NeoCal for Qtopia
User Guide

Version 1.4.4

Published on August 7, 2002

www.hudren.com

Table of Contents

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 and Statistical sections 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
- Large 12-digit display with thousands separators
- Undo/Redo
- Color support (user selectable)
- Labeled results for ease of use and clarity
- 100 general-purpose user registers
- Choice of Simple, Precedence, *and* RPN input methods
- Over 145 functions including:
    - **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

- **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

Requirements

NeoCal 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 Preview SDK and should execute with any compatible QPE / Qtopia 1.5 runtime environment. NeoCal has been tested to work with Zaurus ROM versions 1.10 or later and OpenZaurus beta 2.6.

Installation

The main distribution file, in .zip or .tgz format, contains 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.

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. 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 *Standard*, *Financial*, *Scientific*, *Statistical*, *Programmer*, *Conversion*, 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.

Additionally, selecting the input method changes the bottom five rows of the keypad. This is discussed in the next section.

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 2 + 3 = to get the desired result of 5. A more complex example is 2 + 3 x 5 = 17, which can be entered two ways, 2 + (3 × 5) = or 3 × 5 + 2 =. The first sequence required the use of parentheses to instruct the calculator to multiply before adding the numbers (since most calculators don't know 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 the multiplication and division operations are automatically performed before the addition and subtraction operations. For example, the previous calculation can be entered without the parentheses as 2 + 3 × 5 = to obtain the result of 17.

The benefits of the RPN, or Reverse Polish Notation, input method over the simple input method 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 x 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 x 4 + 3 x 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 x 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. 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.

Preferences

The display is formatted by setting options in the Preferences dialog, which is opened by selecting the Options -> Preferences menu option or tapping on the calculator's display area. The default number format is *General,* 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 in the Preferences dialog are:

- Display 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.
- Integer specifies the size and type of the integer operands in Programmer mode.
- Input specifies the input method. See Comparison of Input Methods for detailed information about the available input methods.
- Stack specifies the size of the stack when using the RPN input method. See RPN Tutorial for more details.
- Checking the Quick switch option displays a row of push buttons for switching between the calculator's functions, in place of the default list.
- Checking the Sticky shift option causes the shift key to remain activated after the next key is pressed.
- Checking 100 Registers option causes the user registers to be addressed using 2 digits, otherwise only 20 registers can be accessed. See User Registers for more information.

- Checking the 2-line display option displays both the X and Y registers in the display area.
- The Comma Radix option displays a comma for the decimal point character.

Formatting Numbers

In conjunction with setting the display format in the Preferences dialog, the gen, 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 +/− button. If you press the +/− 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 EXP, and then enter the exponent. If the number is negative, press +/− *before* pressing EXP. If the exponent is negative, press +/− *after* pressing EXP.

If the ← button is pressed while you're not entering a number, it will have the same effect as pressing CE or CLx. If the ← 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 Colors

The colors or grayscale levels used to draw the keyboard and display can be selected in the Colors dialog. Taping one of the color "swatches" for the text or background will open a color selection dialog used for specifying which color to use for drawing that particular element. The rightmost column previews an example of the current color choices.

Pressing the Default button will return the color selections to their original value.

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.

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 equals character must be entered for the equals button, since the return character is used for the enter button.
- 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, and statistical 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 [1] [2] places the number 12 in the *x* register.
2. Pressing [+] copies the value stored in the *x* register into the *y* register.
3. Pressing [1] [5] stores the number 15 in the *x* register.
4. Pressing [=] 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 ([1/x]) or x and y ([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 [x<>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 [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:

- [CM] is used to clear the *M* register, or set its value to zero.
- [→M] stores the value in *x* into the *M* register.
- [RM] recalls the *M* register by placing its value into *x*.
- [M+] adds *x* to the value already stored in the *M* register.
- [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:

- [STO] is used to store *x* into a specified register. The *STO* indicator is displayed.
- [RCL] is used to recall a register's value into the *x* register. The *RCL* indicator is displayed.
- [x⇆] exchanges or swaps the value stored in the *x* register with the value stored in the specified register. The *SWAP* indicator is displayed.

The [←] 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. Tap on one of the register values to complete the operation using that register.

For example (using 20 registers),

1. Pressing [1] [7] [STO] [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] stores the number 17 in the *x* register.
4. Pressing [3] [3] [x⇆] [.] [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] stores the number 17 in the *x* register.
4. Pressing [3] [3] [x⇆] [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, displays the [STO], [RCL], [x⇆], and [x⇆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 x⇆y button is provided to facilitate moving values between different 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:

- C 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.
- CE 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.
- CLr clears the user, financial, and statistical registers.
- CLf clears the financial registers and displays the number of compounding periods per year. This does not clear the cash flow entries, which are stored in the user registers.
- CLΣ clears the statistical registers.
- CA clears everything.

When using the RPN method, the CLst button is equivalent to the C button, CLx is equivalent to CE, and CLA is equivalent to CA.

Standard 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.
- y^x calculates y raised to the power of x.
- $\sqrt[x]{y}$ calculates the x root of y.

Percentages

Percent

Normally, the [%] 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 [C] [1] [5] [%] displays the number 0.15.
- Pressing [C] [5] [0] [+] [3] [0] [%] displays the number 15 (since 30% of 50 is 15). The result after pressing [=] 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 [Δ%] function calculates the percent difference between two numbers.

In RPN mode, the value of y is preserved.

Percent of Total

The [%T] function calculates the percentage that x is of y. For example,

- Pressing [5] [0] [%T] [3] [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 the preceding 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 are cleared by pressing the CLf button. As a reminder, the value of the P/YR register is also displayed.

In addition, the user registers are used for entering cash flow groups.

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 [N], [I/YR], [PV], [PMT], and [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 ([PV], [PMT], and/or [FV]). In addition,

- The value stored in the [P/YR] register is used to convert the annual nominal rate ([I/YR]) to the periodic rate (which is required for the calculations).
- [×P] stores the displayed number multiplied by the value in the [P/YR] register into the [N] register.
- [÷P] divides the displayed value by the value in the [P/YR] register.
- [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 *BEGIN* indicator is displayed, otherwise *end* mode is in effect.

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 [beg] to set *end* mode.
2. Press [CLf] to clear the financial registers.
3. Press [1] [2] [P/YR] for monthly payments.
4. Press [9] [.] [5] [EFF%] to store the annual effective rate.
5. Press [NM%] to calculate the annual nominal rate of 9.11 percent (which is stored in the [I/YR] register).
6. Press [3] [×P] to store 36 in the [N] register for the total number of payments.
7. Press [1] [6] [5] [0] [0] [PV] to store amount to be financed.
8. Press [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 P/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 an×t 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 = after pressing amrt or an×t 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↓ to display the principal. Pressing R↓ a second time will display the balance. Press an×t 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.

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 [beg] to set *end* mode.
3. Press [CLf] to clear the financial registers.
4. Press [1] [2] [P/YR] for monthly payments.
5. Press [3] [0] [×P] to store the loan duration in the [N] register.
6. Press [7] [I/YR] to store the interest rate.
7. Press [2] [EXP] [5] [PV] to store the loan amount in the present value register.
8. Press [PMT] to calculate the monthly payment amount (-1330.60).
9. Press [amrt] to amortize the first payment and display the interest amount (-1,166.67).
10. Press [=] (in RPN mode, press [R↓]) to display the principal for the first payment (-163.93).
11. Press [=] (in RPN mode, press [R↓]) to display the balance after the first payment (199,836.07).
12. Press [=] (in RPN mode, press [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 [1] [Enter] [1] [2] [amrt] to amortize the first twelve payments and display the interest amount (-13,935.65).
14. Press [anxt] to amortize the second year's payments and display the interest amount (-13,788.78).

At any time, you can press [RCL] [amrt] to recall the starting and ending payments for the range being amortized. (Use the [x$y] button to display the starting payment.)

Cash Flow Analysis

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

All cash flows must occur at regular intervals, but can be of differing amounts. A 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 [CLr] to clear any cash flow entries. This clears all registers except [P/YR].
2. Enter the number of periods per year in the [P/YR] register.
3. Enter the initial cash flow amount and press [CFj].
4. Enter the next cash flow amount and press [CFj].
5. If this cash flow entry occurs more than once consecutively, enter the number of times it occurs and press [Nj].
6. Repeat steps 4 and 5 until all the cash flow entries are entered.

To review all of the cash flow entries,

1. Press [RCL] [0] to display the initial cash flow amount.
2. Press [RCL] [CFj] to display the amount of the next cash flow entry.
3. Press [RCL] [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.

To review and edit the cash flow entries,

- Press [RCL] followed by the number of the cash flow entry to display that amount.
- Press [STO] followed by the number of the cash flow entry to store the that amount.
- Pressing [RCL] [CFj] displays the next cash flow amount.
- Pressing [STO] [CFj] stores the next cash flow amount, but does not change the number of times the cash flow occurs.
- Pressing [RCL] [Nj] 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 [STO] [Nj] 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 CFj button.
- To add a new cash flow entry, first recall the last cash flow amount, then store the new cash flow amount using the CFj 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 CLr to clear the registers.
2. Press 1 2 P/YR to store the number of periods per year.
3. Press 1 0 0 0 +/− CFj to store the initial cash flow.
4. Press CFj to store the first cash flow entry.
5. Press 1 1 Nj to store the number of occurrences for this cash flow.
6. Press 0 CFj to store the second cash flow entry.
7. Press 1 2 Nj to store the number of occurrences for this cash flow.
8. Press 1 6 0 0 0 CFj to store the last cash flow amount.
9. Press IRR to calculate the annual internal rate of return (18.71 percent) and store it in the 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 CLr to clear the registers.
2. Press 1 2 P/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.
- e^x calculates the natural antilogarithm of x.
- log calculates the common log of x.
- 10^x 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.

- [→pol] converts the rectangular coordinate (x, y) into the polar coordinate (r, theta). Theta is returned in units corresponding to the current angular mode.
- [→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.
- [→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 [π] [→deg] to convert pi to 180 degrees.
2. Press [3] [6] [0] [→rad] to convert 360 degrees to 2 pi radians.
3. Press [deg] [2] [Enter] [1] [→pol] to convert the rectangular coordinate (x = 1, y = 2) to the polar coordinate (magnitude = 2.2361, angle = 63.4349 degrees). (Press [x$y] button to view theta.)
4. Press [2] [.] [5] [→hms] to convert 2.5 hours to 2 hours 30 minutes (2.30000 or 2"30'00.0).
5. Press [4] [.] [2] [3] [4] [8] [→hr] to convert 4 hours 23 minutes 48 seconds to 4.3967 hours.

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 [n], [Σx], [Σx²], [Σy], [Σy²], and [Σxy] buttons. However, these registers are not modified using the [STO] button, rather, data is accumulated into these registers by using:

- [CLΣ] to clear the statistical registers.
- [Σ+] 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.
- [Σ−] 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:

- [n] recalls the number of data points accumulated.
- [Σx] recalls the summation of the x values.
- [Σx²] recalls the summation of the squares of the x values.
- [Σy] recalls the summation of the y values.
- [Σy²] recalls the summation of the squares of the y values.
- [Σxy] recalls the summation of the products of the x and y values.
- [RCL] [Σ+] recalls the summation of the x values into the x register and the summation of the y values into the y register.

Let's enter some data,

1. Press [CLΣ] to clear the statistical registers.
2. Press [1] [0] [Enter] [4] [0] [Σ+] to accumulate the first data point.
3. Press [7] [Enter] [3] [5] [Σ+] to accumulate the second data point.
4. Press [1] [5] [Enter] [7] [0] [Σ+] to accumulate the third data point.
5. Press [3] [1] [Enter] [1] [0] [0] [Σ+] to accumulate the fourth data point.

6. Press [Σx] to recall the sum of the x values (245).
7. Press [Σx²] to recall the sum of the squares of the x values (17,725).
8. Press [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:

- [x̄] calculates and displays the mean (average) of the x values.
- [ȳ] calculates and displays the mean (average) of the y values.
- [x̄w] calculates and displays the weighted mean of the x values. The y values are used to determine the weight for the corresponding x value.
- [s] calculates the standard deviation of the x and y values and returns the results in the x and y registers, respectively.
- [σ] 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 [x̄] to calculate the mean or average of the x values (61.25).
10. Press [ȳ] to calculate the mean or average of the y values (15.75).
11. Press [x̄w] to calculate the weighted mean of the x values (76.11).
12. Press [s] to calculate and display the standard deviation of the x values (30.10).
13. Press [x⇄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.

- [x̂,r] estimates x for a given y value. The correlation coefficient (r) is returned in the y register.
- [ŷ,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 [1] [2] [x̂,r] to estimate the x value (50.36) for the given y value.
15. Press [x⇄y] to display the correlation coefficient (0.97).
16. Press [0] [ŷ,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 [x⇄y] to display the slope (0.34) of the fitted line.

Probability

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

- [Cy,x] calculates the number of possible sets of y different items taken x at a time.
- [Py,x] calculates the number of possible different arrangements of y different items taken x at a time.
- [x!] 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 [5] [2] [Cy,x] [5] to calculate the number of different five-card hands that can be dealt from a deck of cards (2,598,960).
2. Press [1] [0] [x!] to calculate the factorial of 10 (3,628,800).
3. Press [1] [0] [Γx] to calculate the gamma of 10 (362,880).
4. Press [ran] to generate a number between zero and one.
5. Press [ran] 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 [+], [−], [×], [÷], and [+/−] buttons operate on the integer part of their operands and return an integer for their result. 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 Programmer mode.

Base Conversions

Use these buttons to convert from one base to another:

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

The [A] through [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 [show] button to reveal the other bits.

Integer Conversions

These buttons are used for converting between different integer types:

- [u8] converts x to an unsigned 8-bit integer.
- [s8] converts x to a signed 8-bit integer.
- [u16] converts x to an unsigned 16-bit integer.
- [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.
- rot 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.

Length Conversions

The following buttons represent the length conversions:

- [in] for inches
- [ft] for feet
- [mi] for miles
- [m] for meters

For example,

1. Press [1] [2] [in] to enter 12 inches.
2. Press [ft] to convert to 1 foot.
3. Press [m] to convert to 0.30 meters.

Area Conversions

The following buttons represent the area conversions:

- [in²] for square inches
- [ft²] for square feet
- [mi²] for square miles
- [m²] for square meters

For example,

1. Press [1] [m²] to enter one square meter.
2. Press [ft²] to convert to 10.76 square feet.
3. Press [in²] to convert to 1500 square inches.
4. Press [m²] to convert back to 1 square meter.

Volume Conversions

The following buttons represent the volume conversions:

- gal for gallons
- fl oz for fluid ounces
- liter for liters
- ml for milliliters

For example,

1. Press 1 gal to enter one gallon.
2. Press liter to convert to 3.79 liters.

Speed Conversions

The following buttons represent the speed conversions:

- in/s for inches per second
- ft/s for feet per second
- cm/s for centimeters per second
- m/s for meters per second
- mi/h for miles per hour
- km/h for kilometers per hour

For example,

1. Press 5 5 mi/h to enter 55 mph.
2. Press km/h to convert to 88.51 km/h.

Weight Conversions

The following buttons represent the weight conversions:

- lb for pounds
- oz for ounces
- kg for kilograms

- gm for grams

For example,

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

Temperature Conversions

The following buttons represent the temperature conversions:

- °F for degrees Fahrenheit
- °C for degrees Celsius

For example,

1. Press 7 5 °F to enter 75 degrees Fahrenheit.
2. Press °C 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 3 ft + 1 5 in = to add fifteen inches to three feet. The result is 4.25 feet, or 4 feet 3 inches.
2. Press 1 gal + 3 liter = to add one gallon and 3 liters. The result is 1.79 gallons.

Multiplication can be performed on two lengths to generate an area. Also, an area divided by a length produces a length. The x^2 and $\sqrt{x}$ functions can also be used for these calculations.

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].

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 [+] 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 [x²] 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.

Stack Manipulation

There are six buttons used only for directly manipulating the stack: [Enter], [R↓], [R↑], [x⇆y], [drop], and [fill]. As noted in the previous example, the [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 [R↓] and [R↑] buttons rotate the values in the stack registers while preserving all values. The [R↓] button is similar to stack drop except the x value is moved into the t register. The [R↑] 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 [drop] button performs a stack drop while preserving the value in the t register. The [fill] button copies the value in the x register into all the registers in the stack.

The [x⇆y] button exchanges the values in the x and y registers.

Choosing the RPN Stack menu option opens a dialog that displays all the registers in the automatic stack. This form also includes the [R↓], [R↑], [drop], and [x⇆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 does NeoCal sometimes return the wrong answer on simple calculations in RPN mode?

RPN mode is not enabled by default and must be turned on in the Preferences dialog. Using the Enter button in non-RPN modes to perform RPN calculations will result in incorrect answers. When RPN mode is enabled, the Enter key will be twice the size of the other keys and the RPN annunciator will be displayed.

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 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.

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 1.4

Version 1.4.2

- Released initial Linux / Qtopia version for Sharp Zaurus

Version 1.4.3

- Added Fixed display format
- Added larger keyboard glyphs
- Added menu button support for Zaurus (now requires ROM >= 1.10)
- Renamed Algebraic input method to Precedence; changed annunciator from ALG to PRC
- Added RPN input method prompt for first time users
- Fixed display problem where some numbers were rounding down when the remaining digits were "50..."

Version 1.4.4

- Fixed incorrect display when using scientific or engineering notation that caused the exponent to be improperly incremented by one when the first digit of the result was rounded upward and a carry was generated.

Version 1.4.4-1

- Released Qtopia-specific documentation
- Repackaged on-device user guide for use with a web browser

Registration

This version of NeoCal is released as freeware as a special thanks to everyone supporting the Linux community. What has been accomplished so far is truly amazing!

You may use this software free of charge without explicit registration.

Support

Support for this product is provided for registered users via the Internet. Please check our website for program and documentation updates before contacting us. Submit any comments or questions to support@hudren.com. Please include the following information:

- Your name
- Your e-mail address
- The 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

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.