy command.

When the Copy command is issued, EES actually copies the data to the clipboard in two formats. The first format uses the number of significant figures that are displayed and this is what appears in an external program. In the second format, EES retains all of the significant digits. If you paste data from one column of a table to another column in EES (or to any other table in EES), the data will be copied with all of the significant figures so that there is no loss in precision during the copy process.

152	600
270.2	800
422.2	1000
607.9	1200
827.5	1400

(a)

DELTAP [kPa]	V_dot [liter/min]
152	600
270.2	800
422.2	1000
607.9	1200
827.5	1400

(b)

1.51981775E+02	6.00000000E+02
2.70189823E+02	8.00000000E+02
4.22171599E+02	1.00000000E+03
6.07927102E+02	1.20000000E+03
8.27456333E+02	1.40000000E+03

(c)

Figure 1-45: Information pasted into Excel using (a) Copy, (b) Copy with Headers, and (c) Copy (E-format) commands.

1.4 Basic Plotting

Section 1.3 described how to create a Parametric Table that includes the values of a dependent variable calculated over a range of values of an independent variable. This information is viewed most conveniently in the form of a plot. EES allows the generation of several types of plots, as discussed in Chapter 9. In this section, only the most basic X-Y type plot is discussed. Data contained in the Parametric Table created in Section 1.3 will be plotted in this section. Data contained in Lookup Tables (see Section 1.8), Arrays (see Section 1.7), and Integral Tables (see Chapter 7) can also be used as the basis of a plot.

Generating a Plot

To generate a plot, select New Plot Window from the Plots menu in order to access the New Plot Setup dialog shown in Figure 1-46. At the top of the dialog you can provide a title (in the Tab Name field) and some descriptive text. The upper right portion of the dialog is used to select the source of the data; here it is a Parametric Table and the name of the table is Table 1. (Note that the New Plot Setup window width can be increased by dragging the right side if needed to display long table names.) As shown in Figure 1-46, the New Plot dialog is set up to plot all 10 rows in the table, but it is also possible to plot only a subset of the rows. The two windows in the dialog allow you to specify the independent (X-Axis) and dependent (Y-Axis) data. Figure 1-46 shows the New Plot dialog set up to plot the pressure loss as a function of the flow rate. Select OK to create the plot, which is shown in Figure 1-47.

Note that in the Professional license, variables may be plotted in their base or alternate set of units. Alternate units can be entered in the Format Table dialog, as shown in Section 1.5.

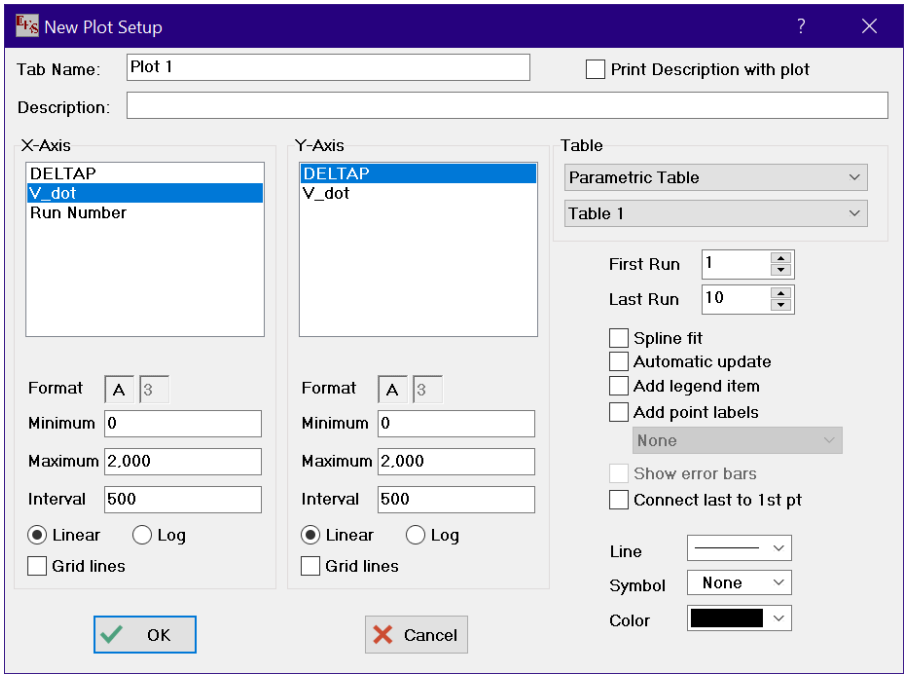


Figure 1-46: New Plot Setup dialog.

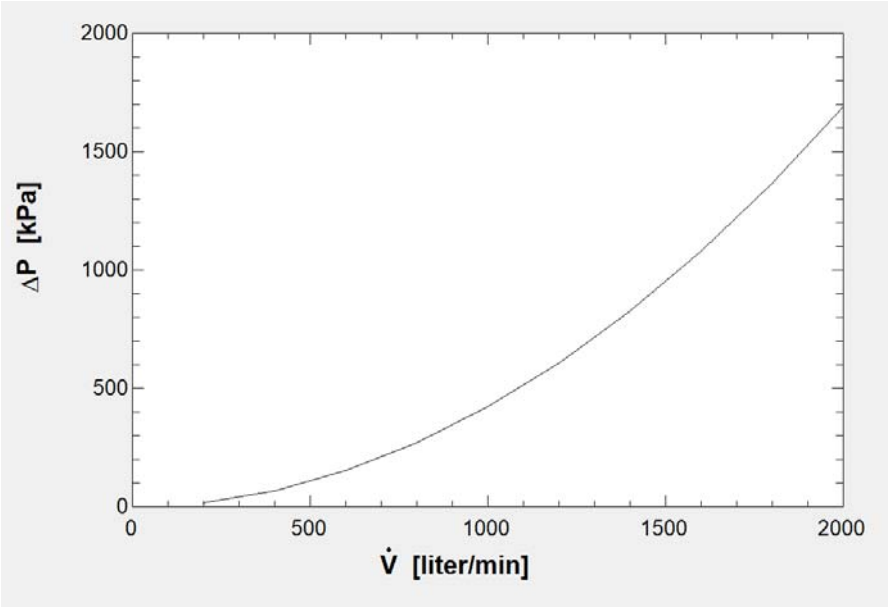


Figure 1-47: Plot showing pressure loss as a function of flow rate.

Modifying Axes

Almost every aspect of the plot can be modified in order to customize or improve it. A detailed discussion of plotting options is provided in Chapter 9. Some of the more commonly used options are described in this section.

Double-click (or right-click) on either axis label to bring up the Format Text Item dialog. You can change the axis label to something descriptive with units and change the font. A gray triangular icon is visible at the bottom right of the plot when the plot toolbar is visible. You can resize the plot by selecting the gray triangle in the bottom right-corner and holding the mouse button down on the icon while moving the mouse to an alternative location. Note that the size of all text and graphic items in the plot changes in proportion to the size of the plot unless the Ctrl key is held down during this process. The axis scale can be adjusted by placing the mouse over any of the axes (left, right, bottom, or top) and clicking the right mouse button. This action will bring up the Modify Axis dialog shown in Figure 1-48, which allows you to make adjustments in the axis scale, add grid lines, etc.

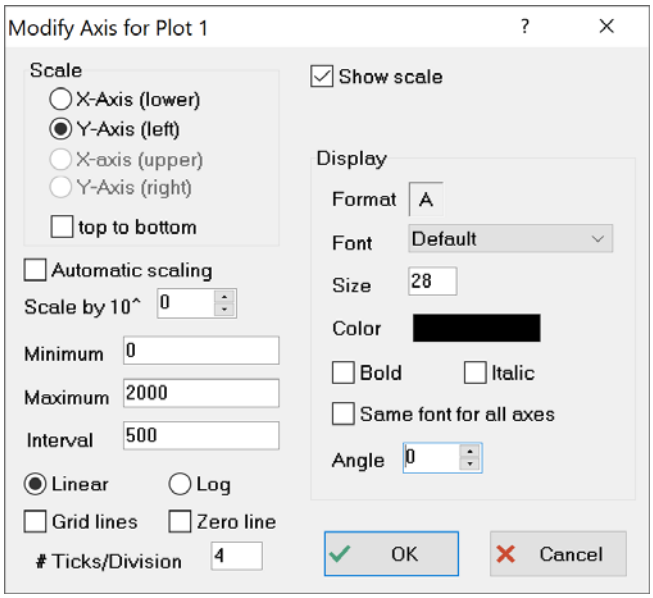


Figure 1-48: Modify Axis dialog.

An improved plot of pressure loss as a function of flow rate is shown in Figure 1-49(a). By selecting the Automatic scaling option in the Modify Axis dialog, the axis scale will be automatically adjusted to encompass the data being presented. The axis labels can be scaled by a factor of 10, 100, 1000, etc. by adjusting the Scale by 10^ option; Figure 1-49(b) shows the result of scaling the labels by 100 (10^2). Select the top to bottom option in the Modify Axis dialog in order to invert the scale, as shown in Figure 1-49(c).

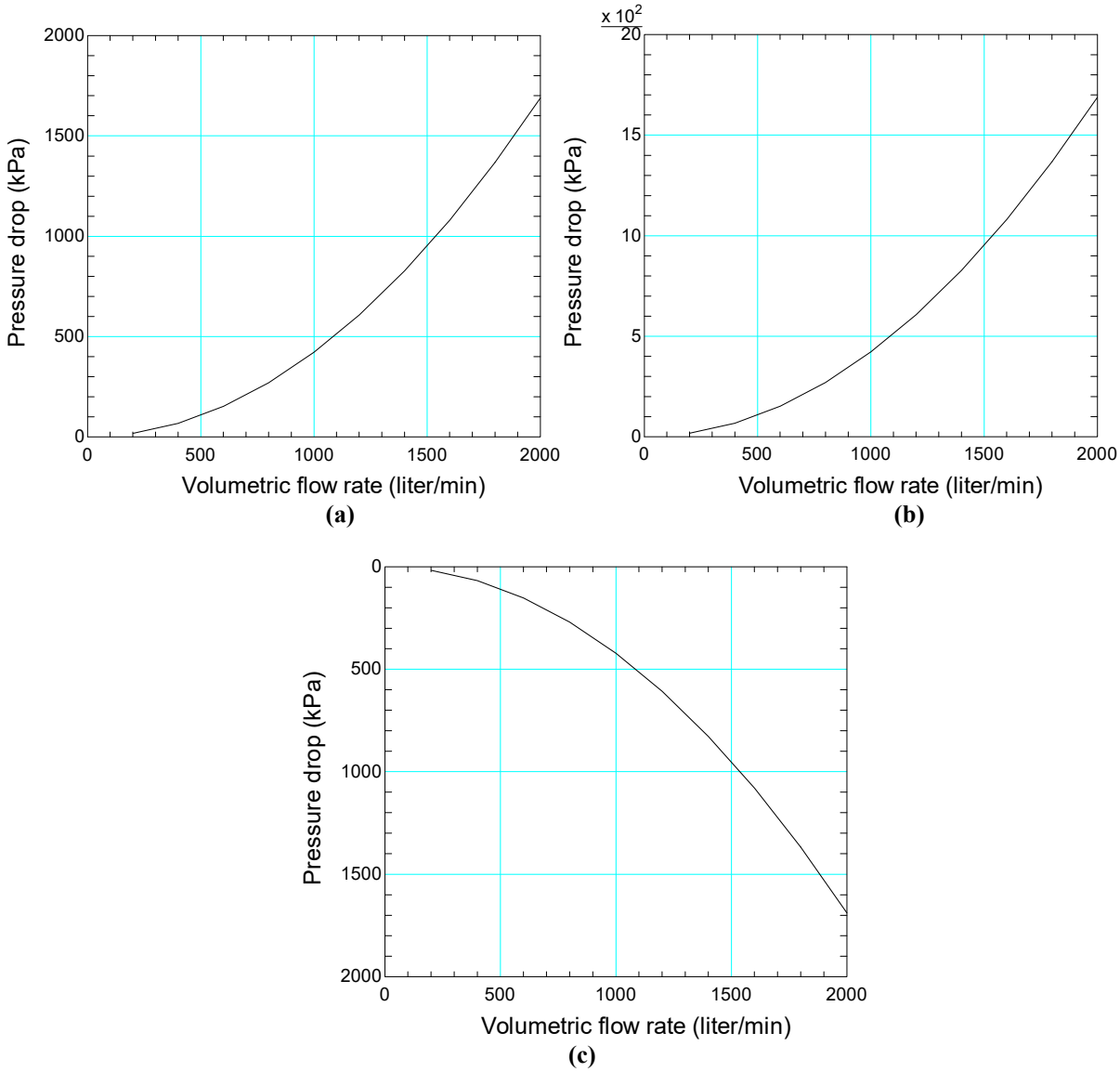


Figure 1-49: Improved plots of pressure loss as a function of flow rate (a) with default y-axis format, (b) with y-axis scaled by a factor of 10^2, and (c) with y-axis scaled from top-to-bottom.

Note that the range of values included on the either axis can also be changed by positioning the mouse cursor on an axis and dragging it. For example, if you wish the starting value of Volumetric flow rate shown in Figure 1-49 at a number other than 0, move the mouse over the left Y axis. The cursor will change to a ↔. Press and hold the left mouse button while moving it slightly to the right or left. Release the mouse when the axis is in the desired position.

Overlaying Plots

Multiple data series can be overlaid onto the same plot. Change the resistance coefficient to K = 0.4 and run the Parametric Table again. Select Overlay Plot from the Plots menu in order to plot pressure loss as a function of flow rate for the adjusted value of K. Your plot should now have two sets of data, as shown in Figure 1-50.

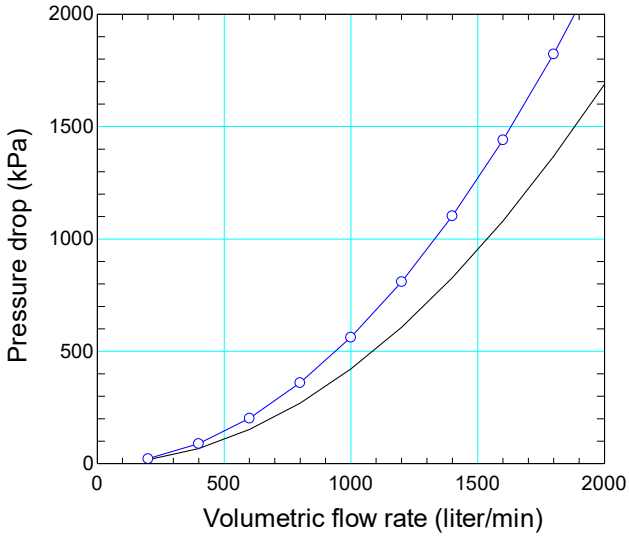


Figure 1-50: Plot with two data series.

Modifying Plots

Double click (or right click) anywhere on the plot (other than on a text or graphic item) in order to access the Modify Plot dialog shown in Figure 1-51. The Modify Plot dialog allows you to add and adjust grid lines and modify the border and size of the plot. The upper window lists all of the data series that appear in the plot. You can delete one or more of these series by selecting Delete button. The underlying data set that is used in the plot to can be extracted to a Lookup Table by selecting Get Data button. The characteristics of each plotted data series (e.g., the line thickness, color, symbols, etc.) can be adjusted. The Smoothing option allows you to adjust the line passing through the data using either a cubic spline, polynomial fit, or moving average. If polynomial fit is selected then a slider appears allowing you to adjust the order of the polynomial. If moving average is selected then a slider appears allowing you to adjust the number of data points that are averaged together.

The parameters in the Border box allow setting the upper left corner of the plot as well as its width and height. The width and height can also be changed by dragging the lower right corner of the plot with the mouse button depressed. However, the settings in the Modify Plot dialog make it easy to set all plots so that they have exactly the same size and position.

At the bottom left of the Modify Plot dialog, controls are provided to allow data to be placed on the primary (X1 and Y1) or secondary axes (X2 and Y2). The primary axes are shown at the bottom (x-axis) and left (y-axis) of the plot. The secondary axes are initially shown at the top (x-axis) and right (y-axis), but they can be moved anywhere as discussed in Section 9-1.

The controls at the bottom right allow the X and Y axes to exchange position and set the background color of the plot rectangle. By default, the plot rectangle has a white background. Clicking in the box to the right of Background color will bring up a color dialog that allows any background color to be selected.

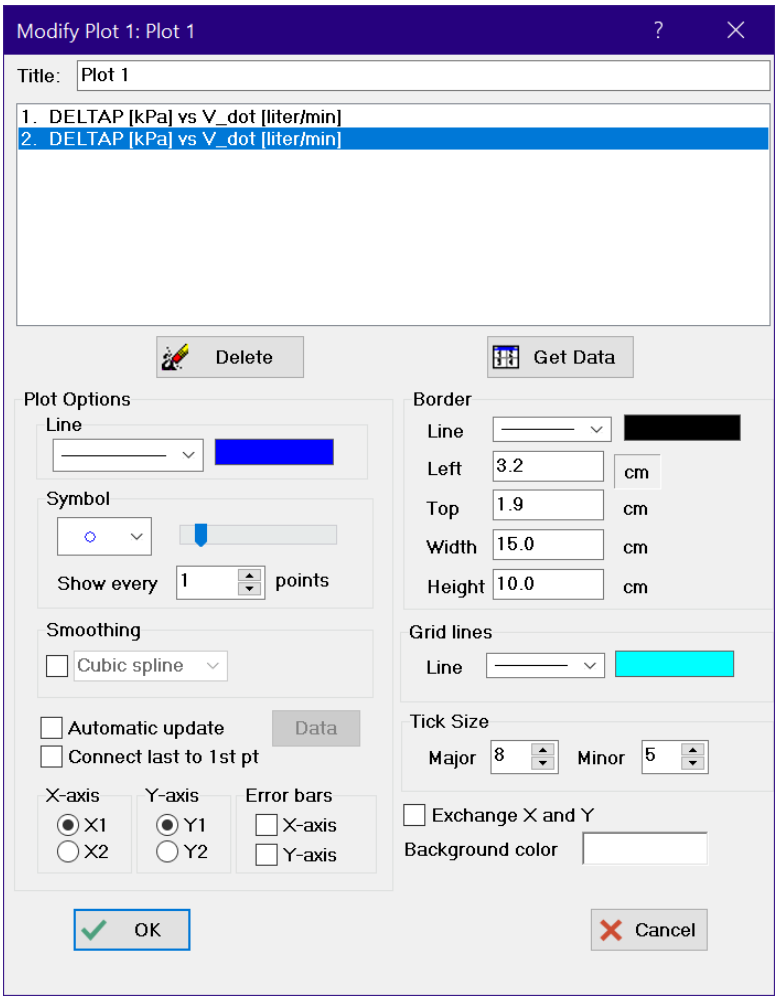


Figure 1-51: Modify Plot dialog.

Automatic Update

The Automatic update option in the Modify Plot dialog causes the data series to be re-plotted each time the data source changes (e.g., each time the data in the Parametric Table are adjusted). This option is useful if you want to adjust parameters in your model and immediately see how they affect a plotted result. Once the Automatic update option is selected, the Data button in the Modify Plot dialog becomes live. Clicking the Data button will bring up the Modify Automatic Plot Data dialog shown in Figure 1-52, which allows you to adjust the range of data that is plotted. Note that this dialog provides a check box for All rows in table. If this box is selected, the range of plotted data will automatically change to include the entire table even as the number of rows in the table change. If this option is not checked, then only the specified rows will be updated.

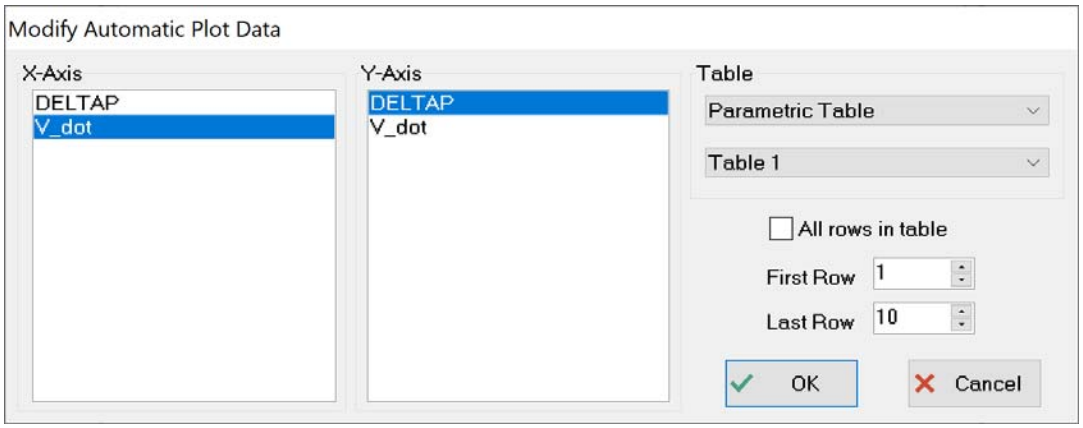


Figure 1-52: Modify Automatic Plot Data after clicking the Data button.

Error Bars

Select Y-axis in the Error Bar section of the Modify Plot dialog to bring up the Specify Error Bar Information dialog shown in Figure 1-53(left). The error bars can be set according to either an absolute or relative uncertainty. Select an Absolute uncertainty of 50 kPa for the y error bar. Repeat the process for the x error bar using an uncertainty of 50 liter/min in order to generate the plot shown in Figure 1-54. Alternatively, the uncertainty values used to generate the error bars can be obtained from a table so that different values of uncertainty are applied to each data point. It is also possible to generate asymmetric error bars by de-selecting the Identical upper and lower values option in the Specify Error Information dialog; this brings up a separate window so that the upper and lower uncertainty values can be selected independently, as shown in Figure 1-53(right).

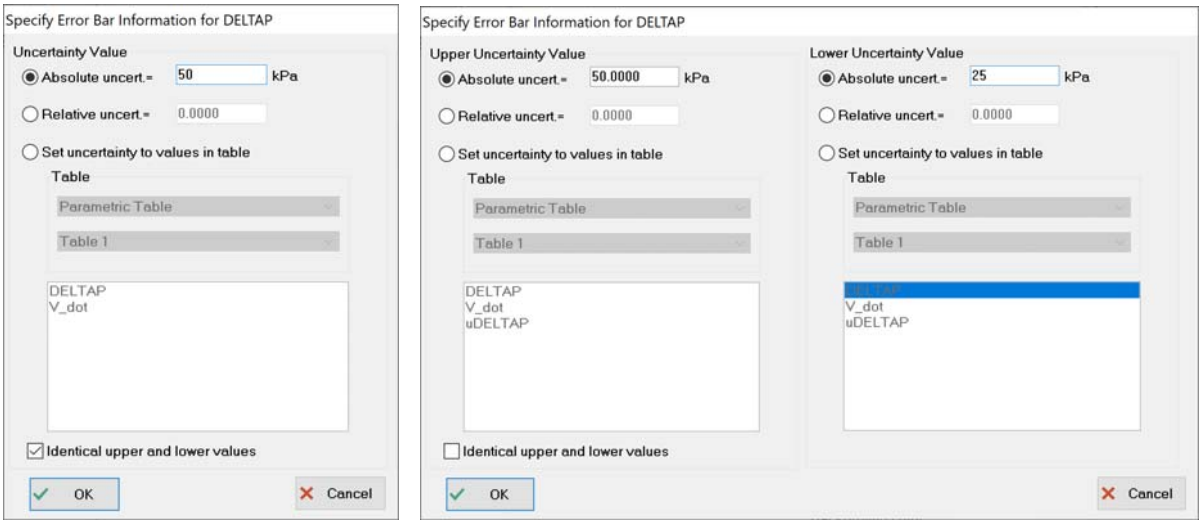


Figure 1-53: Specify Error Bar Information dialog (left) with identical upper and lower values, and (right) with different upper and lower values.

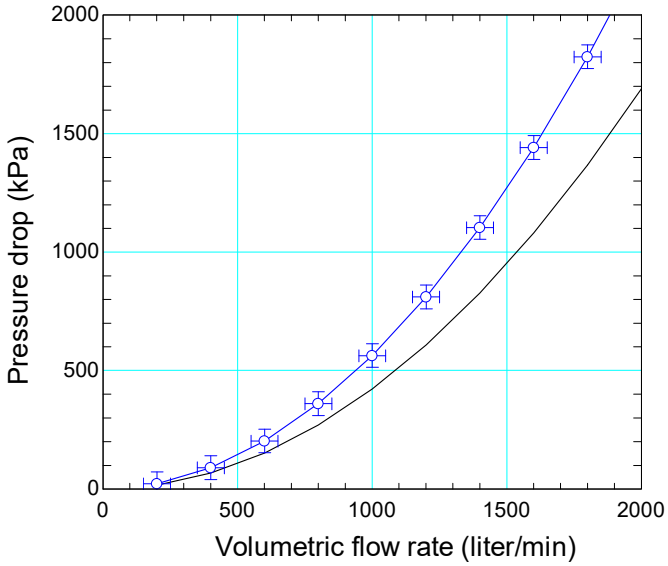


Figure 1-54: Plot with x and y error bars.

The Plot Tool Bar

The plot tool bar (Figure 1-55) should be visible in the plot window; if not, select Show Tool Bar from the Plots menu.

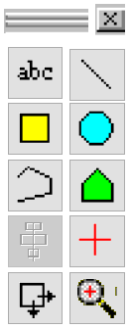


Figure 1-55: Plot tool bar.

Selecting the text tool abc opens the Format Text dialog, shown in Figure 1-56(left). You can type text into the window in order to annotate your figure. The size, font, style, and background color of the text can be selected with the controls in this dialog. Clicking the X_y or X^y button will change the selected text in the text field into a subscript or superscript, respectively. For example, to enter $T_{initial}$, type Tinitial into the text field. Then select initial and click the X_y button. The text field will change to T\dinitial\. The \d instructs EES to display the following text, up to the next \, as a subscript. \u in place of \d will result in a superscript. You can enter these codes directly, without using the buttons if you prefer. Clicking the ¶ button will insert a line break, carriage return code in the text string which will appear as \m in the text edit box. When this option is selected, a check box will appear offering the option of centering the lines of text with respect to each other.

In the Professional license, you can also link the position of the text that you add to a plot to a certain data point using the Link Text to Data Point in Plot option at the bottom of the Format Text dialog. This capability is particularly useful if you have a plot that is automatically updated as you change parameters and you would like the annotations to remain in their correct locations with respect to the data series.

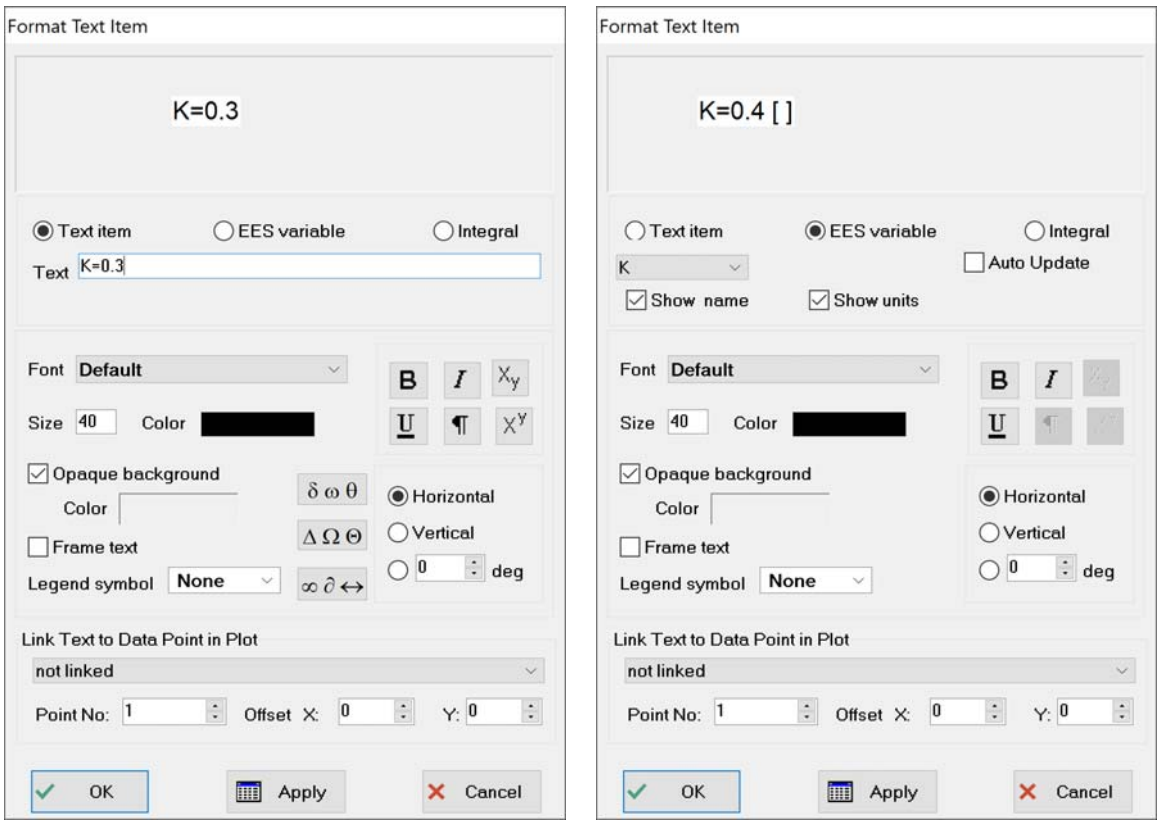


Figure 1-56: Format Text dialog with (left) Text item and (right) EES variable selected.

By default, the text tool is set to place a text item onto the plot; therefore the Text item radio button is automatically selected. However, you can also annotate the plot using the value of an EES variable by selecting the EES variable radio button. The resulting dialog is shown in Figure 1-56(right). Use the pull down menu to select the appropriate variable (in our case K). If the radio button Show name is selected, then both the value and the name of the variable will be shown. If the Show units radio button is selected, then the units of the variable will also be shown provided that they are set, as discussed in Section 1.5. The Integral radio button automatically integrates one of the data series shown in the plots (selected by the user from a dropdown menu) and places the value of the integral on the plot.

Note that strings of international characters can be displayed on the plot by entering the string in the text box. EES will automatically detect if the entered text contains international characters. If so, a blue (International Characters in use) message will appear under the text, as shown in Figure 1-57. The subscript, superscript, and symbol buttons are disabled when international characters are used. Also, a different background color can be set for any text item, as shown in Figure 1-57.

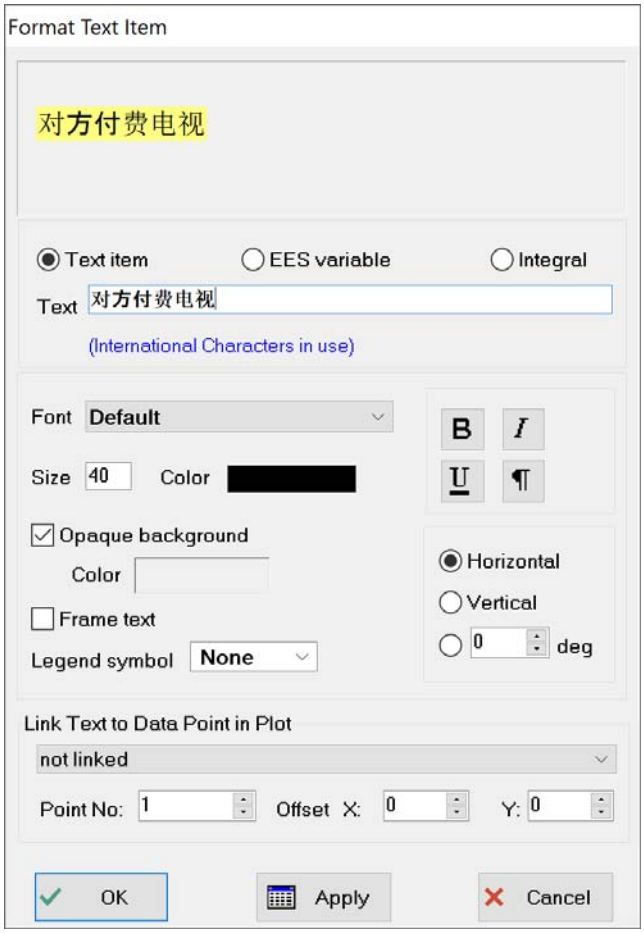


Figure 1-57: Using international text in the Plot window

There are several other useful tools in the Plot tool bar shown in Figure 1-55. The line tool allows you to draw lines on your plot. By holding the shift key down during the drawing process, the lines are constrained to being horizontal, vertical, or 45°. The polyline tool allows you to draw a series of connected lines. The square and circle tools allow rectangular and circular shaped regions to be outlined on the plot; holding the Shift key down while drawing these objects forces them to have a constant aspect ratio. The polygon tool allows you to draw a polygon shaped region; note that the polygon tool is only available in the Professional license.

When the plot tool bar is visible you can also copy graphic objects from other applications (e.g., PowerPoint®) and paste them into the plot window. The pipe elbow shown at the bottom right of Figure 1-58 was inserted in this manner. Right click on any of the objects on a plot in order to change their properties (e.g., to add arrows to lines or to shade regions in opaque or semi-transparent colors) or move objects forward (in front of) or backwards (behind each other). Select multiple objects at once by clicking on them one by one while holding the Shift key down. Alternatively, you can select a rectangular region on the plot by pressing and holding the left mouse button down at the upper left corner of the region and moving the mouse to the lower

right corner of the region. All objects within the region will be selected allowing the properties of all selected objects to be changed at once.

The alignment tool will be activated when multiple objects are selected. The alignment tool can be used to align the positions of a group of selected objects vertically and/or horizontally. The alignment tool is also activated when the X or Y-axis label is selected. In this case, the alignment tool can be used to move the label to the center the axis.

Text and graphic items in Plot windows can be moved with the arrow keys when the Plot tool bar (Figure 1-55) is visible. When an arrow key is pressed the first time, the selected items will move 5 pixels. If the key is held down, the selected items will move quickly. Releasing and pressing the key again (within 3 seconds) will move the selected items by 1 pixel, allowing fine control. If the arrow key is then held down for more than 3 seconds, the speed at which the selected items move in the designated direction will increase again.

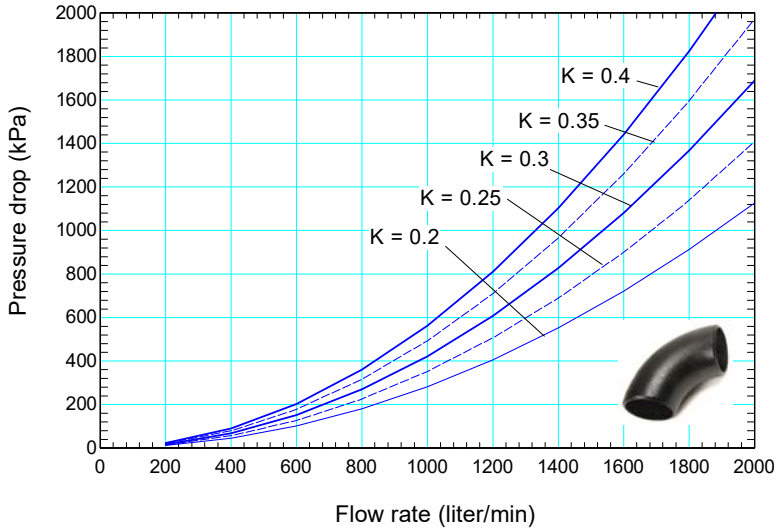


Figure 1-58: Formatted plot of pressure loss as a function of flow rate through a 45º elbow for various values of the resistance coefficient.

The Cross-Hairs, Move, and Zoom Tools

The cross-hairs tool brings up a status bar at the bottom of the plot showing the x and y coordinates of the cursor, as shown in Figure 1-59. See Chapter 9 for a discussion of additional capabilities of the cross-hairs tool.

The plot is normally locked in position in the plot window. However, it can be moved using the Move plot tool (it can also be moved if the Ctrl key is held down). This tool is particularly useful if the plot is located too close to the left or top of the plot window.

The Zoom tool allows a rectangular region of the plot to be selected, as shown in Figure 1-60(left). A new plot is subsequently created, as shown in Figure 1-60(right), which includes the selected region and therefore provides a zoomed-in view of a certain region of the original plot.

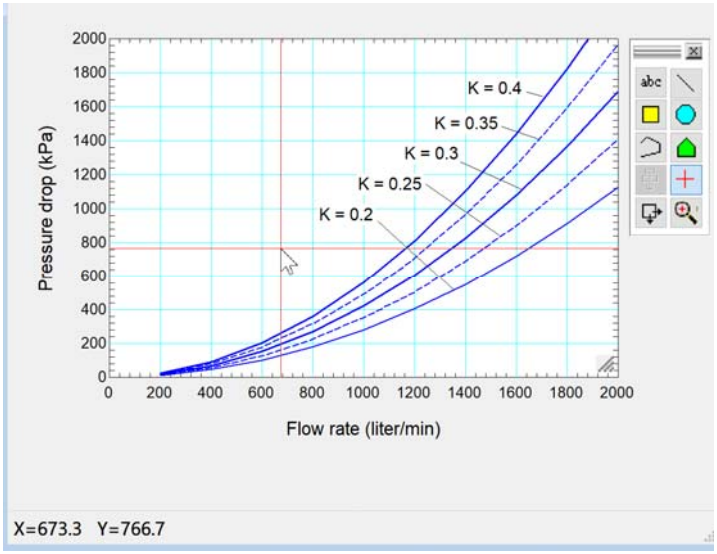


Figure 1-59: Plot window with cross hairs and status bar (at bottom).

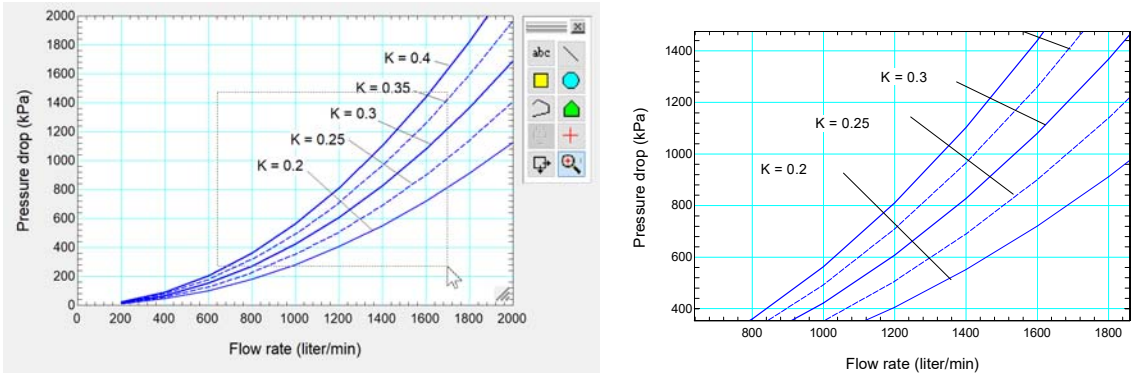


Figure 1-60: (left) Plot with the zoom feature activated and (right) new plot with zoomed in region.

1.5 Units

In Section 1.3 we calculated the pressure loss through a 45º elbow. During the process of solving the problem we found that it was necessary to keep track of and convert between units. Variables in engineering problems typically represent physical quantities and therefore they will have units. EES allows you to assign not only a value to each variable, but also its units. This is an incredibly useful feature of EES because the unit consistency of the equations will be examined by EES and any unit conversion errors will be flagged. Further, EES facilitates the process of converting between units by including a large number of built-in unit conversion factors. EES can also display variables in dual units.

Unit System

Units are commonly categorized as belonging to the English or SI (Systems International) unit system. If you are accessing any of the built-in thermodynamic or transport property functions (discussed in Chapter 4) or any of the trigonometric functions in EES, then it is important that you specify the units system that you will be using for these functions. This is done by selecting Unit System from the Options menu to access the Unit System tab of the Preferences dialog, shown in Figure 1-61. You can select which unit system (English or SI) you will be using.

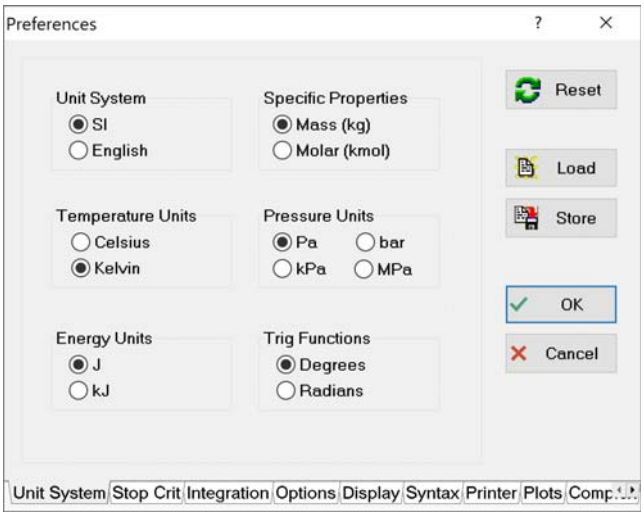


Figure 1-61: Unit System dialog.

Entering Units

Let's return to the example from Section 1.3. The inner diameter of the elbow is $D = 2$ cm and the density of water is $\rho = 1000$ kg/m^3. The dimensionless resistance coefficient is estimated to be $K = 0.3$, allowing the pressure loss can be estimated according to:

$$\Delta P = K \frac{\rho u^2}{2} \tag{1-14}$$

where u is the mean velocity of the water. We will compute the pressure loss when the volumetric flow rate is $\dot{V} = 100$ liter/min. The inputs are entered in EES:

```
$TabStops 0.2 2.5 in

"Inputs"
K=0.3                    "K factor for a 45 degree elbow, dimensionless"
D=2                      "diameter"
rho=1000                 "density"
V_dot=100                "flow rate"
```

Solve the equations and examine the Solution Window, shown in Figure 1-62(left). The value of each variable is indicated, but no units have been set.

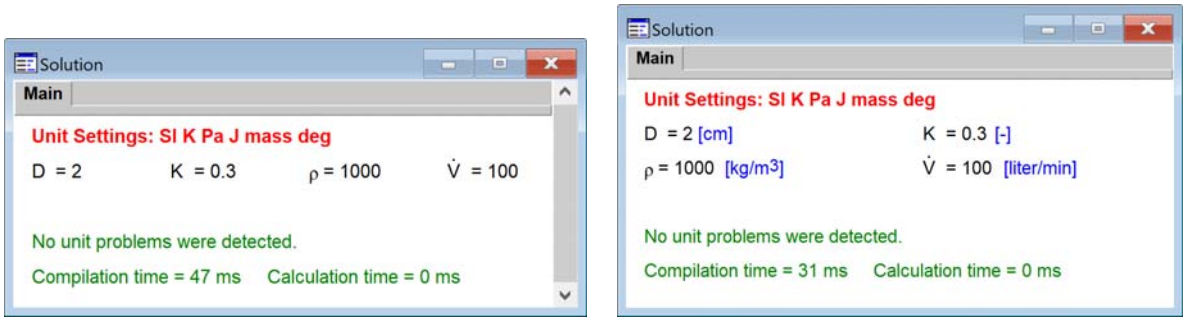


Figure 1-62: Solutions Window showing input parameters (left) with no units set and (right) with units set.

There are several ways to set the units (and alternate units) of each variable. The units of a numerical constant (e.g., 2 or 1000) can be set directly in the Equations Window by placing the unit within square brackets directly after the value. The units of each input variable can therefore be set as they are entered.

```
$TabStops 0.2 2.5 in

"Inputs"
K=0.3 [-]                "K factor for a 45 degree elbow, dimensionless"
D=2 [cm]                 "diameter"
rho=1000 [kg/m^3]        "density"
V_dot=100 [liter/min]    "flow rate"
```

The resulting Solutions Window is shown in Figure 1-62(right). Both the value and the units for each variable are indicated. Notice that using the hyphen alone as a unit specification indicates that the variable K is dimensionless (i.e., it has no units). A space could be used in place of the hyphen. The units for each variable can also be set by right-clicking on the variable in the Solution Window in order to access the Specify Format and Units dialog, shown in Figure 1-63 for the variable D.

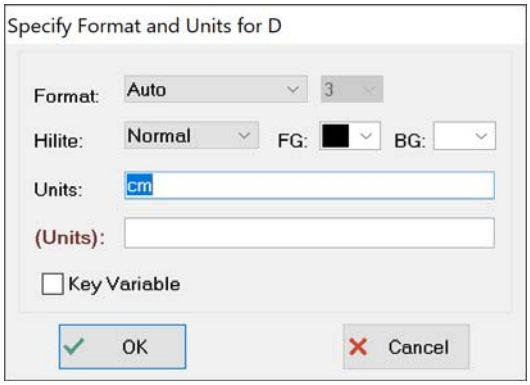


Figure 1-63: Specify Format and Units dialog.

It is also possible to set the units of a variable directly from the Equations window. This is done by clicking on the variable name in the Variable Information palette located to the right of the window. This action will bring up the Variable Information dialog for the selected variable.

Finally, the units for each variable can be set in the Variable Information dialog that is accessed by selecting Variable Info from the Options menu (or by pressing F9 or pressing the Variable Information shortcut key).

Dual Units

Note that two fields are provided for units in the Format and Units dialog shown in Figure 1-63. If a second (alternate) set of units is provided, it must have the same dimensions as the primary set of units. In this case, EES will display the variable in the Solution Window in both sets of units. However, only the value of the variable in its primary units are used in calculations. For example, we can enter in (for inches) in the alternate units field for the variable D, as seen in Figure 1-64(left). Click the OK button and the Solution Window will appear with the value of D shown both in cm and in inches, Figure 1-64(right). EES uses its internal unit conversion routines to determine the value (shown in brown font) of the variable in the second set of units.

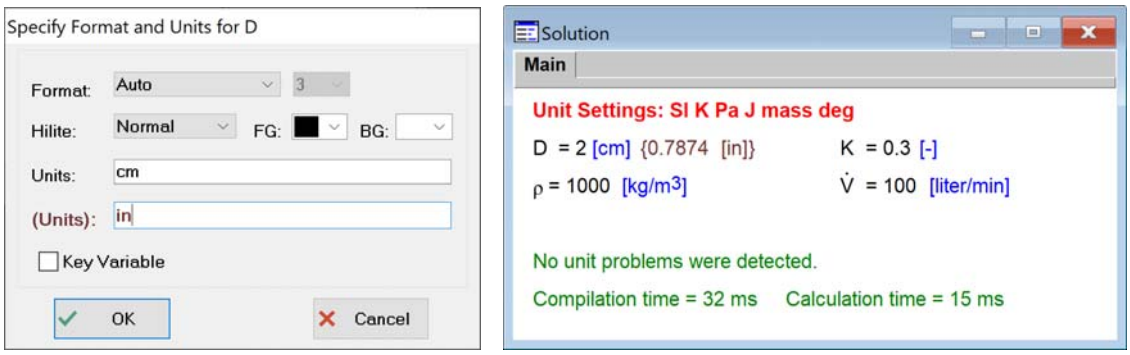


Figure 1-64: (left) Entering a second set of units in the Specify Format and Units dialog and (right) the resulting display in the Solutions Window.

Dual units can be used for variables that are stored in Parametric, Arrays, Lookup, and Integral tables as well. Notice that a field is provided for alternate units in the Format Table dialog shown in Figure 1-40 and again in Figure 1-65 (left). Dual units can be used to create plots with

scales for both unit systems as discussed in Section 9.1. Note that the Format Table dialog includes a check box control named 'Show values in alternate units'. If an alternate set of units is provided with the same dimensions as the primary set and this check box is selected, values will be displayed in both units, as seen in Figure 1-65 (right).

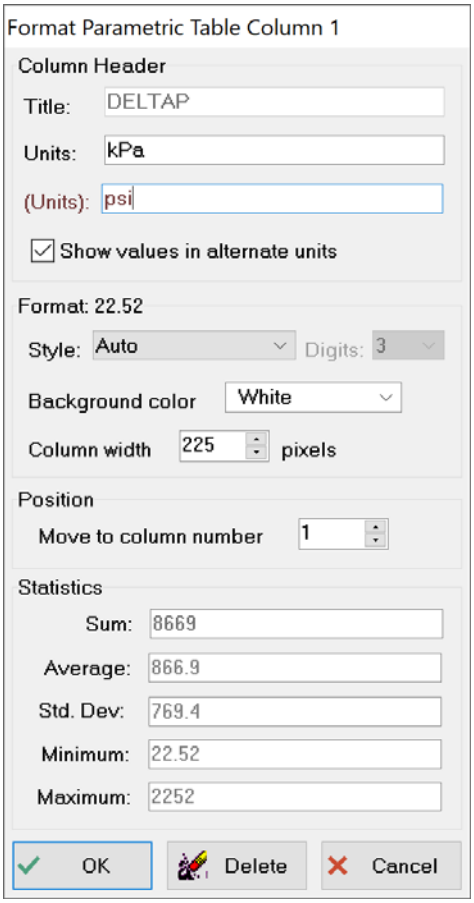


Table 1

1..10	1 ΔP [kPa] {[psi]}	2 V [liter/min]
Run 1	22.52 {3.266}	200
Run 2	90.06 {13.06}	400
Run 3	202.6 {29.39}	600
Run 4	360.3 {52.25}	800
Run 5	562.9 {81.64}	1000
Run 6	810.6 {117.6}	1200
Run 7	1103 {160}	1400
Run 8	1441 {209}	1600
Run 9	1824 {264.5}	1800
Run 10	2252 {326.6}	2000

Figure 1-65: (left) Format Parametric Table dialog and (right) Parametric table showing values displayed in dual units.

The Professional license allows variables to be plotted in their primary or alternate units. See Section 9.1 for an example of this capability.

Units Recognized by EES

The units that EES recognizes can be examined by selecting Unit Conversion Info from the Options menu in order to bring up the Unit Conversion Information dialog, shown in Figure 1-65. The left window lists each dimension (e.g., length) while the right window lists the built in units that are recognized for the selected dimension (e.g., m and ft). If you select two units in the right window, then EES will display the unit conversion in red at the bottom of the dialog, as shown in Figure 1-66 for m and ft.

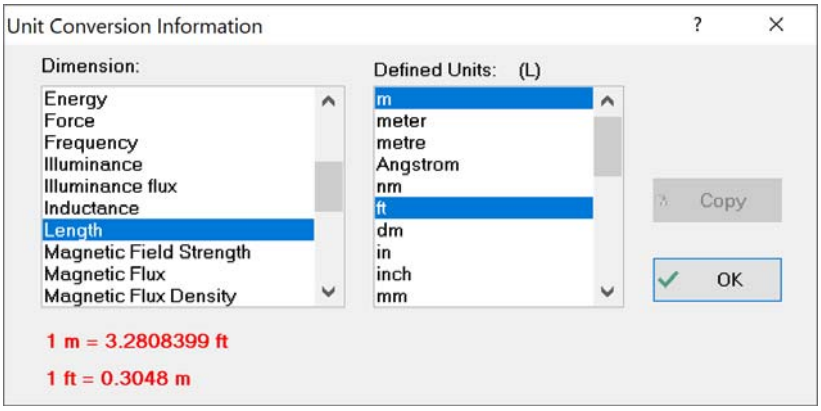


Figure 1-66: Unit Conversion Information dialog.

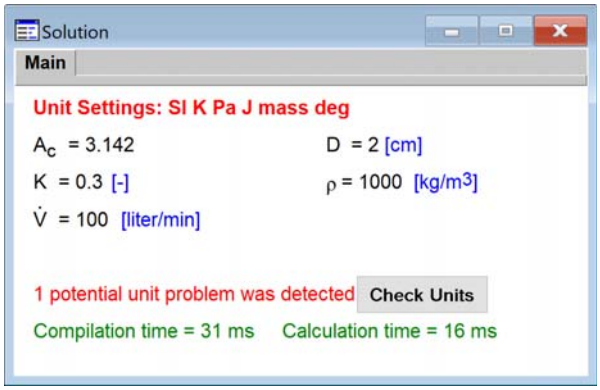
Checking Units

The cross sectional area of the elbow is given by:

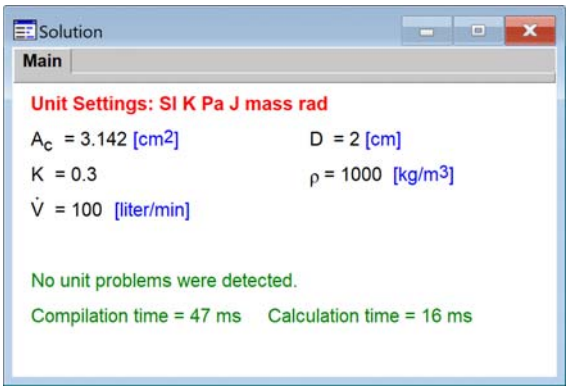
$$A_c = \pi \frac{D^2}{4} \tag{1-15}$$

```
A_c=pi*D^2/4                    "cross-sectional area for flow, in cm^2"
```

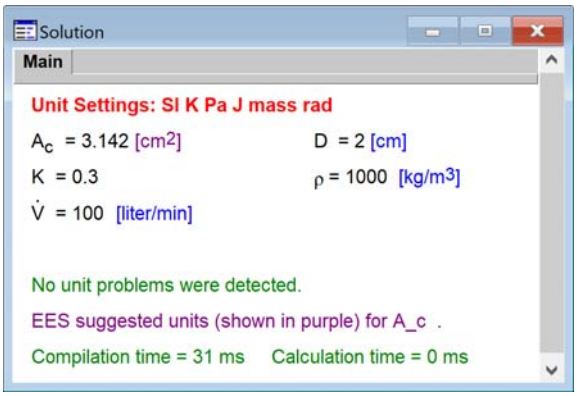
By default, EES will not automatically set the units of A_c. Solve the problem and access the Solution Window, shown in Figure 1-67(a). Notice that there is no unit listed after the variable A_c and also that a red warning message has appeared at the bottom of the Solution Window indicating that EES has detected a potential problem with the unit consistency of the equations.



(a)



(b)



(c)

Figure 1-67: Solution Window with (a) the units of A_c not set, (b) the units of A_c set by the user, and (c) the units of A_c automatically set by EES.

Select Check Units in order to bring up the Check Units window shown in Figure 1-68. EES has identified that Eq. (1-15) is not dimensionally consistent if A_c is dimensionless and D has units cm. It is evident that the units of the variable A_c must be cm^2.

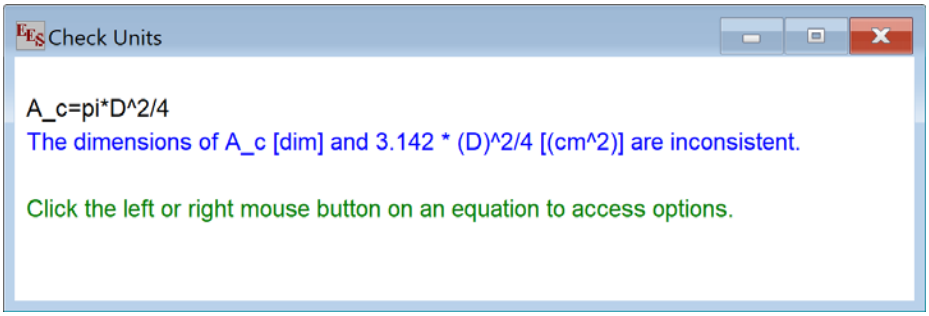


Figure 1-68: Check Units Window.

Clicking the left or right mouse button on an equation that has unit issues will bring up the pop-up menu in Figure 1-69.

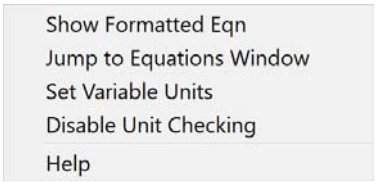


Figure 1-69: Options in the Unit Checking Pop-up Menu.

The first menu option will display the formatted equation showing the units of all variables and constants. This display makes it easy to identify unit problems. The second option will move the focus to the equation in the Equations window. The third option opens the Variable Information dialog where the units of one or more variables in the equation can be specified. The last option disables unit checking for the selected equation (they can be re-enabled in the Residuals Window, discussed in Section 5.2.

By default, EES automatically performs this type of unit check each time the system of equations is solved. Although not recommended, this option can be deactivated by selecting Preferences from the Options menu and selecting the Options tab to access the dialog shown in Figure 1-70. The Check Units Automatically check box can be selected or de-selected. Automatic unit checking can also be controlled with the $CheckUnits directive. Placing the directive $CheckUnits AutoOn at the top of the Equations Window activates automatic unit checking and the directive $CheckUnits AutoOff deactivates it. The $CheckUnits Off and $CheckUnits On directives can be placed before and after, respectively, a set of equations in order to locally deactivate the unit checking for these equations, as discussed in Section 14.2. The unit consistency of the equation set can also be checked at any time by selecting Check Units from the Calculate menu (or by selecting F8).

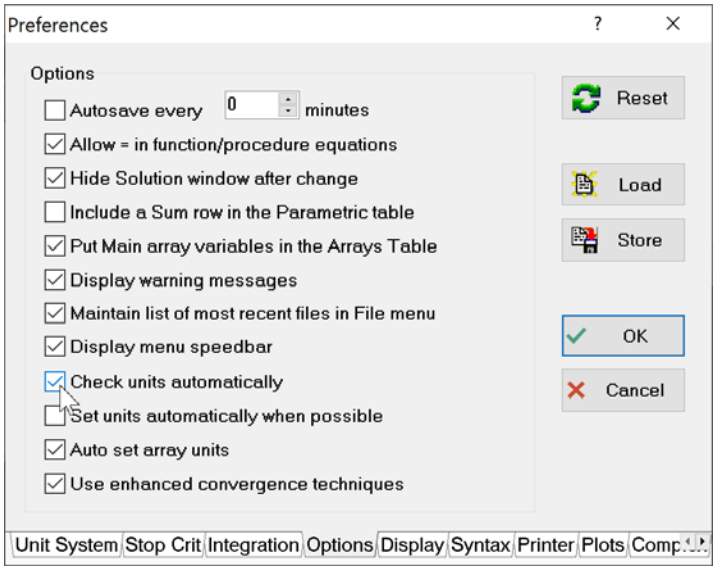


Figure 1-70: Options dialog.

It is not possible to set the units for the variable A_c in the Equations Window using square brackets because the right side of Eq. (1-15) is not a numerical constant. For example, typing:

```
A_c=pi*D^2/4 [cm^2]          "cross-sectional area for flow"
```

will result in EES interpreting the constant 4 in the denominator as having units cm^2. Alternatively, typing:

```
A_c [cm^2] =pi*D^2/4         "cross-sectional area for flow"
```

will result in EES interpreting the square bracket as the subscript in an array (discussed in Section 1.7), leading to an error message. Rather, the units of the variable A_c must be set either by right-clicking on the variable in the Solutions Window and entering the units in the resulting dialog or by using the Variable Information window. The result is shown in Figure 1-67(b). Note that the red unit warning is now gone.

Automatically Setting Units

It is also possible to have EES attempt to discern the appropriate units for each variable based on the equation in which it is used and then set these units automatically. This is not recommended as EES is no longer really checking the unit consistency of your equations and variables but rather forcing unit consistency, if possible, by adjusting the units of the variables. To activate this option, remove the unit specification for the variable A_c and select Set units automatically when possible from the Options tab of the Preferences dialog, shown in Figure 1-70. Solve the equation set to obtain the Solution Window shown in Figure 1-67(c). The units for the variable A_c have been automatically set to cm^2. The units suggested by EES are automatically indicated in purple to highlight the fact that these are not user-specified units. All units shown in purple should be confirmed by the user by right-clicking on these variables in the Solutions window and then making any necessary changes in the dialog window that appears.

Automatic unit setting can also be activated by placing the $AutoSetUnits On directive at the top of the EES Equations Window. The directive $AutoSetUnits Off deactivates this option.

The Convert Function

Let's continue to solve the example problem. The velocity of the water is calculated according to:

$$u = \frac{\dot{V}}{A_c} \tag{1-16}$$

```
u=V_dot/A_c                    "velocity"
```

As written, the units for the variable u must be liter/min (the units for the variable V_dot) divided by cm^2 (the units for the variable A_c). Therefore, the units for u must liter/min-cm^2. (Again, note that in this book and also in EES, the hyphen sign will be used to indicate multiplication on one side of the divisor in the context of units). If we set the units for u to liter/min-cm^2 in the Solutions Window, then EES will find no unit inconsistencies. Of course, the units liter/min-cm^2 don't make a lot of sense for a velocity and it would be better to convert the value to a more reasonable set of units, like cm/min. We could do this manually, looking up each of the various unit conversion factors that are required to arrive at the conclusion that we need to multiply by 1000:

$$\frac{\text{liter}}{\text{min-cm}^2}\left\|\frac{1000\ \text{cm}^3}{1\ \text{liter}} = \frac{\text{cm}}{\text{min}}\right. \tag{1-17}$$

Fortunately, EES has built-in unit conversion information that is easily accessible using the convert function. The convert function simply retrieves the conversion factor between two units and therefore the function requires two arguments; the first argument is the unit to convert *from* while the second is the unit to convert *to*. By adding the convert function to the equation calculating the velocity:

```
u=V_dot/A_c*convert(liter/min-cm^2,cm/min)      "velocity"
```

the unit conversion is easy and transparent to the user. The convert function returned the correct unit conversion multiplier (1000) for the unit conversion that is specified by the two arguments. This is particularly evident when the equation is viewed in the Formatted Equations Window, shown in Figure 1-71.

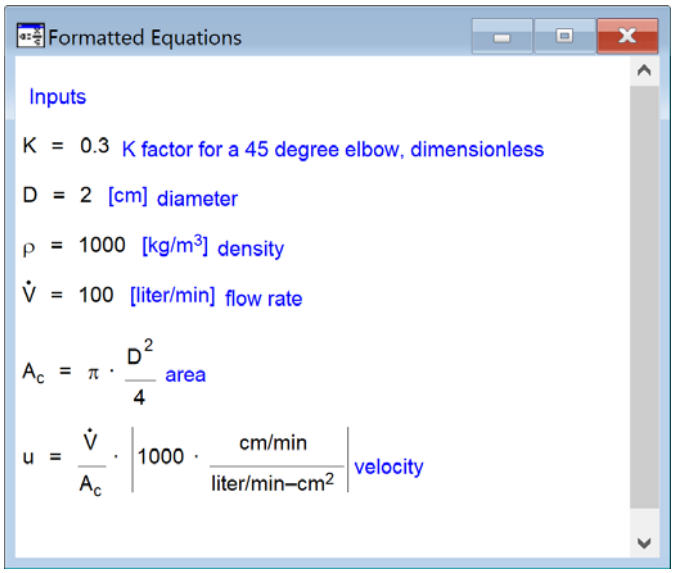


Figure 1-71: Formatted Equations Window.

The units of the variable u should now correctly be set as cm/min in the Solutions Window or in the Variable Information dialog. Finally, Eq. (1-11) is used to compute the pressure loss and the result is converted to kPa using the convert function:

```
DELTAP=K*rho*u^2/2*convert(kg-cm^2/m^3-min^2,kPa)          "pressure loss"
```

The units of the variable DELTAP can be set to kPa in the Solutions Window. The unit assignments of each variable will automatically be stored and the units will be indicated whenever you use the variables, such in a Parametric Table or plot.

If you set a variable to a numerical constant followed by its units in brackets, EES will automatically assign the units to the variable. Starting with version 9.437, EES will also automatically assign the units to variables that have been set to a value using the convert command. For example, the following example will set the value of the variable L to 12 and its units associated with the variable L to ft.

```
L=12 [in] * convert(in, ft)
```

The Units List

The Units List provides a convenient shortcut for entering units that are used often. The units list is activated by right-clicking in any of the areas where you could otherwise manually enter units; this includes the Equations Window, the Units field in the Specify Format and Units dialog in the Solution Window, or the Units column of the Variable Information Window. Select Units List to

access a list of commonly used units, as shown in Figure 1-72. The basic set of units in the list will depend on whether your unit system is set to English or SI (see Figure 1-61). You can click on the desired unit and select Paste or double click on the unit in order to enter it.

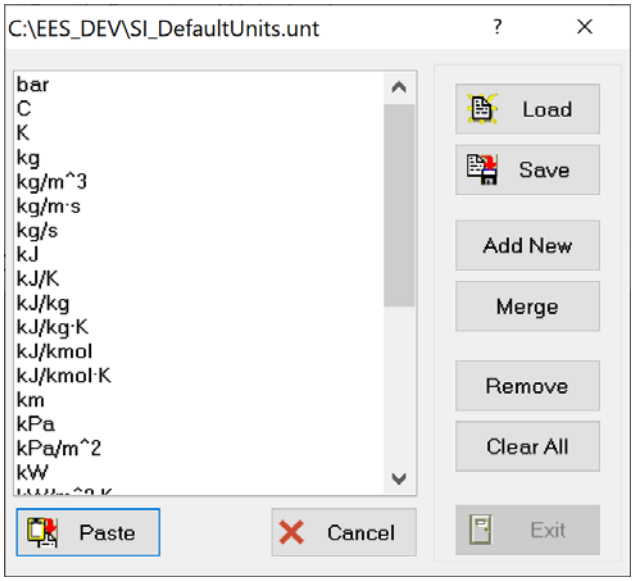
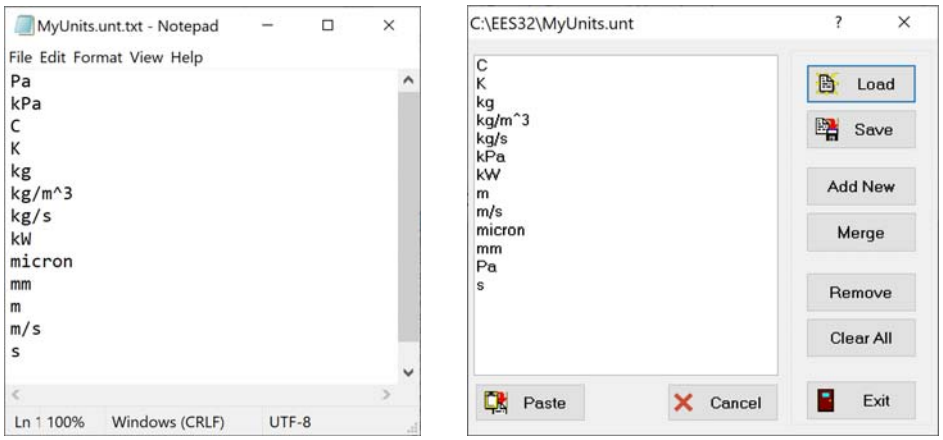


Figure 1-72: The default SI Units List.

The unit list can be modified so that it is most useful to you. You really want to include only a small number of units that you find yourself using often in order for it to save you time. The buttons to the right facilitate this process. Units can be added or removed with the Add New and Remove buttons, respectively. The entire list can be cleared with the Clear button. If you change the units list and want that change to be recorded beyond the current EES session then you should save the new list setup using the Save button.

EES will automatically load the default SI and English units lists (SI_DefaultUnits.unt and Eng_DefaultUnits.unt, respectively) upon opening. You may have your own, separate units list that you have developed and would like to use. This is easy to do because the units list is simply a text file that can be opened and edited with any text editor (e.g., Notepad®). For example, Figure 1-73(a) illustrates a text file that is saved as MyUnits.unt. This units list can be loaded using the Load button in Figure 1-71 and then used for the duration of the EES session, as shown in Figure 1-73(b). Alternatively, a custom units list can be merged with an existing list using the Merge button.



(a) **(b)**

Figure 1-73: A unit list saved as (a) MyUnits.unt file and (b) loaded into the Units List in EES.

If you don't want to load your custom units list each time you start EES then you can use the $Include directive. The $Include directive provides an automatic method of loading a variety of files when you run your EES code. Place the statement:

```
$Include MyUnits.unt
```

at the top of the EES file and you will find that the units list is automatically set as shown in Figure 1-73(b). Note that regardless of how the units in the units list are organized, they are simply a subset of the larger set of units that are recognized by EES.

Suggested Method for Working with Units

Every engineer should have his or her own, carefully thought out method for dealing with units and unit conversions. Unit conversion errors are one of the most common mistakes that engineers make and it is clear that the unit checking and unit conversion features of EES will help to avoid these problems. The authors have developed a specific procedure for solving engineering problems that makes unit consistency easy to maintain. Inputs to an engineering problem are encountered in arbitrary and mixed units. The first step in solving a problem is therefore to convert all inputs to the base SI system (i.e., kg, m, s, K, N, etc.). The calculations required to solve the problem are carried out using the base SI system and unit checking is rigorously applied in order to establish the dimensional consistency of each equation. The results of the calculations can be converted from the base SI system into whatever units are requested or are logical. Again, this procedure is convenient because the units of each variable are self-consistent in the SI unit system; as a result, it is not necessary to constantly worry about applying the correct unit conversion to each equation during the development of a model. If you take this approach and you check the units of your equations then you are actually carrying out a more powerful and complete check on your equations; you are establishing their dimensional (as well as their unit) consistency.

Applying this strategy to our pressure loss problem leads to the following equation set:

```
$TabStops 0.2 2.5 in
```

```
"Inputs - converted to base SI units"
K=0.3 [-]                                   "K factor for a 45 degree elbow, dimensionless"
D=2 [cm]*convert(cm,m)                      "diameter"
rho=1000 [kg/m^3]                           "density"
V_dot=100 [liter/min]*convert(liter/min,m^3/s)  "flow rate"

"Calculations - done in the base SI unit system"
A_c =pi*D^2/4                               "cross-sectional area for flow"
u=V_dot/A_c                                 "velocity"
DELTAP=K*rho*u^2/2                          "pressure loss"

"Results - converted to convenient units"
V_dot_lpm=V_dot*convert(m^3/s,liter/min)    "flow rate, in liter/min"
u_cmpm=u*convert(m/s,cm/min)                "velocity, in cm/min"
DELTAP_kPa=DELTAP*convert(Pa,kPa)           "pressure loss, in kPa"
```

and the Solution Window shown in Figure 1-71. While it is not necessary that you take this approach, it certainly makes solving systems of engineering equations much easier. The settings for the base SI unit system are as shown in Figure 1-61.

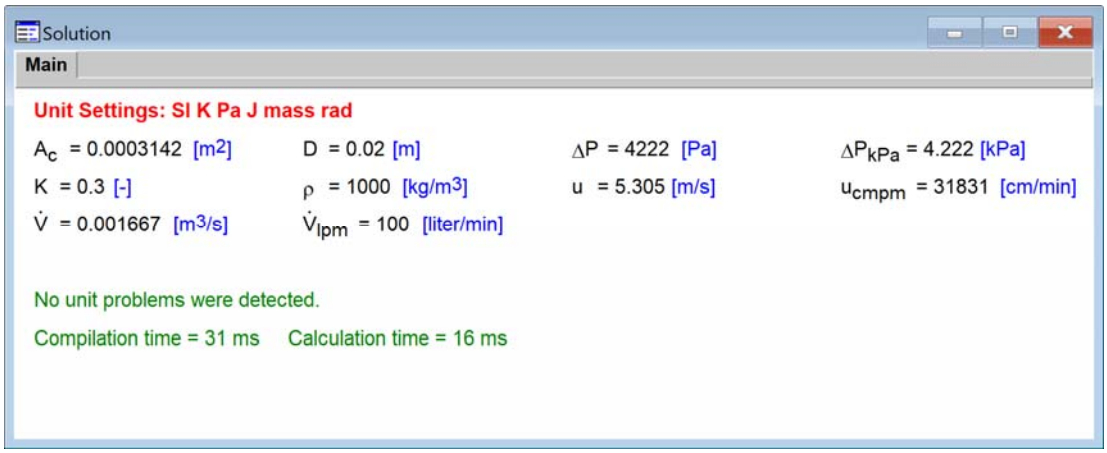


Figure 1-74: Solution Window.

Note that EES allows an alternate set of units to be specified for each variable, as shown in Figure 1-63. The last three lines of the program are not needed if alternate units of liter/min, cm/min, and kPa are provided for variables V_dot, u, and DELTAP, respectively. The Solution window with these alternate units set appears as shown in Figure 1-75.

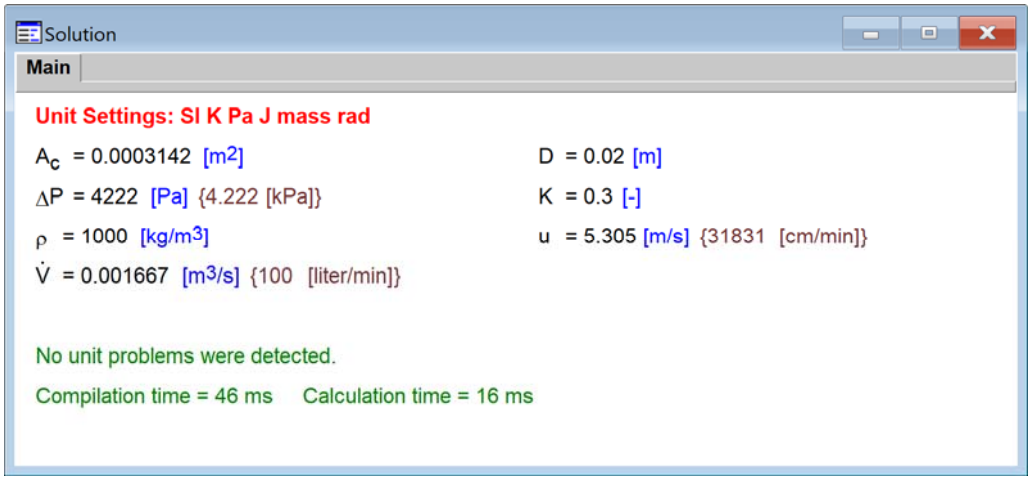


Figure 1-75: Solution Window using alternate units for the variables V_dot, u, and DELTAP.

The ConvertTemp Function

Most unit conversions require only a multiplicative correction that can be provided by the convert function in EES. However, both multiplication and addition operations are required to convert between temperature scales. Therefore, the convert function in EES cannot be used to convert a value from one temperature scale to another temperature scale. Instead, the ConvertTemp function in EES should be used for this purpose. The ConvertTemp function accepts three arguments. The first argument indicates the temperature scale to convert *from*, the second argument indicates the temperature scale to convert *to*, and the third argument is the numerical value, variable, or expression to be converted. The temperature scale indicator can be F, R, C or K, representing the Fahrenheit, Rankine, Celsius, and Kelvin scales, respectively. For example, the following equation converts a temperature of 70°F to the Celsius temperature scale.

```
T_F=70 [F]                          "temperature in F"
T_C=converttemp(F,C,T_F)            "temperature in C"
```

Note that there can be confusion with temperature units. In some cases, you may wish to convert a temperature *difference* in Celsius to an equivalent temperature *difference* in Fahrenheit. In this case you would not use the ConvertTemp function, but instead, use the convert function, e.g., convert(C,F). For clarity, EES provides temperature difference units: DELTAF, DELTAC, DELTAR and DELTAK. For example, a variable representing a temperature difference could be assigned units of DELTAC, which will be displayed as ΔC in the Solutions Window. Use of the temperature difference units is particularly important when using the Dual units option in the Specify Format and Units dialog (Figure 1-63) as EES will automatically assume that a variable having units of F, C, R, or K is a temperature rather than a temperature difference.

Adding Units

EES includes several hundred units and is therefore likely to include any unit you might run across. However, it is possible to add to the list of units recognized by EES. For example, braking distances are often expressed in car lengths where 1 car = 4 m. The unit 'car' is not recognized by EES. The equation:

```
L=4 [car]*convert(car,m)                    "length, in units of car"
```

will result in the error message shown in

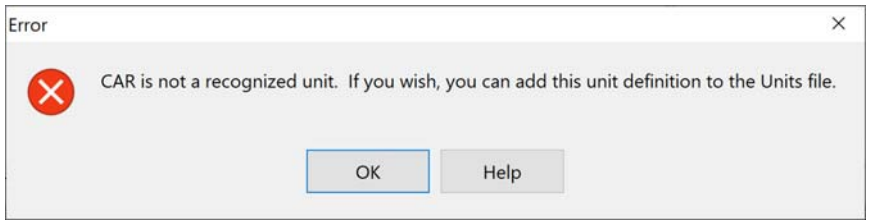


Figure 1-76: Error message.

We can add the unit car to the list of those that are recognized by EES by opening the file Units.txt found in the root EES installation directory using any convenient editor (e.g., Notepad®), as shown in Figure 1-77.

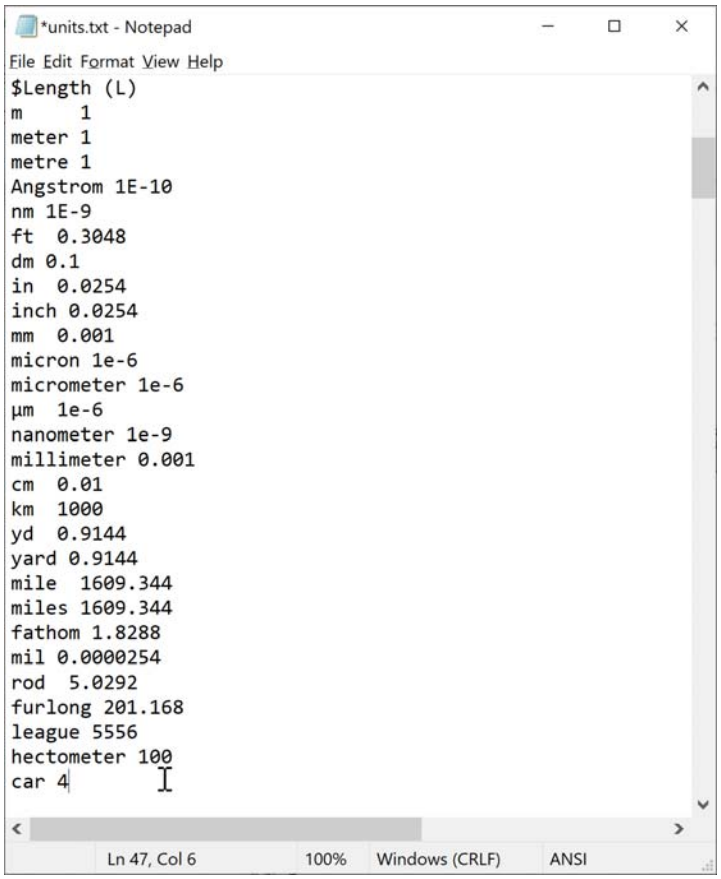


Figure 1-77: Units.txt file opened using Notepad® with the unit car added.

Instructions for adding units are found at the top of the Units.txt file. Unit conversions are organized according to dimension. Locate the dimension of interest (in this case, Length). The first entry in the unit list (in this case, m) is the basis used for conversion of all other units in that dimension. Reading down the list, this should be clear (1 Angstrom = $1x10^{-10}$ m, 1 nm = $1x10^{-9}$

m, etc.). Add the appropriate entry at the bottom of the list, as shown in Figure 1-76, and save the units.txt file. When EES is re-started it will load the new set of unit conversions and the statement:

```
L=4 [car]*convert(car,m)                    "length, in units of car"
```

will not result in an error. The units of the variable L should be set to m in the Solution Window, as shown in Figure 1-78. Note that you must de-select the Overwrite Units files selection when updating EES (see Figure 1-2) or your modified units.txt file will be overwritten when a new version of EES is installed.

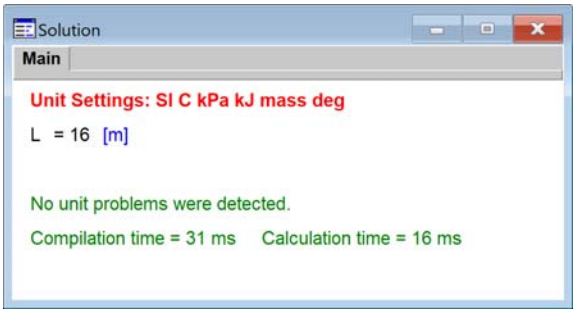


Figure 1-78: Solution Window.

Using String Variables for Units

The units of a variable can be assigned with a string variable. A string variable can be supplied in place of the unit designation within the square brackets that follow numerical constant or provided in the units column of the Variable Information dialog, as shown in Figure 1-79.

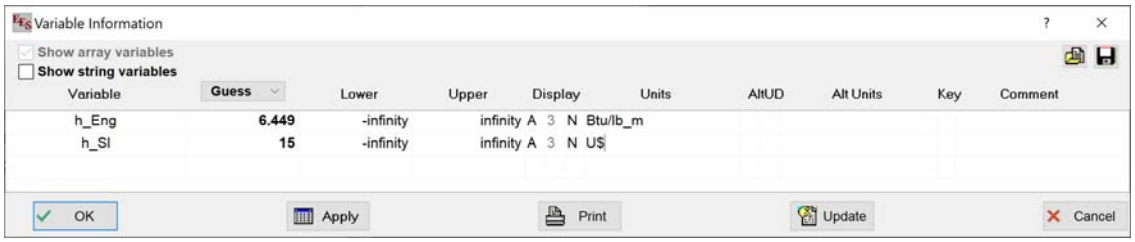


Figure 1-79: Variable Information dialog shown with the string variable U$ used to set the units variable h_SI.

A string variable can be also used in the convert function.

```
U$='kJ/kg'
h_SI=15 [U$]
h_Eng=h_SI*convert(U$, Btu/lb_m)
```

Note that providing alternate units in the Variable Information is an easy way to show results in two sets of units. See Section 15.6 for an example using string variables to allow either English or SI units to be used for a problem that involves the Diagram window.

EXAMPLE 1.1 Oxygen Dewar

This EES example illustrates the process of solving equations, using a Parametric Table, setting units, and making plots. The example can be accessed from the Examples menu in the Mastering EES folder and includes a detailed .pdf file that describes the development of the code.

1.6 Printing

Once you have completed your EES solution, you will need to document the results in some way. This section discusses how you can output information from the EES file.

Printing a Hard Copy

The most direct method of output is simply to print a hard copy of your file to an installed printer. To do this, select Print from the File menu to bring up the Print dialog shown in Figure 1-80(left). The printer and number of copies can be selected. The Print in color option causes EES to attempt to print with the same color scheme that is used for the screen display. The Page breaks option causes EES to print each window on a different page.

Note that, by default, not all of the windows in your EES file will be printed. You need to select from among those shown along the left side of the dialog. While there is only one Equations Window, there may be more than one Parametric, Lookup, Array and/or Integral Table and more than one Plot. You will need to indicate both whether you want these items to be printed as well as which of these items are to be printed. A black check mark indicates that all of the items will be printed, an open box indicates that none will be printed, and a gray check mark indicates that a subset of the items has been selected. For example, if more than one Plot window is present in the EES code, then clicking on the Plots box will bring up the Select Plots to print dialog shown in Figure 1-80(right). Select Preview in the Print dialog to bring up an on-screen display of the printed output.

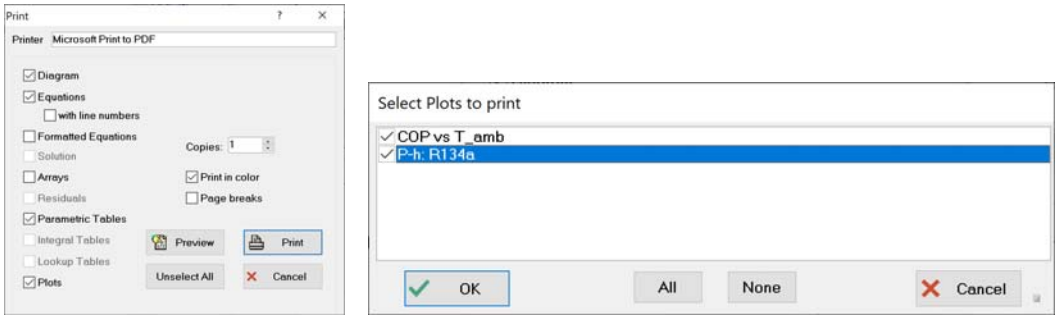


Figure 1-80: (left) Print and (right) Select Plots to print dialogs.

The Print Setup selection from the File menu allows you to set up the details of the printing process (e.g., orientation, paper size, etc.). By default, the font and style used for the printed output is the same as those used for the on-screen display. However, the Printer tab on the Preferences dialog allows you to adjust these values independently.

Copying Equations

It is also possible to copy equations from the Formatted Equations Window in EES so that they can be pasted into various applications. We will use Eq. (1-10), repeated below, as an example.

$$\bar{f} = \frac{4}{Re_D}\left[\frac{3.44}{\sqrt{L^+}} + \frac{\frac{1.25}{4L^+} + \frac{64}{4} - \frac{3.44}{\sqrt{L^+}}}{1 + \frac{0.00021}{\left(L^+\right)^2}}\right] \tag{1-18}$$

This equation is entered in the Equations Window as shown below.

```
f_bar=(4/Re_D)*(3.44/sqrt(L|plus)+(1.25/(4*L|plus)+64/4-3.44/sqrt(L|plus))/(1+0.00021/L|plus^2))
```

Open the Formatted Equations Window, shown in Figure 1-81. Highlight the equation and right click to access the pop-up menu shown in Figure 1-82.

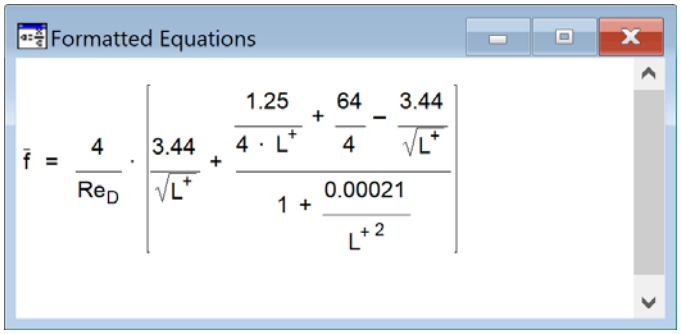


Figure 1-81: Formatted Equations Window.

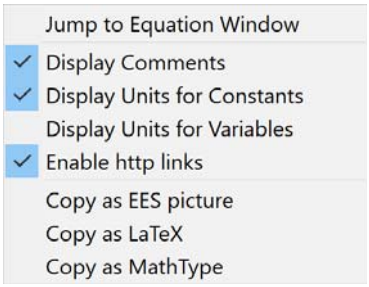


Figure 1-82: Pop-up menu for the formatted equation.

Clicking on a comment in the Formatted Equations Window that contains a hidden or visible hyperlink will automatically start your browser and open it to this page. If you do not wish to have hyperlinks active, de-select the Enable http links menu command from the Formatted Equations pop-up menu.

The three options at the bottom of the pop-up menu allow the equation to be copied and pasted in a variety of formats. If the Copy as EES picture option is selected then the equation will be

placed on the clipboard as a picture (i.e., in windows metafile format), from which it can be pasted into another application.

The next two options are only available in the Professional license. The Copy as LaTeX option translates the equation into LaTeX code that is placed on the clipboard. The Copy as MathType option translates the equation into MathType format that can subsequently be placed in another application such as Microsoft Word® provided that the MathType® software is installed.

LaTeX Report

EES can be used to create a high quality report that includes the Diagram, Equations, and Solution Windows as well as any tables and plots. The report is created in the form of a TeX document (.tex) that can be automatically processed by a LaTeX compiler to produce a device independent (.dvi) output file that can be printed by various utilities. The PDFLaTeX accessory available with the TeX compiler also produces a portable document interface (.pdf) file that is compatible with Adobe Reader. The entire process can be made transparent to the user so that the Latex Report feature effectively becomes a printer to produce .pdf files. The output quality of the Latex Report feature is much higher than what is obtained with the EES Print command.

Both the LaTeX software and Adobe Acrobat Reader (or any equivalent application) must be installed on your computer in order to use this feature. These programs are available at no cost. Download the distribution MiKTeX installation file (i.e., the latest version of the MiKTeX software) from the web page https://miktex.org/download using the instructions provided when you click the Help button in the Create LaTeX/PDF report dialog shown in Figure 1-82. Choose the Basic MiKTeX Installer option for your operating system (32 or 64-bit). Note that EES must eventually know the directory path in which the MiKTeX software is installed so you should make a note of it. In order to configure EES to work with the MiKTeX package, start EES and select Create LaTeX/PDF Report from the File menu in order to access the Create LaTeX/PDF Report dialog (Figure 1-82). Click the Setup button and navigate to the pdfLaTeX executable file (typically found in the miktex/bin subdirectory of the installation directory). If you install MiKTeX in its default directory, EES will likely find it without your intervention.

In order to generate a report, access the dialog shown in Figure 1-83 by selecting the Create LaTeX/PDF Report command in the file menu and enter the desired filename in the edit field at the top of the window. By default, the filename will be the name of the EES file with a .tex extension. The text entered in the Title field will be displayed at the top of the report. The controls at the left side of the dialog allow you to select the items that will be included in the report from among the various items in the EES file. If multiple items of the same type exist then the value within the square brackets indicates the number of items of this type to be printed. For example, there are two plots in the EES file for Figure 1-83 but only one of these is to be printed; clicking on the value within the square brackets will bring up a dialog that allows you to select a subset of the items. The Equations Window selection provides some additional controls such as line numbering and the ability to place each equation and comment on separate lines. Because very long equations may run off of the right side of the report it might be necessary to use the & character to break long equations in the Equations Window. The Solutions and Table group boxes each provide a control that allows you to specify the number of columns per page to

ensure that the text does not run off the page of the report. The diagrams and plots that are to be included with the LaTeX document are stored in separate jpeg files. The resolution of these files can be controlled using the JPG quality control.

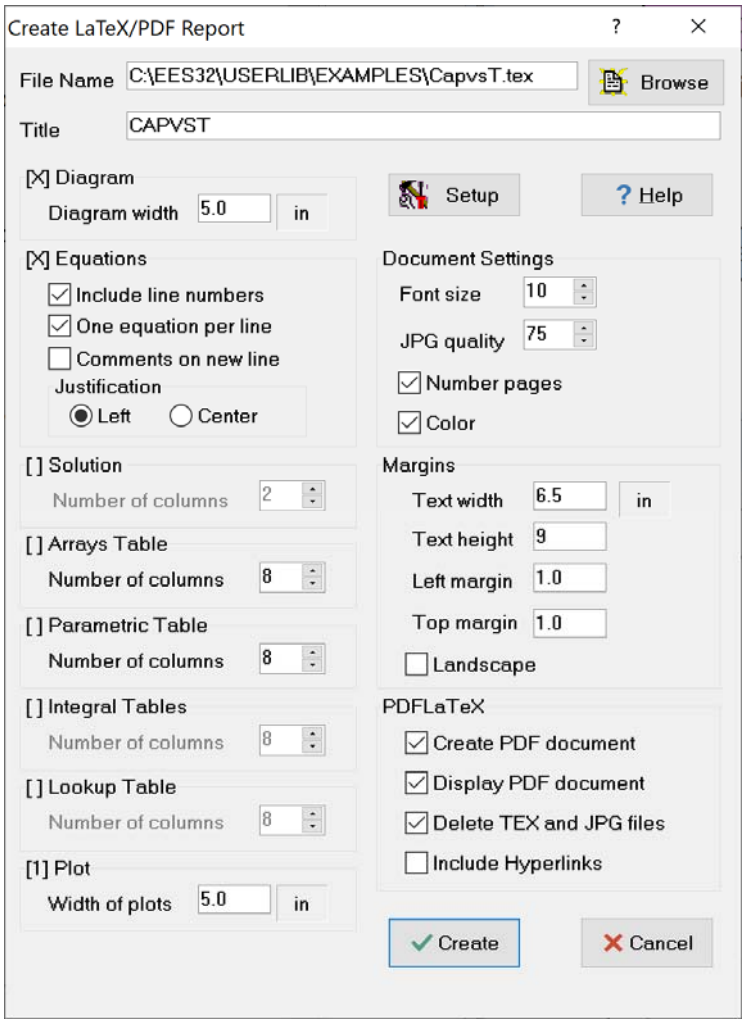


Figure 1-83: Create LaTeX/PDF Report dialog.

At the bottom right of the dialog there are four checkboxes. If the Create PDF document checkbox is selected then EES will attempt to start the PDFLaTeX application and compile the .tex file in order to create a .pdf file. If the Display PDF document checkbox is selected then EES will attempt to start Adobe Reader and display the .pdf file directly after it is created. If the Delete TEX and JPG files checkbox is selected then the .tex file and .jpg files will be deleted, leaving only the .pdf file. The Include Hyperlinks check box will only be enabled if you are using a Professional verson and there are hidden hyperlinks within the Equations window. Clicking this control will enable the hidden hyperlinks within the .pdf document.

Select Create to start the process of generating and compiling the report.

1.7 Arrays

The variables that have been discussed in Sections 1.1 through 1.6 are all scalar variables; that is, they can be assigned only one value. It will often be useful to use array variables, which can be assigned multiple values. Arrays have an array index enclosed in square brackets at the end of the variable name. Data stored in arrays can be used to generate plots in the same way that data stored in Parametric Tables can be used for this purpose, as discussed in Section 1.4.

Assigning Array Variables

The variable T_scalar, assigned below, has one value.

```
T_scalar = 100 [K]                    "a scalar variable"
```

The value and units of the variable T_scalar can be examined in the Solution Window. The array T_array[], assigned below, consists of three separate elements: T_array[1], T_array[2], and T_array[3]. The value in subscripts is the index of the array; therefore, T_array[1] is the first element of the array, and so on.

```
T_array[1]=100 [K]                    "the first element of the array T_array"
T_array[2]=200 [K]                    "the second element of the array T_array"
T_array[3]=300 [K]                    "the third element of the array T_array"
```

The values in the variable T_array[] will be stored in the Arrays Table Window, shown in Figure 1-84 and accessed by selecting Arrays from the Windows menu.

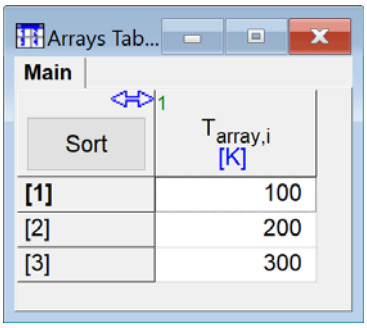


Figure 1-84: Arrays Table Window.

The array elements T_array[1], T_array[2], and T_array[3] each behave just as ordinary EES variables do. They each have units, limits, guess values, etc. One purpose of using arrays is to organize information; for example, each array element may correspond to a thermodynamic state in a system. Also, mathematical operations that are common to each element of an array can be automated easily using the Duplicate command, discussed subsequently.

A string array variable must have the $ character just before the left bracket that holds the array index. For example:

```
A$[1]='Hello'
A$[2]='World'
```

Array Range Notation

A convenient shorthand method for referring to multiple array values uses the array range notation. A continuous range of indices in an array can be referred to by the first index and last index separated by two dots. For example, T_array[1..3] can be used in place of T_array[1], T_array[2], T_array[3]. Array range notation can be used in a variety of contexts including assignment statements. For example, the array T_array[] could be created using the statement:

```
T_array[1..3]=[100 [K], 200 [K], 300 [K]]        "assignment using array notation"
```

An alternative to the above assignment statement is to append the units to the end of the statement, i.e.,

```
T_array[1..3]=[100, 200, 300]  [K]               "assignment using array notation"
```

Note that EES expects the correct number of values (three in the example above) to exist within the square brackets on the right side of the equation above or it will post an error. If more than three values were provided in the square brackets on the right side of the equation then the first three would be used and EES would post a warning.

Two-Dimensional Arrays

If two indices are used for array assignments then the array will be two-dimensional (i.e., a matrix rather than a vector). For example, the statement below:

```
T_array[1,1..3]=[100, 200, 300]  [K]             "first row of array"
T_array[2,1..3]=[400, 500, 600]  [K]             "second row of array"
```

results in the two-dimensional array shown in Figure 1-85.

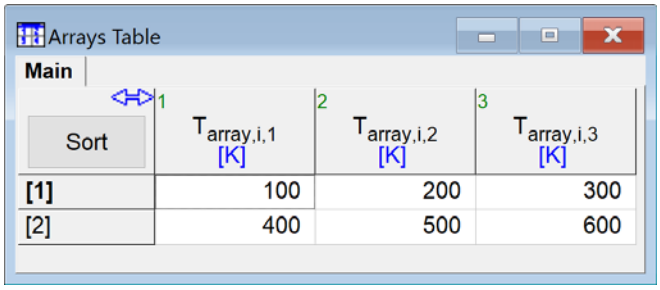


Figure 1-85: Two-dimensional array shown in the Arrays Table Window.

The statement below assigns the value in the 2nd row (the first subscript is 2) and 3rd column (the second subscript is 3) to the variable X. Examining the Solution Window shows that the scalar variable X is equal to 600 K.

```
X=T_array[2,3]                                   "value in 2nd row and 3rd column"
```

The Array Editor

The Array Editor can be selected from the Edit menu in order to provide an easy way to enter or modify EES arrays. Ultimately arrays that are recognized or created by the Array Editor are listed in the Equations Window. In order to be recognized by the Array Editor the array definition must begin with the comment {Array Name} and end with the comment {Array Name end} where Name is the name of the array. For example, the statement below:

```
{Array a}
a[1,1..3]=[1,2,3]
a[2,1..3]=[4,5,6]
{Array a end}
```

would define the array a in a way that is recognized by the Array Editor. When the Array Editor is opened, the dropdown menu at the top of the editor can be used to select from among any of the arrays that have currently been defined in the Equations Window using this syntax, as shown in Figure 1-86. Note that any array defined in the Array editor itself will also use this syntax. A new array variable can be created by selecting Enter array name from the Array drop down menu.

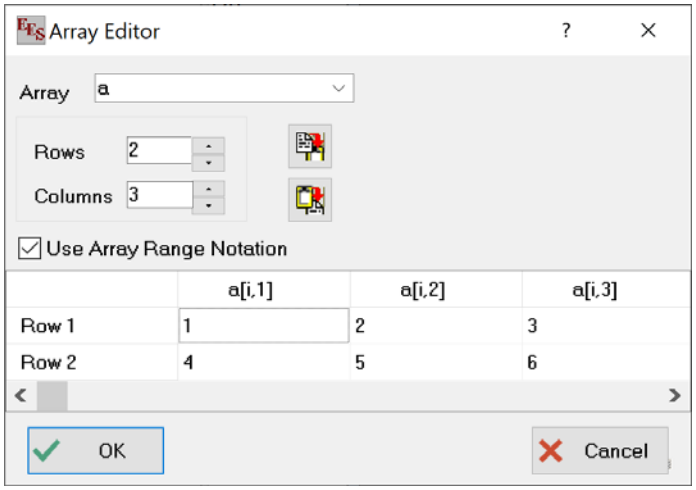


Figure 1-86: Array editor.

Modifications to existing arrays made in the Array Editor will show up in the Equations Window. (Note that the Array Editor will not detect or be able to modify arrays that were not created in the Array Editor.) This feature makes it easy to import data from other applications such as Excel or MATLAB. For example, suppose you want to access a 5x5 array of data generated in Excel, as shown in Figure 1-87.

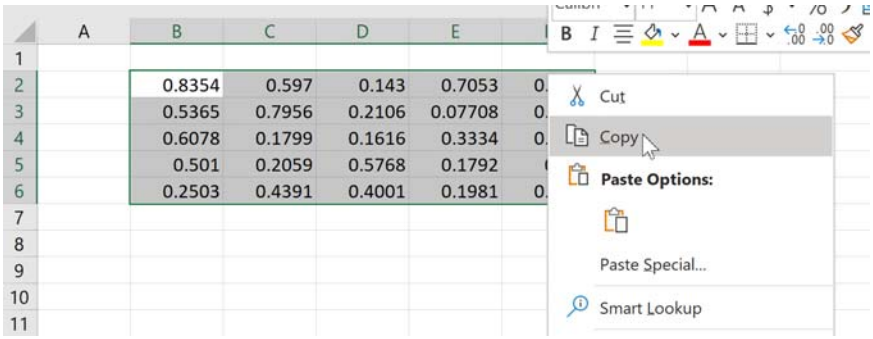


Figure 1-87: An array of data in Excel.

Select Array Editor from the Edit menu. Select Enter Array Name from the drop down menu and then name the array D and choose the right number of rows and columns (5 and 5). Copy the data from Excel and select the Paste button ( ) to place the data into the Arrays Editor, as shown in Figure 1-88.

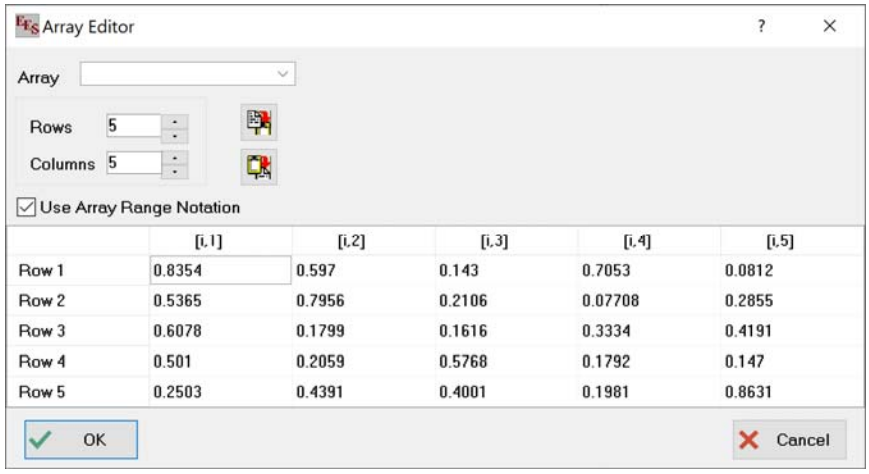


Figure 1-88: Arrays Editor.

Select OK and the array D[] will be created in the Equations Window.

```
{Array D}
D[1..5,1]=[0.835379821,0.536458607,0.607826536,0.50104002,0.250295049]
D[1..5,2]=[0.596997554,0.795630883,0.179900142,0.205868913,0.439109551]
D[1..5,3]=[0.142993304,0.210596416,0.161575547,0.576841329,0.4000961]
D[1..5,4]=[0.70532211,0.077083658,0.333396857,0.179171305,0.198072072]
D[1..5,5]=[0.081197923,0.285540063,0.419114502,0.147024152,0.863134795]
{Array D end}
```

Run the EES program and you will see the array D[] created in the Arrays Table, as shown in Figure 1-89. Again, do not modify or delete the comments that EES places around the array values, as these are used to identify the array to the Array editor.

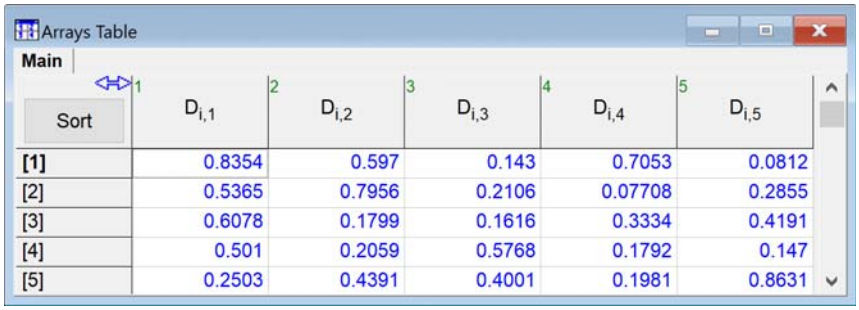


Figure 1-89: Arrays Table.

The Duplicate Statement

The Duplicate statement provides a convenient method for carrying out similar operations on multiple elements of an array. The protocol for a Duplicate statement is:

```
Duplicate index = start,stop
    equation(s) to be duplicated
End
```

The variable index is iterated from the value start to the value stop. The equation(s) contained between the Duplicate and End statements are written for each iteration. For example, the EES code:

```
duplicate i=1,4
    T[i]=i
end
```

is equivalent to writing:

```
T[1]=1
T[2]=2
T[3]=3
T[4]=4
```

When using Duplicate statements it is particularly important to remember that EES uses equations rather than assignments. That is, do not accidentally place equations within a Duplicate statement unless you actually want these equations to be duplicated in the Equations Window. A common mistake that new users of EES make is shown below:

```
duplicate i=1,4
    T[i]=i
    x=4
end
```

The above code is equivalent to writing:

```
T[1]=1
x=4
T[2]=2
```

```
x=4
T[3]=3
x=4
T[4]=4
x=4
```

which leads to 8 equations in 5 unknowns. Solving the EES code will lead to an error message.

Nested Duplicate statements can be used to generate two-dimensional arrays. For example, the code below:

```
duplicate i=1,3
    duplicate j=1,4
        D[i,j]=i+j
    end
end
```

leads to the two-dimensional array shown in Figure 1-90.

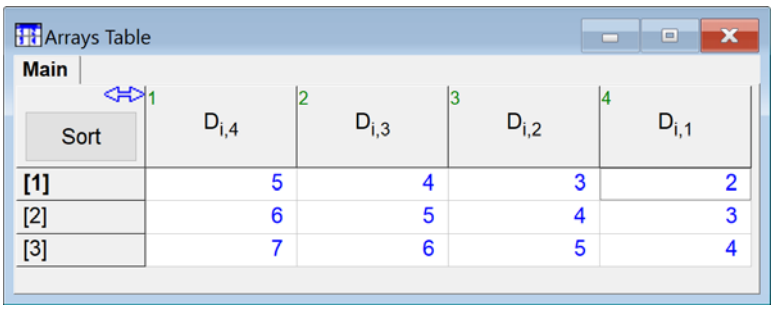


Figure 1-90: Two dimensional array generated using a nested Duplicate statement.

It is possible to define array variables with more than two indices but this practice is not recommended.

Arrays in the Variable Information Dialog

The units of all elements in an array are often the same because a typical array represents different values of the same physical quantity. For example, the array T[]:

```
T[1..3]=[100,200,300]
```

might represent a set of temperatures. Select Variable Info from the Options menu to access the Variable Information Window, shown in Figure 1-91(top). By default, each element of the array is shown and the guess values, limits, and units can be set differently for each element. This arrangement is inconvenient if all elements share a common set of characteristics. De-select the Show array variables check box in the upper left corner in order to collapse the array, as shown in Figure 1-91(bottom). Any characteristic set for the row T[] will be applied to each element of the array. For example, entering K for the units sets the units for all of the elements in the array to K.

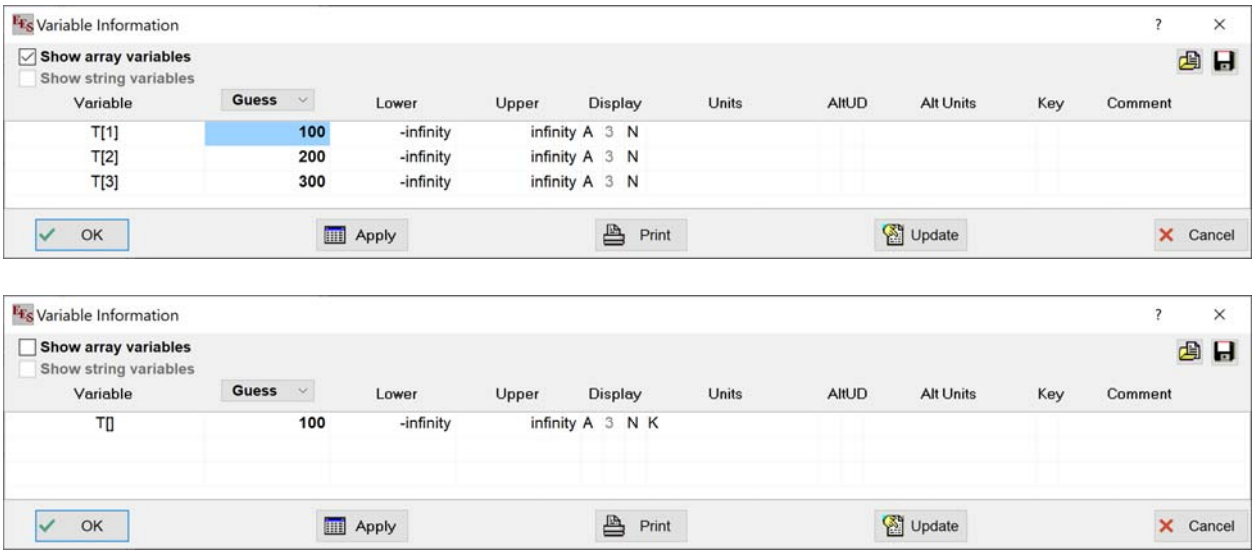


Figure 1-91: Variable Information Window (top) with each element of T shown and (bottom) with the array T collapsed.

It is possible to assign different values of the guess values or limits to each element in an array using the elements of a different array that is setup just for this purpose. This capability is discussed in Section 5.4.

Using the Variable Information Dialog with Large Arrays

It can be cumbersome to locate the row in the Variable Information dialog that corresponds to a particular array element when the EES program contains many large arrays. For example, suppose you have a program that uses arrays x, y, and z, each having a dimension of 1000. You now want to change the guess value for the element z[1]. To do this in the Variable Information dialog, you need to scroll to the first element in the z array, which is row 2001 in the Variable Information Window. A simple way to do this is to collapse the arrays by unchecking Show array variables check box so that the arrays are represented as x[], y[], and z[]. Select z[] by clicking on it. Now check the Show array variables check box to expand the arrays. The list of variables will begin with the first element in the z array.

Purge Unused Variables

EES saves every variable that you introduce as you develop your program. For example, if you start your problem with an array X[] that contains 500 entries and then later change its name to Y[], EES will retain in memory both arrays X[] and Y[] for a total of 1000 variables. This is done so that you don't have to re-enter the characteristics (e.g., units and limits) of the variable X[] should you want to temporarily remove it. Similarly, if at some point your array X[] is defined with 2000 entries and later you reduce its size to 500 entries then EES will retain in memory 2000 variables.

It is often the case that you will want to redefine the size of the arrays that you use in an EES program. For example, arrays are particularly useful in order to solve the system of algebraic equations that result from numerical models. These equations correspond to a set of equations

written for each node or control volume defined within a computational domain. One of the most important things that you should do with a numerical model is to verify that the number of nodes is sufficiently large that the numerical model has converged. This check is typically done by increasing the number of nodes and observing how the solution changes. The final number of nodes that is used is always a compromise between computational time and accuracy.

Based on this discussion, it is clear that you can easily introduce many variables using arrays that you do not ultimately wish to use. In order to clear these variables from memory you must use the Purge Unused Variable command from the Options menu, at which point EES will check if there are unused variables in memory and ask whether you want to purge or keep them.

Solving the EES code below generates a 50x50 two-dimensional array D[]. The EES program includes 2502 variables, all of the entries in the array D[] as well as the variables Nx and Ny.

```
$TabStops 0.25 0.5 in
Nx=50
Ny=50
duplicate i=1,Nx
      duplicate j=1,Ny
            D[i,j]=i*j
      end
end
```

Now reduce the values of the variables Nx and Ny, as shown below, in order to generate a 10x10 array and reduce the number of active variables to 102.

```
$TabStops 0.25 0.5 in
Nx=10
Ny=10
duplicate i=1,Nx
      duplicate j=1,Ny
            D[i,j]=i*j
      end
end
```

Because EES retained the original 2502 variables in memory there are currently 2400 unused variables. In order to purge these variables and reduce the size of the EES file select Purge Unused Variables from the Options menu. You will be asked to confirm that you want to purge the unused variables with the dialog shown in Figure 1-92.

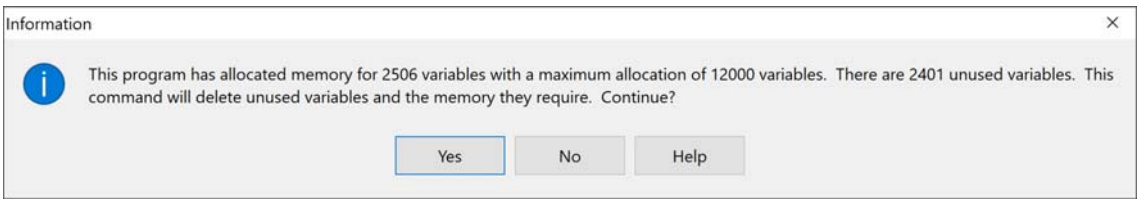


Figure 1-92: Purge unused variables.

It is wise to save the file with a different name before you purge variables.

EXAMPLE 1.2 Extended Surface Problem

This EES example illustrates the process of solving a set of equations using array variables and Duplicate commands. The example can be accessed from the Examples menu in the Mastering EES folder and includes a detailed .pdf file that describes the development of the code.

1.8 Lookup Tables and Lookup Files

Lookup Tables are displayed in an EES program as a table and Lookup Files are stored but not visible in an EES program. Lookup Tables and Files provide a convenient method of saving and accessing tabular data. The data that are available in Lookup Tables and Lookup Files are useful for interpolation and curve fitting, as discussed in Chapter 2

Creating a Lookup Table

To create a Lookup Table, select New Lookup Table from the Tables menu. The name and the number of rows and columns in are entered in the resulting dialog, as shown in Figure 1-93.

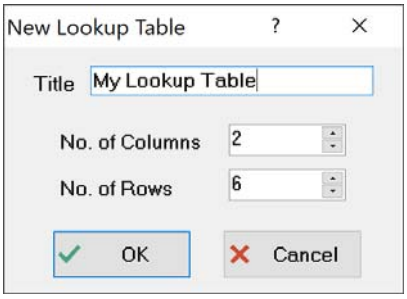
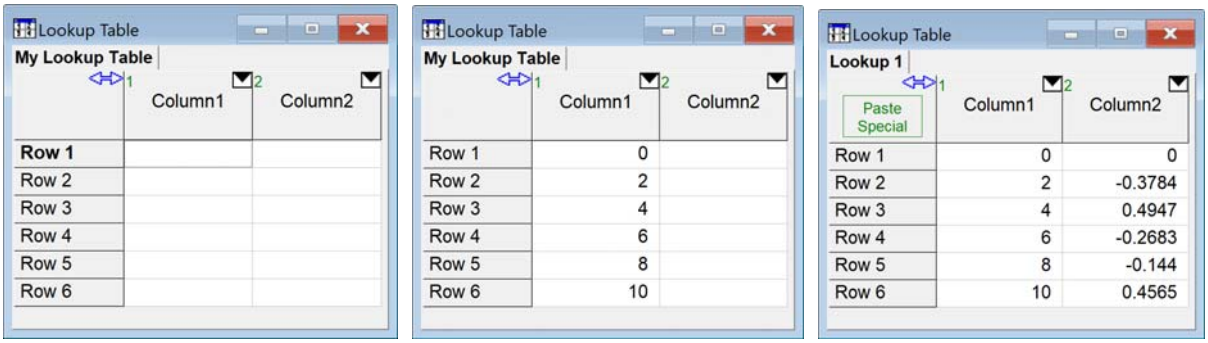


Figure 1-93: New Lookup Table dialog.

Select OK in order to generate the blank Lookup Table shown in Figure 1-94(a).



(a) **(b)** **(c)**

Figure 1-94: Lookup Table (a) blank, (b) with 1st column filled in, and (c) with 2nd column filled in.

Entering Data

Data can be typed directly into the cells of the table or pasted into the table from the clipboard. Alternatively, the values in a selected column can be entered, cleared, or changed by either selecting the triangle icon (▼) or right clicking on the column header and selecting Alter Values; either option brings up the Alter Values dialog. Select Enter Values and then indicate the pattern

that should be used. For example, Figure 1-95(left) shows the Alter Values dialog set up in order to enter values ranging from 0 to 10 in equal increments, leading to Figure 1-94(b).

The column widths can be adjusted manually by dragging the divider between cells in the header (top) row. Also, note the double arrow in the upper left cell above the Paste Special button in the Professional license. Clicking this button will adjust the width of each column in the Lookup Table to fit the data. Clicking the button again will return the column widths to their original values.

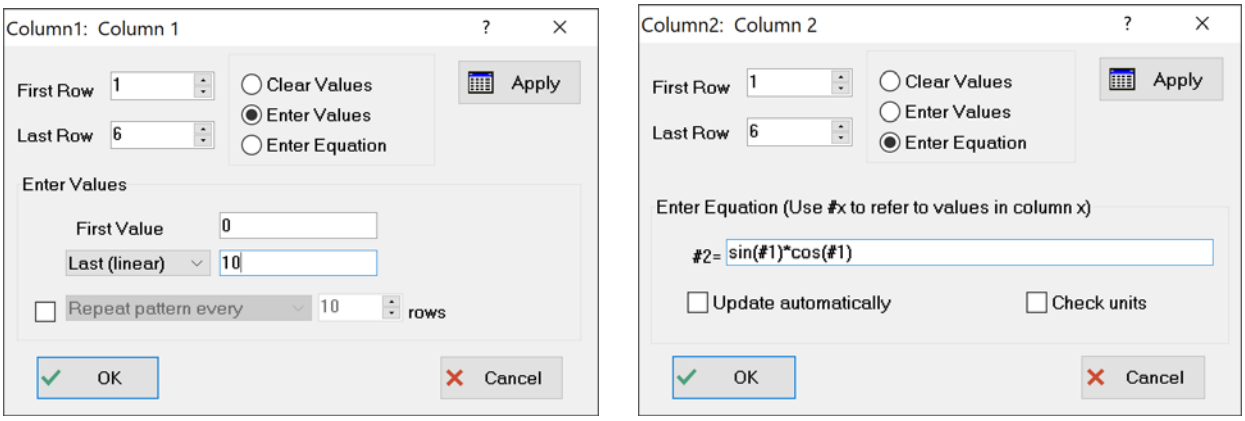


Figure 1-95: Alter Values dialog shown (left) with Enter Values selected and (right) with Enter Equation selected.

In the Professional license, a value greater than the current number of rows can be entered in the Last Row field of the Alter Values dialog (located at the upper left in Figure 1-95). In this case, EES will, after confirmation, increase the number of rows in the Lookup table to this value.

When data are available on the clipboard, the Paste Special (green) button will be visible in the upper left cell of the Lookup table as shown in Figure 1-94(c). Clicking this button will bring up the Paste Special dialog (Figure 1-96). This dialog will allow data that use tabs, commas, spaces, or semicolons to separate values in each column. In addition, the Paste Special dialog makes it possible to paste the column names and/or units into the Lookup table starting at a specified row and column.

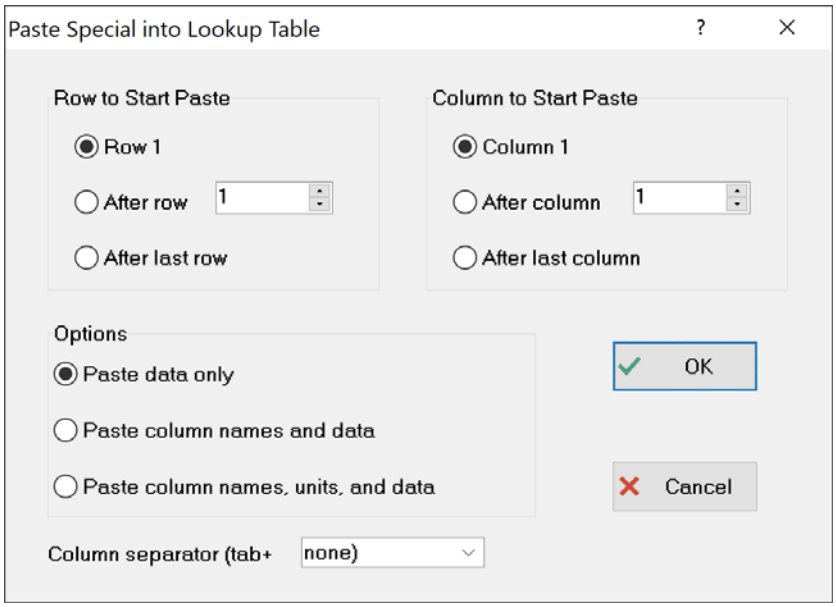


Figure 1-96: Paste Special dialog.

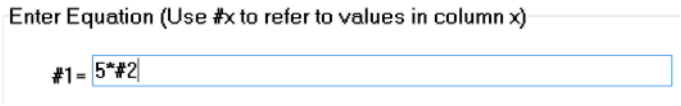
Using Strings in Lookup Tables

Strings can be stored in Lookup Table columns that are designated as having a string format. In order for a column to accept string information the format style of the column must be set to string. To change the format style, right click on the column header and change the Style to String in the Format Column dialog.

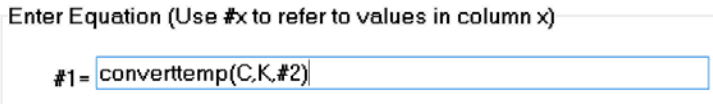
Using Equations in a Lookup Table

Values can also be entered using equations by selecting Enter Equations in the Alter Values dialog, as shown in Figure 1-95(right). The equation can refer to variables in the EES workspace or values in other columns of the Lookup Table. For example, Figure 1-95(right) illustrates the Alter Values dialog setup so that the values in column 2 are calculated as the product of the sine and cosine of the values in column 1. The indicator #X refers to values in column X where X is an integer or the name of an existing column in the Lookup table. Click the Check Units check box (below the equation edit box) to optionally check the equation for unit consistency. Clicking OK results in the Lookup Table shown in Figure 1-94(c).

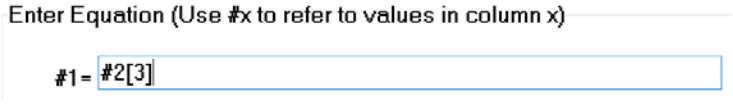
Using the Enter Equation option, values can be entered into the selected column of the table as a function of values in other columns, previously-defined variables in the EES worksheet and any function available in EES. Using this capability results in the Lookup table operating much like as spreadsheet program. The #X sequence is used to refer to values in another column. For example the following entry would set the values in column 1 of the Lookup table to 5 times the values in column 2. Note that, instead of entering #2, you could alternative enter #ColName, where ColName is the name of the column. The changes are applied only to the rows indicated in the edit boxes at the upper left of the dialog.



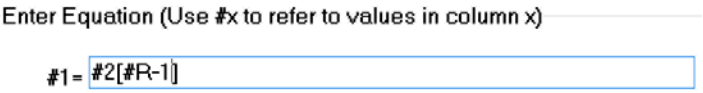
Any of the EES functions can be used in the equation. For example, the following entry converts the values in column 2 which are temperatures in degree Celsius to the Kelvin temperature scale.



In some cases, it is necessary to refer to a particular row of a column in the Lookup table. This capability is available by including the row number in brackets. For example, the following equation sets all of the values in column 1 to the value in row 3 of column 2.



Finally, it is sometimes necessary to refer a row above or below the current row for which a value is being entered. The indicator #R refers to the current row; #R can be used in a subscript. Entering #2[#R] is exactly equivalent to entering #2. However, you can offset the row values by entering the offset to #R. For example, the following equation copies the values in column 2 to column 1, but it offsets the row so that the value in column 1 row 2 is equal to the value in column 2 row 1, and so on for the entire row range.



Note that EES will not provide an error if the row is less than zero or greater than the maximum number of rows.

The Professional license provides the option to automatically update the values in a column that have been set with an equation. Checking automatic update will result in the values assigned with an equation being updated whenever the values that it depends upon are changed.

Saving and Loading Lookup Files

Lookup Tables created in an EES file are saved when the EES file is saved and available when the EES file is reopened. There is no need to separately save a Lookup Table unless you wish to: (1) make the data available to another EES program; (2) make the data available to other applications, e.g., EXCEL; or (3) save the data in a Lookup file so that it can be accessed with Lookup commands without having the data be visible in an EES Lookup Table.

To save an existing Lookup Table, right click on the tab for the Lookup Table to bring up the Information Dialog for the Lookup Table, as shown in Figure 1-97.

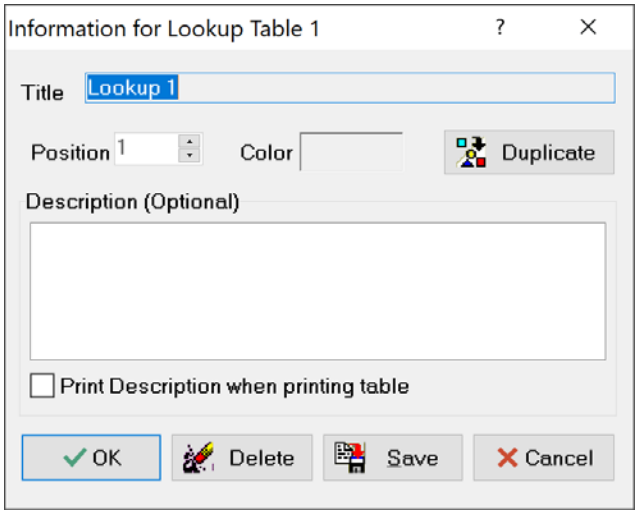


Figure 1-97: Right-click on Lookup Table tab to bring up this dialog

The dialog allows you to edit the name of the Lookup Table, add some descriptive text, duplicate the Lookup Table and alter the position of its tab with regard to other Lookup Tables in the EES file. The Save button allows you to save the Lookup Table in a variety of formats, discussed in Section 1.3 in the context of Parametric Tables. The data alone can be saved in an ASCII text (.txt) or comma separated value (.csv) file that can be read by most other applications. The data with the headers can also be saved in a text file. The table can also be saved in the EES-specific Lookup File or Lookup Table formats. The Lookup Table (.lkt/.lkt64) format is a binary format that can subsequently be opened in other EES files or read directly by EES when using Lookup Table/File functions. This format is typically the easiest one to use and corresponds to a Lookup Table that is saved by and can subsequently be opened by the EES software. Lookup Files saved in the .lkt/.lkt64 format are secure because they are written in a binary format that cannot be read by any application other than EES. All of the information associated with the Lookup Table, including column names, units and column format, is saved in the .lkt/.lkt64 file. See Lookup File Formats in the online help or the subsection Lookup File Formats in this section for details about these data file types.

Deleting Lookup Tables from the Lookup Table Window

Lookup tables are saved with an EES file. They can be saved independently as explained in the previous section. This is useful if the data in the Lookup table are to be used in another program

or you wish to not show the data. An existing Lookup table can be deleted in two ways. One way is to right-click on the window tab to bring up the dialog window shown in Figure 1-97 and then click on the Delete button, A second way to delete one or more Lookup tables is the Delete Lookup Tables command in the Tables menu. Selecting this command will bring up the dialog window shown in Figure 1-98 which lists all Lookup tables in the EES file. Select the tables that are to be deleted, which will place a check mark before the table name and then click Delete.

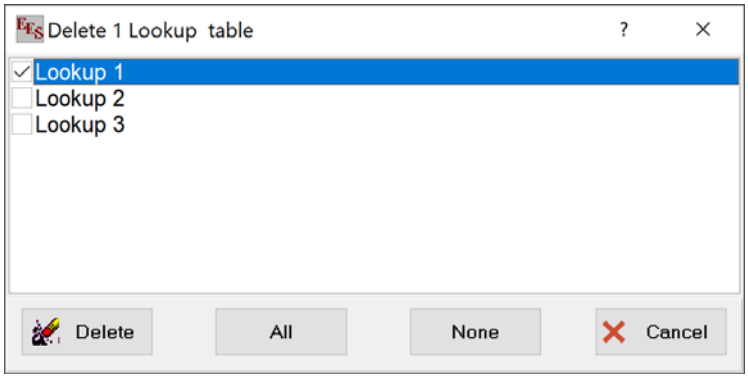


Figure 1-98: The Delete Lookup Tables dialog

Note that deleting the Lookup table means that the data that it contains will not be saved with the EES file the next time the EES file is saved. However, the Delete command does not delete a Lookup table that has been saved separately in Lookup File that has an .lkt, .txt, .csv, or other file name extension.

Lookup Table/File Commands

A complete list of all of the built-in functions that can access data in a Lookup Table or Lookup File is provided in Appendix A, Table A-10. Data in a Lookup Table of File can be used as the basis for interpolation in both 1-D or 2-D , as discussed in Chapter 2. Data in a Lookup Table/File can be retrieved using the Lookup function (for numerical values) or the Lookup$ function (for strings). The calling protocol for the Lookup function is:

```
X = Lookup('TableName',Row,Column)
```

where 'TableName' is a string containing either the name of the table of interest (i.e., the name shown in the tab in the Lookup Window) or the name of the Lookup File stored on disk (including the directory information and file name extension). The 'TableName' argument may be provided with a string variable. Note that all of the built-in functions listed in Appendix A, Table A-10 can access a Lookup File stored on disk (rather than a Look Table displayed in EES) by providing the name of the file rather than the name in the Lookup Table tab. The argument Row is the number of the row in the table and Column is either the number or name of the column. For example, the command:

```
X=Lookup('My Lookup Table', 3, 2)
```

will result in the variable X being assigned 0.4947 if the Lookup Table shown in Figure 1-94(c) is available in the EES file. Non-integer values of the column or row will result in linear

interpolation; however, the interpolation functions discussed in Chapter 2 are more appropriate for this type of operation and provide the option of higher order interpolation. The Lookup$ function operates similar to the Lookup function except that it returns a string value from a column that has been designated as having a string format.

Data can also be written to a Lookup Table using the Lookup command in a Function or Procedure or with the $CopytoLookup directive. Functions and Procedures are described in Chapter 3.

Lookup File Formats

A Lookup file is a two-dimensional table of data that has been stored in a disk file. A name and display format for each column of data may also be stored, depending on the file format. Lookup files can be accessed by any of the Lookup commands listed in Appendix A-10 in the same manner as Lookup tables; these include the Differentiate, Interpolate, Lookup, LookupCol, and LookupRow, SumLookup, AvgLookup, MinLookup, MaxLookup, and StdDev commands as well as others. The only difference is that the name of the file (with the file name extension) is provided to the command instead of the name of the table. EES recognizes both binary and ASCII forms for Lookup files. Binary files are identified with a .lkt filename extension. ASCII Lookup files can have a .txt, .csv, or .dat filename extension. Each file format has advantages and disadvantages. The binary form is read in more quickly and requires a smaller file size. The ASCII form is easier to edit and it can be read and written by spreadsheet or other applications. Details for each file type follow:

Binary Lookup files (.lkt and .lkt64)

Binary Lookup files store all of the information that appears in a Lookup Table window in a binary file on disk, including the data, and the column name, units, and display format for each file type. A binary Lookup file is created using the Save Lookup Table command or with the $SaveTable directive. Once created, the Lookup file can be opened into the Lookup Table window using the Open Lookup Table menu command, the $OpenLookup directive, or a Link button (type 6) on the Diagram window. The .lkt file format is used by 32-bit versions of EES. However, it is not necessary to open the file in order to access the data in the file. The file is opened internally (without needing the Lookup table) when commands such as Interpolate, Lookup etc. are provided with the name of the file. The .lkt64 file format can only be read or written by the 64-bit license.

ASCII Lookup files (.txt)

There are several variations for the .txt Lookup file format. In the basic form, the first line of the file contains the number of rows and columns in the table. Each following line provides the data for one row with the value for each column separated by one or more spaces or a tab character. The basic form does not provide a means of specifying the names, units, or display format for the data. EES assigns the names 'Column1', 'Column2', etc. and these column names should be used when the file is used with Lookup commands like Interpolate or Differentiate. Automatic formatting is used to display the data if the file is read into the Lookup Table with the Open Lookup Table command. The following example shows the .txt data needed for a Lookup file that has five rows and three columns, with no column header information.

```
5  3
1 11 111
2 22 222
3 33 333
4 44 444
5 55 555
```

If the number of rows is provided a negative number, EES will determine the number of rows of data in the file when the file is read.

If the number of columns is provided as a negative number then EES will expect to find the format specification. A format specification consists of one or two characters. The first character is the type of format and the second character indicates the number of digits provided for this format. The number of digits is usually an integer between 0 and 9 but the letter 'M' can be supplied to indicate the maximum possible number of digits. Valid format specifications are:

A auto format. A followed by a number, e.g., A3, is accepted but the number is ignored.
F# fixed decimal format with # digits after the decimal point, e.g., F3.
E# exponential format with # digits after the decimal point or #+1 significant figures, e.g., E2
N# N significant figure format with # being the number of significant figures, e.g., N4
G# engineering format with # significant figures, e.g., G4
T# Time format; if # is 0, the default format is used; if # is 1, the format is hours:minutes:seconds; if # is 2, the format is hours:minutes
D# Date format; # must be between 0 and 9 representing different date formats. 0 is the default format
S string format. S followed by a number, e.g., S0, is accepted but the number is ignored.

The format specification is followed by one or more spaces and then the column heading and units for each column. If a column contains string data, then the format specification must be S (or S#) and the string data must be contained in single quotes. Units are optional but if supplied, they must be enclosed in square brackets with a least one space separating them from the column name. The following lines contain the .txt data for each row, separated by one or more spaces or a tab. The example below would create a table with 2 rows and 3 columns. The columns would be formatted with E4, F1, and S format specifications and the column names will be ColA, ColB, and Name.

```
2  -3
E4 ColA [Btu]
F0 ColB
S Name
1.23E-12 2.2 'A'
2.34E-11 4.8 'B'
```

The Save Table command in the Tables menu and the $SaveTable directive provides a way to save a Lookup table as a .txt file including the column header information.

ASCII Lookup files (.csv)

The .csv file provides only data without any information concerning the column names or data format. The number of values in the first row of the file determines the number of columns in the table. The number of rows in the table is equal to the number of rows of data. Values on each row are separated with the list separator character. Each row ends with a linefeed - carriage return.

ASCII Lookup files (.dat)
Like the .csv file, files with the .dat filename extension provide only data without any information concerning the column names or data format. The number of values in the first row of the file determines the number of columns in the table. The number of rows in the table is equal to the number of rows of data. Values on each row are separated with a tab character. Each row ends with a linefeed - carriage return.

Files having filename extensions other than .txt, .lkt, .lkt64, .csv, or .dat can be read provided that they are ASCII files with numerical values separated by a space or comma.

Lookup files with ASCII data may be read in with specifications provided by a Format (.fmt) file. See the online help for details relating to this type of file.

EXAMPLE 1.3 Pump Performance Model

This EES example illustrates the process of using a Lookup Table to store information about different pumps so that a model can be developed that easily switches between pump models by pulling the relevant information from a Lookup Table. The example can be accessed from the Examples menu in the Mastering EES folder and includes a detailed .pdf file that describes the development of the code.

1.9 Other Features

There are a few additional basic features of EES that are worth discussing before closing out our introduction to the program.

Built-In Constants

Constants in EES are denoted by a # sign after the name; constants are like variables in EES except that their values cannot be changed in the Equations Window. A large number of common physical constants are pre-assigned in EES. These entries can be viewed by selecting Constants from the Options menu in order to access the Constants dialog, shown in Figure 1-99.

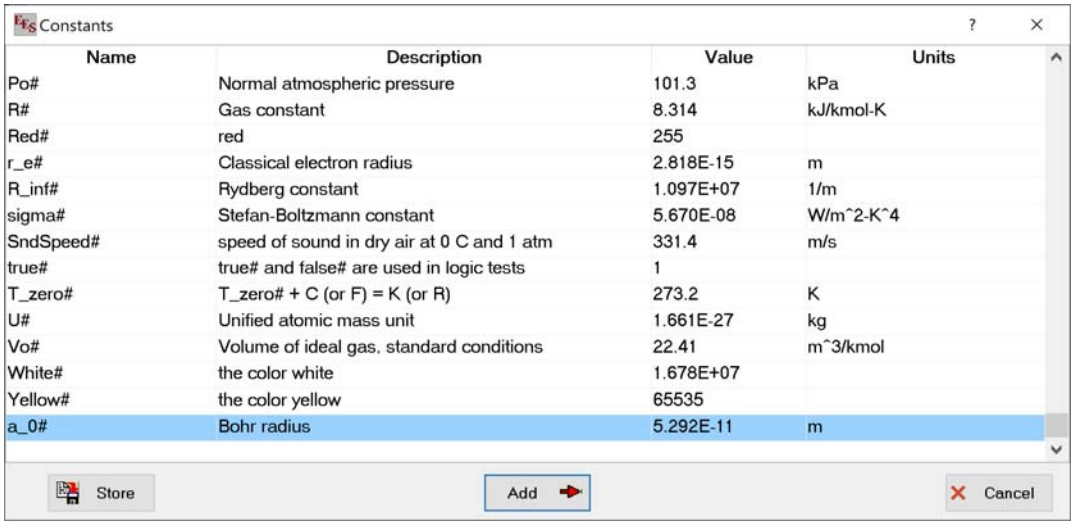


Figure 1-99: Constants dialog.

Many physical constants of interest are available; for example, pi# is the value of π (actually, pi also refers to π and is the only built-in constant in EES that does not require the use of #), R# is the universal gas constant, k# is Boltzmann's constant, etc. These constants are provided in the unit system that has been specified by the user, as discussed in Section 1.5. The order of the constants appearing in this dialog can be changed by clicking on the column header. For example, clicking on Name at the top of column 1 will rearrange the order of the table so that the entries are ordered alphabetically in terms of their names. Clicking a second time on Name will reverse the order.

Adding Constants

It is possible to add constants by selecting Add from the Constants dialog in Figure 1-99 in order to access the Add New Constant dialog shown in Figure 1-100.

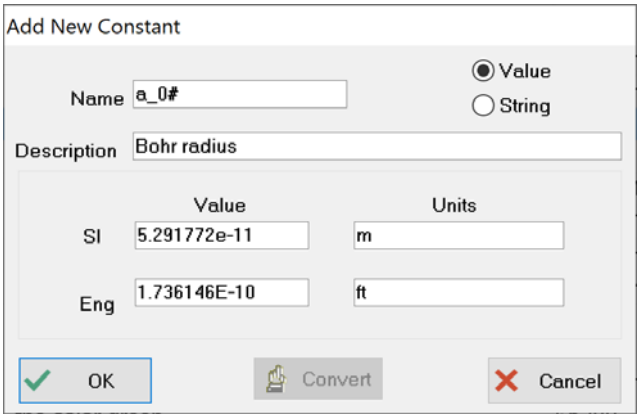


Figure 1-100: Add New Constant dialog.

The Add New Constant dialog prompts the user for the name, description, and value of the constant in both SI and English units. For example, suppose we want to enter the Bohr radius, a_0, to our EES program. The value of the Bohr radius is 5.291772×10^{-11} m (or 1.736146×10^{-10} ft)

according to Mohr et al. (2011). The Add New Constant dialog should be filled out as shown in Figure 1-100. If you select OK and access the Constant dialog by selecting Constants from the Options menu then you will see that the constant a_0# has been added to the bottom of the dialog, as shown in Figure 1-99. The new constants list can be saved by selecting the Store button in the Constants dialog so that it is available each time EES is opened. Note that you will need to de-select the Overwrite Constants.txt file in the installation process, shown in Figure 1-2, in order to keep your added constants when you update or reinstall the EES program. Alternatively, make a copy of the Constants.txt file to ensure that it is not overwritten when you install a new version of EES.

The user can also specify the value of constants in an EES program using the $Constant directive. The calling protocol for the $Constant directive is:

```
$Constant Name# = ValueSI [UnitsSI] ValueEng [UnitsEng]
```

where Name# is the name of the constant, ValueSI and ValueEng are the values of the constant in the SI and English units specified by [UnitsSI] and [UnitsEng], respectively. For example, the Bohr radius can be entered in EES according to:

```
$Constant a_0#=5.291772e-11 [m] 1.736146 [ft]
```

See Section 14.3 for more detail relating the $Constant directive. Finally, it is possible to edit the Constants.txt file directly to add a constant that is loaded each time EES is opened.

The Calculator Window

The Calculator Window allows you to quickly carry out simple calculations without interfering with the equations that are entered in the Equations Window. This capability is often very useful when you are trying to debug an EES code or understand a solution. The Calculator Window allows you to enter and solve single-line expressions that involve any of the variables that have previously been defined and evaluated in the Equations Window. Any of the built-in functions in EES can be accessed from the Calculator Window. For example, enter and solve the following equations:

```
a=b+2
b^2=4+c
c=a-b
```

which should lead to the Solution Window shown in Figure 1-101.

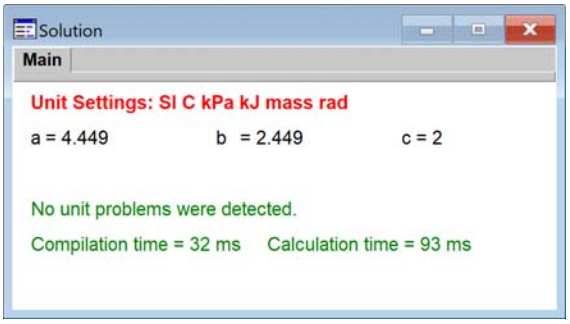


Figure 1-101: Solution Window.

Open the Calculator Window by selecting Calculator from the Windows menu (or selecting the button), as shown in Figure 1-102.

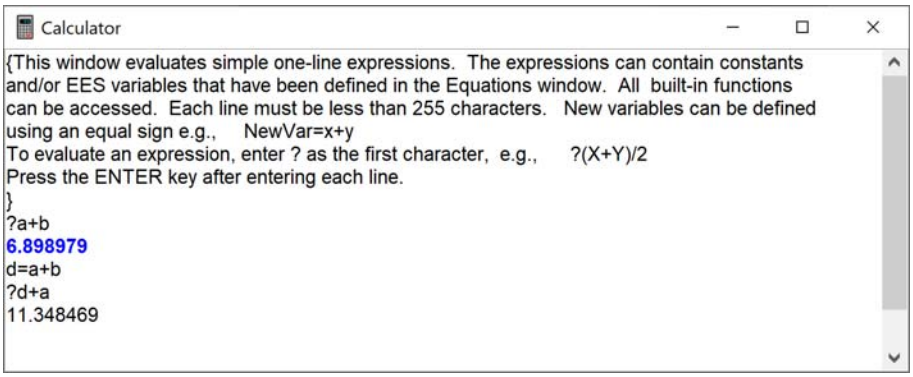


Figure 1-102: Calculator Window.

The result of any command preceded by a question mark will be displayed in the Calculator Window. For example, the command:

```
?a+b
```

will lead to 6.899, as shown in Figure 1-103. New variables can also be defined in the Calculator Window (provided that they don't already exist in the Equations Window). For example, we can define the variable d in the Calculator Window:

```
d=a+b
```

and then use the variable d in a new expression, also shown in Figure 1-102. Use % to refer the last calculated variable.

Preferences and Syntax Highlighting

Select Preferences from the Options menu in order to access the Preferences dialog shown in Figure 1-102. The Preferences dialog allows the user to specify a large number of characteristics of the EES program that are categorized according to the tabs at the bottom of the dialog. For example, in the Options tab shown in Figure 1-103 you can specify how often the file is auto-saved, whether units are checked automatically, whether or not to maintain a list of recently opened files, and a number of other options. Some of these options can be set by Directives as discussed in Appendix C. For example, the option to display a Sum row in the Parametric table can be enabled with the directive $SumRow On. The option to automatically check units can be enabled with the directive $CheckUnits AutoOn. Each tab contains choices of this nature that will be discussed throughout this book.

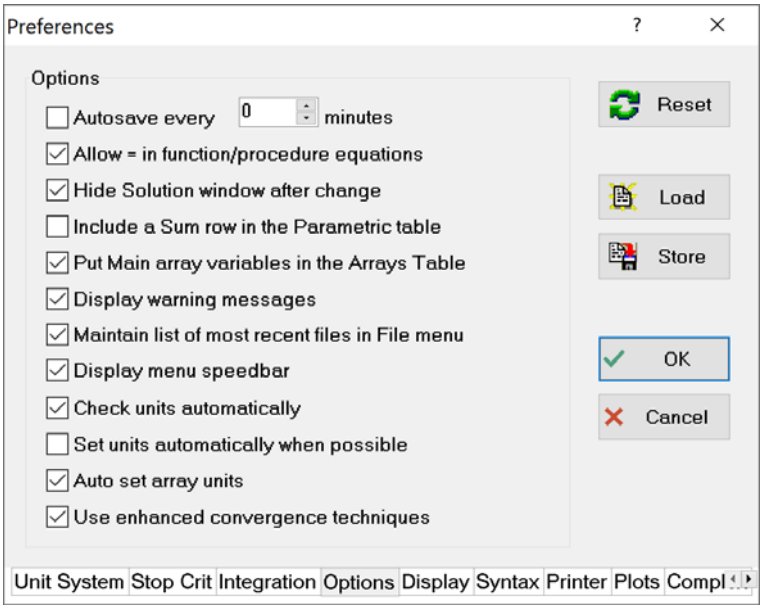


Figure 1-103: Preferences dialog showing general options.

The Display preferences are shown in Figure 1-104. The fields in this window control some of the basic display options. These options are applied independently of the setting of the Syntax Highlighting control in the status bar of the Equations window. One of the most useful options is the size of the font. The default size is 10, but the drop-down menu allows sizes to be selected from 8 to 24. Large font sizes are useful when projecting EES in a classroom or presentation. The screen background can be switched from white to black using the control at the bottom of this dialog.

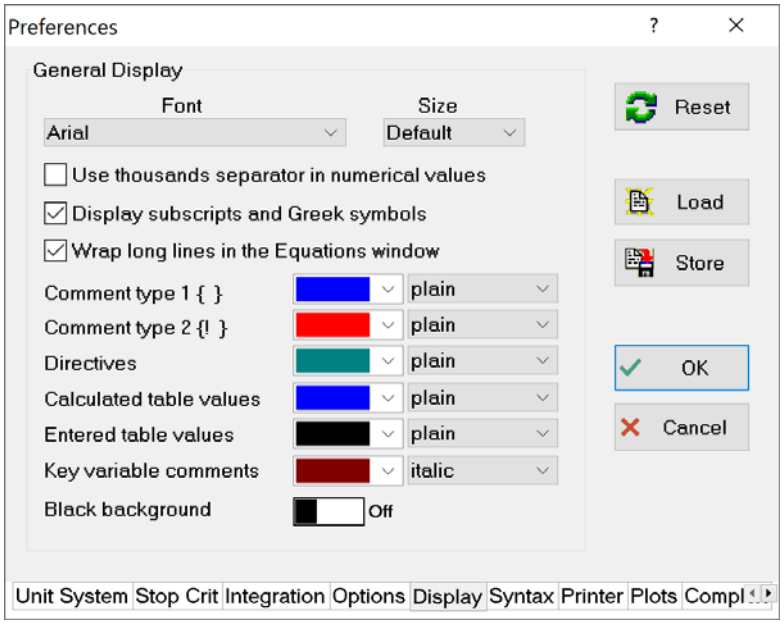


Figure 1-104: The Display options in the Preferences Dialog.

The options in the Syntax tab are shown in Figure 1-106. The Tab Stops option can place up to 25 tabs at the specified locations, which are useful for organizing programs with Duplicate loops or comments. This option can be also specified with the $TabStops directive as described in Appendix C. All of the other options affect the appearance of the content in the Equations window when the Syntax Highlighting control in the status bar at the bottom of the Equations window (Figure 1-105).

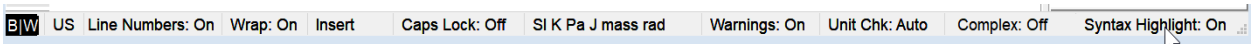


Figure 1-105: Equation Window status bar.

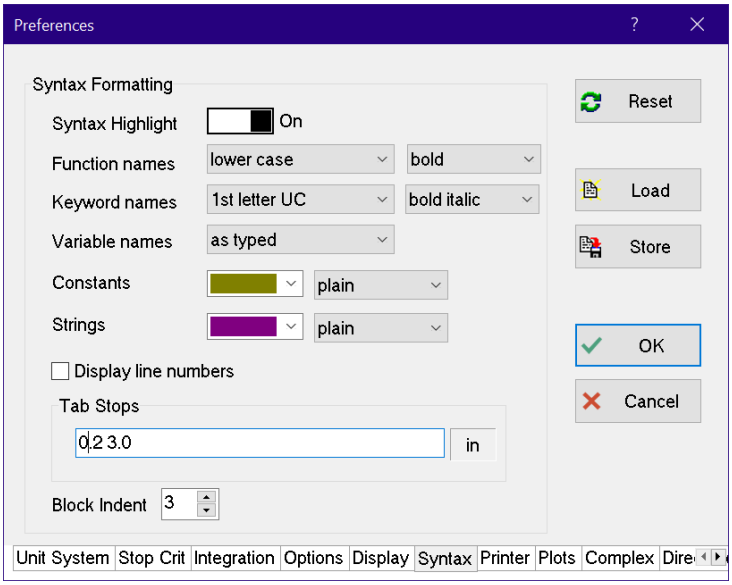


Figure 1-106: The Syntax Highlighting options in the Preferences Dialog.

Syntax highlighting affects the display of keywords and both built-in and user-supplied Functions, Procedures, Modules and Subprograms. The display options allow the both the case and style of Function names and EES Keywords to be independently controlled. The case of EES variables can be AsType (i.e., no control) or AsFirstEntered. The font color and style of constants and strings can be specified. Syntax highlighting can enhance the display and make the equations easier to read as shown in Figure 1-107, where a section of the Equations window is shown both without and with syntax highlighting using the default options.

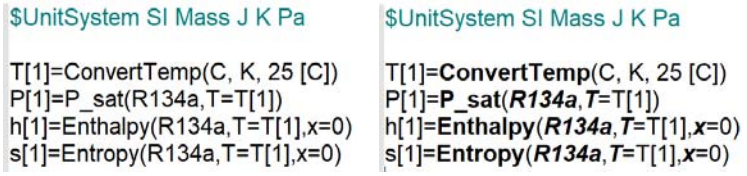


Figure 1-107: Equation Window (left) without and (right) with syntax highlighting.

Syntax highlighting must be on in order to have hyperlinks active. **Note:** Syntax highlighting can increase the time required for EES to refresh the display. If you have a large EES program and the display is slow, you may want to turn off syntax highlighting.

Clicking the Reset button in the Preferences Dialog will reset all preferences to their original (as-shipped) settings.

Pressing OK will set the preferences you selected; these will be in effect for only the current session. However, selecting the Store button will cause the preferences to be saved in an EES Preferences (.prf) file. If you save the file as EES.prf in the EES working directory (which is the default selection that is used when you select Store), then these preferences will automatically be loaded whenever EES is started. Alternatively, you can save a .prf file that you want to apply only to the current file in an alternative directory and subsequently load it using the Load button in Figure 1-103.

Even more convenient is to load the preferences automatically using the $Include directive. The $Include directive is discussed more completely in Section 14.5. Briefly, it provides a method to automatically load various types of external files in an EES program. One type of file that can automatically be loaded is the Preferences file.

For example, select Preferences from the Options dialog and then select the Display tab. Adjust the font, size, etc. as you see fit and select Store. Save the .prf file in a directory of your choosing (e.g., C:\temp\My Preferences.prf). Place an appropriate $Include directive at the top of your EES file.

```
$Include C:\temp\My Preferences.prf
```

If you close and re-open the file you should find that the selections that you made in the Preferences dialog and saved in the file My Preferences.prf are automatically loaded and used in your EES file.

References

Mohr, P.J., B.N. Taylor, and D.B. Newell, "The 2010 CODATA Recommended Values of the Fundamental Physical Constants (Web Version 6.0)," NIST, Gaithersburg, MD, 20899, (2011).

Nellis, G.F. and S.A. Klein, *Heat Transfer*, Cambridge University Press, New York, (2009).

Shah, R.K. and A.L. London, *Laminar Flow Forced Convection in Ducts*, Academic Press, New York, (1978).