

NeoCal for Qtopia User Guide

Version 2.0.3

Published on February 21, 2005



www.hudren.com

Copyright (c) 2000-2005 Hudren Andromeda Connection. All rights reserved.

NeoCal and the HAC logo are trademarks of Hudren Andromeda Connection. Other company and product names may be trademarks or registered trademarks of their respective holders.

Table of Contents

INTRODUCTION	5
FEATURES.....	5
REQUIREMENTS	7
INSTALLATION.....	8
UPGRADING NEOCAL	9
UNINSTALLING NEOCAL.....	9
GETTING STARTED	10
KEYBOARD LAYOUT	10
COMPARISON OF INPUT METHODS.....	10
THEMES	12
DYNAMIC KEYBOARDS.....	12
PREFERENCES	13
<i>General</i>	13
<i>Display</i>	14
<i>Keyboard</i>	15
FORMATTING NUMBERS.....	15
ENTERING NUMBERS.....	15
CUSTOMIZABLE KEYBOARD.....	16
UNDO/REDO SUPPORT	16
MACRO RECORD AND PLAYBACK	17
ALTERNATIVE INPUT	17
USING NEOCAL WITH OTHER APPLICATIONS	18
USING REGISTERS.....	19
WORKING REGISTERS	19
THE M REGISTER	20
USER REGISTERS	20
USER REGISTERS DIALOG	21
CLEARING THE REGISTERS.....	22
GENERAL FUNCTIONS	23
ONE-NUMBER FUNCTIONS.....	23
TWO-NUMBER FUNCTIONS	23
PERCENTAGES	23
<i>Percent</i>	24

<i>Percent Change</i>	24
<i>Percent of Total</i>	24
FINANCIAL FUNCTIONS.....	25
FINANCIAL REGISTERS	25
MARGIN AND MARKUP	26
INTEREST CONVERSION.....	26
TIME VALUE OF MONEY	27
AMORTIZATION	28
CASH FLOW ANALYSIS	30
SCIENTIFIC FUNCTIONS	33
TRIGONOMETRY.....	33
LOGARITHMS	33
CONVERSIONS.....	33
PHYSICAL CONSTANTS	34
STATISTICAL FUNCTIONS.....	36
STATISTICAL REGISTERS	36
MEAN AND STANDARD DEVIATION	37
LINEAR REGRESSION	38
PROBABILITY	38
PROGRAMMER FUNCTIONS	40
BASE CONVERSIONS.....	40
INTEGER CONVERSIONS	40
BITWISE OPERATIONS	41
CONVERSION FUNCTIONS.....	42
LENGTH CONVERSIONS	42
AREA CONVERSIONS.....	42
VOLUME CONVERSIONS	43
SPEED CONVERSIONS	44
WEIGHT CONVERSIONS	44
TEMPERATURE CONVERSIONS	45
UNIT ARITHMETIC	45
DATE AND TIME FUNCTIONS.....	47
DURATIONS	47
MOMENTS	49
DATE ARITHMETIC.....	51

RPN TUTORIAL	53
AUTOMATIC MEMORY STACK	53
STACK MANIPULATION.....	54
FREQUENTLY ASKED QUESTIONS	55
GENERAL FAQ	55
QTOPIA-RELATED FAQ.....	55
REVISION HISTORY	57
VERSION 2.0	57
REGISTRATION.....	59
CLIPBOARD METHOD.....	60
SUPPORT	61
PROGRAM UPDATES	61
DOCUMENTATION UPDATES	61

Introduction

NeoCal™ is a calculator, and like many other calculators, you can start using it without reading a manual. This guide was written to provide a reference for its advanced features which may work similarly to, but not exactly the same as, the calculators you've already used.

NeoCal was designed to operate consistently and logically, with only a few exceptions added to match expectations of how other calculators work. If you come across something that doesn't make sense, please check this guide to get an understanding of NeoCal. The Financial Functions and Statistical Functions chapters are especially helpful for understanding these more complex applications. They're easy to use once you have a general idea of how NeoCal was designed.

Throughout this guide, examples are often provided in place of explanations, so it's important to work through them. Using the calculator while following the examples may improve your understanding of both the examples and the operation of the calculator.

Features

Here is a summary of some of NeoCal's advanced features:

- Efficient keyboard layout
- 12-digit display with thousands separators
- Multi-level undo/redo support
- Keyboard macro record and playback
- Customizable colors and grayscale levels
- Labeled results for ease of use and clarity
- 100 general-purpose user registers
- Choice of Simple, Precedence, and RPN input methods

NeoCal includes over 220 functions:

- **Financial:** time value of money, amortization; discounted cash flow analysis for internal rate of return and net present value; interest conversion between nominal and effective rates; margin and markup calculations

- **Scientific:** trigonometric, hyperbolic, and logarithmic functions; degrees/radians conversions; time and angle conversions; physical constants
- **Statistical:** two variable statistical analysis including mean, weighted mean, standard deviation; linear regression forecasting, correlation coefficient; probability calculations
- **Programmer:** base number conversions between binary, octal, decimal, and hexadecimal; 32-bit, 16-bit, and 8-bit signed and unsigned integer math calculations and conversions; logical and bit-manipulation operations
- **Conversions:** length, area, volume, speed, weight, temperature; unit arithmetic
- **Date and Time:** date arithmetic; duration arithmetic and conversions including frames for digital video editing; moment conversions

Requirements

NeoCal for Qtopia is designed to run on Linux-based handhelds with Qtopia installed. Currently supported devices include:

- Sharp Zaurus SL series

NeoCal was developed using the Qtopia Commercial SDK and should execute with any compatible QPE / Qtopia 1.5 runtime environment. NeoCal has been tested to work with Zaurus SL-series ROM versions 1.10 or later.

Installation

The NeoCal download file (**neocal_qtopia_setup.exe**) is a Windows installation or setup program that will install the NeoCal software and related documentation files on the desktop computer. Open or execute this file on your desktop computer and follow the instructions to proceed with the installation.

This installation program will install the NeoCal files onto the hard drive of your desktop computer and create a program group called "NeoCal for Qtopia" that can be used to access the NeoCal documentation.

If you have downloaded a file without 'setup' in the name, you will have to manually decompress the archive.

The installed or decompressed directory will contain this user guide in .pdf format and both the NeoCal application and user guide in .ipk format. The user guide .ipk file can be installed and removed independently of the application itself and is not required for use of the calculator. Due to current technical limitations, the user guide must be installed to RAM, not an expansion card.

To transfer the software to the Zaurus using Qtopia Desktop on the PC,

1. Click on the Documents icon on the left side of the window.
2. Choose the File -> Upload File menu option to upload the neocal-qtopia ipk file to the Zaurus. Optionally, upload the neocal-guide-qtopia ipk file to install the HTML user guide.

To install the software on the Zaurus,

3. Tap the Settings tab of the Qtopia home screen.
4. Tap on Add / Remove Software.
5. If a previous version of NeoCal is already installed, uninstall it now.
6. Tap the Install packages button.
7. Tap on the neocal-qtopia package, then click OK on the Install dialog.
8. Optionally, tap on the neocal-guide-qtopia package and then click OK on the Install dialog.
9. Close the Package Installer window.

10. Close the Add/Remove Software window, Qtopia will restart. You should now see the NeoCal icon on the Applications tab of the Qtopia home screen.
11. Delete the neocal-qtopia and neocal-guide-qtopia ipk files from the Zaurus. This can be done either on the Zaurus or in the Qtopia Desktop documents view.

For additional information, please refer to the documentation supplied with your handheld.

Upgrading NeoCal

You must first remove NeoCal from the device before you can install a newer version, see Uninstalling NeoCal. Your preferences and registration information are not removed and will be available when the new version of NeoCal is installed.


Uninstalling NeoCal

You can remove NeoCal from the device using the Add/Remove Software application. Tap the **Uninstall packages** button, then the **neocal-qtopia** package, then the **Uninstall** button. If you have also installed the user guide, select the **neocal-guide-qtopia** package and tap the **Uninstall** button to also remove it. Your preferences and registration information will not be removed.

If you used the Windows installation program, the NeoCal files can be removed from your desktop by running the Start -> Programs -> NeoCal for Qtopia -> Uninstall program to remove all files from your desktop computer. This does not remove the software from the handheld device.

Getting Started

Keyboard Layout

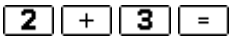
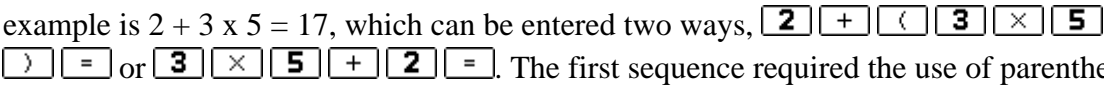

NeoCal contains far too many functions to place every button on the screen at once, and so several methods are provided for accessing these additional buttons on the advanced keyboard. The first method is to push the shift button () showing alternate functions for the buttons currently being displayed. Pressing the shift button a second time will restore the buttons to their non-shifted function.

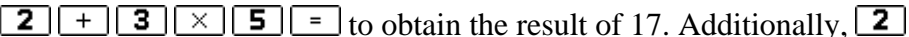
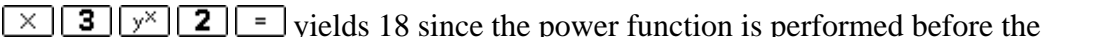
The second method involves choosing a functional group of buttons by selecting *General*, *Financial*, *Scientific*, *Statistical*, *Programmer*, *Conversion 1*, *Conversion 2*, *Date & Time*, or *User* from the popup list in the upper right corner of the display. This changes the functions for the top two rows of buttons. Switching between function groups during a calculation does not have an effect on that calculation or the state of the calculator, and so calculations can involve operations from different functional groups.

Selecting the input method changes the bottom five rows of the advanced keypad. This is discussed in the Comparison of Input Methods section.

The basic keyboard layout is provided for doing simple calculations using larger, finger-friendly buttons.

Comparison of Input Methods

This calculator was designed to work equally well using either the *Simple*, *Precedence*, or *RPN* input methods. The simple input method is most familiar and allows entering calculations like  to get the desired result of 5. A more complex example is $2 + 3 \times 5 = 17$, which can be entered two ways,  or . The first sequence required the use of parentheses to instruct the calculator to multiply before adding the numbers (since the Simple input method does not conform to this algebraic convention). The second sequence avoided the use of parentheses by re-ordering the operations to generate the correct result of 17.

The Precedence input method is an improvement over the simple input method in that binary operations are automatically performed before multiplication and division operations which are performed before addition and subtraction operations, reducing the need for parentheses. For example, the previous calculation can be entered without the parentheses as  to obtain the result of 17. Additionally,  yields 18 since the power function is performed before the multiplication.

The benefits of the RPN, or Reverse Polish Notation, input method over the simple and precedence input methods become apparent when calculating complex expressions. Fewer mistakes are made since fewer keystrokes are required and the intermediate results are always displayed (providing feedback during the calculation, not just at the end). The order of the calculation is changed to avoid using parentheses, as in the previous example, so that the expression is evaluated from the inside working outward.

The general rule of thumb for RPN calculators is to enter the values, then the operation. Thus, $2 + 3 =$ becomes 2 **Enter** 3 **+**. The **Enter** key is used to separate numbers when entering more than one in succession. In the example, $2 + 3 \times 5 = 17$, the RPN method could be evaluated using either 2 **Enter** 3 **Enter** 5 **×** **+**, or 3 **Enter** 5 **×** 2 **+**. Looking at these two sequences, we can isolate 3 **Enter** 5 **×** which always shows the result of 15. The two sequences then become 2 **Enter** 15 **+** and 15 2 **+** which are more easily recognizable as 17.

Keystroke savings are realized in calculations that cannot be reordered to a form without parentheses. Let's look at the example $2 \times 4 + 3 \times 5 = 23$. Using the simple input method we could calculate this expression as 2 **×** 4 **+** $($ 3 **×** 5 **)** **=**, or without using parentheses, 2 **×** 4 **=** **→M** 3 **×** 5 **+** **RM** **=**. Using RPN, the sequence becomes 2 **Enter** 4 **×** 3 **Enter** 5 **×** **+**. Saving one keystroke is not that significant, but this is still a relatively simple calculation. One important distinction between the two input methods is demonstrated by the sub expression 2×4 . Using the simple input method, this sometimes needs to be enclosed in parentheses (when not located at the beginning of the expression) and sometimes not. Using RPN, this sub expression is always entered the same way. This consistency also contributes to fewer mistakes being made.

One final point is that most of the calculations require the operand to be entered before the operation is executed. For example, calculating the square root of 2 is always entered 2 **√x** and the inverse of 10 is entered 1 0 **1/x** regardless of the current input method. The RPN input method just extends this methodology to all of the calculator's functions.

The input method can be chosen in the Preferences dialog. If you'd like to learn more about how an RPN calculator works, check out the RPN Tutorial chapter of this guide. With all of the input methods, all of the calculator functions are available allowing you to work the way you're most comfortable.

The input method can be temporarily changed using the **simp**, **prc**, or **rpn** buttons. The clear operation restores the input method to the setting specified in the Preferences dialog.

Themes

NeoCal contains built-in themes that can be selected to alter the overall appearance of the main display and keyboard. Themes do not affect the operation of NeoCal in any way, but selecting a theme that is visually appealing can increase your efficiency when using NeoCal.

Generally, a theme specifies most of the visual aspects of the calculator including colors, margins, spacings, display lines, font size, border type, and dynamic keyboard settings. Some of these characteristics can be specified or overridden in the Preferences dialog to further customize your interaction with NeoCal.

Dynamic Keyboards

The keyboards are constructed dynamically to best take advantage of the available screen space while maintaining efficient operation based on user preferences which are specified by the current theme and possibly overridden by options in the Preferences dialog.

Several different sets of glyphs are available for drawing the keys and NeoCal will always select the largest glyph for any given key. In other words, the bigger the key, the bigger the writing on it and the smaller the key, the smaller the writing. If the key is large enough, NeoCal can display both the primary and shifted functions at the same time. The appearance and behavior of the keyboard is controlled by a number of settings:

The *Show* option determines whether the shifted functions should be shown along with the primary functions and, if so, where they should be displayed. *None* specifies they should never be shown even when there is plenty of room, *Above* specifies the shifted functions should be displayed above and outside the key boundaries, and *Within* specifies the shifted functions should be drawn on the face of the key (above the primary function). If there is not enough room to display all of the shifted functions, none of them will be displayed regardless of the current setting. To increase the likelihood that shifted functions will be shown, try reducing the border, font size, and/or number of display lines.

If the shifted functions should be shown but there is not enough room to display them, NeoCal may change the primary function glyph to a smaller size to provide the room necessary to display the shifted functions. In this case, the *Size* option specifies the smallest acceptable size for the primary functions.

The *Keys* option specifies how the shifted keyboards should be displayed when the shifted functions are shown on the primary keyboard. The *Static* setting will not display a

separate shifted keyboard, but continue to display the primary keyboard when the shift state is active. The *Swapped* setting switches the shifted function with the primary function indicating the lower (shifted) function is active and pressing the shift button again will access the primary function. The *Zoomed* setting will display keys with only the active (shifted) function visible. If the shifted functions are not displayed on the primary keyboard, then the shifted keyboards will be displayed with the equivalent of the *Zoomed* setting.

The *Color* option specifies which color to use when drawing the shifted function on the primary keyboard. *Shift* indicates the shift color should be used, overriding the foreground color for that particular key. *Foreground* indicates the normal glyph color for that key should be used. *Background* causes the glyph to be drawn using the background color for that function. If the shifted functions are displayed within the key and this setting is set to *Background*, the glyph will be drawn using the shift color.

The *Highlight* option specifies which color should be used for displaying the shifted (active) function on the shifted keyboard. The *Shift* setting specifies the glyph should be drawn using the shift color and the *Foreground* setting specifies the normal color used for drawing that function. The *Swapped* setting will draw the shifted function using the foreground color of the primary function, and draw the primary function using the shift color.

Preferences

There are many settings available in the Preferences dialog to control and customize the operation and appearance of NeoCal. This dialog can be opened by selecting the Preferences menu option from the Options menu, or by tapping on the calculator's display area.

General

The default number format is *Standard*, specifying numbers to be displayed in a natural format, with scientific notation used only when the number is too small or too large to fit on the display. *Fixed* format also displays the number in a natural format but only switches to scientific notation when the number is too large for the display (when the number is too small, it displays zero). Choosing the *Scientific* format will cause all numbers to be displayed using scientific notation. The *Engineering* format also uses scientific notation whereas the exponent is adjusted to a multiple of three.

Additionally, the number of digits that should be shown to the right of the decimal place can be specified. Selecting *All* causes all significant digits to be displayed (without trailing zeros, up to the limit of the display). Specifying a number for this field causes the

display to show that number of digits past the decimal point including trailing zeros. If there are significant digits beyond what is being displayed, the last digit is rounded. This setting does not effect the accuracy of the calculations and is only used for displaying the results.

Pressing the `[show]` button will temporarily format the display showing all significant digits. Pressing `[show]` a second time, or performing any other operation, will return the format of the display to the specified setting. The *round* function, `[rnd]`, will modify the internally stored number to be exactly equal to the number shown on the display. This is useful when you don't want internally stored, but not displayed, digits to be a part of ongoing calculations.

Other options are:

- Input specifies the input method or keystroke order, discussed in Comparison of Input Methods. Use Simple for chained calculations, Precedence to perform binary operations before multiplication and division before addition and subtraction, and RPN for post-fix calculations without parentheses (see the User Guide for more details). The numeric value specifies the size of the stack used for RPN stack operations
- Shift specifies the behavior of the shift button where Sticky causes the shift state to remain active whereas Locking remains active only after being pressed twice in a row
- Action assigns an operation to the action (or center) hardware button
- Checking 100 Registers option causes the user registers to be addressed using 2 digits, otherwise only the first 20 registers can be accessed
- Checking the Backspace Drops option switches the operation from clear x to drop when pressed and not entering a number
- Checking the Comma Radix option displays a comma for the decimal point character

Display

- *Tap* specifies which dialog should be presented when the calculator's display area is tapped. The display is divided into three regions: left, center, and right. First select the region of the display area, then select the dialog to be displayed for that region.
- Theme changes the overall appearance of the calculator including colors, dimensions, and keyboard behavior

- Lines specifies the number of lines displayed on the main calculator screen
- Font specifies the relative size of the font used to display results
- Checking Basic Also specifies the Lines and Font settings be used for the Basic keyboard
- Border specifies the type of border to draw around the display and keyboard
- Checking the Quick Switch option displays a toolbar for switching between the Advanced keyboard function groups

Keyboard

The Keyboard set of options are used to control the dynamic keyboard. See Dynamic Keyboards for a detailed explanation of the dynamic keyboard and these settings.

- Show determines if and where the shifted functions should be shown on the main keyboard
- Size specifies the smallest acceptable size the main function can be reduced to in order to show the shifted functions
- Keys specifies how the shifted keyboard should be displayed
- Color specifies which color should be used to display the shifted functions on the main keyboard
- Highlight specifies which color should be used to display the shifted functions on the shifted keyboard
- Shadow specifies the type of drop shadow used for displaying the current function

Formatting Numbers

In conjunction with setting the display format in the Preferences dialog, the `std`, `fix`, `sci`, and `eng` buttons can be used to format the numbers in the display. Press one of these buttons to choose the format type followed by a single digit to specify the number of digits past the decimal point. Use the `.` button to specify All digits.

Entering Numbers

Pressing any digit, the decimal point, or the exponent button causes the calculator to go into a special mode for entering numbers. The display format is not used; rather, the buttons pressed to enter the number are displayed. The rightmost character in the display gets erased when the `←` button is pressed.

When entering negative numbers, first enter the digits for the number, then press the $\boxed{+/-}$ button. If you press the $\boxed{+/-}$ button first, it will change the sign of the number currently being displayed, not the number you're about to enter.

When entering numbers in exponential format, first enter the mantissa, then press $\boxed{EE\%}$, and then enter the exponent. If the number is negative, press $\boxed{+/-}$ *before* pressing $\boxed{EE\%}$. If the exponent is negative, press $\boxed{+/-}$ *after* pressing $\boxed{EE\%}$.

If the $\boxed{\leftarrow}$ button is pressed while you're not entering a number, it will have the same effect as pressing \boxed{CE} or $\boxed{CL\%}$. If the $\boxed{\leftarrow}$ button is pressed after a prefix key, the prefix is cancelled.

Pressing any key to perform an operation will terminate the "entering a number" mode and the display will be formatted using the display settings.

Customizable Keyboard

The Keyboard dialog is used to customize the which keys should be used for the *User* function group or mode. The top portion of the dialog shows the customized key layout while the bottom portion displays the collection of keys that can be assigned to the custom layout.

The Integer Mode checkbox specifies whether NeoCal should enter integer mode when the user functions are selected. This behavior is required for Programmer mode and is duplicated in User mode when this option is checked. Toggling the Shifted option shows the keys in the specified shifted state.

Any of the keys from the other function groups can be assigned in User mode:

1. Press the user key to be assigned.
2. Locate the desired key from the available keys by selecting the function group and shift state in the lower half of the dialog.
3. Press the located key to assign that function to the currently depressed user key.


The Clear button clears the assignment for all of the user keys in the current shift state. The Copy button assigns all of the displayed assignable keys to the displayed user keys in the same location.

Undo/Redo Support

The last ten operations can be reverted by selecting the Undo menu option from the Edit menu. An undo operation can be undone by selecting the Redo menu option. The entire

state of the calculator, including stack and register values, are restored to the state before the previous operation was performed. Continue selecting the Undo operation to continue reverting the calculator to previous states.





When another operation is performed after an undo operation, the redo information is lost. Also, the undo information is lost when switching to another application.

The  button should be used to correct mistakes while entering numbers, as the undo operation will undo the entering number operation.

Settings in the Preferences dialog are not affected by the Undo/Redo operations.

Macro Record and Playback

A macro stores a sequence of keystrokes or button presses that can be played back at a later time for simple automation of a common calculation. Only a single macro may be recorded or stored at a time.

- Press  to start recording a new macro. Any previously recorded macro is permanently deleted.
- Press  to pause recording or playback of the macro. Press it a second time to resume recording or playback.
- Press  to stop recording or playback of a macro.
- Press  to playback any previously recorded macro. It may be pressed again to resume a macro that has been paused for user input. The macro is played back using the same input method used to record it.

Using pause while recording a macro will pause the macro during playback allowing for input from the user. Keystrokes are not recorded while the macro is paused during recording, however, the user should provide similar keystrokes during playback before resuming the macro. Mistakes made during recording can be corrected using the undo operation.

Alternative Input

The basic operations of the calculator like entering numbers and performing addition and subtraction, can be entered from the Input Method area or from an external keyboard.

Here are some tips:

- The enter or return character can be used for the Equals button in non-RPN modes and for the Enter button when using RPN mode.

- The period or comma character can be used for the radix mark.
- The asterisk (*) or the x character can be used for multiplication.
- The slash (/) character is used for division.
- The S and R characters are used for the STO and RCL operations.
- The backspace key is used for the Backspace operation.
- The delete key is used for the Clear (Stack) operation.

Using NeoCal with Other Applications

The complete state of the calculator is maintained so that you can switch from NeoCal in the middle of a calculation, bring up another application, and then return to NeoCal and finish the calculation.

NeoCal also supports the copy and paste commands allowing you to exchange numbers with other applications. When pasting into NeoCal, the first recognizable number is extracted from any surrounding text.

Using Registers

An important concept of any calculator is the register. Each register can store a single value and is used for performing complex calculations or for remembering values for later use. These storage registers are further grouped into user (or numbered), financial, statistical, and date registers.

Working Registers

NeoCal uses several registers for performing calculations with the most commonly used labeled x , y , and *last x*. The x register contains the number shown in the display, while the y register is paired with the x register when two values are required for a calculation or returned from a calculation. The *last x* register stores the value of the x register before the last numeric operation was performed.

As an example, let's add the numbers 12 and 15:

1. Pressing $\boxed{1}$ $\boxed{2}$ places the number 12 in the x register.
2. Pressing $\boxed{+}$ copies the value stored in the x register into the y register.
3. Pressing $\boxed{1}$ $\boxed{5}$ stores the number 15 in the x register.
4. Pressing $\boxed{=}$ adds the numbers in the x and y registers and places the result in the x register (which is shown on the display). The y register contains the value it contained before we started this calculation. The *last x* register contains the number 15.

Many keys are labeled with x ($\boxed{1/x}$) or x and y ($\boxed{y^x}$) referring to the x and y registers. The general rule is to enter the y value first, then the x value. Also,

- Pressing the $\boxed{x\leftrightarrow y}$ button will exchange or swap the values stored in the x and y registers. This is often used when two values are returned from an operation and you want to view the additional value.
- Pressing the \boxed{LSTx} button recalls the value stored in the *last x* register into the x register. This allows a number to be used repeatedly without having to re-enter it.

The RPN input method also recognizes two more registers, z and t , which are discussed in RPN Tutorial.

The M Register

The *M* register is provided as a convenience when using the Simple or Precedence input method and is generally used to store intermediate values during the calculation. The following keys are used to access this register:

- \boxed{CM} is used to clear the *M* register, or set its value to zero.
- $\boxed{\rightarrow M}$ stores the value in *x* into the *M* register.
- \boxed{RM} recalls the *M* register by placing its value into *x*.
- $\boxed{M+}$ adds *x* to the value already stored in the *M* register.
- $\boxed{M-}$ subtracts *x* from the value currently stored in the *M* register.

The *MEM* indicator is displayed when the value of this register is nonzero.

User Registers

NeoCal has 100 storage registers for you to keep numbers beyond the current calculation. They are also useful for storing intermediate results for the current calculation or a series of calculations. If the 100 Registers options is not checked in the Preference Dialog, only twenty registers can be accessed and are numbered from 0 through 9 and .0 through .9. If the 100 Registers option is checked, then all 100 registers can be accessed by pressing two digits 00 through 99. In either case, the registers are accessed using the these prefix buttons followed by the register number:

- \boxed{STO} is used to store *x* into a specified register. The *STO* indicator is displayed.
- \boxed{RCL} is used to recall a register's value into the *x* register. The *RCL* indicator is displayed.
- $\boxed{\times \leftrightarrow}$ exchanges or swaps the value stored in the *x* register with the value stored in the specified register. The *SWP* indicator is displayed.

The $\boxed{\leftarrow}$ button can be used to cancel the register prefix. Also, pressing a register prefix button twice in a row is a shortcut for displaying the User Registers dialog where tapping on one of the register values completes that operation.

For example (using 20 registers),

1. Pressing $\boxed{1} \boxed{7} \boxed{STO} \boxed{0}$ stores the number 17 in both the *x* register (by entering the number) and register 0.

2. Pressing **2** **4** **STO** **.** **5** stores the number 25 in both the *x* register and register 15 (sometimes referred to as register .5).
3. Pressing **RCL** **0** recalls the number 17 into the *x* register.
4. Pressing **3** **3** **×\$** **.** **5** stores 33 in register 15 and recalls 24 into the *x* register.

The same example using 100 registers,

1. Pressing **1** **7** **STO** **0** **0** stores the number 17 in both the *x* register (by entering the number) and register 0.
2. Pressing **2** **4** **STO** **1** **5** stores the number 25 in both the *x* register and register 15.
3. Pressing **RCL** **0** **0** recalls the number 17 into the *x* register.
4. Pressing **3** **3** **×\$** **1** **5** stores 33 in register .5 and recalls 24 into the *x* register.

The store and recall operations can be used in conjunction with the add, subtract, multiply, and divide operations. Continuing the previous example,

5. Pressing **3** **STO** **+** **0** **0** stores 3 into the *x* register and 20 (17 + 3) into register 0.
6. Pressing **RCL** **-** **1** **5** recalls -30 (3 - 33) into the *x* register.
7. Pressing **1** **1** **STO** **÷** **1** **5** stores 11 in the *x* register and -3 (-33 / 11) in register 15.
8. Pressing **RCL** **×** **0** **0** recalls 220 (11 x 20) into the *x* register.

In addition to these general-purpose storage registers, there are ten financial registers and six statistical registers. They are discussed in the Financial Functions and Statistical Functions chapters of this guide, respectively.

User Registers Dialog

The User Registers dialog displays the values of all 20 or 100 user registers. The values are displayed in groups of ten which are selected at the top of the form.

Invoking the dialog through the menu or by tapping in the display area, shows the **STO**, **RCL**, **×\$**, and **×\$y** buttons. This allows multiple operations to be performed before the dialog is dismissed using the Done button. To perform an operation, simply tap on the

register prefix button and then tap on the register value. The $\boxed{\times\div y}$ button is provided to facilitate moving values between multiple (more than two) user registers.

When the dialog is invoked by pressing the same register prefix button twice, tap on the register value to complete the operation and dismiss the dialog. You can select the register group being displayed, to locate the register needed to complete the operation, without dismissing the dialog.

Clearing The Registers

Storing zero into any register clears that register. The following keys are used to clear groups of registers:

- \boxed{C} or \boxed{CLst} clears the x , y , z , t , and *last x* registers. This is also referred to as clearing the stack and should be performed before starting a new set of calculations. This operation also returns the current input method to the setting specified in the Preferences dialog.
- \boxed{CE} or \boxed{CLx} clears the x register. This should be used when an incorrect number has been entered and you'd like to start entering the number again.
- \boxed{CR} or \boxed{CLR} clears or resets all registers, clears the keyboard macro, and resets the calculator state. It does not modify the settings in the Preferences dialog.
- \boxed{CLr} clears the user, financial, and statistical registers.
- \boxed{CLf} clears the financial registers and displays the number of compounding periods per year. This function also clears any user registers that currently contain a cash flow entry, leaving all other user registers unmodified.
- $\boxed{CL\text{E}}$ clears the statistical registers used to hold a summary of the entered data values.
- \boxed{CLd} resets the date conversion registers to their default values (except \boxed{fps}) and displays the number of frames per second.

General Functions

It's always a good idea to clear the calculator before starting new calculations by pressing either the `CLst` or `C` buttons. All of the functions in this section store the value of the x register into the *last x* register before performing the operation.

One-Number Functions

The following list of functions simply modify the value of the x register:

- `1/x` calculates the inverse of x , or x^{-1} .
- `x2` calculates the square of x , or x multiplied by x .
- `√x` calculates the square root of x .
- `int` returns the integer portion of x .
- `frac` returns the fractional portion of x .
- `abs` returns the absolute value of x .
- `ceil` returns the smallest integral value not less than x .
- `floor` returns the largest integral value not greater than x .

Two-Number Functions

These functions require two numbers as input and store their result in the x register:

- `+` calculates the addition of x and y .
- `-` calculates the subtraction of x from y .
- `×` calculates the multiplication of x and y .
- `÷` calculates y divided by x .
- `mod` calculates the remainder of y divided by x .
- `yx` calculates y raised to the power of x .
- `∛y` calculates the x root of y .

Percentages

Percent

Normally, the $\boxed{\%}$ function returns x divided by 100. However, it does perform a more useful function when the previous operation was addition or subtraction, then the number returned is the specified percentage of y . For example,

- Pressing \boxed{C} $\boxed{1}$ $\boxed{5}$ $\boxed{\%}$ displays the number 0.15.
- Pressing \boxed{C} $\boxed{5}$ $\boxed{0}$ $\boxed{+}$ $\boxed{3}$ $\boxed{0}$ $\boxed{\%}$ displays the number 15 (since 30% of 50 is 15). The result after pressing $\boxed{=}$ is 65.

In RPN mode, the result is always the specified percentage of y . The value of y is preserved so that subsequent operations can be performed without having to re-enter the base number.

Percent Change

The $\boxed{\Delta\%}$ function calculates the percent difference between two numbers.

In RPN mode, the value of y is preserved.

Percent of Total

The $\boxed{\%T}$ function calculates the percentage that x is of y . For example,

- Pressing $\boxed{5}$ $\boxed{0}$ $\boxed{\%T}$ $\boxed{3}$ $\boxed{0}$ results in 60 (since 30 is equal to 60% of 50).

In RPN mode, the value of y is preserved.

Financial Functions

Financial Registers

NeoCal contains ten registers used specifically for the financial applications:

- **cost** is used for cost
- **price** is used for price
- **mar** and **mu** are used for margin and markup
- **P/YR** is used for the number of periods or payments per year
- **N** is used for the total number of periods or payments
- **I/YR** and **NM%** are used for the annual nominal interest rate
- **PV** is used for present value
- **PMT** is used for the periodic payment
- **FV** is used for future value
- **EFF%** is used for the annual effective rate

Just like the user registers, these registers are accessed using the **STO** and **RCL** buttons. However, the financial registers perform a calculation when their button is pressed without being preceded by **STO** or **RCL**. As a convenience, pressing one of these buttons *while entering a number* will cause any pending operation to be performed and then store that number in the register.

For example,

- Pressing **STO** **FV** will store the displayed number in the future value register.
- Pressing **RCL** **price** will display the value stored in the price register.
- Pressing **7** **.** **5** **I/YR** will store 7.5% in the annual interest rate register.
- Pressing **4** **×** **1** **5** **N** will store 60 in the N register.
- Pressing **PV** will calculate, store, and display the present value.

The financial registers and any user registers currently holding a cash flow entry are cleared by pressing the **CLF** button. As a reminder, the value of the **P/YR** register is displayed.

Margin and Markup

This application uses the **cost**, **price**, **mar**, and **mu** registers. Keeping in mind that the margin and markup registers are actually the same, any of these values can be calculated from the other two registers.

For example,

1. Pressing **1** **0** **cost** stores 10 in the cost register.
2. Pressing **2** **0** **price** stores 20 in the price register.
3. Pressing **mar** calculates the margin to be 50 percent.
4. Pressing **mu** calculates the markup to be 100 percent.
5. Pressing **3** **0** **mar** stores 30 percent in the margin register.
6. Pressing **price** calculates the price to be 14.29 (30% margin with a cost of 10, rounded to two decimal places).
7. Pressing **RCL** **cost** displays the cost of 10.
8. Pressing **RCL** **mu** displays the margin of 30 percent. Since the margin/markup register was last used to store the margin, recalling its value (using either button) will display the margin.
9. Pressing **mu** calculates the markup to be 42.86 percent.

Interest Conversion

This application uses the **NM%**, **EFF%**, and **P/YR** registers to convert between the annual nominal and effective interest rates. The effective rate takes compounding into consideration based on the number of periods per year.

1. Pressing **1** **2** **P/YR** stores 12 in the periods per year register. This is equivalent to monthly compounding.
2. Pressing **6** **NM%** stores 6 percent into the nominal interest rate register. This is equivalent to 0.5 percent per month.
3. Pressing **EFF%** calculates and stores 6.17 percent for the annual effective rate.

Since **NM%** and **I/YR** share the same register, the annual effective rate can easily be converted for use by the time value of money application. In the United States, the annual effective rate is equivalent to the annual percentage rate (APR).

Time Value of Money

This application can calculate any single value of the \boxed{N} , $\boxed{I/YR}$, \boxed{PV} , \boxed{PMT} , and \boxed{FV} registers based on the values of the other four registers. This is useful for compound interest calculations involving regular, uniform payments such as loans, mortgages, leases, and annuities. There must be at least one negative and one positive cash flow (\boxed{PV} , \boxed{PMT} , and/or \boxed{FV}). In addition,

- The value stored in the $\boxed{P/YR}$ register is used to convert the annual nominal rate ($\boxed{I/YR}$) to the periodic rate (which is required for the calculations).
- $\boxed{\times P}$ stores the displayed number multiplied by the value in the $\boxed{P/YR}$ register into the \boxed{N} register.
- $\boxed{\div P}$ divides the displayed value by the value in the $\boxed{P/YR}$ register.
- \boxed{beg} switches between *begin* and *end* modes and is used to specify whether the payments occur at the beginning or the end of the period. When *begin* mode is selected, the *BEG* indicator is displayed, otherwise *end* mode is in effect.

The TVM dialog displays the current values of all the registers used by the various Time Value of Money calculations. Use the Preferences dialog to assign a section of the display for quicker access to this dialog.

For an example, let's consider a car loan. What would the payments be for a three-year loan of \$16,500 with an APR of 9.5 percent?

1. If the calculator is in *begin* mode, press \boxed{beg} to set *end* mode.
2. Press \boxed{CLf} to clear the financial registers.
3. Press $\boxed{1} \boxed{2} \boxed{P/YR}$ for monthly payments.
4. Press $\boxed{9} \boxed{.} \boxed{5} \boxed{EFF\%}$ to store the annual effective rate.
5. Press $\boxed{NM\%}$ to calculate the annual nominal rate of 9.11 percent (which is stored in the $\boxed{I/YR}$ register).
6. Press $\boxed{3} \boxed{\times P}$ to store 36 in the \boxed{N} register for the total number of payments.
7. Press $\boxed{1} \boxed{6} \boxed{5} \boxed{0} \boxed{0} \boxed{PV}$ to store amount to be financed.
8. Press \boxed{PMT} to calculate -525.54 as the monthly payment amount. By convention, positive amounts designate money you receive and negative amounts designate money you pay.

As a comparison, let's calculate the payment for a four-year loan:

9. Press **4** **×P** to store 48 for the total number of payments.
10. Press **PMT** to calculate -411.46 as the monthly payment amount.

Amortization

This application uses the **I/YR**, **PV**, **PMT**, and **FV/YR** registers to calculate the amount of interest and principal for a range of payments. The remaining balance is also calculated.

- Pressing **amrt** while entering a number will amortize for the range of payments starting with y and ending with x (inclusive).
- Pressing **amrt** by itself will calculate the interest, principal, and balance for the range of payments currently entered (which defaults to the first payment, or range 1-1).
- Pressing **anxt** will amortize the next range of payments. For instance, if the current range of payments is 1-12 (the first year), then the next range is 13-24 (the second year).
- Pressing **apr'v** will amortize the previous range of payments and is (currently only) available in the Amortization dialog.
- Pressing **=** after pressing **amrt** or **anxt** will display the principal. Pressing **=** a second time will display the balance. As a convenience, pressing **=** a third time will calculate the amortization for the next range of payments.
- In RPN mode, press **R↓** or **drop** to display the principal. Pressing **R↓** a second time will display the balance. Press **anxt** to calculate the amortization for the next range of payments.
- Pressing **STO** **amrt** will store y and x as the starting and ending payments. (Pressing **amrt** a second time will perform the calculation.)
- Pressing **RCL** **amrt** recalls the starting and ending payments into the y and x registers, respectively.
- **beg** switches between *begin* and *end* modes and is used to specify whether compounding occurs at the beginning or the end of the period. When *begin* mode is selected, the *BEGIN* indicator is displayed, otherwise *end* mode is in effect.

The amortization calculation is sensitive to rounding errors and may not match your financial institution's calculations exactly (different institutions use different rounding rules). NeoCal will round the intermediate results based on the display setting for the

number of digits to the right of the decimal place, therefore, this setting affects the results.

The Amortization dialog provides a convenient way of viewing all the registers used in the amortization calculation along with the calculated interest, principal, and balance. Use the Preferences dialog to assign a section of the display for quicker access to this dialog.

Let's look at a mortgage example, \$200,000 for 30 years at an annual nominal rate of 7 percent with payments and compounding occurring monthly.

1. Set the display to show two decimal places in the Preferences dialog.
2. If the calculator is in *begin* mode, press $\boxed{\text{beg}}$ to set *end* mode.
3. Press $\boxed{\text{CLf}}$ to clear the financial registers.
4. Press $\boxed{1} \boxed{2} \boxed{\text{P/YR}}$ for monthly payments.
5. Press $\boxed{3} \boxed{0} \boxed{\times\text{P}}$ to store the loan duration in the $\boxed{\text{N}}$ register.
6. Press $\boxed{7} \boxed{\text{I/YR}}$ to store the interest rate.
7. Press $\boxed{2} \boxed{\text{EEX}} \boxed{5} \boxed{\text{PV}}$ to store the loan amount in the present value register.
8. Press $\boxed{\text{PMT}}$ to calculate the monthly payment amount (-1330.60).
9. Press $\boxed{\text{amrt}}$ to amortize the first payment and display the interest amount (-1,166.67).
10. Press $\boxed{=}$ (in RPN mode, press $\boxed{\text{R}\downarrow}$) to display the principal for the first payment (-163.93).
11. Press $\boxed{=}$ (in RPN mode, press $\boxed{\text{R}\downarrow}$) to display the balance after the first payment (199,836.07).
12. Press $\boxed{=}$ (in RPN mode, press $\boxed{\text{anxt}}$) to amortize the second payment and display the interest amount (-1,165.71). This process can be repeated through the term of the loan.

To calculate the amount of interest paid during the first year of the loan,

13. Press $\boxed{1} \boxed{\text{SPC}}$ (in RPN mode, use $\boxed{\text{Enter}}$) $\boxed{1} \boxed{2} \boxed{\text{amrt}}$ to amortize the first twelve payments and display the interest amount (-13,935.65).
14. Press $\boxed{\text{anxt}}$ to amortize the second year's payments and display the interest amount (-13,788.78).

At any time, you can press $\boxed{\text{RCL}} \boxed{\text{amrt}}$ to recall the starting and ending payments for the range being amortized. (Use the $\boxed{\text{xy}}$ button to display the starting payment.)

Cash Flow Analysis

This application uses the $\boxed{\text{I/YR}}$ and $\boxed{\text{P/YR}}$ registers. In addition, the user registers are used to store the cash flow entries. The net present value ($\boxed{\text{NPV}}$) function uses the $\boxed{\text{I/YR}}$ register to calculate the present value of a series of cash flows and stores the result in the $\boxed{\text{PV}}$ register. The internal rate of return ($\boxed{\text{IRR}}$) function calculates the annual nominal interest rate required to give a net present value of zero and stores the result in the $\boxed{\text{I/YR}}$ register.

All cash flows must occur at regular intervals, but can be of differing amounts. Each cash flow entry consists of an amount and the number of times it occurs consecutively. Up to 99 cash flow entries can be stored after the initial cash flow. The amount of the initial cash flow is stored in register 0, the amount of the first cash flow is stored in register 1, the amount of the second cash flow is stored in register 2, and so on. When you enter a series of cash flows, you must account for every period, even those with a zero amount.

To enter the cash flows,

1. Press $\boxed{\text{CLF}}$ to clear the financial registers and any cash flow entries.
2. Enter the number of periods per year in the $\boxed{\text{P/YR}}$ register, if necessary.
3. Enter the initial cash flow amount and press $\boxed{\text{CFj}}$.
4. Enter the next cash flow amount and press $\boxed{\text{CFj}}$.
5. If this cash flow entry occurs more than once consecutively, enter the number of times it occurs and press $\boxed{\text{Nj}}$.
6. Repeat steps 4 and 5 until all the cash flow entries have been entered.

To review all of the cash flow entries,

1. Press $\boxed{\text{RCL}} \boxed{0}$ (or $\boxed{\text{RCL}} \boxed{0} \boxed{0}$ if the 100 Registers option is enabled in the Preferences dialog) to display the initial cash flow amount.
2. Press $\boxed{\text{RCL}} \boxed{\text{CFj}}$ to display the amount of the next cash flow entry.
3. Press $\boxed{\text{RCL}} \boxed{\text{Nj}}$ to view the number of times that cash flow amount occurs.
4. Repeat steps 2 and 3 until all the cash flow entries have been reviewed.

The Cash Flow dialog can also be used to review the cash flow entries. Use the Preferences dialog to assign a section of the display for quicker access to this dialog.

To review and edit the cash flow entries,

- Press $\boxed{\text{RCL}}$ followed by the number of the cash flow entry to display that amount.
- Press $\boxed{\text{STO}}$ followed by the number of the cash flow entry to store that amount.
- Pressing $\boxed{\text{RCL}} \boxed{\text{CF}_j}$ displays the next cash flow amount.
- Pressing $\boxed{\text{STO}} \boxed{\text{CF}_j}$ stores the next cash flow amount, but does not change the number of times the cash flow occurs.
- Pressing $\boxed{\text{RCL}} \boxed{\text{N}_j}$ displays the number of times the current cash flow occurs. The current cash flow entry is the entry whose amount was last stored or recalled.
- Pressing $\boxed{\text{STO}} \boxed{\text{N}_j}$ stores the number of times the current cash flow occurs.
- The cash flow amount can be changed by storing a new amount into the corresponding register, provided that the amount was initially stored using the $\boxed{\text{CF}_j}$ button.
- To add a new cash flow entry, first recall the last cash flow amount, then enter the new cash flow amount using the $\boxed{\text{CF}_j}$ button.

Let's suppose that you purchase \$1000 worth of stock at the beginning of each of the next 12 months and then sell the stock two years from now for \$16,000. What is the annual internal rate of return?

1. Press $\boxed{\text{CLF}}$ to clear the registers.
2. Press $\boxed{1} \boxed{2} \boxed{\text{P/YR}}$ to store the number of periods per year.
3. Press $\boxed{1} \boxed{0} \boxed{0} \boxed{0} \boxed{+/-} \boxed{\text{CF}_j}$ to store the initial cash flow.
4. Press $\boxed{\text{CF}_j}$ to store the first cash flow entry.
5. Press $\boxed{1} \boxed{1} \boxed{\text{N}_j}$ to store the number of occurrences for this cash flow.
6. Press $\boxed{0} \boxed{\text{CF}_j}$ to store the second cash flow entry.
7. Press $\boxed{1} \boxed{2} \boxed{\text{N}_j}$ to store the number of occurrences for this cash flow.
8. Press $\boxed{1} \boxed{6} \boxed{0} \boxed{0} \boxed{0} \boxed{\text{CF}_j}$ to store the last cash flow amount.
9. Press $\boxed{\text{IRR}}$ to calculate the annual internal rate of return (18.71 percent) and store it in the $\boxed{\text{I/YR}}$ register.

As another example, let's calculate the present value of twelve monthly payments of \$1000 assuming 8 percent nominal annual interest rate and the first payment starting in one month from now:

1. Press **CLf** to clear the registers.
2. Press **1** **2** **F/YR** to store the number of periods per year.
3. Press **8** **I/YR** to store the interest rate.
4. Press **0** **CFj** to store the initial cash flow.
5. Press **1** **0** **0** **0** **CFj** to store the amount of the initial cash flow entry.
6. Press **1** **2** **Nj** to store the number of occurrences.
7. Press **NPV** to calculate the net present value (11,495.78) and store the result in the **PV** register.

Scientific Functions

Trigonometry

The angular mode buttons (`deg`, `rad`, and `grad`) specify whether angles are measured in degrees, radians, or grads. The *RAD* and *GRAD* indicators are displayed to indicate those modes, while the absence of these indicators implies degrees mode.

The following functions modify the number in the *x* register:

- `sin` calculates the sine of *x*.
- `cos` calculates the cosine of *x*.
- `tan` calculates the tangent of *x*.
- `asin` calculates the arc sine of *x*.
- `acos` calculates the arc cosine of *x*.
- `atan` calculates the arc tangent of *x*.

Pressing `hyp` before any of the above functions accesses the hyperbolic functions. The `π` button recalls the value of pi into the *x* register.

Logarithms

The following functions modify the number in the *x* register:

- `ln` calculates the natural log of *x*.
- `ex` calculates the natural antilogarithm of *x*.
- `log` calculates the common log of *x*.
- `10x` calculates the common antilogarithm of *x*.

Conversions

The following functions provide useful conversions:

- `→deg` converts an angle measured in radians to degrees.
- `→rad` converts an angle measured in degrees to radians.
- `→rec` converts the polar coordinate (*r*, *theta*) into the rectangular coordinate (*x*, *y*).
Theta should be expressed in units corresponding to the current angular mode.

- $\boxed{\rightarrow\text{pol}}$ converts the rectangular coordinate (x, y) into the polar coordinate (r, theta). Theta is returned in units corresponding to the current angular mode.
- $\boxed{\rightarrow\text{hms}}$ converts a number from a decimal-fraction (H.h) to minutes-second (H.MMSSs) format. This is useful for time expressed in hours and angles expressed in degrees.
- $\boxed{\rightarrow\text{hr}}$ converts a number in minutes-second format to decimal-fraction format. All of the calculator's functions (except this one) require numbers to be in decimal-fraction format.

For example,

1. Press $\boxed{\pi}$ $\boxed{\rightarrow\text{deg}}$ to convert pi to 180 degrees.
2. Press $\boxed{3}$ $\boxed{6}$ $\boxed{0}$ $\boxed{\rightarrow\text{rad}}$ to convert 360 degrees to 2 pi radians.
3. Press $\boxed{\text{deg}}$ $\boxed{2}$ $\boxed{\text{SPC}}$ (in RPN mode, use $\boxed{\text{Enter}}$) $\boxed{1}$ $\boxed{\rightarrow\text{pol}}$ to convert the rectangular coordinate (x = 1, y = 2) to the polar coordinate (magnitude = 2.2361, angle = 63.4349 degrees). (Press $\boxed{\times\div}$ button to view theta.)
4. Press $\boxed{2}$ $\boxed{.}$ $\boxed{5}$ $\boxed{\rightarrow\text{hms}}$ to convert 2.5 hours to 2 hours 30 minutes (2.30000 or 2"30'00.0).
5. Press $\boxed{4}$ $\boxed{.}$ $\boxed{2}$ $\boxed{3}$ $\boxed{4}$ $\boxed{8}$ $\boxed{\rightarrow\text{hr}}$ to convert 4 hours 23 minutes 48 seconds to 4.3967 hours.

Physical constants

The Constants dialog lists many common physical constants in SI units that can be recalled into the x register. Use the Preferences dialog to assign a section of the display for quicker access to this dialog.

The following constants are listed in the Constants dialog:

- Avogadro's number, NA: $6.0221367\text{e}23 \text{ gmol}^{-1}$
- Boltzmann constant, k: $1.380658\text{e}-23 \text{ J/K}$
- Molar volume, Vm: 22.4141 l/gmol
- Universal gas constant, R: $8.31451 \text{ J/(gmol K)}$
- Standard temperature, StdT: 273.15 K
- Standard pressure, StdP: 101.325 kPa
- Stefan-Boltzmann constant: $5.67051\text{e}-8 \text{ W/(m}^2 \text{ K}^4)$

- Speed of light in vacuum, c: 299792458 m/s
- Permittivity of vacuum: 8.85418781761e-12 F/m
- Permeability of vacuum: 1.25663706144e-6 H/m
- Acceleration of gravity, g: 9.80665 m/s²
- Gravitational constant, G: 6.67259e-11 m³/(s² kg)
- Planck's constant, h: 6.6260755e-34 J s
- Dirac's constant, hbar: 1.05457266e-34 J s
- Electron charge, q: 1.60217733e-19 C
- Electron rest mass, me: 9.1093897e-31 kg
- q/me, qme: 175881962000 C/kg
- Proton rest mass, mp: 1.6726231e-27 kg
- mp/me, mpme: 1836.152701
- Fine structure constant: 0.00729735308
- Magnetic flux quantum: 2.06783461e-15 Wb
- Faraday constant, F: 96485.309 C/gmol
- Rydberg constant: 10973731.534 m⁻¹
- Bohr radius, a0: 0.0529177249 nm
- Bohr magneton: 9.2740154e-24 J/T
- Nuclear magneton: 5.0507866e-27 J/T
- Photon wavelength (ch/e): 1239.8425 nm
- Photon frequency (e/h), f0: 2.4179883e14 Hz
- Compton wavelength: 0.00242631058 nm
- Wien displacement constant, c3: 0.002897756 m K
- Dielectric constant: 11.9
- SiO₂ dielectric constant: 3.9
- Reference intensity, I0: 0.000000000001 W/m²

Statistical Functions

Statistical Registers

This application uses the statistical registers and performs one- and two-variable statistical calculations such as mean, standard deviation, and linear regression. The statistical registers are represented by the \boxed{n} , $\boxed{\Sigma x}$, $\boxed{\Sigma x^2}$, $\boxed{\Sigma y}$, $\boxed{\Sigma y^2}$, and $\boxed{\Sigma xy}$ buttons. However, these registers are not modified using the \boxed{STO} button, rather, data is accumulated into these registers by using:

- $\boxed{CL\Sigma}$ to clear the statistical registers.
- $\boxed{\Sigma+}$ to accumulate a data point into the statistical registers. The data point can be either a single x value or an (x, y) pair. The number of accumulated data points is displayed. As with all NeoCal operations, the y value is entered before the x value.
- $\boxed{\Sigma-}$ to remove a data point from the statistical registers. This is useful for correcting mistakes.

The accumulated statistics can be recalled by simply pressing these buttons:

- \boxed{n} recalls the number of data points accumulated.
- $\boxed{\Sigma x}$ recalls the summation of the x values.
- $\boxed{\Sigma x^2}$ recalls the summation of the squares of the x values.
- $\boxed{\Sigma y}$ recalls the summation of the y values.
- $\boxed{\Sigma y^2}$ recalls the summation of the squares of the y values.
- $\boxed{\Sigma xy}$ recalls the summation of the products of the x and y values.
- $\boxed{RCL} \boxed{\Sigma+}$ recalls the summation of the x values into the x register and the summation of the y values into the y register.

The Statistical dialog displays the current values of all the summary registers used by the statistical calculations. Use the Preferences dialog to assign a section of the display for quicker access to this dialog.

Let's enter some data,

1. Press $\boxed{CL\Sigma}$ to clear the statistical registers.

2. Press $\boxed{1} \boxed{0} \boxed{\text{SPC}}$ (in RPN mode, use $\boxed{\text{Enter}}$) $\boxed{4} \boxed{0} \boxed{\Sigma+}$ to accumulate the first data point.
3. Press $\boxed{7} \boxed{\text{SPC}} \boxed{3} \boxed{5} \boxed{\Sigma+}$ to accumulate the second data point.
4. Press $\boxed{1} \boxed{5} \boxed{\text{SPC}} \boxed{7} \boxed{0} \boxed{\Sigma+}$ to accumulate the third data point.
5. Press $\boxed{3} \boxed{1} \boxed{\text{SPC}} \boxed{1} \boxed{0} \boxed{0} \boxed{\Sigma+}$ to accumulate the fourth data point.
6. Press $\boxed{\Sigma x}$ to recall the sum of the x values (245).
7. Press $\boxed{\Sigma x^2}$ to recall the sum of the squares of the x values (17,725).
8. Press \boxed{n} to recall the number of accumulated data points.

Mean and Standard Deviation

The mean and standard deviation calculations are based upon the values accumulated in the statistical registers. These buttons are used to perform the calculations:

- $\boxed{\bar{x}}$ calculates and displays the mean (average) of the x values.
- $\boxed{\bar{y}}$ calculates and displays the mean (average) of the y values.
- $\boxed{\bar{x}_w}$ calculates and displays the weighted mean of the x values. The y values are used to determine the weight for the corresponding x value.
- \boxed{s} calculates the standard deviation of the x and y values and returns the results in the x and y registers, respectively.
- $\boxed{\sigma}$ calculates the population standard deviation of the x and y values and returns the results in the x and y registers, respectively.

Continuing the previous example,

9. Press $\boxed{\bar{x}}$ to calculate the mean or average of the x values (61.25).
10. Press $\boxed{\bar{y}}$ to calculate the mean or average of the y values (15.75).
11. Press $\boxed{\bar{x}_w}$ to calculate the weighted mean of the x values (76.11).
12. Press \boxed{s} to calculate and display the standard deviation of the x values (30.10).
13. Press $\boxed{\times s y}$ to display the standard deviation of the y values (10.69).

Linear Regression

Linear regression involves fitting a straight line through the set of two or more data points accumulated in the statistical registers. Linear estimation uses this line to calculate an estimate of either x or y, based on the other.

- \hat{x},r estimates x for a given y value. The correlation coefficient (r) is returned in the y register.
- \hat{y},m estimates y for a given x value. The slope (m) of the line is returned in the y register. The y intercept of the line can be calculated by estimating y for x = 0.

Continuing the previous example,

14. Press $\boxed{1} \boxed{2} \hat{x},r$ to estimate the x value (50.36) for the given y value.
15. Press $\times\hat{y}$ to display the correlation coefficient (0.97).
16. Press $\boxed{0} \hat{y},m$ to estimate and display the y value (-5.34) for the given x value. This is the y-intercept of the fitted line.
17. Press $\times\hat{y}$ to display the slope (0.34) of the fitted line.

Probability

The input for these calculations is restricted to non-negative integers.

- $C_{y,x}$ calculates the number of possible sets of y different items taken x at a time.
- $P_{y,x}$ calculates the number of possible different arrangements of y different items taken x at a time.
- $\times!$ calculates the factorial of the displayed number.
- Γx calculates the gamma of the displayed number.
- ran returns a pseudo-random number between 0 and 1.

For example,

1. Press $\boxed{5} \boxed{2} C_{y,x} \boxed{5}$ to calculate the number of different five-card hands that can be dealt from a deck of cards (2,598,960).
2. Press $\boxed{1} \boxed{0} \times!$ to calculate the factorial of 10 (3,628,800).
3. Press $\boxed{1} \boxed{0} \Gamma x$ to calculate the gamma of 10 (362,880).
4. Press ran to generate a number between zero and one.

5. Press to generate a different number between zero and one.

Programmer Functions

Selecting the *Programmer* group of functions places the calculator into integer mode, with the type of integer specified in the Preferences dialog. The $\boxed{+}$, $\boxed{-}$, $\boxed{\times}$, $\boxed{\div}$, and $\boxed{+/-}$ buttons operate on the integer part of their operands and return an integer for their result. The $\boxed{\text{rnd}}$ button also works differently in integer mode by converting x to the type of integer specified in the Preferences dialog.

Switching away from the *Programmer* function group restores these functions to work with real numbers and returns the display to decimal mode. These buttons are marked with a dot in the upper right corner as a reminder that they function differently in integer mode.

Integer mode can also be set for the *User* function group by checking the Integer option in the Keyboard dialog.

Base Conversions

Use these buttons to convert from one base to another:

- $\boxed{\text{bin}}$ displays the number in binary (base 2) format and displays the *BIN* indicator.
- $\boxed{\text{oct}}$ displays the number in octal (base 8) format and displays the *OCT* indicator.
- $\boxed{\text{dec}}$ displays the number in decimal (base 10) format. This is the default and no indicator is displayed.
- $\boxed{\text{hex}}$ displays the number in hexadecimal (base 16) format and displays the *HEX* indicator.

The $\boxed{\text{A}}$ through $\boxed{\text{F}}$ buttons are used for additional digit entry in hexadecimal mode. When large numbers are displayed in binary mode and don't fit on the display, use the $\boxed{\text{show}}$ button to reveal the other bits.

Integer Conversions

These buttons are used for converting between different integer types:

- $\boxed{\text{u8}}$ converts x to an unsigned 8-bit integer.
- $\boxed{\text{s8}}$ converts x to a signed 8-bit integer.
- $\boxed{\text{u16}}$ converts x to an unsigned 16-bit integer.
- $\boxed{\text{s16}}$ converts x to a signed 16-bit integer.

- `u32` converts x to an unsigned 32-bit integer.
- `s32` converts x to a signed 32-bit integer.
- `rnd` converts x to the type of integer specified in the Preferences dialog.

Bitwise Operations

The following operations can be used to manipulate the bits of the stored number:

- `and` calculates y AND x .
- `or` calculates y OR x .
- `xor` calculates y exclusive OR x .
- `not` calculates the 1's complement of x .
- `+/-` calculates the 2's complement of x .
- `rotr` rotates y by x bits. A positive x rotates to the right, while a negative x rotates to the left.
- `«` shifts y to the left by x bits.
- `»` shifts y to the right by x bits.
- `asr` performs an arithmetic shift of y to the right by x bits.

Conversion Functions

The conversions are divided into six types: length, area, volume, speed, weight, and temperature. The conversion buttons are pressed to either assign a unit (when the number does not have a unit, or the unit is not of the same type) or convert to another unit of the same type. In either case, the units assigned to the number are the same as the button.

The conversions are separated into two function groups with the Length, Area, and Speed conversions in the first group and the Volume, Weight (or Mass), and Temperature conversions in the second group. In addition, duration conversions are available in the Date & Time function group and discussed in the Date and Time Functions section.

Length Conversions

The following buttons represent the length conversions:

- for inches
- for feet
- for yards
- for miles
- for nautical miles
- for millimeters
- for centimeters
- for meters
- for kilometers

For example,

1. Press to enter 12 inches.
2. Press to convert to 1 foot.
3. Press to convert to 0.30 meters.

Area Conversions

The following buttons represent the area conversions:

- for square inches

- $\boxed{\text{ft}^2}$ for square feet
- $\boxed{\text{yd}^2}$ for square yards
- $\boxed{\text{mi}^2}$ for square miles
- $\boxed{\text{acre}}$ for acres
- $\boxed{\text{mm}^2}$ for square millimeters
- $\boxed{\text{cm}^2}$ for square centimeters
- $\boxed{\text{m}^2}$ for square meters
- $\boxed{\text{km}^2}$ for square kilometers

For example,

1. Press $\boxed{1}$ $\boxed{\text{m}^2}$ to enter one square meter.
2. Press $\boxed{\text{ft}^2}$ to convert to 10.76 square feet.
3. Press $\boxed{\text{in}^2}$ to convert to 1500 square inches.
4. Press $\boxed{\text{m}^2}$ to convert back to 1 square meter.

Volume Conversions

The following buttons represent the volume conversions:

- $\boxed{\text{tsp}}$ for teaspoons
- $\boxed{\text{tbsp}}$ for tablespoons
- $\boxed{\text{fl.oz}}$ for fluid ounces
- $\boxed{\text{cup}}$ for cups
- $\boxed{\text{pint}}$ for pints
- $\boxed{\text{qt}}$ for quarts
- $\boxed{\text{gal}}$ for gallons
- $\boxed{\text{fl.oz}^{\text{UK}}}$ for fluid ounces UK
- $\boxed{\text{gal}^{\text{UK}}}$ for gallons UK
- $\boxed{\text{ml}}$ for milliliters
- $\boxed{\text{liter}}$ for liters

- $\boxed{\text{in}^3}$ for cubic inches
- $\boxed{\text{ft}^3}$ for cubic feet
- $\boxed{\text{yd}^3}$ for cubic yards
- $\boxed{\text{cm}^3}$ for cubic centimeters
- $\boxed{\text{m}^3}$ for cubic meters

For example,

1. Press $\boxed{1}$ $\boxed{\text{gal}}$ to enter one gallon.
2. Press $\boxed{\text{liter}}$ to convert to 3.79 liters.

Speed Conversions

The following buttons represent the speed conversions:

- $\boxed{\text{ft/s}}$ for feet per second
- $\boxed{\text{mi/h}}$ for miles per hour
- $\boxed{\text{knot}}$ for knots
- $\boxed{\text{cm/s}}$ for centimeters per second
- $\boxed{\text{m/s}}$ for meters per second
- $\boxed{\text{km/h}}$ for kilometers per hour

For example,

1. Press $\boxed{5}$ $\boxed{5}$ $\boxed{\text{mi/h}}$ to enter 55 mph.
2. Press $\boxed{\text{km/h}}$ to convert to 88.51 km/h.

Weight Conversions

The following buttons represent the weight conversions:

- $\boxed{\text{oz}}$ for ounces
- $\boxed{\text{lb}}$ for pounds
- $\boxed{\text{ton}}$ for tons
- $\boxed{\text{mg}}$ for milligrams
- $\boxed{\text{g}}$ for grams

- for kilograms

For example,

1. Press to enter one pound.
2. Press to convert to 16 ounces.
3. Press to convert to 0.45 kilograms.

Temperature Conversions

The following buttons represent the temperature conversions:

- for degrees Fahrenheit
- for degrees Celsius

For example,

1. Press to enter 75 degrees Fahrenheit.
2. Press to convert to 23.89 degrees Celsius.

Unit Arithmetic

Addition and subtraction can be performed on measurements with the same type of unit (except temperature) or with constants. If the units are not the same type, the result will not have a unit assigned to it. If necessary, the second operand's units will be converted to the first operand's units.

For example,

1. Press to add fifteen inches to three feet. The result is 4.25 feet, or 4 feet 3 inches.
2. Press to add one gallon and 3 liters. The result is 1.79 gallons.

Multiplication can be performed on two lengths to generate an area and an area divided by a length will produce a length. The and functions can also be used for these calculations.

Multiplication between an area and a length yields a volume, while a volume divided by an area or length yields the appropriate unit.

For example,

1. Press **3** **ft** **×** **4** **ft** **=** to generate the result of 12 square feet.
2. Press **÷** **6** **ft** **=** for the result of 2 feet.
3. Press **3** **ft** **x²** to generate 9 square feet.

Units are preserved through the following numeric functions: **+/-**, **int**, **frac**, **abs**, and **rnd**.

Date and Time Functions

A *duration* represents a length of time, without starting or ending points. The duration units include years, months, weeks, days, hours, minutes, seconds, and frames. Frames are useful for digital video or film editing applications. Examples of durations include 2.5 years, 13 weeks, 8 days, and 75 minutes.

A *moment* represents a point in time and is expressed as a *date* (August 14, 2002), a *time* (5:15 pm), or a *datetime* (August 14, 2002 5:15 pm) value. NeoCal does not differentiate between timezones or daylight savings time adjustments, nor are leap seconds taken into consideration.

Durations

The duration conversion buttons either assign a duration unit (when the value in the x register is not already a duration) or convert a duration to another unit. The following buttons represent the duration conversions:

- for years
- for months
- for weeks
- for days
- for hours
- for minutes
- for seconds
- for frames

These conversions are determined by values stored in the following date conversion registers:

- for the number of days per year (365.25)
- for the number of days per month (30.4375)
- for the number of days per week (7)
- for the number of hours per day (24)
- for the number of frames per second (30)

Just like the user registers, these registers are accessed using the **STO** and **RCL** buttons. As a convenience, pressing one of these buttons *while entering a number* will cause any pending operation to be performed and then store that number in the register. Otherwise, pressing a date register button will recall its value into the x register.

The **CLd** button resets the date conversion registers to their default values, except for the **fps** register whose value is displayed as a reminder. It is important to clear the date registers before performing a series of date calculations to ensure consistent results. (The **CLd** button also clears the date moment registers.)

The **fp**, or fractional part, button takes the fractional component of a duration and converts it to the next smaller unit.

For example,

1. Press **CLd** to reset the date registers.
2. Press **1** **year** to enter 1 year.
3. Press **12** **month** to convert to 12 months.
4. Press **52.18** **week** to convert to 52.18 weeks.
5. Press **365.25** **day** to convert to 365.25 days.
6. Press **31,557,600** **sec** to convert to 31,557,600 seconds.

This example shows duration conversions based on a forty-hour work week,

1. Press **CLd** to reset the date registers.
2. Press **8** **hpd** to store 8 in the hours per day register.
3. Press **5** **dpw** to store 5 in the days per week register.
4. Press **1** **3** **4** **hour** to enter 134 hours.
5. Press **3.35** **week** to convert to 3.35 weeks.
6. Press **1.75** **fp** to calculate 1.75 days (0.35 weeks).
7. Press **6** **fp** to calculate 6 hours (0.75 days). This shows that 130 hours is the same as 3 weeks, 1 day, and 6 hours.

Duration arithmetic can also be performed, for example:

1. Press **CLd** to reset the date registers.

2. Press **1** **hour** **+** **3** **0** **mnt** **=** to add 1 hour and 30 minutes to generate the result of 1.5 hours.
3. Press **1** **day** **+** **4** **8** **hour** **=** to add 1 day and 48 hours to generate the result of 3 days.
4. Press **3** **week** **-** **1** **2** **day** **=** yields the result of 1.29 weeks. Pressing **fp** shows this to be 1 week and 2 days.

In addition, there are three time conversion functions:

- **→hms** converts hours in decimal-fraction (H.h) format to minutes-second (H.MMSSs) format.
- **→hr** converts hours in minutes-second format to decimal-fraction format.
- **→fr** converts hours in minutes-second format to frames in decimal-fraction format.

For example,

1. Press **2** **.** **5** **→hms** to convert 2.5 hours to 2 hours 30 minutes (2.30000).
2. Press **4** **.** **2** **3** **4** **8** **→hr** to convert 4 hours 23 minutes 48 seconds to 4.3967 hours.
3. Press **0** **.** **0** **1** **3** **0** **→fr** to convert 1 minute 30 seconds to 2,700 frames (assuming 30 fps).
4. Press **1** **0** **0** **0** **fr** **→hms** to convert 1,000 frames to 33.3 seconds (0.00333).

Moments

Moments are stored internally as the number of seconds elapsed since January 1, 1904. Currently, only date and times through December 31, 2031 are supported. This is subject to change in a future version of NeoCal.

There are three date moment registers which are used for entering date and time values. They are:

- The **date** button is used for entering dates using a Y.MMDD format. For convenience, years less than 32 are assumed to be in the 21st century and years between 32 and 99 are assumed to be in the 20th century.
- The **time** button is used for entering times using a 24 hour H.MMSS format.

- The `now` button recalls the current date and time.

In addition to resetting the date conversion registers, the `CLd` also clears or resets the date moment registers to January 1, 1904 12:00 am.

Pressing the `date` or `time` buttons *while entering a number* will convert the number already in the x register. Otherwise, pressing one of these buttons will display a dialog for entering the desired value. In both cases, the value will be stored in that register for later recall.

For example,

1. Pressing `2` `.` `0` `8` `2` `8` `date` enters August 28, 2002 and stores it into the *date* register.
2. Pressing `1` `4` `.` `3` `0` `time` enters 2:30 pm and stores it into the *time* register.
3. Pressing `now` generates August 28, 2002 2:30 pm (the current date and time) and stores it into the *now* register.
4. Pressing `date` displays a dialog for entering a date and stores that date into the *date* register.
5. Pressing `RCL` `now` recalls the last generated date and time value (August 28, 2002 2:30 pm) from the *now* register.

There are two moment conversion functions:

- `dp` for extracting the date part of a moment
- `tp` for extracting the time part of a moment

For example,

1. Pressing `now` generates August 28, 2002 2:45 pm.
2. Pressing `dp` returns August 28, 2002 as a date value.
3. Pressing `RCL` `now` recalls August 28, 2002 2:45 pm into the x register.
4. Pressing `tp` returns 2:45 pm as a time value.

In addition, the following duration conversion functions will extract a portion of a moment:

- `year` for extracting the year from a date or datetime value

- `mth` for extracting the month from a date or datetime value
- `day` for extracting the day from a date or datetime value
- `hour` for extracting the hour from a time or datetime value
- `mnt` for extracting the minute from a time or datetime value
- `sec` for extracting the second from a time or datetime value

For example,

1. Pressing `now` generates August 28, 2002 3:00 pm.
2. Pressing `year` returns 2002.
3. Pressing `RCL now` recalls August 28, 2002 3:00 pm into the x register.
4. Pressing `hour` returns 15.

Date Arithmetic

Date arithmetic involves the addition or subtraction of a moment with either a duration or another moment. The following combinations are supported:

- A moment plus or minus a duration results in a similar moment.
- A moment minus a similar moment results in a duration.
- A date plus a time results in a datetime.

Date arithmetic involving dates and datetimes assume duration units to be days, while calculations involving times assume hours. Durations of other units are converted to these units, however, the duration conversion registers are not used. Instead, these conversions always use the default values of these registers. Explicit conversions to days and hours must be performed to use the values stored in these registers.

For example,

1. Press `date` and enter today's date (August 28, 2002).
2. Press `+ 90 =` to calculate 90 days into the future (November 26, 2002).
3. Press `- 90 =` to calculate 90 days into the past (May 30, 2002).
4. Press `RCL date` to recall today's date (August 28, 2002).
5. Press `+ 3 week =` to calculate 3 weeks into the future (September 18, 2002).

To calculate the day of year:

6. Press **RCL** **date** to recall today's date (August 28, 2002).
7. Press **-** **1** **.** **1** **2** **3** **1** **date** **=** to calculate the number of days since December 31, 2001 (240).

To time an event:

1. Press **now** to generate the starting datetime (August 28, 2002 9:15 am).
2. At the end of the event, press **RCL** **now** to recall the starting datetime.
3. Press **-** **now** to generate the ending datetime (August 28, 2002 2:30 pm).
4. Press **=** **+/-** to calculate the length of time (5.25 hours).

To construct a datetime value, add separate date and time values:

1. Press **2** **.** **0** **8** **2** **8** **date** to enter the date part (August 28, 2002).
2. Press **+** **1** **5** **.** **1** **5** **time** to enter the time part (3:15 pm).
3. Press **=** to combine the datetime value (August 28, 2002 3:15 pm).

RPN Tutorial

Automatic Memory Stack

The RPN input method is based upon working with numbers stored in the automatic memory stack, which consists of the x, y, z, and t registers. The stack *lift* and *drop* operations are performed automatically and are key to understanding its use. Imagine the registers stacked on top of each other with t (on the top) above z, which is above y, which is above x. Both the lift and drop operations cause values to be shifted to an adjacent register, but shift in opposite directions.

The stack lift operation is performed to make room for a new value in the x register: the value in the t register is thrown away, the t register gets the z value, the z register gets the y value, and the y register gets the x value. This allows the new x value to be stored while preserving the original x value (in the y register).

The stack drop operation is performed when an operation requiring two operands is executed: the t register remains unaffected, the z register gets the t value, the y register gets the z value, and the x register gets the y value. The result of the calculation is then stored in the x register, effectively combining (or consuming) the original x and y values. Notice that the value in the t register is propagated downward. This is useful for operations involving multiple uses of a constant value.

Calculator operations can be categorized as either lifting the stack, dropping the stack, or neither lifting nor dropping the stack. Also, some operations disable stack lift, meaning that if the next operation normally lifts the stack, then the stack won't be lifted. Or, if the next operation lifts the stack twice, the stack will only be lifted once.

Let's look at a simple example to calculate the square of the sum of two numbers,

1. Press **4** to enter the number 4. Entering a number is a stack lifting operation, so the previously displayed value is moved into the y register.
2. Press **Enter** to separate the two numbers. This operation lifts the stack and then disables stack lift (see the next section for details on the enter button). The number 4 is now stored in both the x and y registers.
3. Press **5** to enter the second number. Since stack lift was disabled, 5 is simply stored in the x register.

4. Press $\boxed{+}$ to add the numbers stored in the x and y registers, drop the stack, and then store the result in the x register. The number that was displayed before we started this example is now stored in the y register.
5. Press $\boxed{\times^2}$ to calculate the square of the displayed number. This operation does not lift nor drop the stack; only the value in the x register is modified.

By default the stack contains four values, but the stack size can be increased in the Preferences dialog.

Stack Manipulation

There are eight buttons used only for directly manipulating the stack: $\boxed{\text{Enter}}$, $\boxed{\text{R}\downarrow}$, $\boxed{\text{R}\uparrow}$, $\boxed{\times\text{\$}y}$, $\boxed{\text{drop}}$, $\boxed{\text{fill}}$, $\boxed{\text{rot}}$, and $\boxed{\text{over}}$. As noted in the previous example, the $\boxed{\text{Enter}}$ button lifts the stack and then disables lift for the next operation. This is primarily used for entering two numbers in succession, but can also be used for propagating a single value into all four stack registers by pressing it three times.

The $\boxed{\text{R}\downarrow}$ and $\boxed{\text{R}\uparrow}$ buttons rotate the values in the stack registers while preserving all values. The $\boxed{\text{R}\downarrow}$ button is similar to stack drop except the x value is moved into the t register. The $\boxed{\text{R}\uparrow}$ button is similar to stack lift except the t value is moved into the x register. Pressing either of these buttons four times in a row will result in the register values stored in their original locations.

The $\boxed{\text{drop}}$ button performs a stack drop while preserving the value in the t register. The $\boxed{\text{fill}}$ button copies the value in the x register into all the registers in the stack.

The $\boxed{\times\text{\$}y}$ button exchanges the values in the x and y registers.

The $\boxed{\text{rot}}$ operation rotates the bottom three (x, y, and z) values upward. The $\boxed{\text{over}}$ button pushes the y value onto the stack, moving all values upward and duplicating the y value.

Choosing the RPN Stack menu option opens a dialog that displays all the registers in the automatic stack. This form also includes the $\boxed{\text{rot}}$, $\boxed{\text{over}}$, $\boxed{\text{drop}}$, and $\boxed{\times\text{\$}y}$ stack manipulation buttons.

Frequently Asked Questions

If you don't find an answer to your question here, please check the web for an updated FAQ at <http://www.hudren.com/products/neocal/guide/faq.html>.

General FAQ

Why can't I find a button mentioned in the documentation?

Many buttons are only accessible by first pressing the shift button which looks like an up arrow and is located above the 7 button.

Why does the I/YR and IRR calculations return 'error'?

These solutions are found using a 'solve' algorithm which sometimes fails to find the correct solution. The current value in the I/YR register is used as a starting point or estimate, and if this value is too different from the answer, the algorithm might fail to find a correct solution and return 'error'. Try providing an estimate or guess by storing an approximate value in the I/YR register before performing the calculation.

Why is a physical constant being recalled as zero?

Many of the physical constants are extremely small numbers and display as zero when using the Fixed number format even though the number recalled into the x register is non-zero. Choose a different number format in the Preferences dialog to prevent this.

Why do some buttons appear in more than one location?

This is done strictly for convenience, providing easy access to the buttons you need, when you need them. The duplicated buttons operate the same regardless of their location.

Why am I having problems printing the PDF version of this user guide?

This user guide was created using the latest version of Adobe Acrobat, so earlier versions may have a problem printing or viewing it. You can download the most recent version of Acrobat Reader for free from the Adobe website at <http://www.adobe.com/products/acrobat/readstep.html>.

Qtopia-related FAQ

How can I view the NeoCal User Guide on a Zaurus device?

The user guide is included as a separate install package for the Sharp Zaurus and may be installed and removed independently of the NeoCal application. Once installed, the user guide can be viewed using a web browser. See the NeoCal help screen for an appropriate URL.

Newer versions of the Qtopia Help Browser do not follow the links in the User Guide properly, what are other ways to view the NeoCal User Guide?

Versions of the user guide published on or after August 7, 2002 can be viewed using a web browser. After downloading and installing a current user guide, see the NeoCal help screen for an appropriate URL to copy to your favorite browser.

Can the User Guide be stored on an expansion card?

The user guide contains a large number of files which some versions of Qtopia have problems with when installing to locations other than RAM. In theory, it should work fine.

Revision History

Version 2.0

- Implemented Themes for selecting overall calculator appearance and keyboard behavior
- Implemented Dynamic Keyboard Technology for controlling the look and feel of the keyboard
- Modified store, recall, and swap to handle decimal point when addressing 100 registers
- Modified backspace to clear format prefix
- Modified [Sum-] to return error when $n < 1$
- Added shift annunciator to show current keyboard shift state
- Added Edit > Cut menu item (copies to clipboard, then drops)
- Added support for action hardware button (Jog Dial push or 5-way center)
- Added preference for locking shift button
- Added preference to have the backspace key ([<-]) perform the drop stack operation instead of clear x ([CLx], [CE]) when not entering a number
- Added preference to display up to four lines
- Added preference to always display a single line for the Basic keyboard
- Added preferences to display shifted functions on non-shifted buttons
- Renamed RPN Stack dialog as Stack
- Relabeled [CLst] button as [clear]
- Relabeled [frac] button as [fp], [int] as [ip], and [fp] as [fpr]
- Relabeled [->M] button as [SM] (store memory)
- Added dot to store, recall, and swap buttons to indicate addressing of only 20 registers

Version 2.0.1

- Restored the default number of display lines to 1 for all themes

Version 2.0.2

- Improved accuracy when tapping buttons
- Fixed bug when using the decimal point to address registers in 20 registers mode

Version 2.0.3

- Improved registration process to compensate for common entry errors
- Fixed bug where display font was not restored properly during startup

Registration

NeoCal is distributed as a shareware program, which means that you may use this program for a brief trial or evaluation period. If you continue to use this program for more than thirty (30) days, however, you are required to register this software for a fee.

Full registration details for this software are available on our website at <http://www.hudren.com/register>.

Once we have been notified of your registration (which can sometimes take up to 24 hours or longer), we will send you a registration confirmation email containing your registration code. The Reminder dialog will no longer be displayed once this information is entered into the Registration dialog on your handheld. Saving this information in a safe place is a good idea as it may become necessary to enter it again when re-installing the software.

Unfortunately, email delivery is not 100% reliable, and increased anti-spamming efforts often block the registration email without notifying us that the email was not delivered. Also, we occasionally receive email addresses that were not entered correctly or addresses for email accounts with full quotas that are not accepting new emails. Many people have even deleted the registration email incorrectly thinking it was spam.

We take the responsibility of delivering your registration code very seriously and most are delivered within twenty (20) minutes of an order being placed (may be longer if the order was placed on a website other than our own). If you have not received your registration code within two (2) business days, please contact us so that we can resend it or make other arrangements to get it to you.

There is no need to download different software when registering, the "trial" and "registered" versions are identical with the latter having the registration information entered in the Registration dialog on the handheld device. It is recommended that you download the latest version if you do not already have it installed (version information and download links are provided in the registration confirmation email).

You must enter the registration information exactly as it appears in the confirmation email. The code will not work if they do not match exactly, including letter case, accents, spaces, and punctuation (including hyphens). Please do not hesitate to contact us at support@hudren.com if you have any questions or experience any problems entering your registration code.

Starting with NeoCal 1.7, an alternative method for entering the registration information is available by copying all of the registration information (in particular, the three lines containing the name, email, and code values) to the clipboard of the handheld device and then opening the NeoCal Registration dialog where the information will be automatically entered into the appropriate fields.

If you have misplaced or discarded your registration information and need it again, you may visit our website at <http://www.hudren.com/support> to request that it be emailed to you.

Registered users are entitled to use all future versions of NeoCal, so be sure to periodically check our website for free updates.

Thank you for supporting this product by registering the software.

Clipboard Method

An alternative method for entering the registration information is available by copying all of the registration information from the registration email to the clipboard of the handheld device and then opening the Registration dialog in NeoCal where the registration information will be automatically entered for you.

Support

Support for NeoCal is provided via email. Please check our website for program and documentation updates as the latest version might solve any problems you're experiencing or answer any questions you may have. Remember, registered users are entitled to use all future versions free of charge.

Please send any questions or comments to support@hudren.com and include the following information:

- Your name
- Your e-mail address
- The NeoCal platform and version
- The brand and model of your handheld
- If you're reporting a bug in the software, please include the steps needed to recreate the problem

We will respond as quickly as possible. Thank you for your patience.

Program Updates

Registered users are entitled to use all future versions of NeoCal, so be sure to periodically check our website for free updates.

The latest version of this software can be found on the web at <http://www.hudren.com/products/neocal>.

Documentation Updates

The latest NeoCal documentation can be found at <http://www.hudren.com/products/neocal/guide>.

Adobe Acrobat Reader is required to view or print PDF files and can be downloaded for free from the Adobe [website](#).