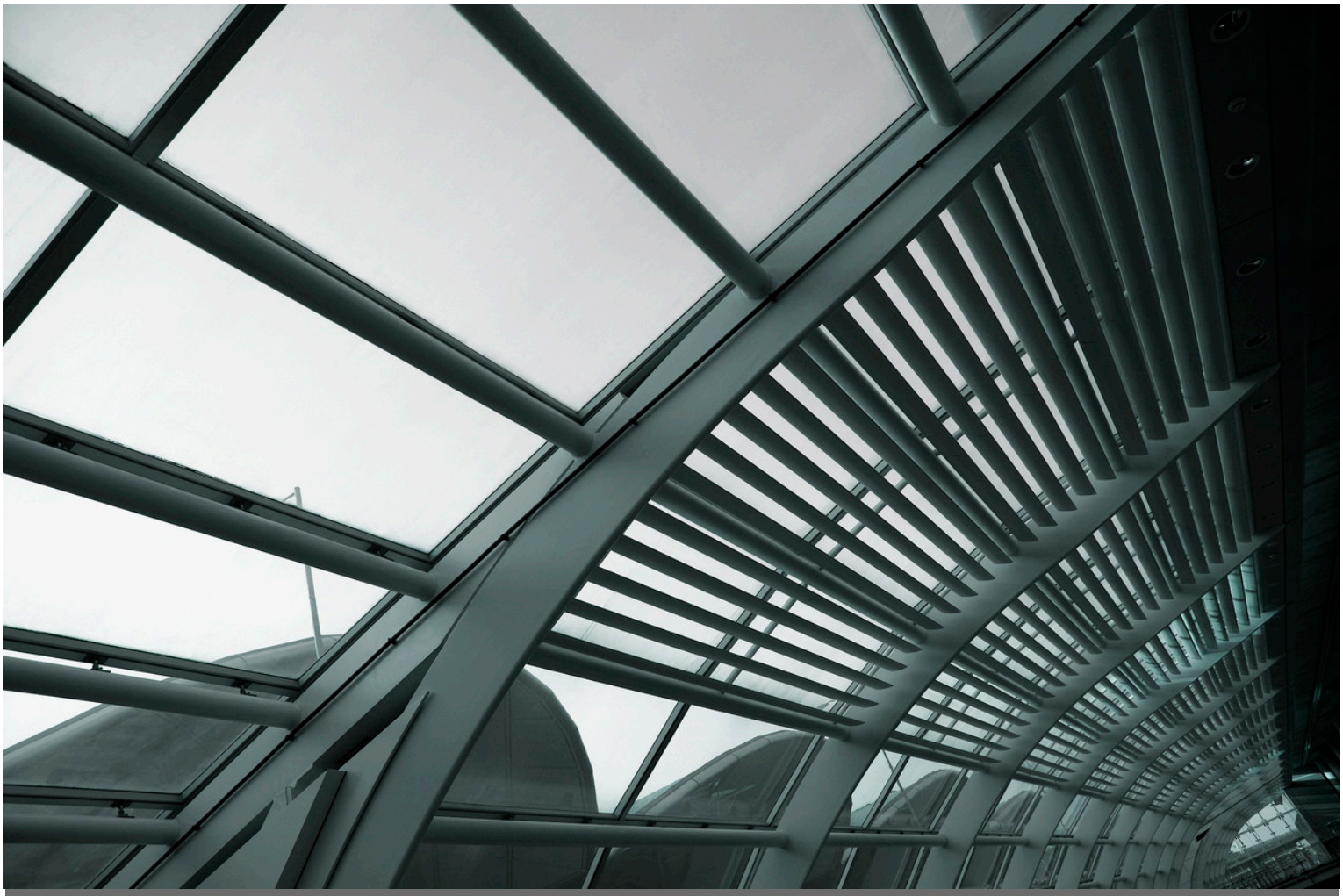


BuildCalc 2.1: Users Manual

Advanced Construction Calculator for Professionals

BuildCalc.com A Division of 42nd Parallel.

Table of Contents

Introduction: Welcome to BuildCalc 2.0

Hello, my name is Ben Askren, the developer of BuildCalc, and I want to thank you. In March of 2009, I set out on a mission to make BuildCalc the best construction calculator there is. With the guidance of construction pros who know what works, I dedicated myself to a lot of late nights and lost weekends. With so much of myself wrapped up in BuildCalc, it is a great satisfaction when someone finds it to be one of their favorite tools. So it is my hope that you quickly discover the benefits of BuildCalc and find it one of your favorite tools.

Sincerely,

Ben Askren

Groups of Functions

Below are groups for the functions that can be found in this manual. If you can't find the help you seek here, please don't hesitate to contact me at help@BuildCalc.com and I will work to make it right.

Fractional Inch Math	*Function*
Fraction Entry	[/] – the Solidus Function
Switching Denominators	[^{1}/2] [^{1}/4] [^{1}/8] [^{1}/16] [^{1}/32] [^{1}/64]
Fractional / Decimal Switch	[Feet] [Inches]
What is the default denominator?	[^{1}/?]

Layouts	*Function*
Arch Framing	[Arc]
Balusters	[Balstr]
Compound Miters	[CmpMtr]
Equilateral Polygons	[Polygn]
Rafters	[Diag] – *Common* [Hip/V] [IrPtch]– *Hip/Valley* [Jack] – *Regular & Irregular Jack Rafters*
Rakewall Framing	[R/Wall]
Staircases	[Stair]

Material Estimation	*Function*
Brick, Block, Pavers, Tile	[Msnry]
Board Feet Conversion	[Bd Ft]
Concrete	[Footng] – *footers* [Height] – *rectangle slabs* [ColCon] – *columns / cones*
Cost	[Cost]
Excavation	[Height] – *"box" volume* [col/cone] – *column/cone volume* [wt/vol] – *volume to weight conversion*
Fence Posts, Rails, Pickets, Panels	[Fence] [Rails]
Drywall	[DryWal]
Shingles, Sheathing, Underlayment	[Roof]
Siding	[Height] – *"boxes"* [Diag] - *gables*
Studs, Posts, Pylons, Pillars	[qty@oc]

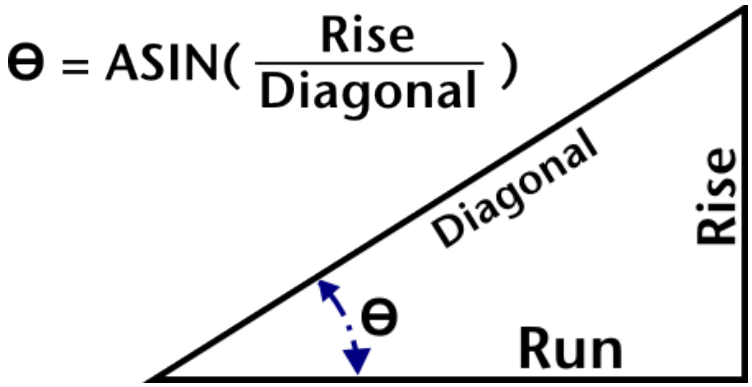
Geometry / Trigonometry	*Function*
Arches, Arcs	[Arc]
Boxes, Rooms and Slabs	[Height]
Circles	[Radius] [Circle]
Columns & Cones	[ColCon]
Equilateral Polygons	[Polygn]
Rectangles	[Width]
Right Triangles	[Pitch] [Rise] [Run] [Diag]
Trig Functions	[SIN] [COS] [TAN]
Inverse Trig Functions	[ASIN] [ACOS] [ATAN]

Memory	*Function*
Cumulative Memory	[M+] [M-] [M-R/C]
Permanent Memory	[M1] [M2] [M3]
Functional Memory	[Pitch] [Rise] [Run] [Diag] [Radius] [Circle] [Arc] [Length] [Width] [Height] [Rails] [o.c.] [R/Wall]
Conversion Memory	[wt/vol] [tons]
Operation Memory	[Tape] [Store] [Recall] *for advanced functions*

[ACOS] Arccosine Function

1. Calculate the arccosine for a given angle.

2. The arccosine of a right triangle is the inverse of the cosine for that triangle.



Example:

1. Calculate the arccosine for a rise of 12 and a diagonal of 23.

Input	*Display*
clear temporary memory [C] [C]	0
If necessary, swipe green keys to reveal trig keys	
12 ÷ 23 = [Conv] [ASIN]	31.45°

**Note: The Arccosine function is one of six trigonometric functions found on BuildCalc. You can access the trig functions by sweeping your finger across the green keys (switching [Length] [Width] [Height] for [SIN] [COS] [TAN]). Pressing the yellow [Conv] button will switch [SIN] [COS] [TAN] to [ASIN] [ACOS] [ATAN].*

[Acre] Acre Function

1. Set the units for a quantity to acres.
2. Convert an area to acres.

Examples:

1. Set the units for a quantity to acres.

Input	*Display*
clear temporary memory	0
125 [Conv] [Acre]	125acre

2. Convert an area to acres.

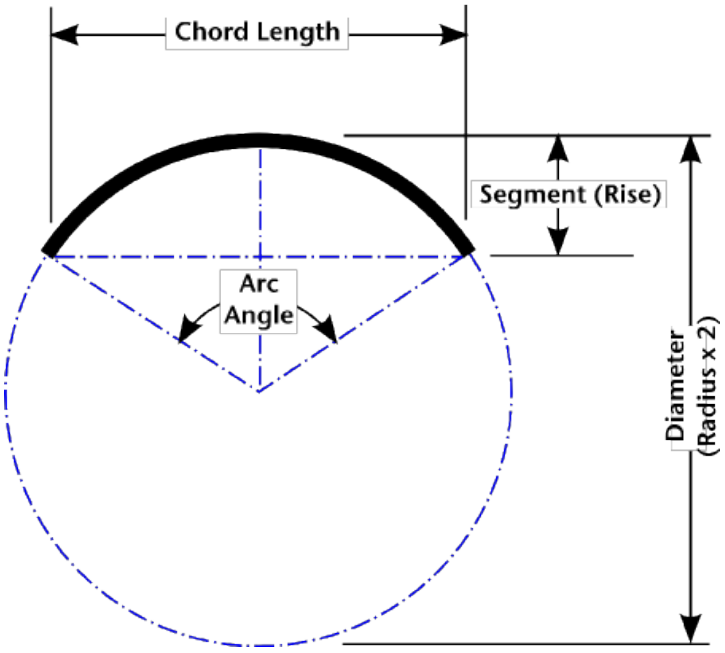
Input	*Display*
clear temporary memory	0
200 [Feet] [x] 80 [Feet] [=]	16000ft^2
[Conv] [Acre]	0.367309acre

[Arc] Arc Geometry & Framing Function

Starting with version 2.0, BuildCalc's Arc function has been enhanced to display the most information with the least number of keystrokes. Because the first press of the [Arc] key is often used to store an arc length or arc angle value for other calculations, no change has been made in function for the first key press. However, if the [Arc] key is pressed a second time, all Arc function results are displayed in a list.

The Arc function calculates the descriptive geometry for an arc when given two of the following as inputs:

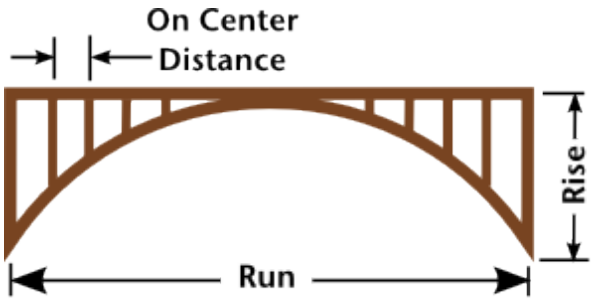
- Arc Length or Angle (enter directly)
- Diameter (as entered into [Circle] or [Radius])
- Chord Length (as entered into [Run])
- Segment Length (as entered into [Rise])



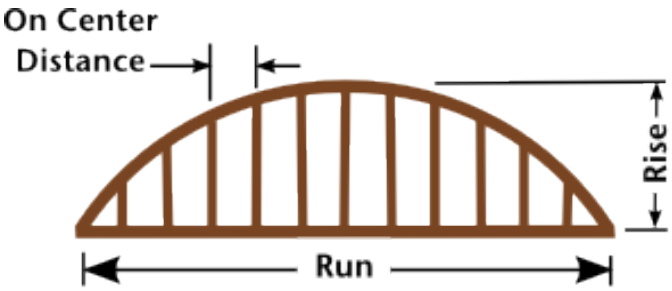
Arc Illustration

The Arc function also calculates the length of each stud required to frame an arch. In addition to two of the above values, this portion of the calculation depends upon the following stored values or settings: [o.c.], [Prefs] and [R/Wall].

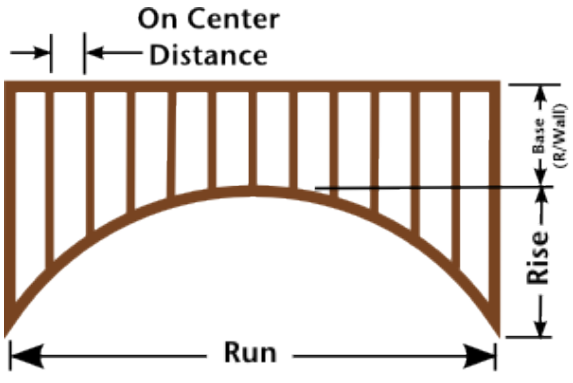
Key	*Display*
clear temporary memory © ©	0
[o.c.]	Stud on-center spacing
[Prefs] *Arched Wall*	Switches between arches that are framed on the **outside** vs. framed on the **inside**
[R/Wall]	The extra base length added to every stud



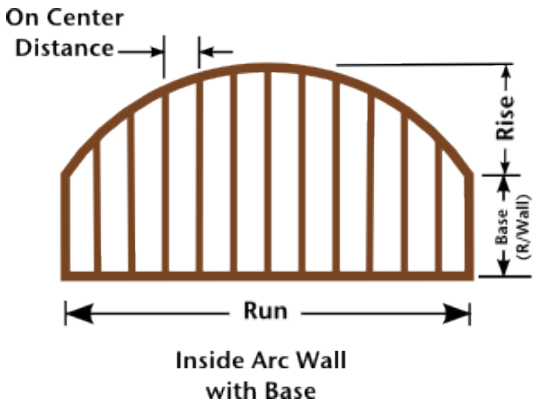
Outside Arc Wall



Inside Arc Wall



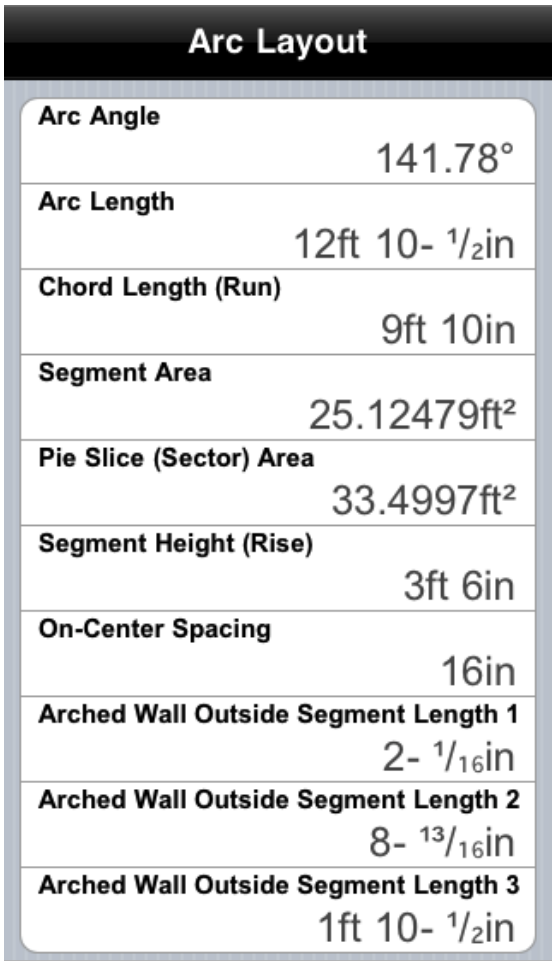
Outside Arc Wall
with Base



Example:

Calculate the framing for an arc with a run of 9’10” and a rise of 3’6”.

Input	***Display***
clear temporary memory	0
9 [Feet] 10 [Inches]	9ft 10in
[Run]	Run 9ft 10in
3 [Feet] 6 [Inches] [Rise]	Rise 3ft 6in
[Arc]	Arch Angle 141.78°
[Arc]	*see results, below*

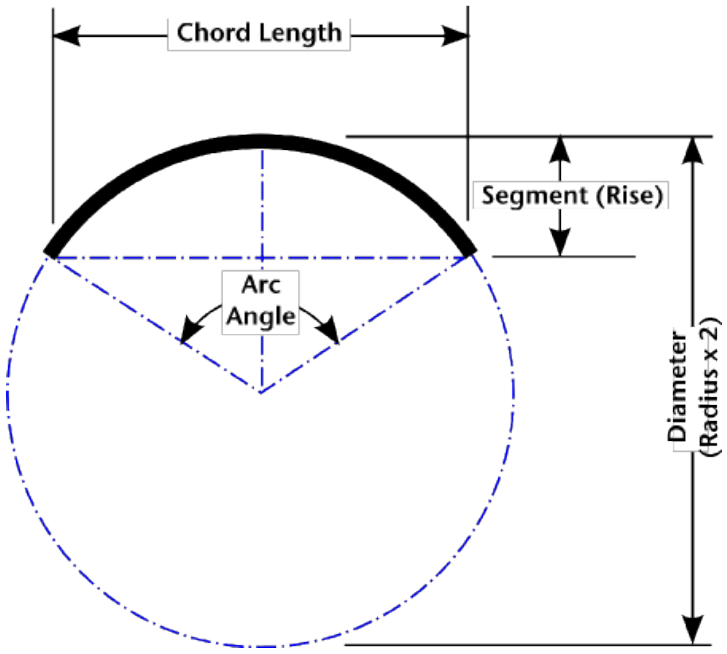


Note: this example assumes a 16" on-center spacing stored in [o.c.].

Finding the Radius or Diameter of an Arc

The diameter or radius for an arc can be easily found when given two of the following* as inputs:

- [Arc] Angle
- Chord Length ([Run])
- Segment Length ([Rise])



Arc Illustration

Examples:

1. Find the radius of an arc given a Segment Height (Rise) of 2m and a Chord Length (Run) of 6m

Input	*Display*
clear temp memory 	0
6 [m]	6m
[Run]	Run 6m
2 [m] [Rise]	Rise 2m
[Conv] [Radius]	Radius 3.25m

2. Find the diameter of an arc given a Segment Height (Rise) of 6ft 2in and a 120° arc angle.

Input	*Display*
clear temporary memory ⓒ ⓒ	0
6 [Feet] 2 [Inches]	6ft 2in
[Rise]	Run 6ft 2in
120 [Arc]	Arc Angle 120.00°
[Circle]	Diameter 24ft 8in

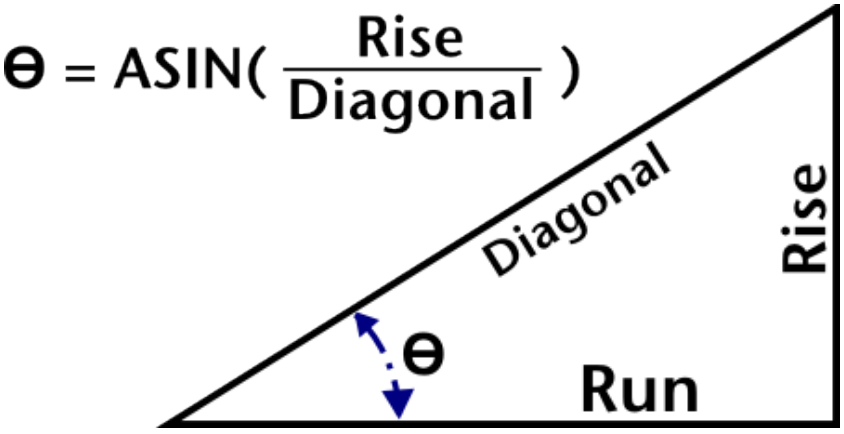
3. Find the radius and diameter of an arc given a Chord Length (Run) of 36 inch and a 30° arc angle.

Input	*Display*
clear temporary memory ⓒ ⓒ	0
36 [Inches] [Run]	Run 36in
30 [Arc]	Arc Angle 30.00°
[Conv] [Radius]	Radius 69- $^{9}/_{16}$in
[Circle]	Diameter 139- $^{1}/_{16}$in

Note that the [Arc] function can be used to calculate the length of each stud required to frame an arch.

[ASIN] Arcsine Function

1. Calculate the arcsine for a given angle.
2. The arcsine of right triangle is the inverse of the sine for that triangle.



Example:

1. Calculate arcsine for rise of 12 and diagonal of 23.

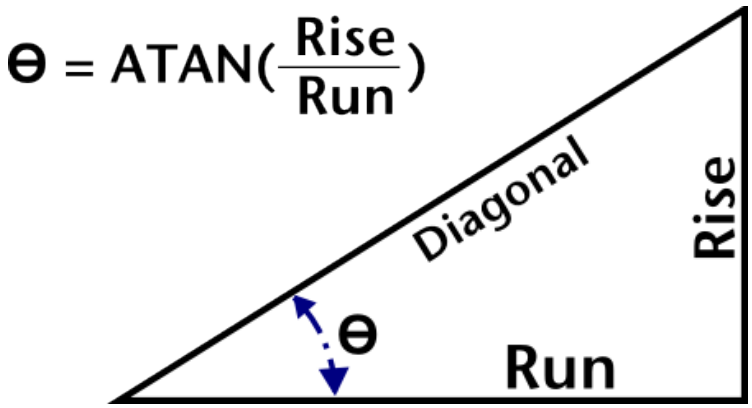
Input	*Display*
clear temporary memory ⓒ ⓒ	0
If necessary, swipe green keys to reveal trig keys	
12 ÷ 23 = [Conv] [ASIN]	31.45°

**Note: The Arccosine function is one of six trigonometric functions found on BuildCalc. You can access the trig functions by sweeping your finger across the green keys (switching [Length] [Width] [Height] for [SIN] [COS] [TAN]). Pressing the yellow [Conv] button will switch [SIN] [COS] [TAN] to [ASIN] [ACOS] [ATAN].*

[ATAN] Arctangent Function

1. Calculates the arctangent for a given angle.

2. The arctangent of a right triangle is the inverse of the tangent for that triangle.



Example:

1. Calculate arctangent for rise of 12 and run of 23.

Input	*Display*
clear temporary memory [C] [C]	0
If necessary, swipe green keys to reveal trig keys	
12÷23[ATAN]	27.55°

**Notes:*

- *The Arccosine function is one of six trigonometric functions found on BuildCalc. You can access the trig functions by sweeping your finger across the green keys (switching [Length] [Width] [Height] for [SIN] [COS] [TAN]).*
- *Pressing the yellow [Conv] button will switch [SIN] [COS] [TAN] to [ASIN] [ACOS] [ATAN].*

[⌫] Backspace Key

Delete the last digit(s) entered.

Example:

1. Remove and replace the denominator of a fraction just entered.

Input	*Display*
clear temporary memory [C] [C]	0
7 [/] 16	0- $^{7}/_{16}$in
[⌫] [⌫]	0- $^{7}/$in
8	0- $^{7}/_{8}$in

[Balstr] Baluster Function

The Baluster Function calculates horizontal and inclined baluster layouts – which can also be useful for more complex layout of other vertical members like pickets, studs and spindles.

BuildCalc has three Baluster Layout calculation modes:

- **Limit Opening**: Calculate the number of members and their locations with open space between members being no greater than an input value you provide.
- **Evenly Space**: For a given run and number of members, calculate their layout and maximum open spacing.
- **Best-Fit**: Given a desired on-center spacing and run, calculate the layout that is closest to the desired on-center spacing

Inputs to the [Balstr] function are shown in the "Input Parameters" section and can be given by one of the three following ways:

- Tapping on the parameter on the right of any cell in the "Input Parameters" section. In the example below, tap on 12ft in the "Run:" cell to change the run.
- Value stored in the [Run] memory function is automatically input into the appropriate Input Parameter field.
- If values have been stored using the Store button (at the bottom left), these values and results can be recalled using the Recall button. Note that the Recall button will not be present if values have not been stored.

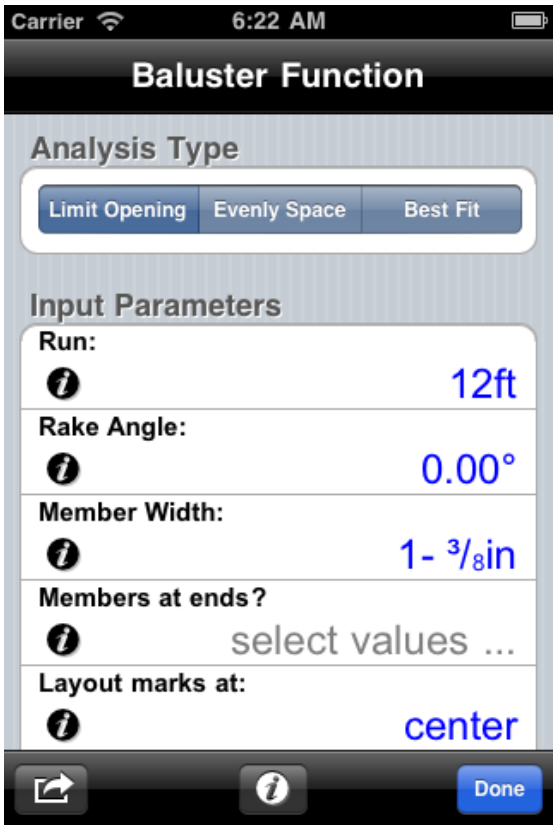
For more information on input parameters, tap the on-the-spot help at the bottom left of each input parameter cell.

Note: The [Balstr] function is not available when BuildCalc's Advanced Function mode is "OFF" (via [Conv] [Prefs]).

Examples:

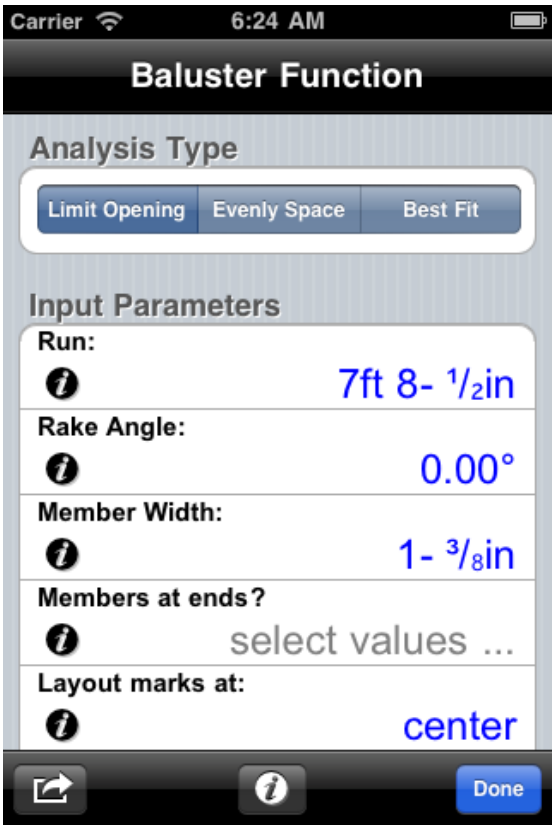
Example 1. You have a handrail you wish to install between two posts on a deck. The distance between the posts is 7' 8- ½". You are going to use 2- ¼" x 2- ¼" balusters. To meet code, the space between each baluster must be less than 4 inches. What is the layout that will use the least number of balusters?

Input	*Display*
clear temporary memory C C	0
[Conv] [Balstr]	*see screens below*



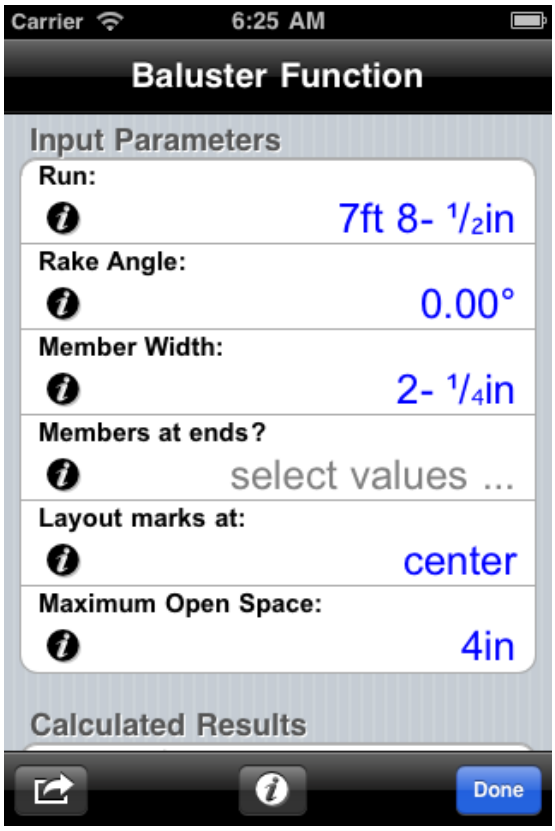
So first, let's enter the run for this baluster layout. Start by tapping on the **Run:** cell.

And now enter `7 [Feet] 8 [Inches] 1 / 2 [Done]`

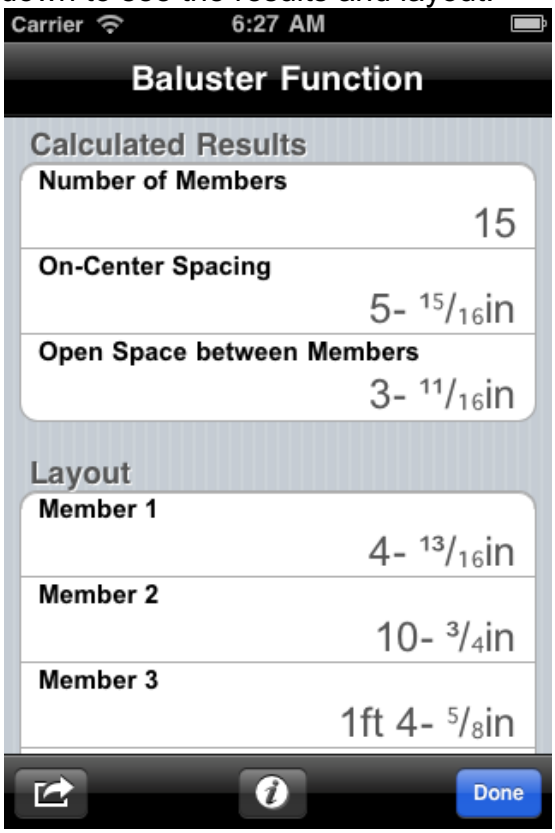


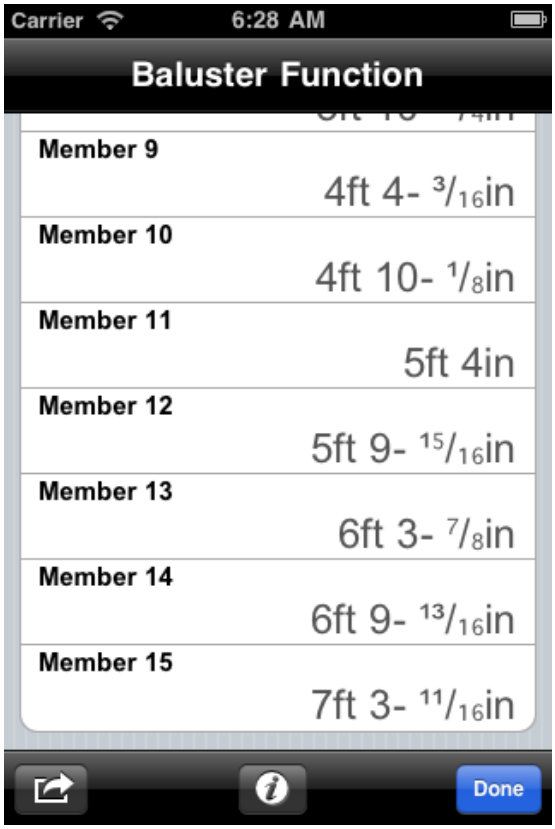
And now, tap on the **Member Width:** cell to enter the baluster size.

Enter `2 [Inches] 1 / 4 [Done]` for the baluster size.



The remaining input parameters appear to be what is needed for this analysis. For example, the **Maximum Open Space:** is already set to 4 inches. Scroll down to see the results and layout.



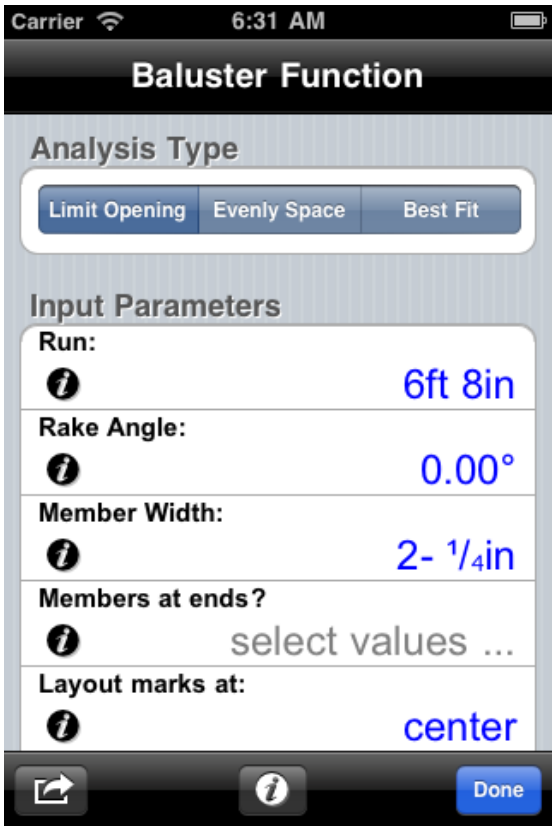


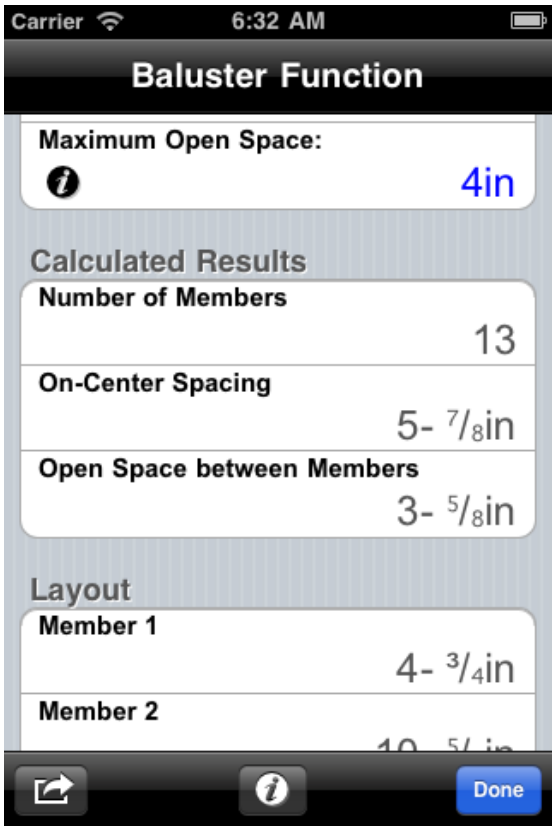
Example 2: You have three sections on a deck, each of a different span (6’ 8”, 7’ 9” and 4’ 5”), and you want all of them to have the same spacing for the balusters. Code requires that the space between balusters be no greater than 4” and you will be using 2- ¼” x 2- ¼” balusters. What is the layout for each span?

This is a little more complicated example but a common problem. First we will solve for each section and note the calculated on-center spacing. Then, using the smallest on-center spacing calculated from the three, we will calculate the baluster layout for each section.

Calculate the spacing for the 6’ 8” section:

Input	*Display*
6 [Feet] 8 [Inches]	6ft 8in
[Conv] [Balstr]	*see screens below*

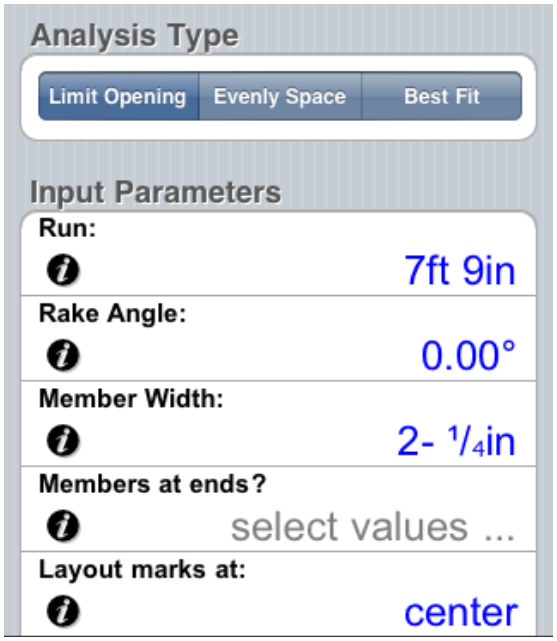


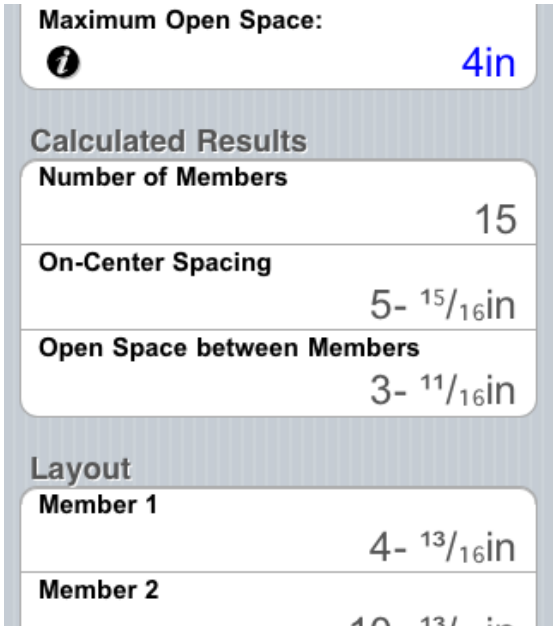


Here we can see that the calculated On-Center Spacing is 5- 7/8””. Be sure that the Analysis type is [Limit Opening] and the Maximum Open Space is 4”. If you want, you can store the results of this calculation using the [Store] button (first, tap the actions button, at the bottom left).

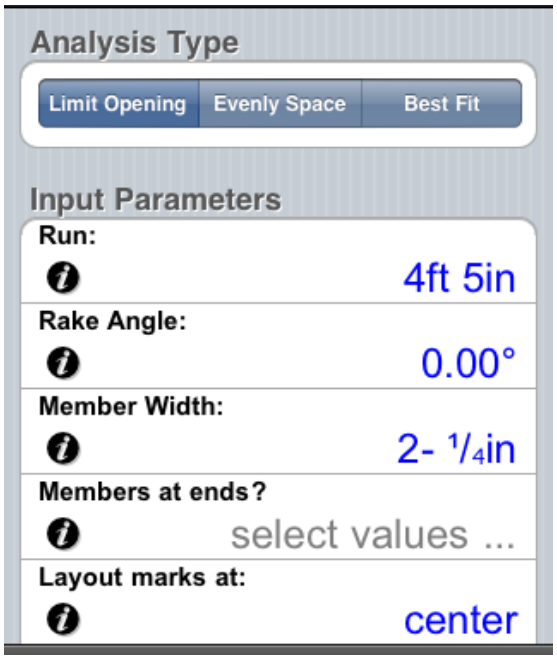
To calculate the On-Center Spacing for the other sections, tap on the Run cell, enter the run value and then note the On-Center spacing. Here are the results:

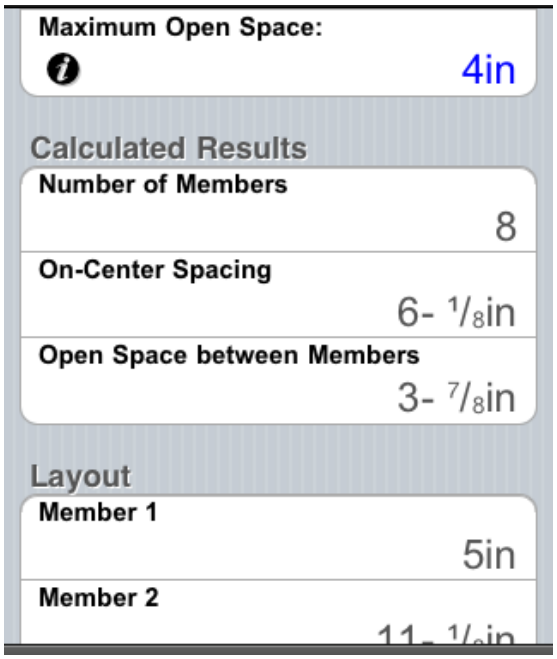
Section 2: 7’ 9”





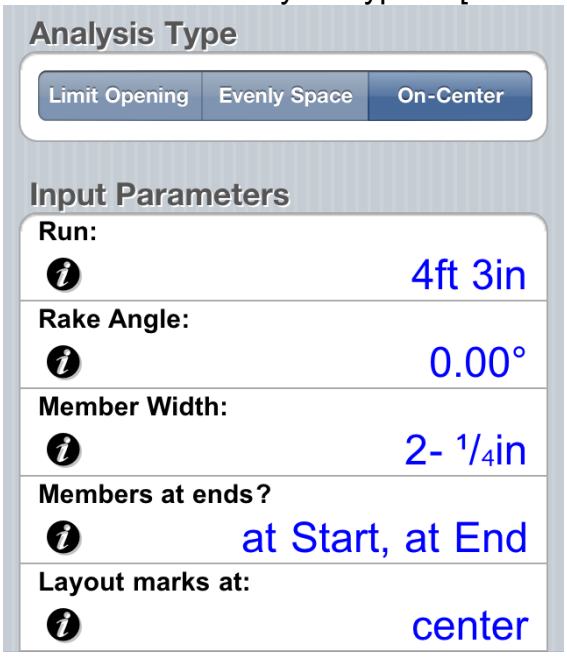
Section 3: 4’ 5”



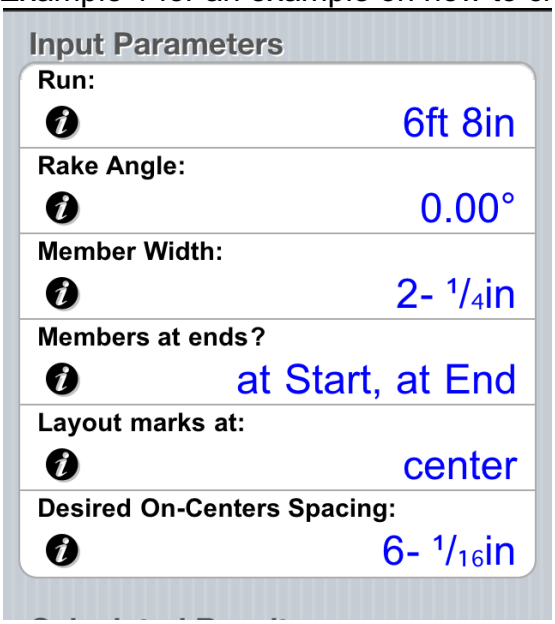


So the smallest On-Center spacing is 5- 7/8” and the largest 6- 1/8”. The range of opening sizes is 3- 5/8 to 3- 7/8”. Any of these will pass code. But to minimize the amount of difference between sections (in spacing between the posts and the balusters at the ends) let’s use the on-center value between the two extremes: 6 inches.

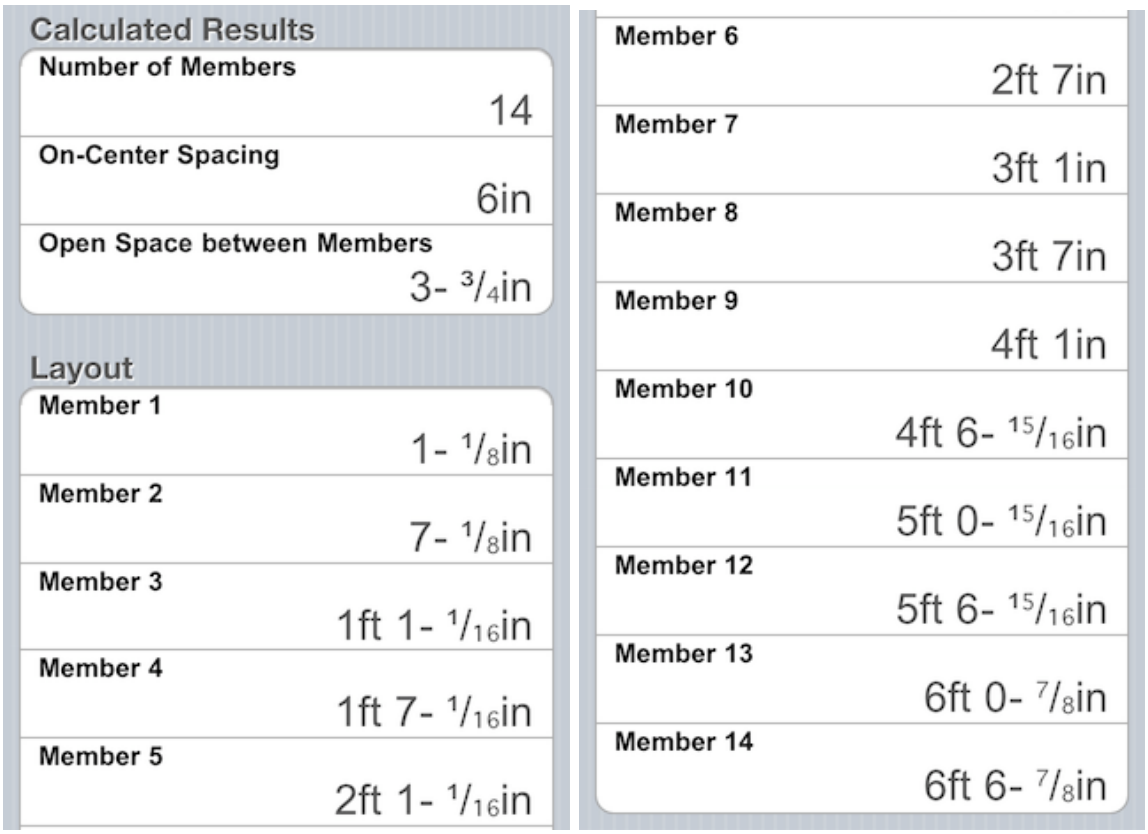
Now lets calculate the layout for each section for a 6” On-Center spacing. First, let’s switch the Analysis Type to [Best-Fit].



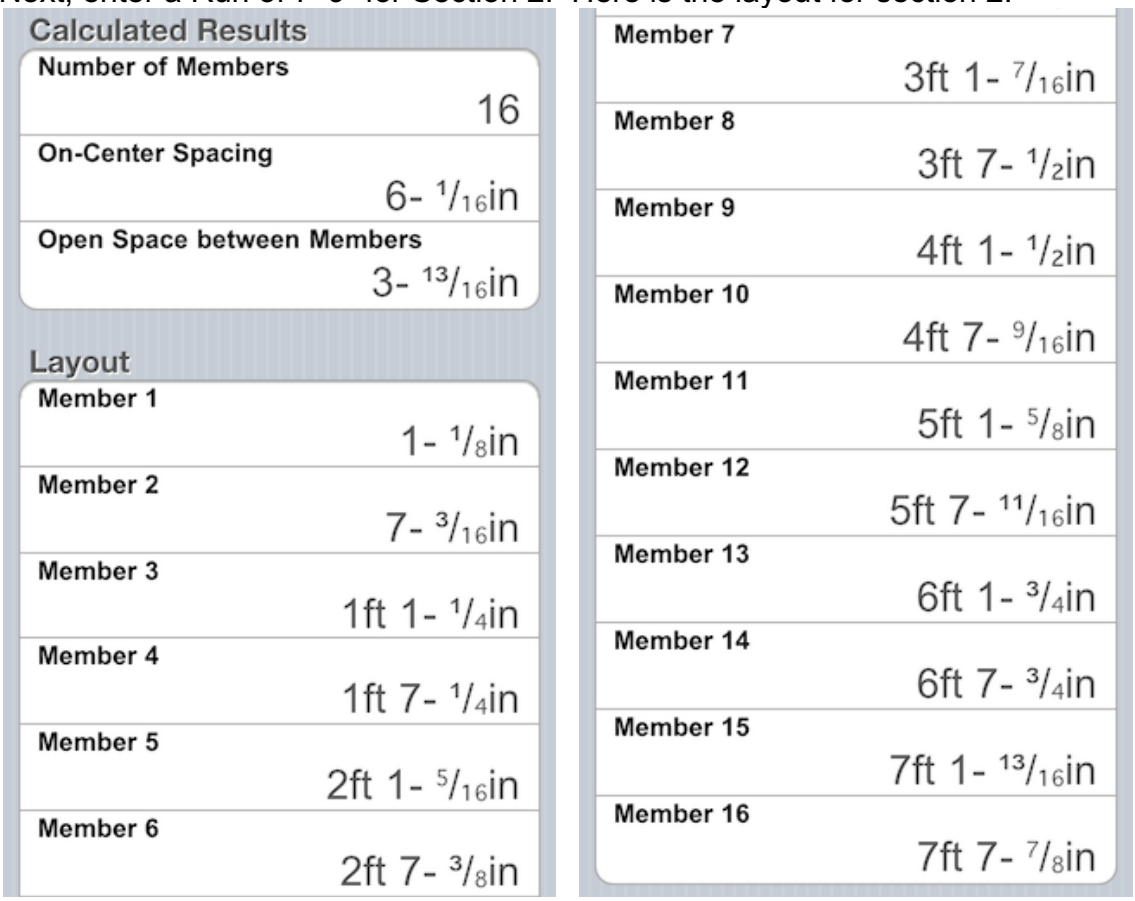
Next, change the Run to 6’ 8” and the On-Center Spacing to 6- 1/16” (see Example 1 for an example on how to change these values).



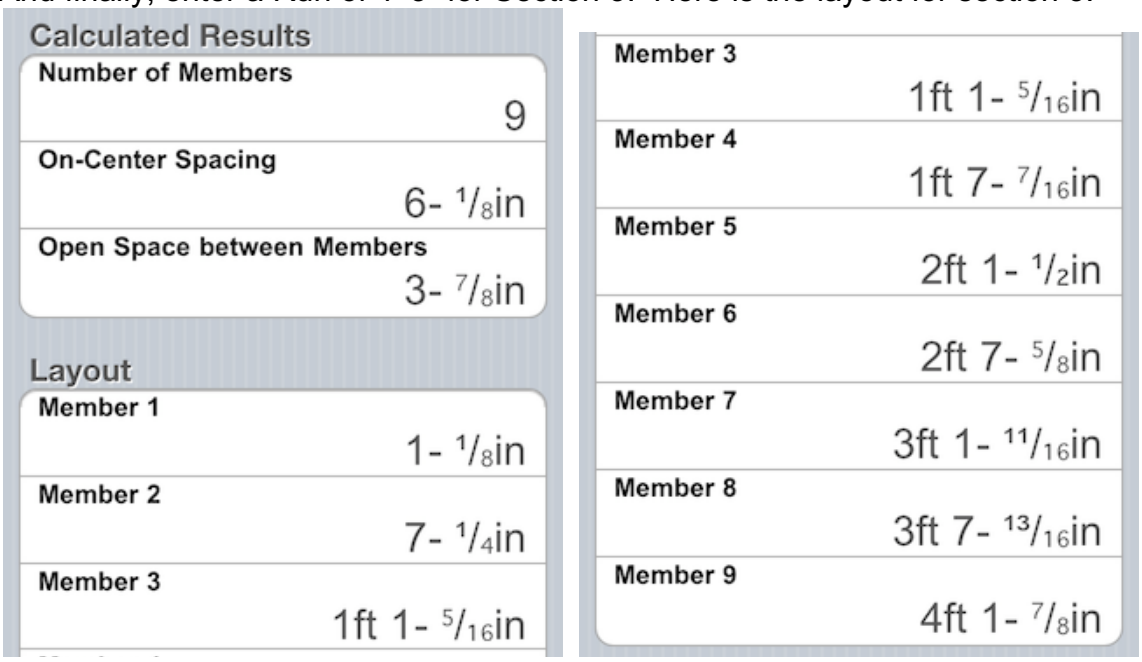
Now you can scroll down to see the layout for section 1:



Next, enter a Run of 7’ 9” for Section 2. Here is the layout for section 2:



And finally, enter a Run of 4’ 3” for Section 3. Here is the layout for section 3:

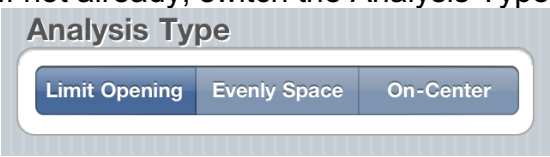


Example 3: You have a staircase for a deck that you need to run the balusters on two rails (the upper hand rail and a lower rail to support the balusters). The spacing between balusters can be no more than 4", the Run of the staircase is 6' and the pitch of the stairs (Angle of Incline or the Rake Angle) is 33.27° (something you calculated using the [Stairs] function). You will be using 2- ¼" x 2- ¼" balusters. What is the layout for these balusters?

Enter the Baluster Calculation Screen

Input	*Display*
[Conv] [Balstr]	*see screens below*

If not already, switch the Analysis Type to "Limit Opening".



Now enter 6' for the Run, 33.27° for the Rake Angle, and 2- ¼" for the Member Width.

Scroll down for the layout of the Balusters. Note that the layout values represent where to place the Balusters along the length of the Rails. Placing your tape measure from the start of the Rail to the end, you can mark the Rail using this layout and your balusters will be properly spaced.

[Bd Ft] Board Feet Function

1. Set the units for a quantity to board feet.
2. Convert a volume to board feet.
3. Convert a weight to board feet (using the density stored in [wt/vol]).

Examples:

1. **Set the units for a quantity to board feet.**

Input	*Display*
clear temporary memory 	0
35.2 [Conv] [Bd Ft]	35.2bf

2. **Convert a volume to board feet.**

Input	*Display*
clear temporary memory 	0
4 [m] x 37 [Conv] [cm] x 8 [Conv] [cm] = [m]	0.1184m^3
[Conv] [Bd Ft]	50.17508bf

3. Convert a weight to board feet (using the density stored in [wt/vol]).

Input	*Display*
clear temporary memory	0
465 [Conv] [lbs]	465lb
[Conv] [Bd Ft]	50.22bf

Notes:

- ** Assumes 1.5 tons per cubic yard is stored in* [wt/vol]
- *When multiplying or dividing with mixed units, BuildCalc's display of results is dependent upon your region. If in North America, BuildCalc displays the results for lengths and areas in feet, and volumes in yards. Elsewhere, BuildCalc displays the results for lengths, areas, and volumes in meters. The default display units for lengths, areas and volumes can be changed via [Conv] [Prefs].*

[Circle] Circle Function

Starting with version 2.0, BuildCalc's Circle function has been enhanced to display the most information with the least number of keystrokes. Because the first press of the [Circle] key is often used to store a diameter value for other calculations, no change has been made in function for the first key press. However, if the [Circle] key is pressed a second time, all Circle function results are displayed in a list.

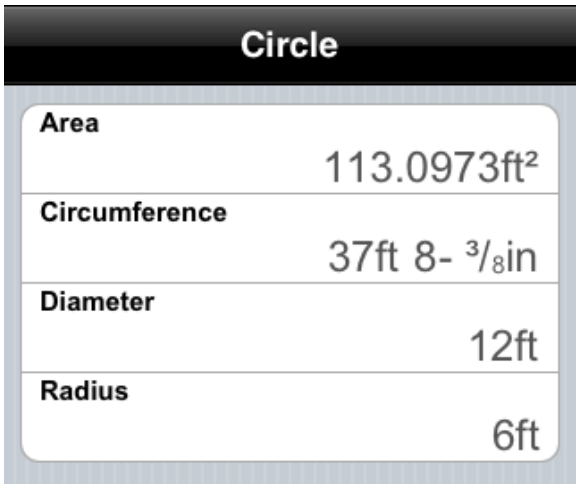
The Circle function calculates the descriptive geometry for a circle given the following input scenarios:

1. Input a [Radius].
2. Input a Diameter ([Circle]).
3. Given two of the following for an arc:
    - [Arc] Angle
    - Chord Length ([Run])
    - Segment Length ([Rise])

Examples:

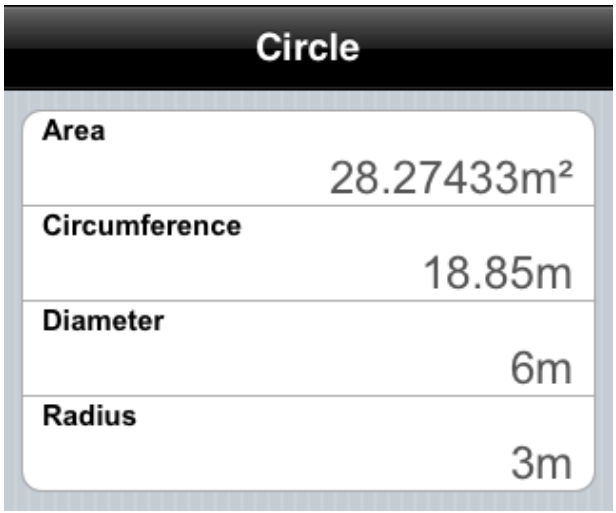
1. Input a Radius to calculate descriptive geometry for a circle.

Input	*Display*
clear temporary memory	0
6 [Feet] [Conv] [Radius]	Radius 6ft
[Circle]	Diameter 12ft
[Circle]	*See results, below*



2. Input a Diameter to calculate the descriptive geometry for a circle.

Input	*Display*
clear temporary memory [C] [C]	0
6 [m] [Circle]	Diameter 6m
[Circle]	*See results, below*



***Note:** See **"Finding the Radius or Diameter of an Arc"** in the arc function section.*

[ClrAll] Clear All Function

In addition to those values cleared with ⓒ ⓒ, [ClrAll] (or [Conv]x) resets the values stored at the following keys to their default values:

[M1]	[M2]	[M3]
[M+]	[FtArea]*	[wt/vol]
[o.c.]	[MsnSz]*	[SprAng]*
[TreadW]*	[RiserH]*	[FloorH]*

**Note: These keys are not available by default starting in BuildCalc 2.1 since they are redundant with the new functionality available in the [Footng], [Msnry], [CmpMtr] and [Stair] keys. To regress to the functionality of BuildCalc prior to version 2.0, switch "Advanced Function Mode" to OFF in BuildCalc's preferences ([Conv] [Prefs]).*

[cm] Centimeter Function

1. Set the units for a quantity to centimeters.
2. Convert a length, area or volume to centimeters.
3. Convert a weight to cubic centimeters (using the density stored in [wt/vol].

Examples:

1. Set the units for a quantity to centimeters.

Input	*Display*
clear temporary memory [C] [C]	0
3 [Conv] [cm]	3cm
[Conv] [cm]	$3cm^2$
[Conv] [cm]	$3cm^3$

2. Convert a length to centimeters.

Input	*Display*
clear temporary memory [C] [C]	0
56 [Inches]	56in
[Conv] [cm]	142.24cm

3. Convert a weight to cubic centimeters (using the density stored in [wt/vol]).

Input	*Display*
clear temporary memory 	0
0.125 [Conv] [lbs]	0.125lb
[Conv] [cm]	31.85645cm^3

**Note: Assumes 1.5 tons per cubic yard is stored in [wt/vol].*

[CmpMtr] Compound Miter Function

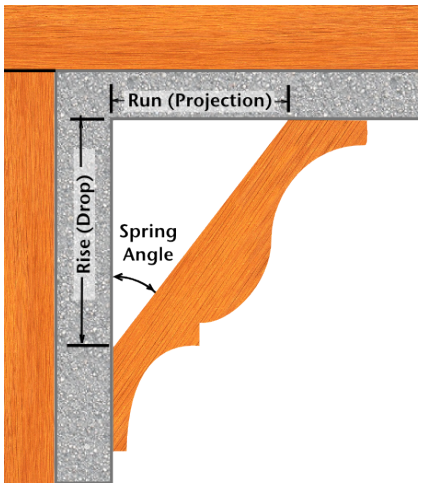
BuildCalc's version 2.0 Advanced Compound and Simple Miter Function is a bit different (and a lot simpler to use) than the Compound Miter function in earlier versions. The biggest change is that the results are interactive - meaning that you can make changes to inputs without having to start your calculation from the beginning. Just tap on the parameter on the right of any cell in the "Input Parameters" section and then modify that parameter.

However, for those familiar with older versions, [CmpMtr] works as it did before … but there is no need for the [SprAng] button. So now you can enter the Spring Angle along with the number of Corners and the Corner Angle any time you wish in the Input Parameters section of the Compound Miter Function.

And one more thing. Confused about how to layout Miter and Bevel results? You're not alone. Starting with BuildCalc 2.0, you can switch between Miter Saw results and Protractor "on the board" results (what you would measure using a hand held protractor). No more confusion and super easy flexibility.

The Compound Miter function calculates the table angle (Miter Angle) and the blade tilt angle (Bevel Angle) settings using the Spring and the Wall Corner Angle. In addition to interactively changing the input values of the [CmpMtr] function, Wall Corner Angle can be input prior to the press of the [CmpMtr] key using one of the following ways:

1. Input Wall Corner Angle directly (if it is greater than or equal to 25°).
2. Input number of room corners (if all of the rooms corner angles are equal and there are less than 25 corners).

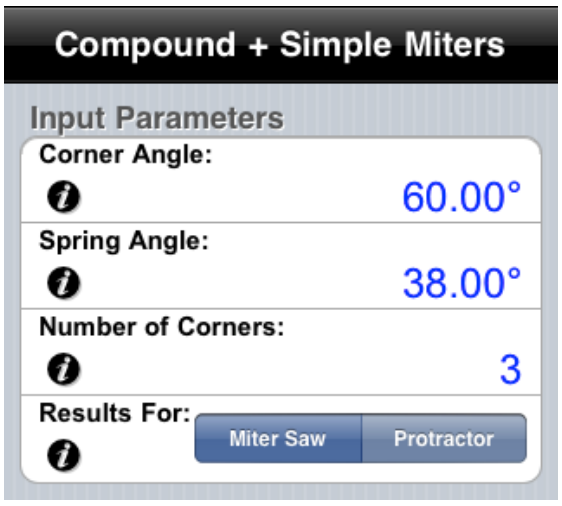


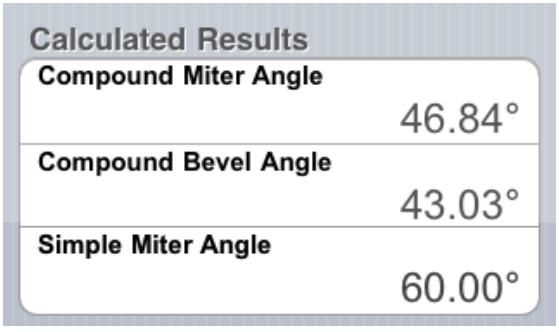
Compound Miter Parameter Illustration

Examples:

1. Input Wall Corner Angle directly (if Wall Corner Angle is ≥ 25˚).

Input	*Display*
clear temporary memory	0
60 [CmpMtr]	*See results, below*

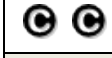




2. Input number of room corners (if all corner angles are equal and there are less than 25 corners).

Input	*Display*
clear temporary memory 	0
5 [CmpMtr]	*See results, below*

3. **Use default corner angle** (90˚).

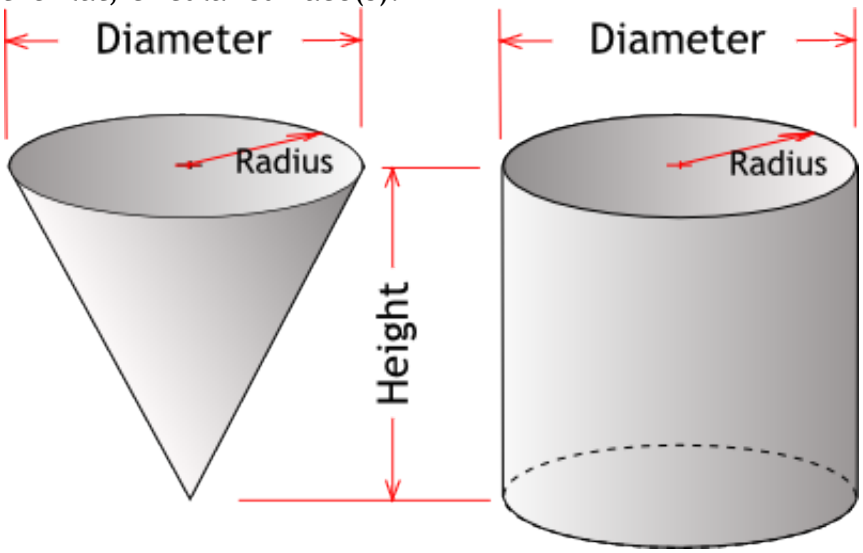
Input	*Display*
clear temporary memory 	0
[CmpMtr]	*See results, below*

[ColCon] Column / Cone Function

Starting with version 2.0, BuildCalc's Column/Cone function has been enhanced to display the most information with the least number of keystrokes. One press of the [ColCon] key and all Column/Cone function results are displayed in a list.

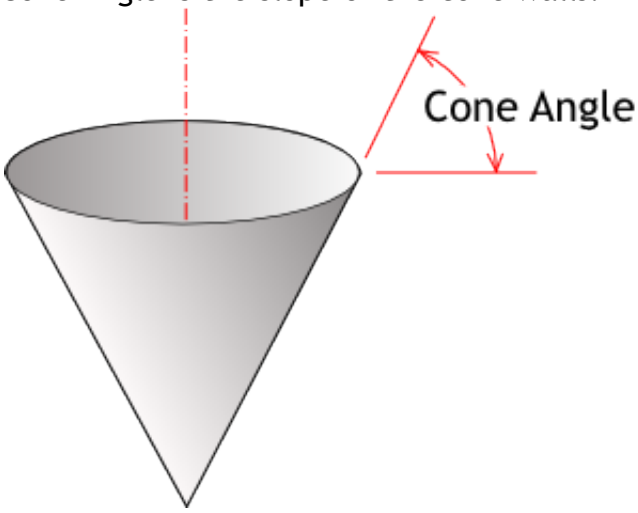
The Column/Cone function calculates descriptive geometry for a column and cone based upon a height value (stored in [Height]) and either a diameter (stored in [Circle]) or a [Radius].

Area Results:
Area results are for the curved surface (walls) of a column or cone and exclude the flat, circular surface(s).



Volume Results:
Think of the volume of a Column or Cone as the amount of something (like concrete) required to fill the column or cone.

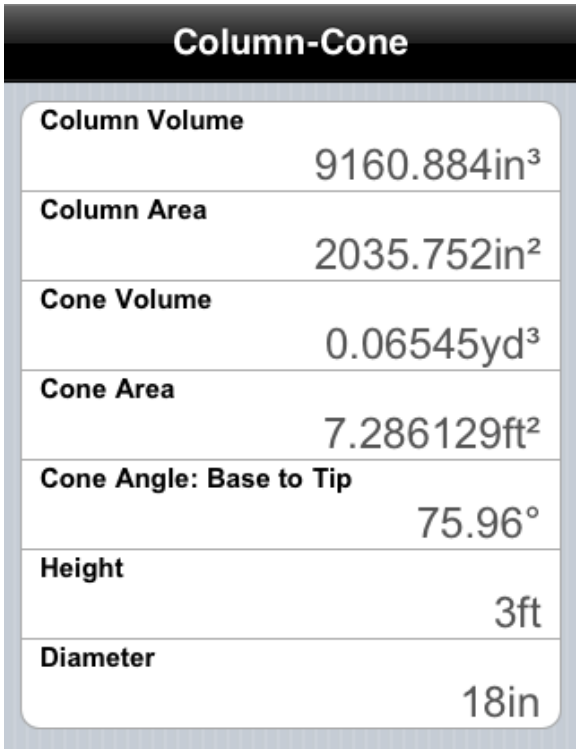
Cone Angle Result:
Cone Angle is the slope of the cone walls.



Examples:

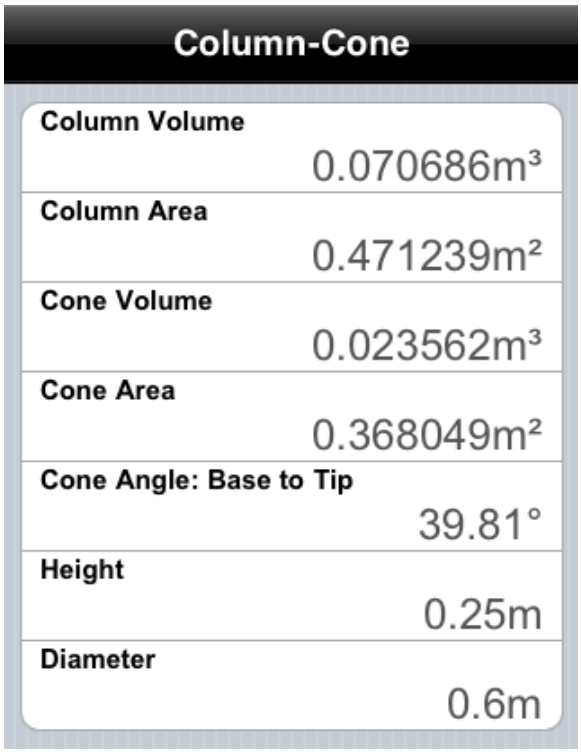
1. Input a [Radius].

Input	*Display*
clear temporary memory [C] [C]	0
3 [Feet] [Height]	Height 3ft
9 [Inch] [Conv] [Radius]	Radius 9in
[Conv] [ColCon]	*See results, below*



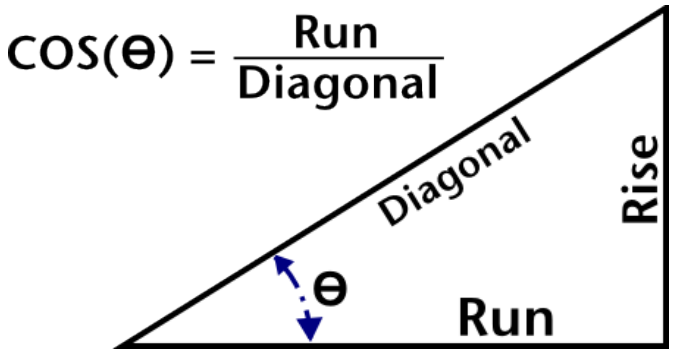
2. Input a Diameter ([Circle]).

Input	*Display*
clear temporary memory	0
0.6 [m] [Circle]	Diameter 0.6m
0.25 [m] [Height]	Height 0.25m
[Conv][ColCon]	*See results, below*


Column-Cone
Column Volume
0.070686m³
Column Area
0.471239m²
Cone Volume
0.023562m³
Cone Area
0.368049m²
Cone Angle: Base to Tip
39.81°
Height
0.25m
Diameter
0.6m

[COS] COSINE Function

Calculate the cosine for a given angle.



Example:

Calculate the cosine for 38°.

Input	*Display*
clear temporary memory C C	0
38 [COS]	0.788011

[cost] Cost Function

Displays a cost calculation result with the local currency symbol (see note below).

Example:

1. Calculate the cost of a given amount of board feet.

Input	*Display*
clear temporary memory [C] [C]	0
56 [Conv] [Bd Ft]	56bf
[x] 412 [Conv] [Cost]	56bf x $412 per 1000bf = $23.07
[C] [C]	0
9 [yard] [yard] [yard]	$9yd^3$
[x] 123 [Conv] [Cost]	$9yd^3$ x $123 per yd^3 = $1107.00

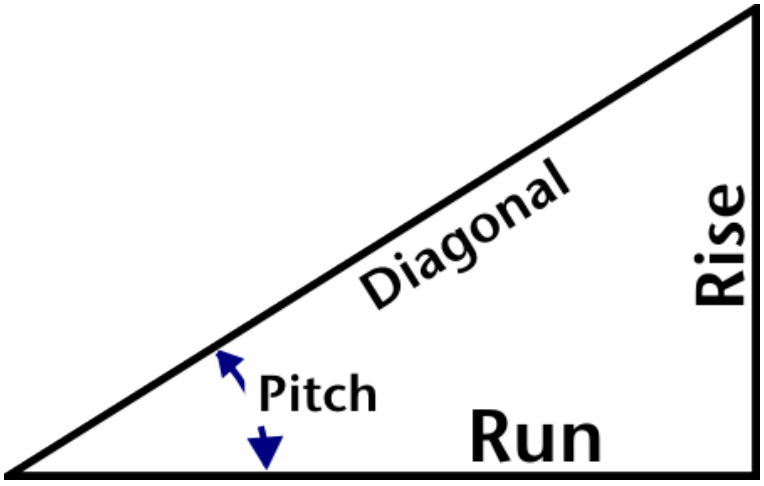
Notes:

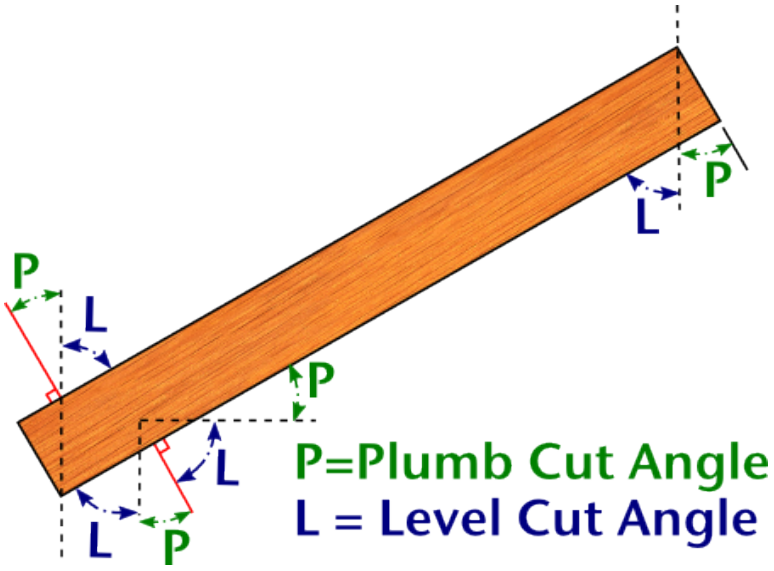
- *If units are Board Feet (*[BdFt]*), calculator assumes unit price is local currency per 1000 board feet.*
- *BuildCalc uses the local currency symbol based on your iPhone or iPod's regional configuration. This configuration value is set in the Settings App at General / International / Regional Format.*

[Diag] Diagonal Function

The Diagonal key can either:

- Calculate a Diagonal from any two of the following values stored:
  - [Pitch], [Rise], [Run]
- Calculate the Plumb and Level Cut Angles for a Common Rafter using two of the above stored values.
- Store a Diagonal value for other calculations.
- Recall a Diagonal value for display.



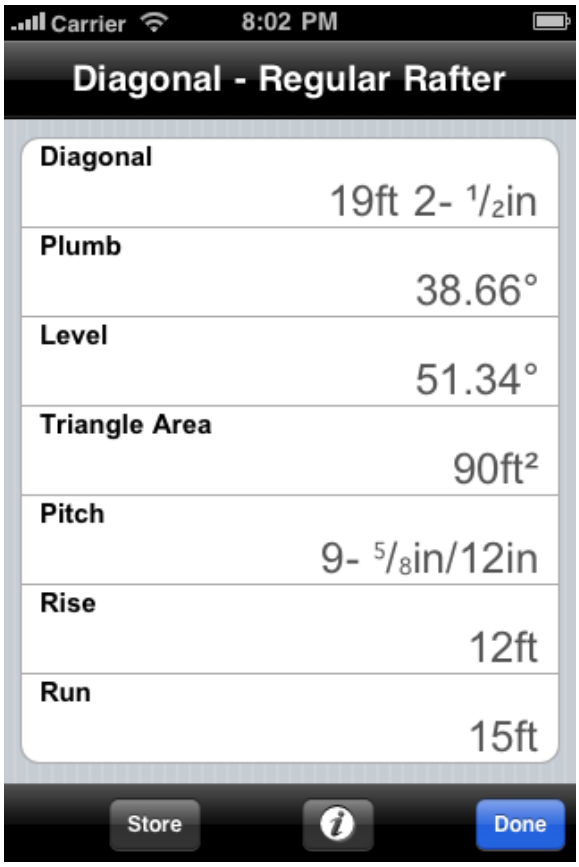


Examples:

1. Calculate the Common Rafter geometry from a given [Rise] and [Run].

Input	*Display*
clear temporary memory © ©	0
12 [Feet] [Rise]	Rise 12ft
15 [Feet] [Run]	Run 15ft
[Diag]	Diagonal 19ft 2- $^{1}/_{2}$in
[Diag]	*See results, below*

***Note**: Diagonal values will be cleared from memory upon © © or [ClrAll].*



2. Enter and Recall a Diagonal.

Input	*Display*
clear temporary memory 	0
12 [Feet] [Diag]	Diag 12ft
	0
[Diag]	Diag 12ft

[dmsdeg] Angular Conversion Function

1. Provides for entry of an D:M:S value
2. Conversion of an angular value

Example:

Enter an angle in D:M:S (degree, minute, second) format and then convert it to other angular formats.

Input	*Display*
clear temporary memory 	0
Enter an angle in D:M:S format 23.16.45	DMS 23:16:45
[Conv] [dmsdeg]	23.28°
[dmsdeg]	Pitch 5- 3/16in
[dmsdeg]	% Pitch 43.02371
[dmsdeg]	% Slope 0.430237
[dmsdeg]	Radians 0.406298rad

[Drywal] Drywall Function

The Enhanced Drywall Function calculates the number of sheets of Drywall*, Sheathing or Plywood based on the below input scenarios. The Drywall Function has been enhanced to allow you to view multiple sheet sizes and edit the list of sheet sizes to suit your needs. See the examples below to learn how this is done.

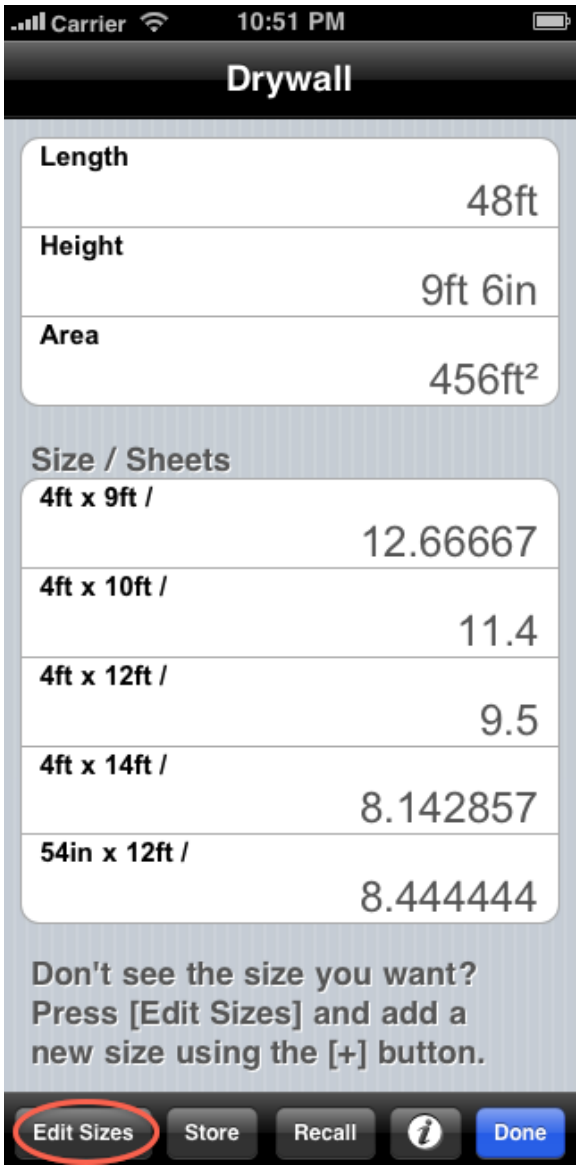
The Drywall function calculates the number of sheets of Drywall*, Sheathing or Plywood based on the following inputs:

1. Stored [Length] and [Height].
2. Stored [Length] and [Width].
3. Stored [Length] only.
4. Input Area.
5. Input Length.
6. Room Mode: Stored [Length], [Width] and [Height].

Example:

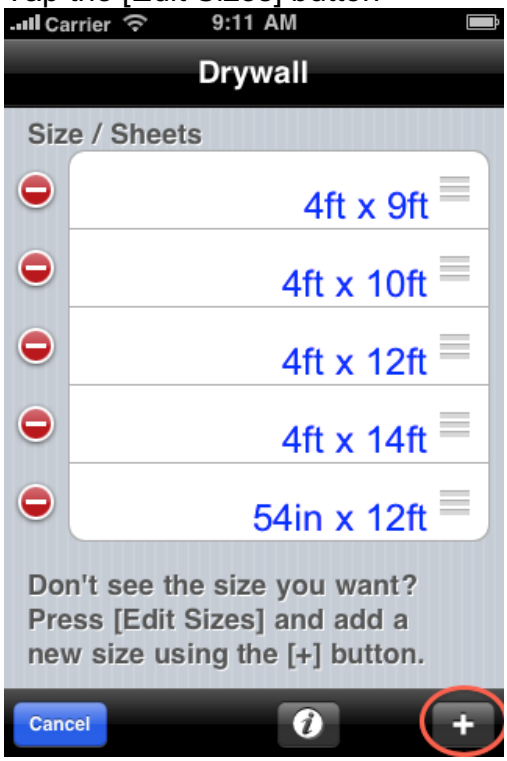
1. Stored [Length] and [Height].

Input	*Display*
clear temporary memory 	0
48 [Feet] [Length	Length 48ft
9 [Feet] 6 [Inch] [Height]	Height 9ft 6in
	0
[Conv] [Drywall]	*See results, below*



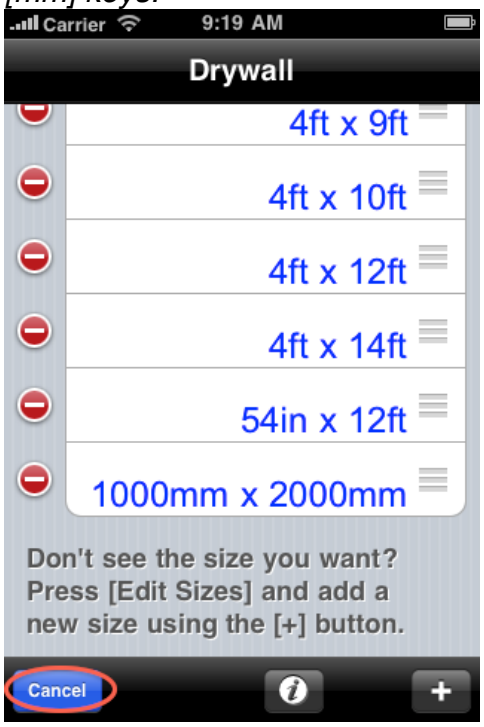
But let's say that the local building supply has a really good deal on Chinese drywall. It's size is 1000mm x 2000mm. How many sheets of that would you need? This is where the [Edit Sizes] button comes in handy. Try the following to get the result you want:

- Tap the [Edit Sizes] button



- Tap the [+] button to add a new size

- Enter `1000 [Conv] [mm] x 2000 [Conv] [mm] [done]` *Note: if in metric mode, you do not need to press [Conv] to access the metric unit keys. Also, you can quickly switch between metric and imperial mode by swiping your finger across the [Yards] [Feet] [Inches] keys or the [m] [cm] [mm] keys.*

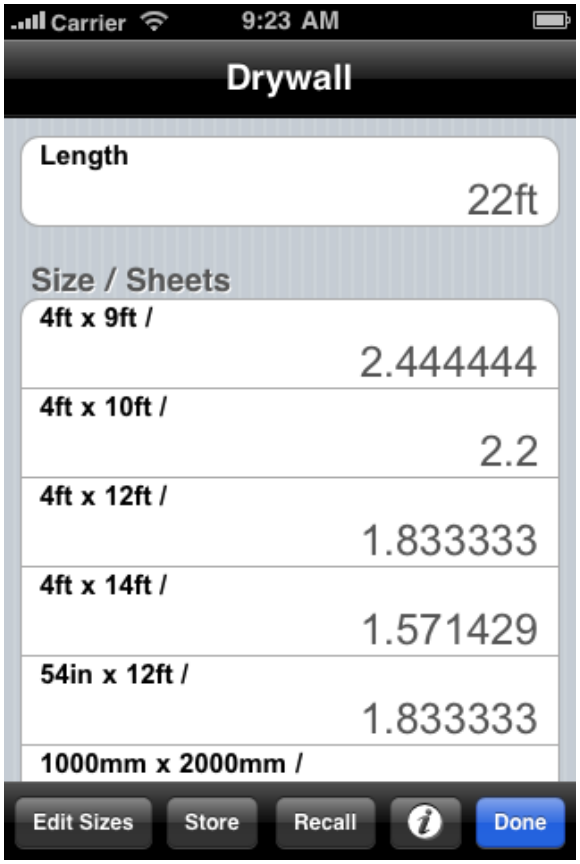


- Tap [Cancel] to exit the size editor

The result: you'll need 22 sheets of this Chinese drywall for this job.

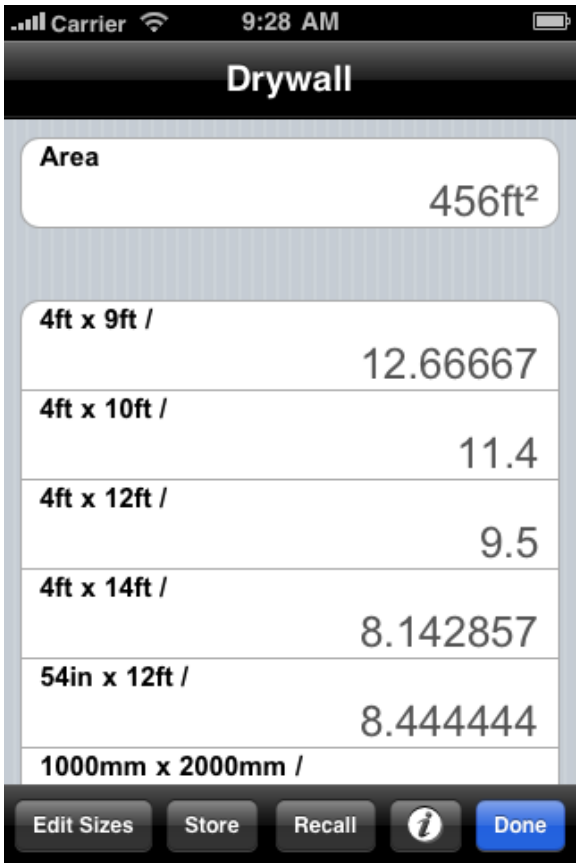
2. Stored [Length] only. How many 4'x10' sheets will be required to cover a length of 22'? *Note that, since drywall widths tend to have similar widths, BuildCalc uses the drywall length to make this calculation.*

Input	*Display*
clear temporary memory 	0
22 [Feet] [Length]	Length 22ft
	0
[Conv] [Drywall]	*See results, below*



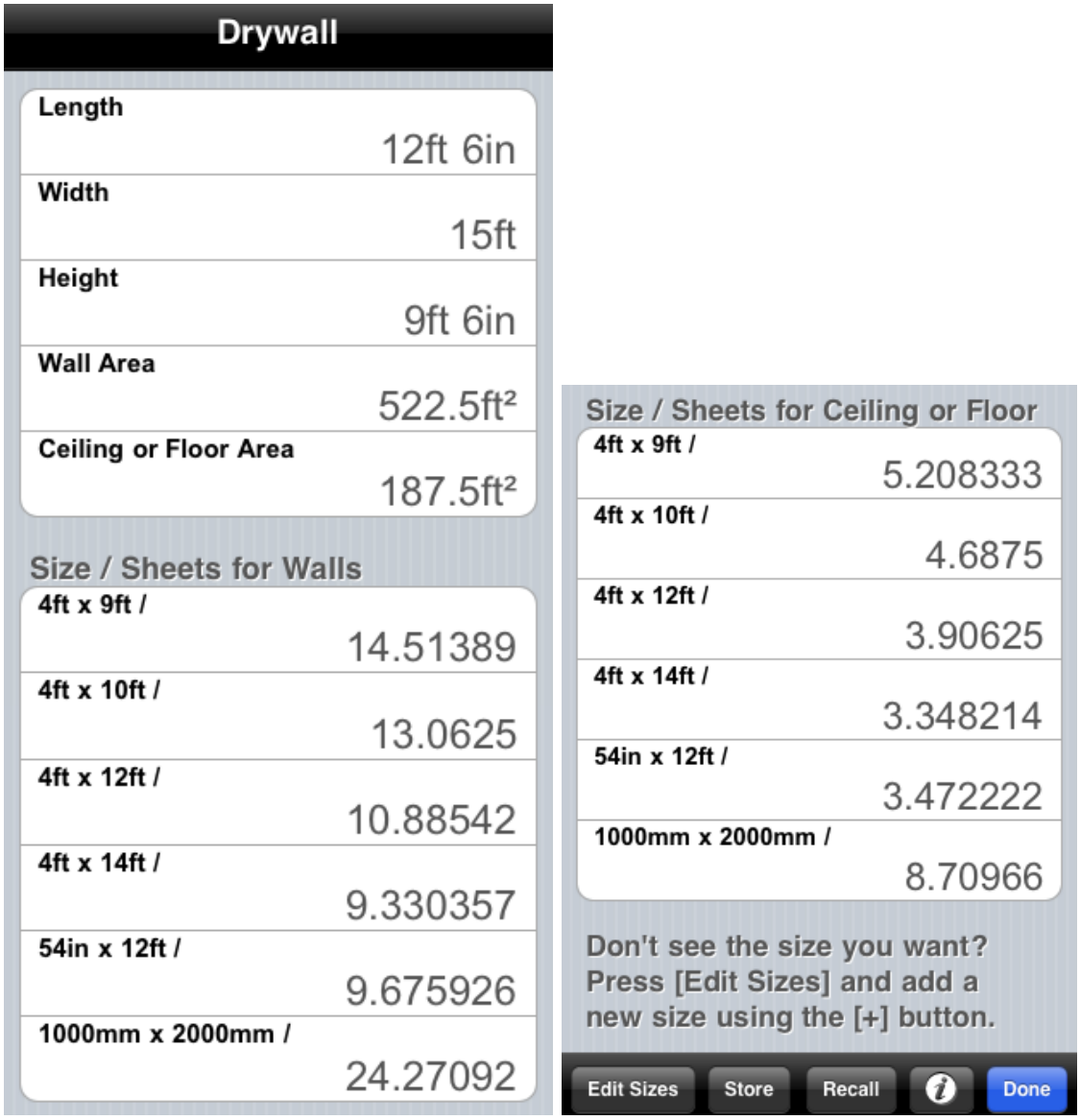
3. Input Area. You have a 48' x 9'6" area to cover with drywall. How many 4'x12' sheets are needed?

Input	*Display*
clear temporary memory 	0
48 [Feet] [x]	48ft
9 [Feet] 6 [Inch] =	$456ft^2$
[Conv] [Drywall]	*See results, below*



Room Mode: You have a 12' 6" x 15' room. The ceiling is 9' 6". You want to use 3/8" drywall on the walls and 1/2" drywall on the ceilings. In both cases, you plan on buy 4'x10' sheets.

Input	*Display*
clear temporary memory 	0
12 [Feet] 6 [Inches] [Length]	Length 12ft 6in
15 [Feet] [Width]	Width 15
9 [Feet] 6 [Inches]	Height 9ft 6in
[Conv] [Drywall]	*See results, below*



NOTE: Drywall is also called gypsum board, wallboard, plasterboard, rock lath, rigips, alçipan, and placoplatre – as well as a number of commercial names.

[Feet] Feet Entry and Conversion

1. Set the units for a quantity to feet
2. Convert a length, area or volume to feet
3. Switch between decimal and fractional display
4. Convert a weight to cubic feet

Examples:

1. Set the units for a quantity to feet.

Input	*Display*
clear temporary memory 	0
6 [Feet]	6ft
[Feet]	$6ft^2$
[Feet]	$6ft^3$

2. Convert a length, area or volume to feet.

Input	*Display*
clear temporary memory 	0
18 [Inch]	18in
[Feet]	1.5ft

3. Switch between decimal and fractional display.

Input	*Display*
clear temporary memory 	0
33 [Inch] 7 [/] 16	33- $^{7}/_{16}$in
[Feet]	2ft 9- $^{7}/_{16}$in
[Feet]	2.786458ft

4. Convert a weight to cubic feet (using the density stored in [wt/vol]).

Input	*Display*
clear temporary memory 	0
1.25 [Conv] [tons]	1.25T
[Feet]	22.5ft^3

**Note: Assumes 1.5 tons per cubic yard is stored in* [wt/vol].

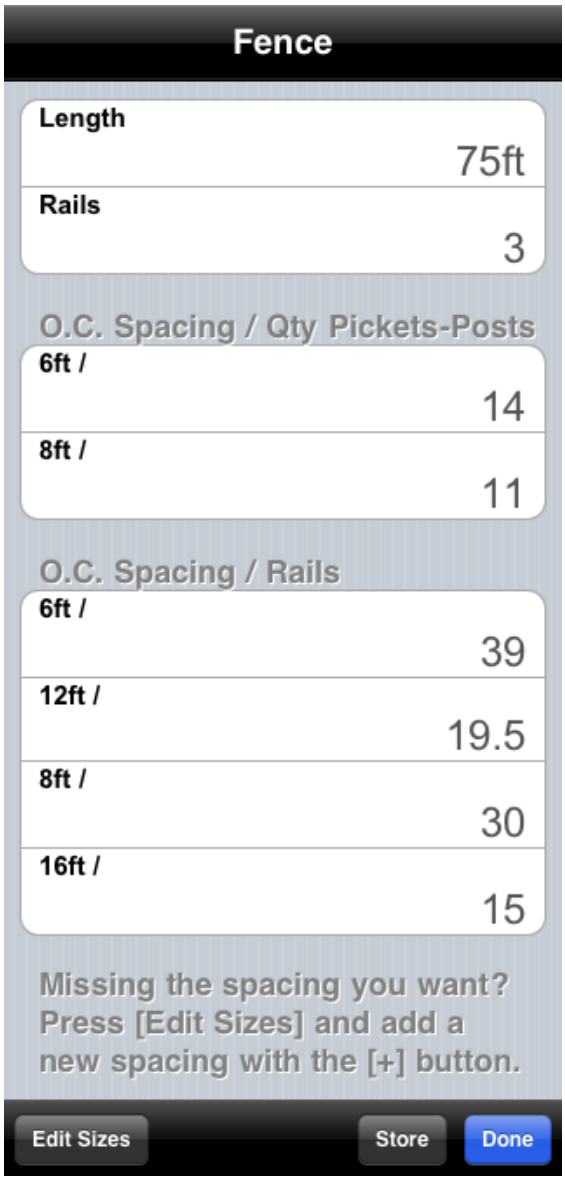
[Fence] Fence Material Estimation Function

Calculates the number of fence elements (pickets, panels, posts, rails) based on your input length for the fence row. Use the [Rails] key to store the number of rails per section used for this calculation.

Examples:

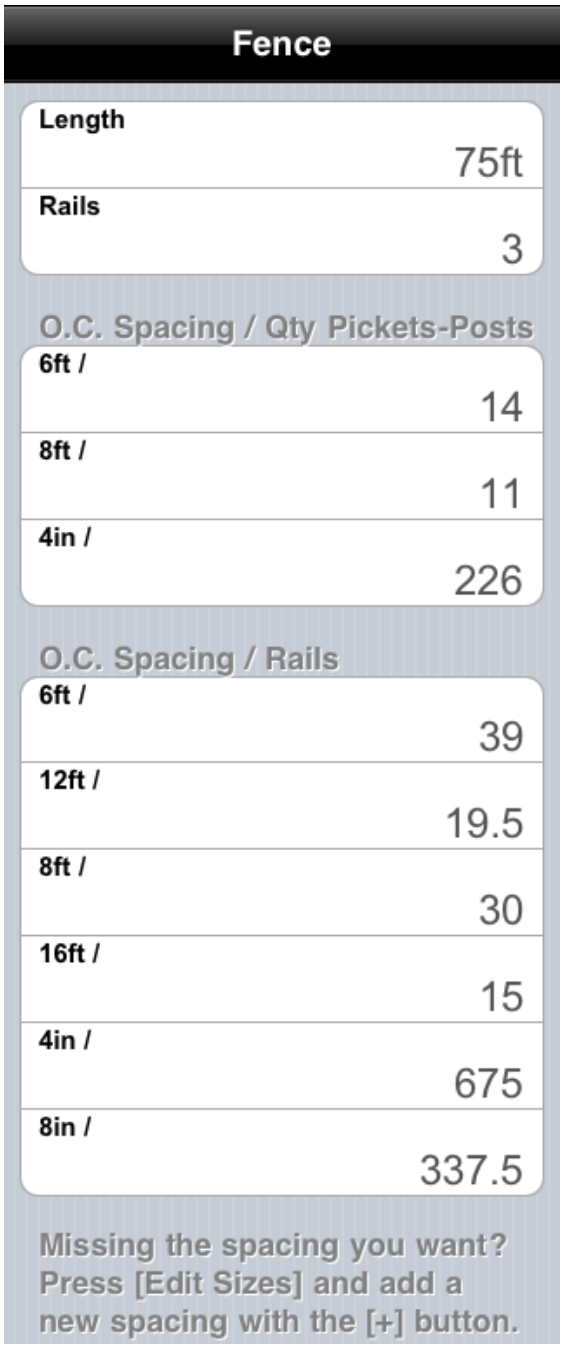
1. How many Pickets, Post and Rails are needed for a 75' fence row with 3 rails per section, posts spaced at 8' and with 3" pickets spaced at 4" centers.

Input	*Display*
clear temporary memory [C] [C]	0
3 [Store] [Rails]	Rails 3
75 [Feet] [Length]	Length 75ft
[Conv] [Fence]	*See results, below*



So you will need 11 Posts and 30 Rails … but where is the result for 3” pickets spaced at 4” centers? Here’s how to add them:

- First, tap the [Edit Sizes] button
- Then tap the [+] button
- Next, enter **`4[Inches]`** (this is the on-center spacing for these 3" pickets) and **`[Done]`**
- Now that you're done adding your additional spacing, click [Cancel].



2. You have 33 8’ fence panels. How far will that go?

Input	*Display*
clear temporary memory [C] [C]	0
33 [Conv] [Fence]	*See results, below*

Notes:

- The above example is based on the inputs of example 1.
- "So what is up with there being more results in the rails section than in the Pickets-Posts section?" Well, for some fences, you might have your posts spaced at 8' but you want to use 16' rails. Considering this, BuildCalc automatically calculates these results for you.
- "Argh! Why does all of my picket lengths show up in the rails count section. I mean, why can't BuildCalc see that it is a short distance and thus, it's not a rail?". Fair question. Unfortunately, it turns out to be harder to do this than it should be ... which means that I am stretching the current code too far. So, check back again with BuildCalc version 2.1. This will be fixed!

[Footng] Footing Function

Calculates the volume of a footing based on the following input scenarios:

1. Stored Length
2. Input Length

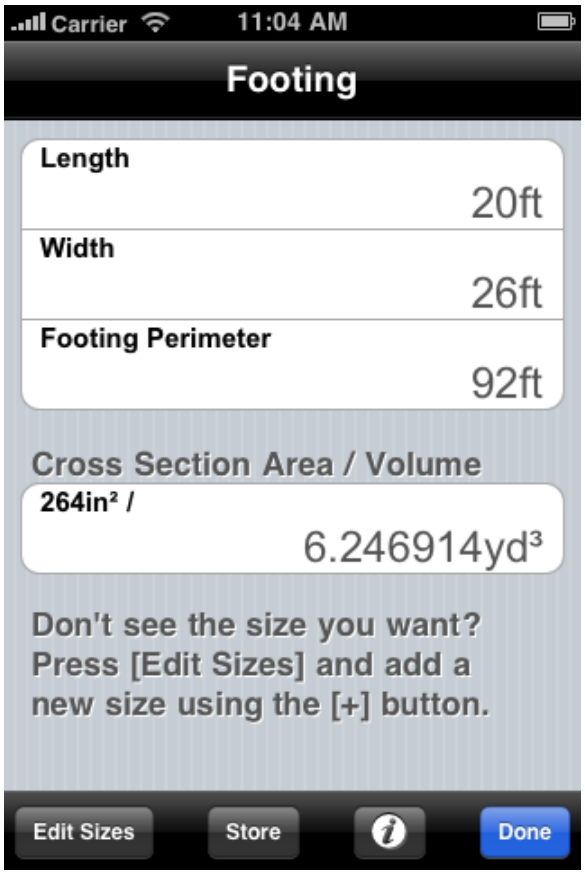
Example:

1. Volume of a footing based upon Stored Length.

Input	*Display*
clear temporary memory [C] [C]	0
22 [Feet] [Length]	Length 22ft
[Conv] [Footng]	*See results, below*

2. You have a 26' x 20' garage that calls for a 24" x 18" footer.

Input	*Display*
clear temporary memory 	0
26 [Feet] [Width]	Width 26ft
20 [Feet] [Length]	Length 20ft
[Conv] [Footng]	*See results, below*

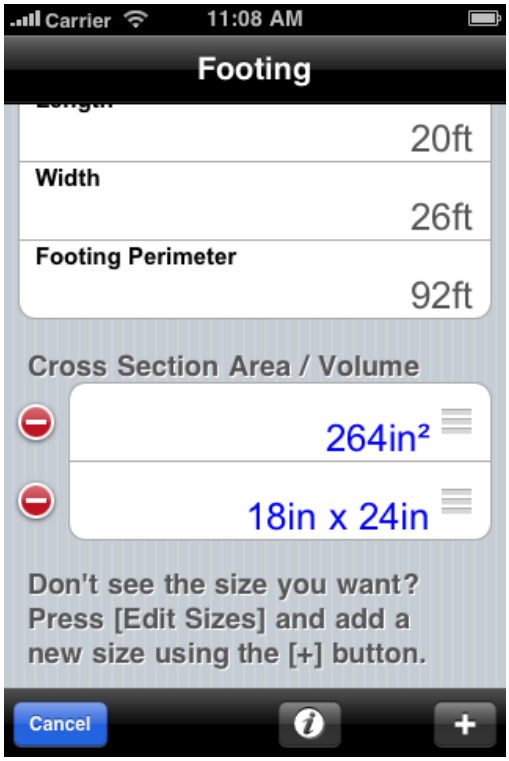


But wait a minute! 264in^2 isn't the size of foot needed for this calculation. How do we get results for a 18"x24" footer cross section? Try the following:

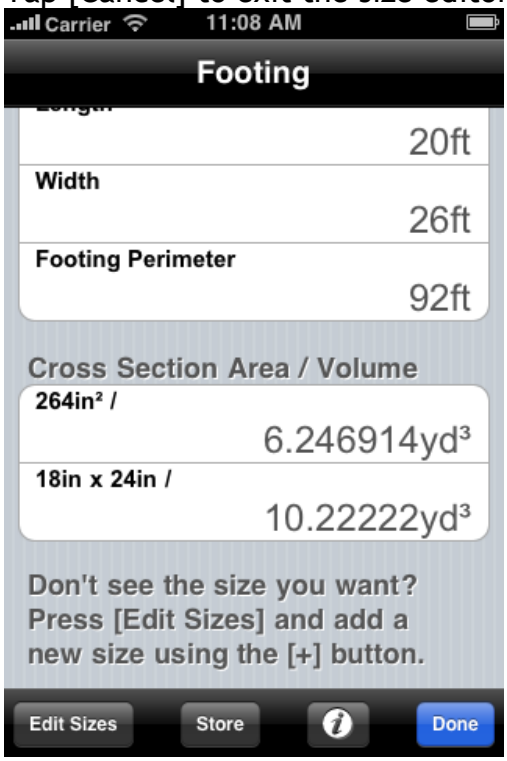
- Tap [Edit Sizes] to start the size editor

- Tap [+] to add a new size

- Enter `18[Inches]x24[Inches][done]`



- Tap [Cancel] to exit the size editor



Fractions! How to work with them in BuildCalc

There are a couple of ways for BuildCalc to give you fractional results. Here are some examples:

Entering Fractions:
Let's say that you have a run that is 13' 6-3/4" and you want to mark it into 8 equal sections. How would you do that?

Input	*Display*
13 [Feet] 6 [Inches] 3/4	13ft 6- $^{3}/_{4}$in
÷ 8 =	1ft 8- $^{3}/_{8}$in

Converting Decimals to Fractions:
So, that is a pretty simple example. Now, let's try something more complicated. This time you have 1 yard of concrete and you want to see how long of 5" deep by 36" wide sidewalk can be poured with that concrete.

Input	*Display*
1 [Yards] [Yards] [Yards]	1yd^3
÷ 5 [Inches]	1yd^3 ÷ 5in
÷ 3 [Feet]	64.8ft^2 ÷ 3ft
=	21.6ft
[Inches]	259.2in
[Inches]	259- $^{3}/_{16}$in

But, let say you want to see the results in a different denominator. In other words, instead of the results being in 16ths of an inch, you want to see the results in quarters of an inch. There are a couple of ways of doing this as well. First, if you have a result of a calculation, it is pretty easy to see that result in a different denominator with a couple of keystrokes. Using the results of the previous example ...

Input	*Display*
from the previous example	259- $^{3}/_{16}$in
[Conv] [1/4]	259- $^{1}/_{4}$in
and for grins … [Conv] [1/64]	259- $^{13}/_{64}$in
[Conv] [1/32]	259- $^{3}/_{16}$in

Notice that the results in 32nds is the same as it is in 16ths. This is because 259- $^{3}/_{16}$" (the same as 259- $^{6}/_{32}$") is more precise than 259- $^{5}/_{32}$" or 259- $^{7}/_{32}$" ... and for convenience, BuildCalc rounds 259- $^{6}/_{32}$" to 259- $^{3}/_{16}$" for you.

The second way in which you can influence the display of fractional results is to change the default denominator. This is done via BuildCalc's preferences.

Input	*Display*
[Conv] [Prefs]	*BuildCalc's preferences menu*
scroll down and select "Fractional Resolution"	
select the denominator to display your fractional results	

[Height] Height Function

1. Store and Recall a height for use by the [perArea], [DryWal] and [ColCon] (Column/Cone) functions.
2. Calculate Volume, Wall Area and Room Area (walls + ceiling).

Examples:

1. Store and Recall a height for use by the [Msnry]**, [DryWal], and [ColCon] (Column/Cone) functions.**

Input	*Display*
clear temporary memory [C] [C]	0
[Recall] [Height]	0
9 [Feet] 6 [Inch] [Height]	Height 9ft 6in
[C]	0
[Height]	Height 9ft 6in

2. Calculate Volume, Wall Area and Room Area (walls + ceiling).

Input	*Display*
clear temporary memory 	0
22 [Feet] [Length]	Length 22ft
18 [Feet] 8 [Inch] [Width]	Width 18ft 8in
9 [Feet] 6 [Inch] [Height]	Height 9ft 6in
[Height]	*See results, below*

[Hip/V] Hip/Valley Rafter Function

In other constructions calculators, setting up a calculation for a Regular or Irregular hip roof can be a real puzzle. Starting with BuildCalc 2.0, this is no longer the case. Inputs are all in one place, clearly labeled, and all have "on the spot" help to provide you with the missing pieces. In addition, BuildCalc's Hip/Valley function has been enhanced to display the most information with the least number of keystrokes. One press of the [Hip/V] key and all Hip/Valley function results are displayed in a list – and fast. And the [Hip/V] function is now interactive. What to change input values? Change the Minor Pitch? Switch between Regular and Irregular Hip Roof analysis? You can do this and more. Lastly, your results will be updated automatically - without you needing to remember what key to press. All the examples below illustrate this new feature.

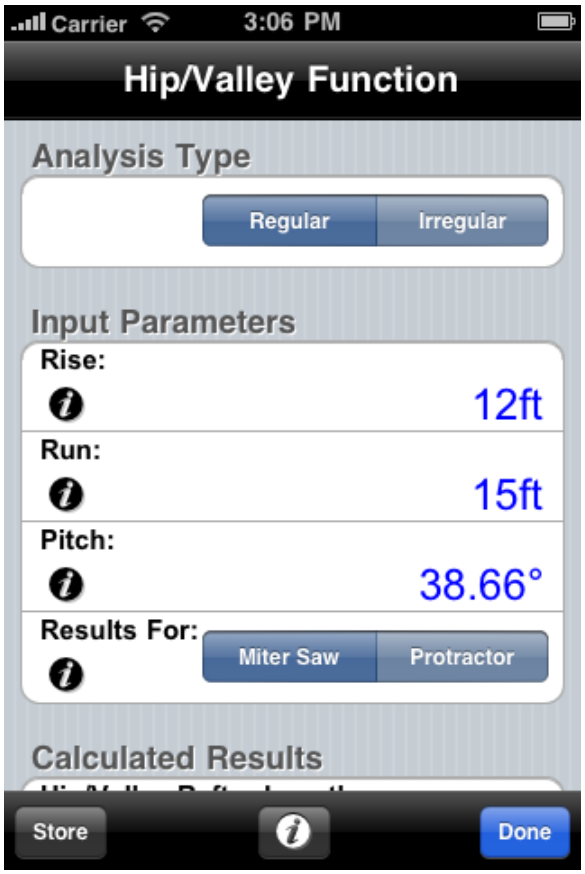
For those experienced with BuildCalc, we made sure that you would have to make no compromises - the functionality is still there (see example 3) in both keystroke and interactive form. Version 2.0's [Hip/V] key can still calculate the layout geometry for a Hip or Valley Rafter based on the Regular and Irregular roof geometry, as described by the values saved at two of the following: [Pitch], [Rise], [Run] and [Diag]. See the examples below to see how this works.

And one more thing. Confused about how to layout Miter and Bevel results? You're not alone. Starting with BuildCalc 2.0, you can switch between Miter Saw results and Protractor "on the board" results (what you would measure using a hand held protractor). Less confusion and more flexibility.

Examples:

1. **Regular Hip/Valley Rafter Calculation** with a 15' rise and a 7- ½" pitch.

Input	*Display*
clear temporary memory C C	0
[Hip/V]	*See results, below*

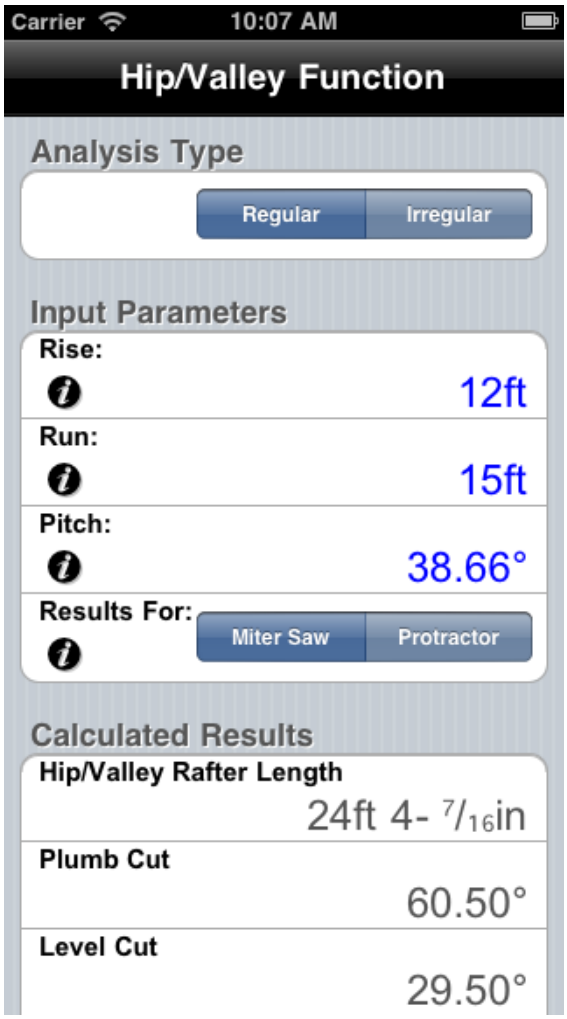


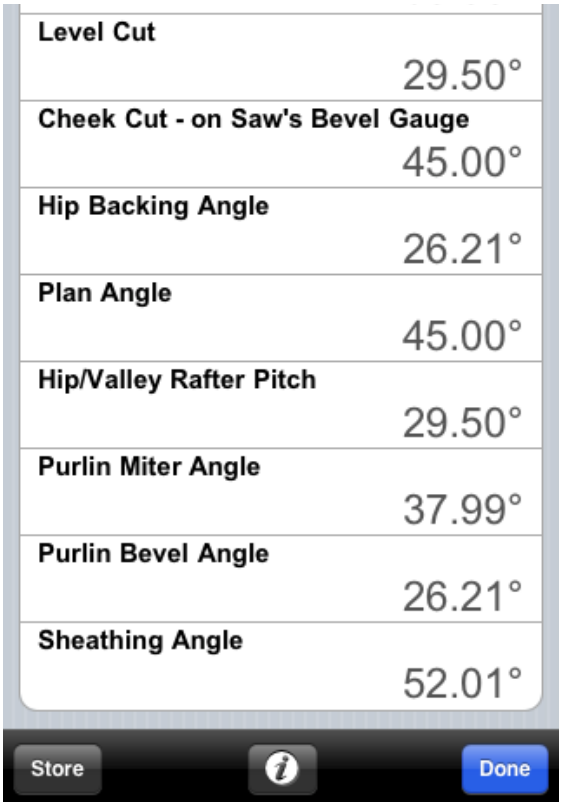
In the above interactive screen, enter 15 for the rise and 7[Inches]1/2 for the Pitch. Notice that, if the units stay the same (like when you changed the Rise from 12' to 15') then you don't have to enter the units! On the flip side, when you changed the pitch from degrees to inches per 12" pitch, you changed units - and thus had to enter the units. Below are the results:

Also note that you can quickly switch between Regular and Irregular Hip/Valley Rafter Calculations using the [Regular / Irregular] switch at the top of the Hip/Valley Function screen.

2. Irregular Hip/Valley Rafter Calculation

Input	*Display*
clear temporary memory C C	0
12 [Feet] Rise	Rise 12ft
15 [Feet] Run	Run 15ft
[Hip/V]	*See results, below*

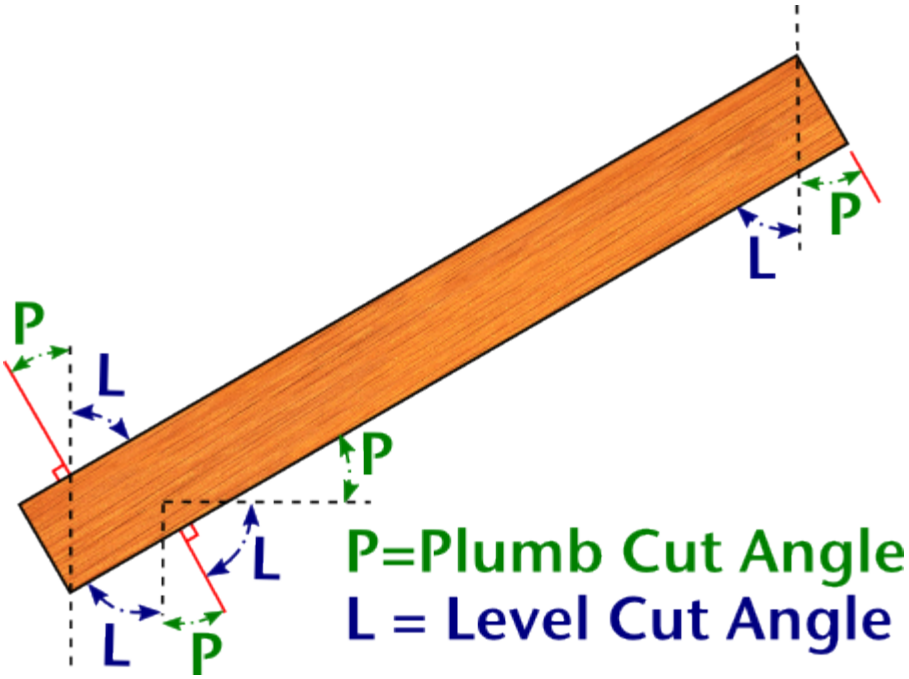


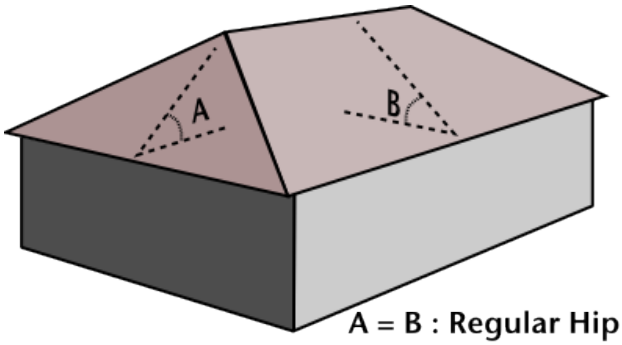


3. Irregular Hip/Valley Rafter Calculation assuming an Irregular Pitch of 30˚.

Input	*Display*
clear temporary memory C C	0
12 [Feet] Rise	Rise 12ft
15 [Feet] Run	Run 15ft
30 [Hip/V]	*See results, below*

Definitions:



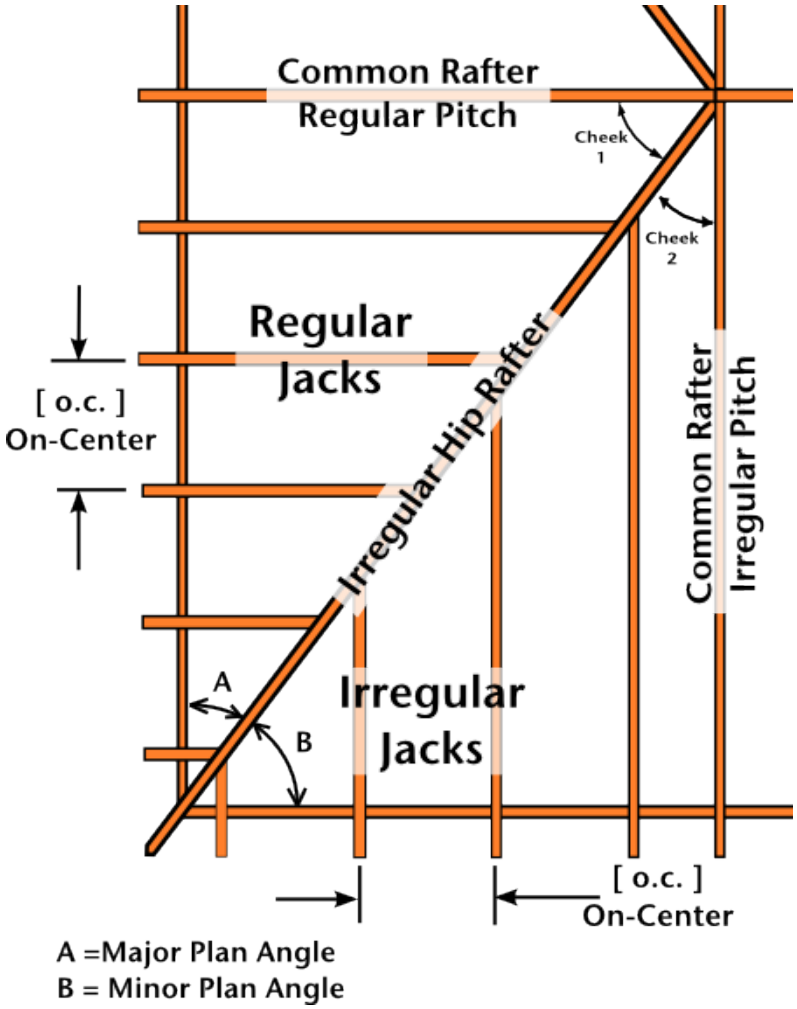


A = B : Regular Hip Roof
A ≠ B : Irregular Hip Roof

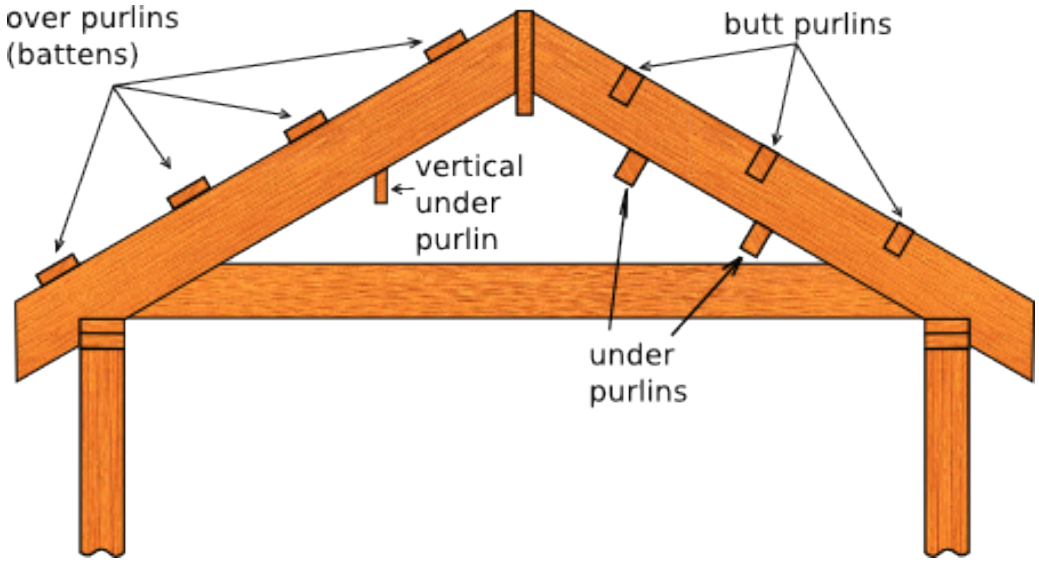
If the Hip or Valley is the junction of two roofs of different pitches, then the second (irregular or minor) roof pitch is entered into the [IrPitch] location prior to using the [Hip/V] function.

Notes:

- *The above image illustrates the Cheek Cut Angles for an Irregular Pitch Roof.*
- *For a Regular Pitch Roof, Cheek Cut Angle 1 = Cheek Cut Angle 2.*



Plan View of Hip/Valley Layout



Types of Purilins (BuildCalc calculates Under and Butt Purlins)

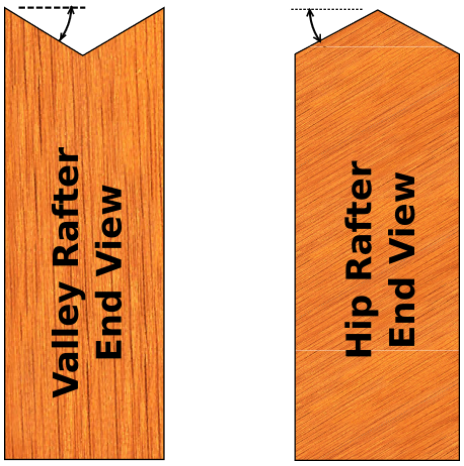


Illustration of Hip/Valley Backing Angles

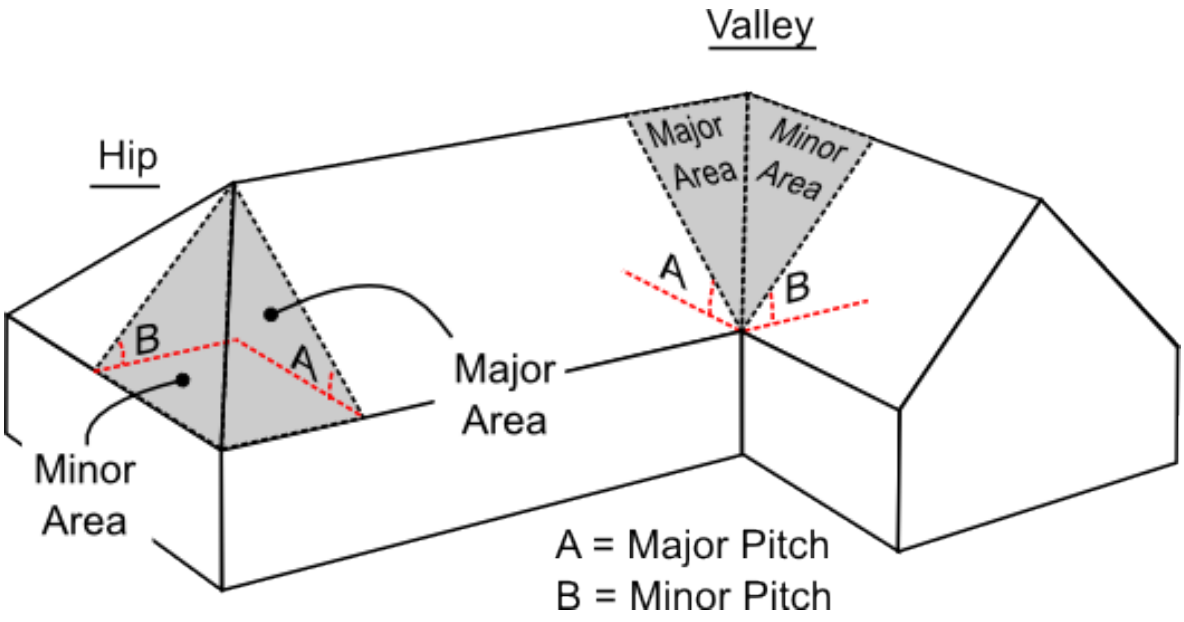


Illustration of Hip/Valley Roof Areas

Notes:

- *BuildCalc's purlin calculations are for purlins that are at right angles to rafters (common purlins, butt purlins and under purlins). For vertical purlins, instead use the associated plan angle for the miter angle and 90° for the bevel angle. For battens (or over-purlins) switch the side (Miter) and edge (Bevel) results.*
- *Quickly switch between Regular and Irregular Hip/Valley calculations using the [Regular-Irregular] switch at the top of the Hip/Valley Function view.*

[Inch] Inch Entry and Conversion

1. Set the units for a quantity to inches
2. Convert a length, area or volume to inches
3. Switch between decimal and fractional display
4. Convert a weight to cubic inches

Examples:

1. Set the units for a quantity to inches.

Input	*Display*
clear temporary memory [C] [C]	0
4 [Inch]	4in
[Inch]	$4in^2$
[Inch]	$4in^3$

2. Convert a length, area or volume to inches.

Input	*Display*
clear temporary memory [C] [C]	0
18 [Conv][cm]	18cm
[Inch]	$7-\ ^{1}/_{16}in$

3. Switch between decimal and fractional display.

Input	*Display*
clear temporary memory [C] [C]	0
33 [Inch] 7 [/] 16	33- $^{7}/_{16}$in
[Inch]	33.4375in

4. **Convert a weight to cubic inches (using the density stored in [wt/vol]).**

Input	*Display*
clear temporary memory [C] [C]	0
25 [Conv] [lbs]	25lb
[Inch]	338.8in^3

**Note: Assumes 1.5 tons per cubic yard is stored in* [wt/vol].

[IrJack] Irregular Jack Rafter Function

The Irregular Jack Rafter Layout Function provides the following layout information for hip roofs beyond the information provided by the [Hip/V] (Hip/Valley Rafters) Function – with minimum hassle for you.

- Incremental change in Irregular Jack Rafter Lengths
- Lengths of each Irregular Jack Rafter
- Irregular Jack Rafter Plumb Cut Angle
- Irregular Jack Rafter Level Cut Angle
- Irregular Jack Rafter Cheek Cut Angle

Information required for this function includes:

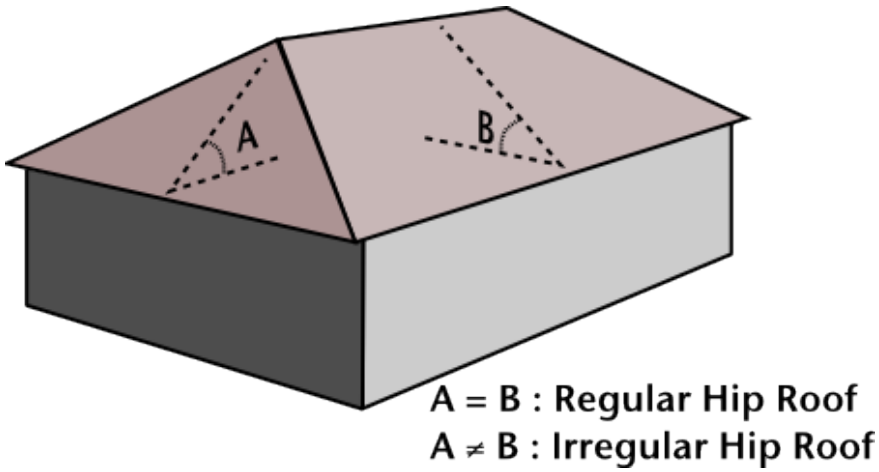
1. An Irregular (Minor) Common Rafter Pitch, entered with the [IrPitch] key.

2. Two of the following values to describe the Regular Common Rafter geometry:
    - [Pitch]
    - [Rise]
    - [Run]
    - [Diag]

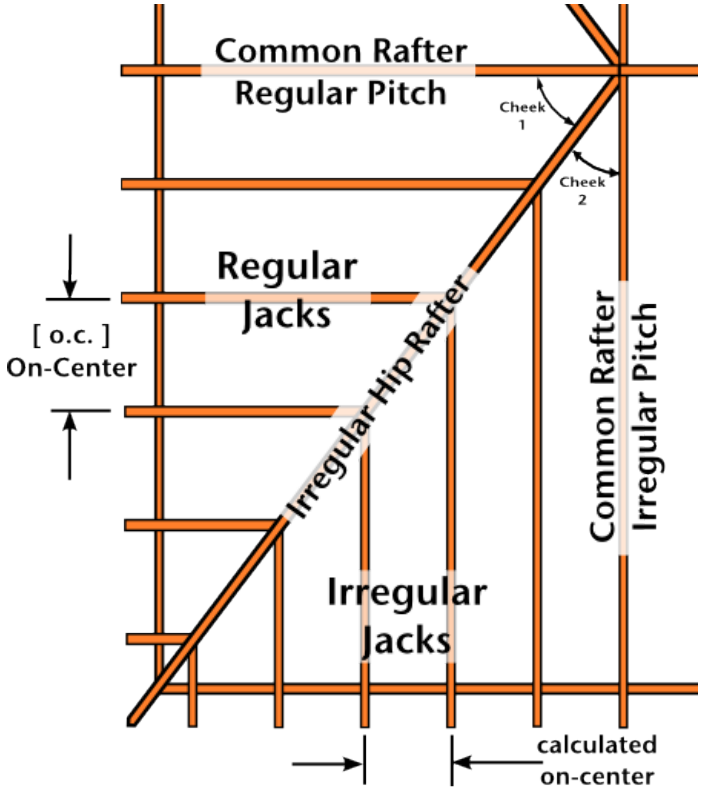
3. On-Center spacing stored in [o.c.]

4. [Prefs] *Jack Rafters*, either:
    - Largest to Smallest
    - Smallest to Largest

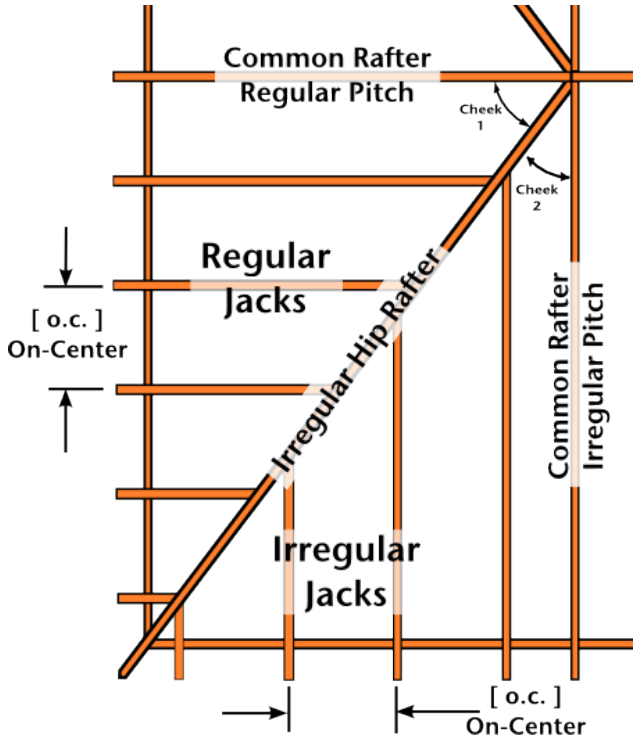
5. [Prefs] *Irregular Jack Spacing* setting, either **On-Centers** or **Mating**. See illustrations, below, for more explanation.



Also, if an Irregular Pitch is stored, then subsequent presses of the [IrJack] key will switch between Irregular and Regular Jack Rafter results. If no Irregular Pitch is stored, the IrJack key will function the same as the [Jack] key
In addition to [IrPitch], calculations for Irregular Jack Rafters are based upon if the [Prefs] Irregular Jack Spacing is set to either On-Centers or Mating. The next two illustrations help explain the difference.



Plan View of Irregular Hip/Valley layout with Rafters **mating** at the hip/valley.

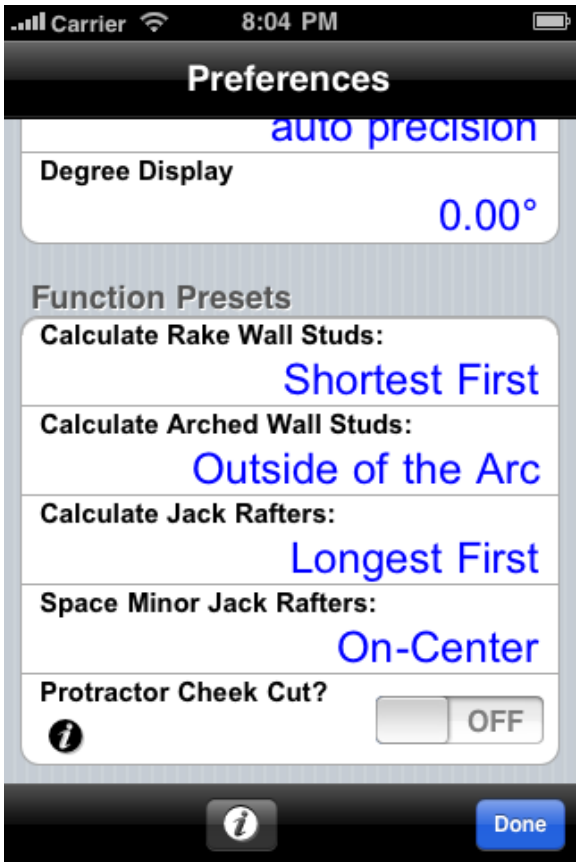


Plan View of Irregular Hip/Valley layout with rafters spaced at the **on-center** distance.

***Example of an Irregular Hip/Valley and Jack Rafter calculation*: See [Jack] for Regular Hip/Valley examples.**

Part A: Setup [Prefs] for Ascending Jack Order and Mating Jack Spacing. *Note: Don't forget to undo these changes!*

Input	*Display*
clear temporary memory [C] [C]	
[Conv] [Prefs]	*See below*

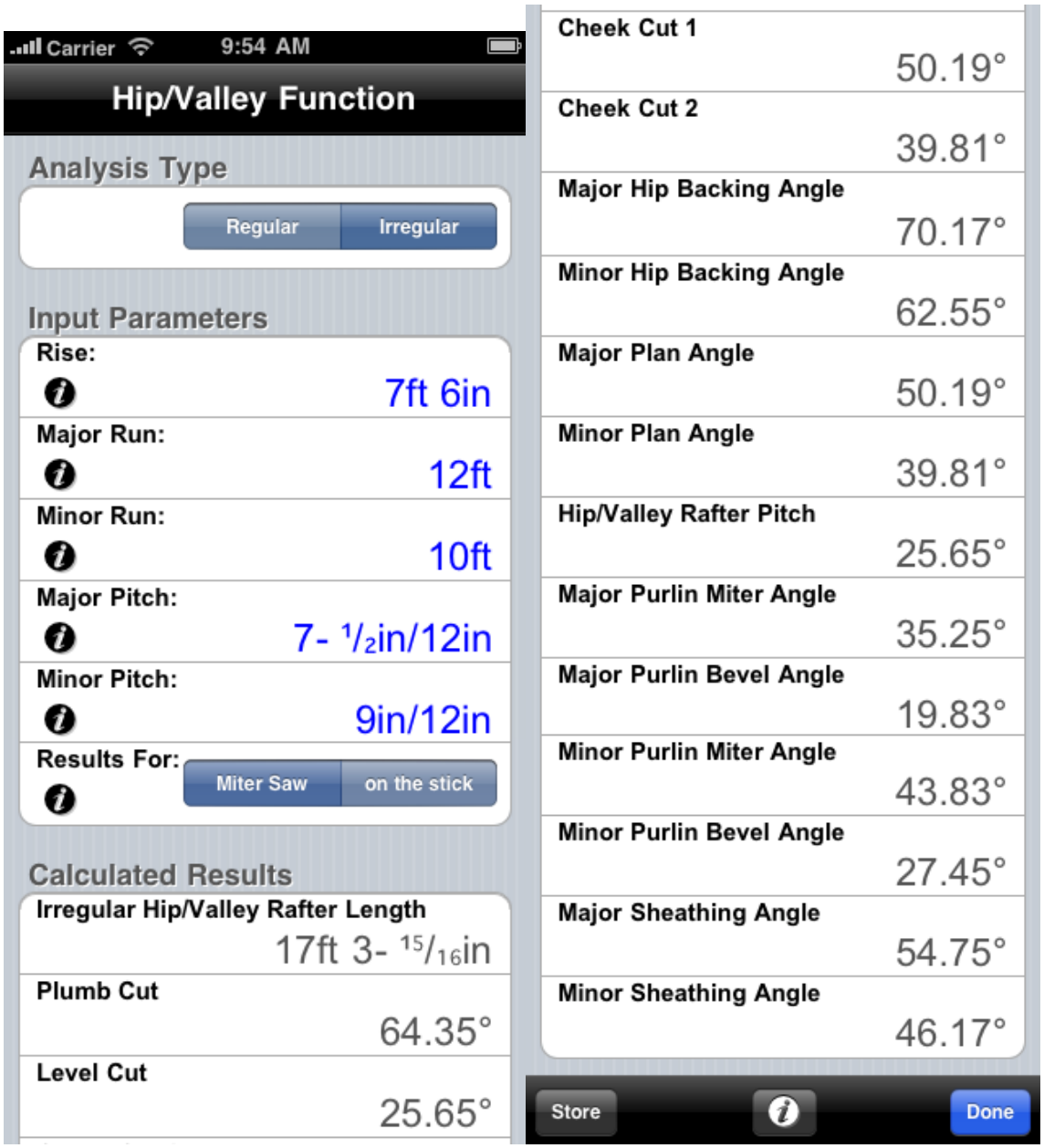


Part B: Common Rafter Layout.

Input	*Display*
clear temporary memory [C] [C]	0
40 [Inch] [Run]	Run 40in
7 [Inch] 1 [/] 2 [Pitch]	Pitch 7- $^{1}/_{2}$in
[Diag]	Diagonal 47- $^{3}/_{16}$in
[Diag]	*See results below*

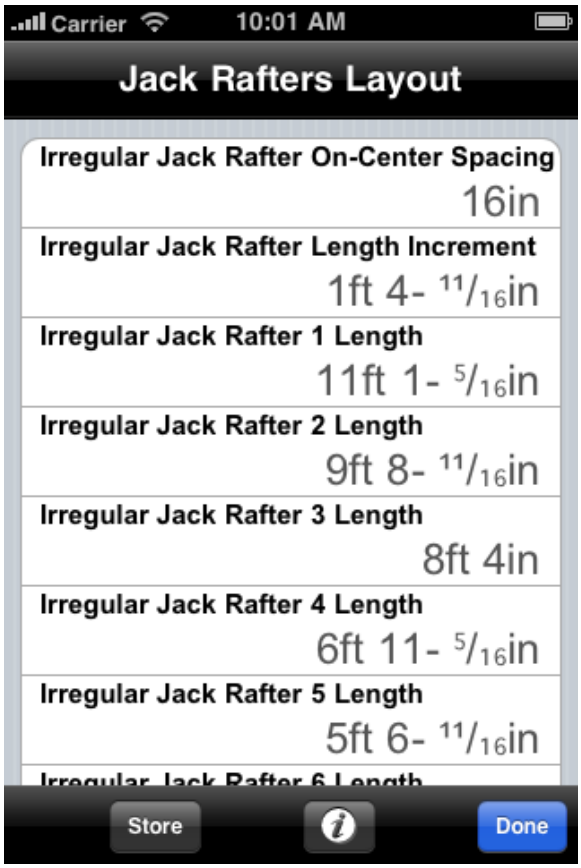
Part C: Enter Irregular Pitch and calculate the Irregular Hip/Valley rafter length and cut angles.

Input	*Display*
Note: Do not clear from above. Use stored values.	57.99°
9 [Inch] [Conv] [IrPitch]	*See results below*


Carrier
9:54 AM
Hip/Valley Function
Analysis Type
Regular
Irregular
Input Parameters
Rise:
7ft 6in
Major Run:
12ft
Minor Run:
10ft
Major Pitch:
7- 1/2in/12in
Minor Pitch:
9in/12in
Results For:
Miter Saw
on the stick
Calculated Results
Irregular Hip/Valley Rafter Length
17ft 3- 15/16in
Plumb Cut
64.35°
Level Cut
25.65°
Cheek Cut 1
50.19°
Cheek Cut 2
39.81°
Major Hip Backing Angle
70.17°
Minor Hip Backing Angle
62.55°
Major Plan Angle
50.19°
Minor Plan Angle
39.81°
Hip/Valley Rafter Pitch
25.65°
Major Purlin Miter Angle
35.25°
Major Purlin Bevel Angle
19.83°
Minor Purlin Miter Angle
43.83°
Minor Purlin Bevel Angle
27.45°
Major Sheathing Angle
54.75°
Minor Sheathing Angle
46.17°
Store
Done

Part D: Display the Irregular Jack Rafter Layout*.

Input	*Display*
Note: Do not clear from above. Use stored values.	
[Conv] [IrJack]	*See results below*



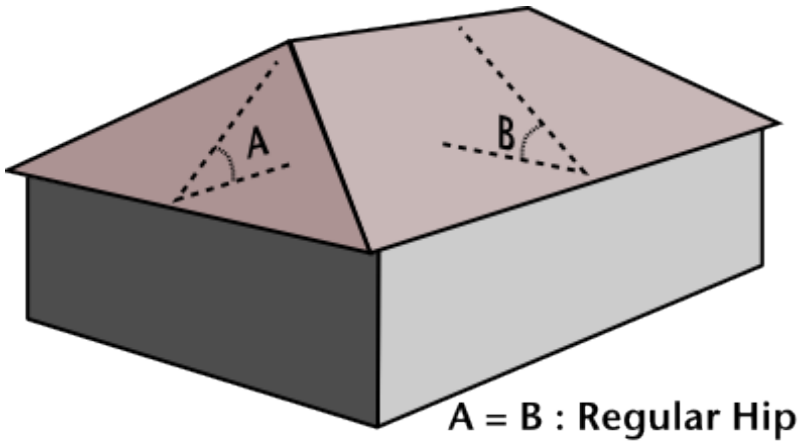
Scroll down to see all of the results. Where are the regular jack rafter layout dimensions? Tap [Done] and then tap [Jack] to see them. Repeated taps of [Jack] switch between Irregular and Regular Jack Rafter Layout display.

[IrPitch] Irregular Pitch Function

The Irregular Pitch Function is for the storage and recall of a Minor (Irregular) Roof Pitch. Its value is used by the [Hip/V] and [Jack] Functions.

Note that, starting with BuildCalc 2.0, Minor (Irregular) Pitch can be directly entered into the [Hip/V] function.

See [Hip/V] for more instruction.



A = B : Regular Hip Roof
A ≠ B : Irregular Hip Roof

[Jack] Jack Rafter Function

The Enhanced Jack Rafter Layout Function provides the following layout information for hip roofs beyond the information provided by the [Hip/V] (Hip / Valley Rafters) Function - with the minimum amount of hassle for you.

- Incremental change in Jack Rafter Lengths
- Lengths of each Jack Rafter
- Plumb Cut Angle
- Level Cut Angle
- Cheek Cut Angle

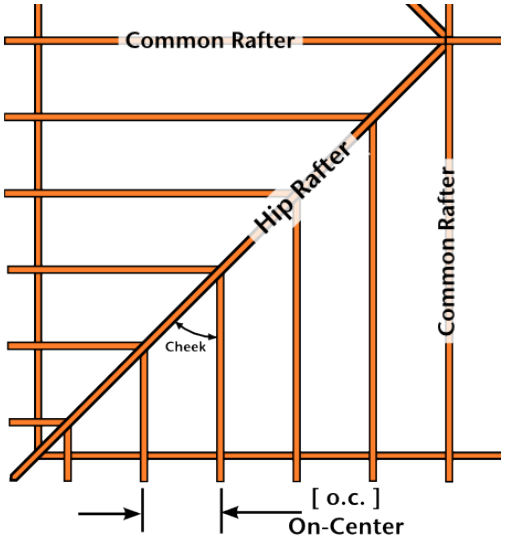
Information required for this function includes:

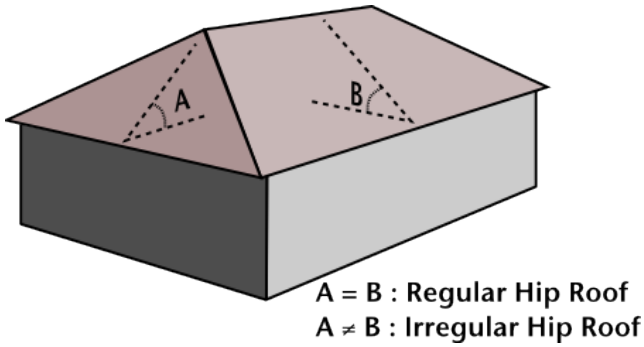
1. Two of the following values to describe the Regular Common Rafter geometry:
    - [Pitch]
    - [Rise]
    - [Run]
    - [Diag]

2. On-Center spacing stored in [o.c.]

3. [Prefs] *Jack Rafters*, either:

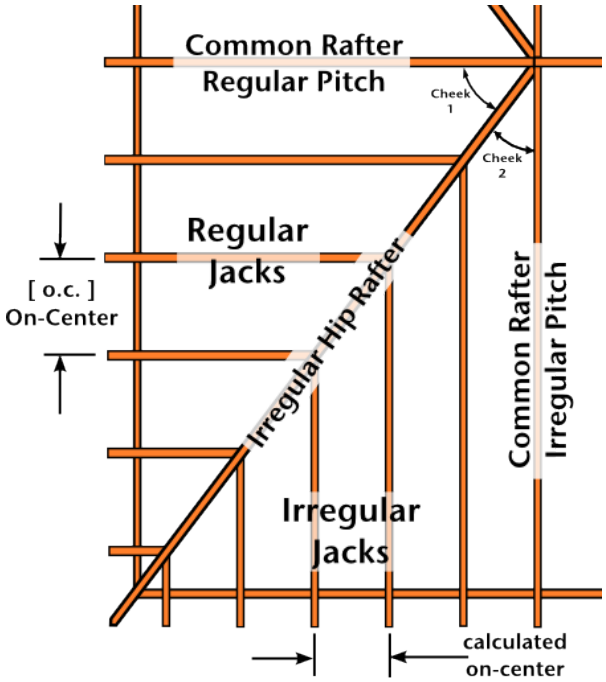
    - Largest to Smallest
    - Smallest to Largest



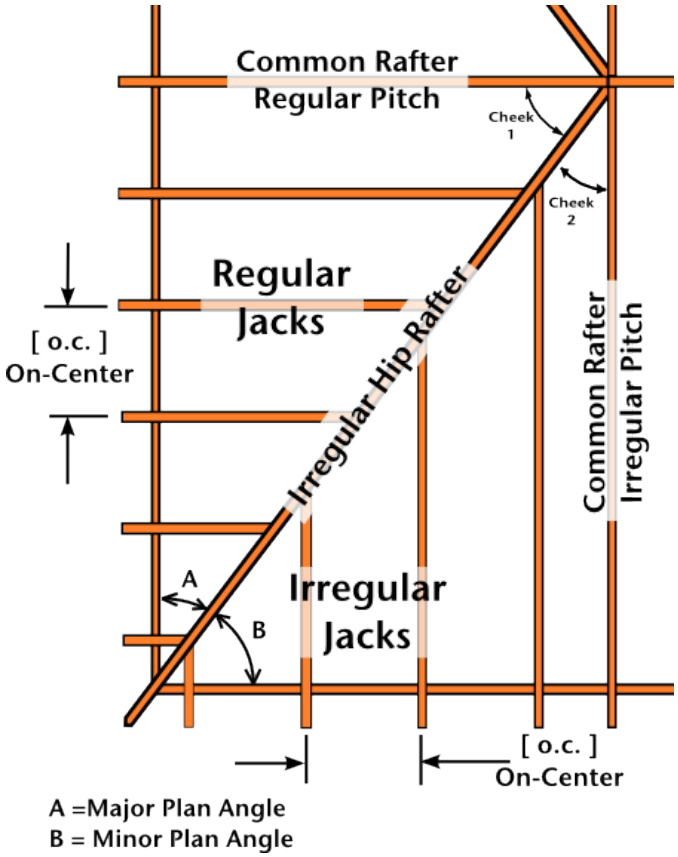


Also, if an Irregular Pitch is stored in [IrPitch] then more presses of the [Jack] key will switch between Minor (Irregular) and Common (Regular) Jack Rafter results.

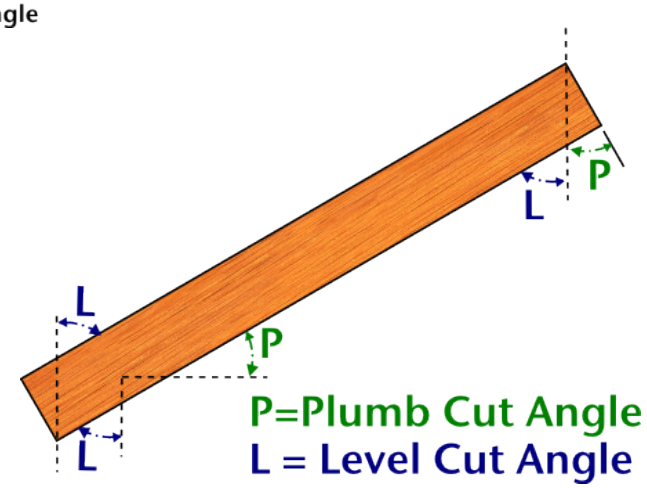
In addition to [IrPitch], calculations for Irregular Jack Rafters are based upon if the [Prefs] *Irregular Jack Spacing* is set to either **On-Centers** or **Mating**. See illustrations, below for more explanation.



Irregular Hip/Valley layout with Rafters **mating** at the hip/valley.



Irregular Hip/Valley layout with rafters spaced at the **on-center** distance.



***Example of a Regular Hip/Valley and Jack Rafter calculation*: (see [IrJack] for an Irregular Hip/Valley example.)**

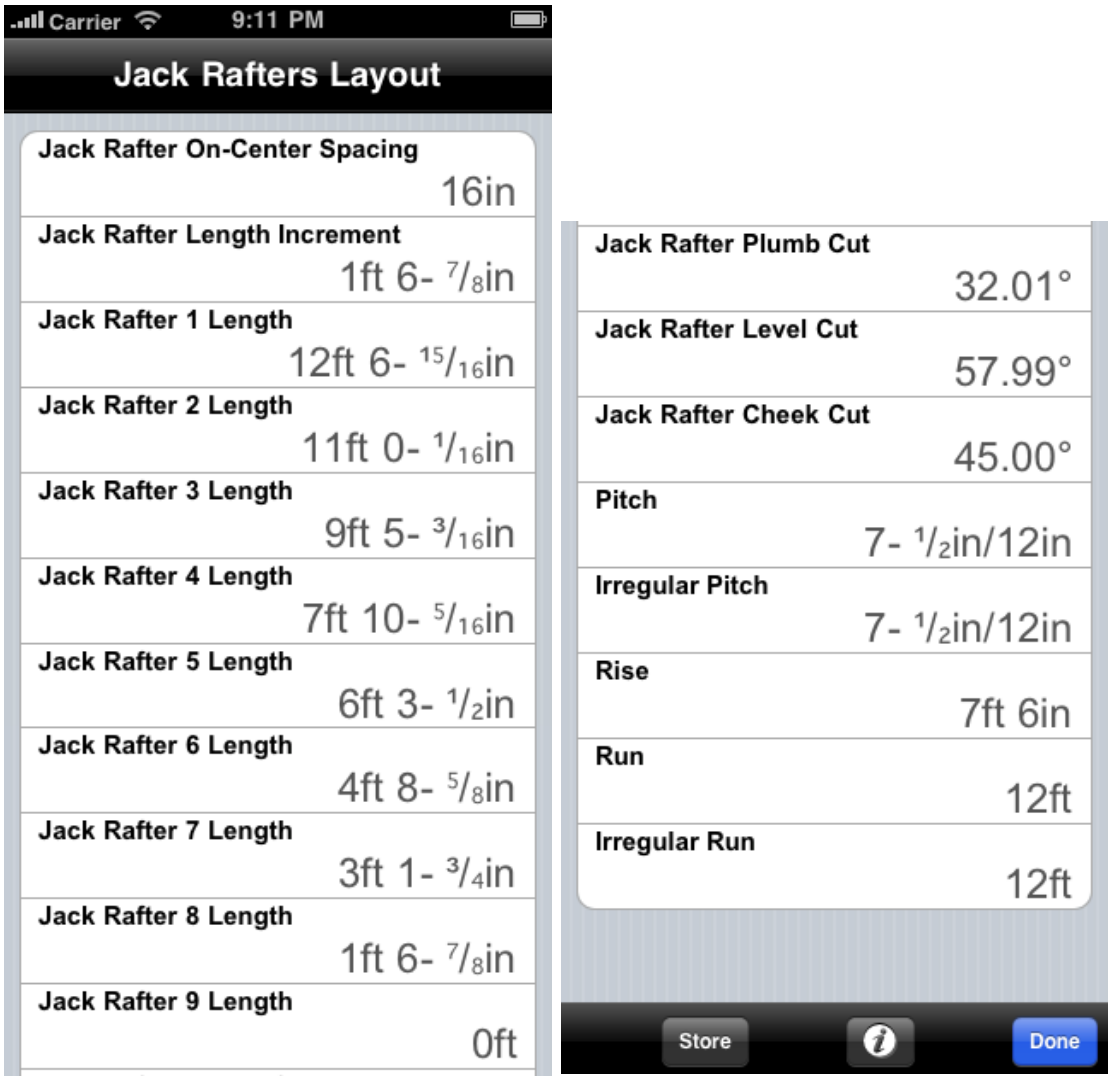
Input	*Display*
clear temporary memory 	0
12 [Feet] [Run]	Run 12ft
7 [Inch] 1 [/] 2 [Pitch]	Pitch 7- $^{1}/_{2}$in
[Diag]	Diagonal 14ft 1- $^{13}/_{16}$in
[Diag]	*See results below*

Part B: Hip/Valley rafter length and cut angles.

Input	*Display*
Note: Do not clear from above. Use stored values	
[Hip/V]	*See results below*

Part C: Jack Rafter Layout.

Input	*Display*
Note: Do not clear from above. Use stored values	
[Jack]	*See results below*



[kg] Kilogram Function

1. Set the units for a quantity to kilograms
2. Convert a weight to kilograms
3. Convert a volume to kilograms (using the density stored in [wt/vol])

Examples:

1. Set the units for a quantity to kilograms.

Input	*Display*
clear temporary memory 	0
3 [Conv] [kg]	3kg

2. Convert a weight to kilograms.

Input	*Display*
clear temporary memory 	0
56 [Conv] [lbs]	56lb
[Conv] [kg]	25.40117kg

3. Convert a volume to kilograms (using the density stored in [wt/vol]).

Input	*Display*
clear temporary memory [C] [C]	0
11 [Yard] [Yard] [Yard]	$11yd^3$
[Conv] [kg]	14968.55kg

**Note: Assumes 1.5 tons per cubic yard is stored in* [wt/vol].

[lbs] Pounds Function

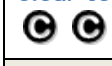
1. Set the units for a quantity to pounds
2. Convert an weight to pounds
3. Convert a volume to pounds (using the density stored in [wt/vol])

Examples:

1. Sets the units for a quantity to pounds.

Input	*Display*
clear temporary memory 	0
3.7 [Conv] [lbs]	3.7lb

2. Convert a weight to pounds.

Input	*Display*
clear temporary memory 	0
12 [Conv] [kg]	12kg
[Conv] [lbs]	26.45547lb

3. Convert a volume to pounds (using the density stored in [wt/vol]).

Input	*Display*
clear temporary memory [C] [C]	0
3.5 [Feet] [x] 4 [Feet] [x] 4 [Inch] [=]	0.17283yd^3
[Conv] [lbs]	518.5185lb

*__Note__: *Assumes 1.5 tons per cubic yard is stored in* [wt/vol].

[Length] Length Function

1. Enter a length value to be used in other functions*
2. Recall the stored length value

Examples:

1. Enter a length value to be used in other functions*.

Input	*Display*
clear temporary memory [C] [C]	0
13 [Feet] 6 [Inch] 3 [/] 8 [Length]	Length 13ft 6- $^{3}/_{8}$in

2. Recall the stored length value.

Input	*Display*
clear temporary memory [C] [C]	0
[Recall] [Length]	Length 13ft 6- $^{3}/_{8}$in

***Length is used in the following functions:**

Function	*Calculates*
[Width]	Area, square-up and perimeter
[Height]	Volume, wall area, and room area
[Msnry]	Number pieces of masonry, of size [MsnSz], that can fill a length or area.
[Footng]	Volume of a footing
[Drywall]	Sheets of 4’x8’, 4’x9’, 4'x10', and 4’x12’ drywall.
[Roof]	Area, Squares, and 4’x8’ sheets to cover a roof
[qty@oc]	The number of Vertical Members, spaced at [o.c.], that will span a length.

[m tons] Metric Tons Function

1. Set the units for a quantity to metric tons.
2. Convert a weight to metric tons.
3. Convert a volume to metric tons (using the density stored in [wt/vol])

Examples:

1. Set the units for a quantity to metric tons.

Input	*Display*
clear temporary memory 	0
3 [Conv] [m tons]	3MT

2. Convert a weight to metric tons.

Input	*Display*
clear temporary memory 	0
12 [Conv] [ton]	12T
[Conv] [m ton]	10.88622MT

3. Converts a volume to metric tons (using the density stored in [wt/vol])

Input	*Display*
clear temporary memory C C	0
11 [Yard] [Yard] [Yard]	$11yd^3$
[Conv] [m tons]	14.96855MT

*__Note__: *Assumes 1.5 tons per cubic yard is stored in* [wt/vol].

[M-] Negative Cumulative Memory Function

- [M+] Adds the current value to the cumulative memory value.
- [M-] Subtracts the current value from the cumulative memory.
- [Recall] [M+] Recalls cumulative memory value, average and count.
- [M-R/C] or [Recall] [Recall] clears the cumulative memory.

Example:

1. **First, sum three numbers (5, 10, 15) and get the cumulative statistics on those numbers.**
2. **Next, add a negative number (-8) to this cumulative sum and then get the cumulative statistics on this new list of numbers.**

Input	*Display*
clear all memory [Conv] [ClrAll]	0
5[M+] 10[M+] 15[M+]	M+ 15
[Recall] [M+]	M+Total 30
[M+]	M+Avg 10
[M+]	M+Count 3
[clr] [clr]	0
[M-]	M+ 8
[Recall] [M+]	M+Total 22
[M+]	M+Avg 5.5
[M+]	M+Avg 4
clear cumulative memory [Conv] [M-R/C]	4

[M-R/C] Clear Cumulative Memory

- [M+] Adds the current value to the cumulative memory value
- [M-] Subtracts the current value from the cumulative memory
- [Recall] [M+] Recalls cumulative memory value, average and count
- [M-R/C] or [Recall] [Recall] clears the cumulative memory

Example:

1. **First, sum three numbers (5, 10, 15) and get the cumulative statistics on those numbers.**
2. **Next, add a negative number (-8) to this cumulative sum and then get the cumulative statistics on this new list of numbers.**

graInput	*Display*
clear all memory [Conv] [ClrAll]	0
5[M+] 10[M+] 15[M+]	M+ 15
[Recall] [M+]	M+Total 30
[M+]	M+Avg 10
[M+]	M+Count 3
[clr] [clr]	0
[M-]	M+ 8
[Recall] [M+]	M+Total 22
[M+]	M+Avg 5.5
[M+]	M+Avg 4
clear cumulative memory [Conv] [M-R/C]	4

[m] Meter Function

1. Set the units for a quantity to meters
2. Convert a length, area or volume
3. Convert a weight to cubic meters (using the density stored in [wt/vol])

Example:

1. Set the units for a quantity to meters

Input	*Display*
clear temporary memory 	0
3 [m]	3m
[m]	$3m^2$
[m]	$3m^3$

2. Convert a length, area or volume to meters

Input	*Display*
clear temporary memory 	0
[m]	56in
[m]	1.4224m
[m]	142.24cm
[m]	1422.4mm
[m]	1.4224m

3. Convert a weight to cubic meters (using the density stored in [wt/vol]).

*Input**	*Display*
clear temporary memory	0
1.25 [Conv] [m tons]	1.25MT
[m]	0.702315m^3

**Note: Assumes 1.5 tons per cubic yard is stored in* [wt/vol].

[M+] Add to Cumulative Memory

- [M+] Adds the current value to the cumulative memory value
- M-] Subtracts the current value from the cumulative memory
- [Recall] [M+] Recalls cumulative memory value, average and count
- [M-R/C] or [Recall] [Recall] clears the cumulative memory

Example:

1. First, sum three numbers (5, 10, 15) and get the cumulative statistics on those numbers.

2. Next, add a negative number (-8) to this cumulative sum and then get the cumulative statistics on this new list of numbers.

Input	*Display*
clear all memory [Conv] [ClrAll]	0
5[M+] 10[M+] 15[M+]	M+ 15
[Recall] [M+]	M+Total 30
[M+]	M+Avg 10
[M+]	M+Count 3
[clr] [clr]	0
[M-]	M+ 8
[Recall] [M+]	M+Total 22
[M+]	M+Avg 5.5
[M+]	M+Avg 4
clear cumulative memory [Conv] [M-R/C]	4

[M1] Permanent Memory Function

1. Stores a value to Permanent Memory
2. Recalls a value from Permanent Memory

Examples:

1. Stores a value to Permanent Memory.

Input	*Display*
clear temporary memory 	0
11 [Yard] [Yard] [Yard]	$11yd^3$
[Store] [M1]	$11yd^3$

2. Recalls a value from Permanent Memory.

Input	*Display*
clear temporary memory 	0
Recall] [M1]	$11yd^3$

[M2] Permanent Memory Function

1. Stores a value to Permanent Memory
2. Recalls a value from Permanent Memory

E***xamples:***

1. Stores a value to Permanent Memory.

Input	*Display*
clear temporary memory [C] [C]	0
11 [Yard] [Yard] [Yard]	$11yd^3$
[Store] [M2]	$11yd^3$

2. Recalls a value from Permanent Memory.

Input	*Display*
clear temporary memory [C] [C]	0
[Recall] [M2]	[Recall] [M2]

[M3] Permanent Memory Function

1. Stores a value to Permanent Memory
2. Recalls a value from Permanent Memory

Examples:

1. Stores a value to Permanent Memory.

Input	*Display*
clear temporary memory 	0
11 [Yard] [Yard] [Yard]	$11yd^3$
[Store] [M3]	$11yd^3$

2. Recalls a value from Permanent Memory.

Input	*Display*
clear temporary memory 	0
[Recall] [M3]	$11yd^3$

[mm] Millimeter Function

1. Set the units for a quantity to millimeters
2. Convert a length, area or volume to millimeters.
3. Convert a weight to cubic millimeters (using the density stored in [wt/vol])

Example:

1. Set the units for a quantity to millimeters.

Input	*Display*
clear temporary memory © ©	0
3 [Conv] [mm]	3mm
[Conv] [mm]	$3mm^2$
[Conv] [mm]	$3mm^3$

2. Convert a length, area or volume to millimeters.

Input	*Display*
clear temporary memory © ©	0
56 [Inch]	56in
[Conv] [mm]	1422.4mm

3. Convert a weight to cubic millimeters (using the density stored in [wt/vol]).

Input	*Display*
clear temporary memory 	0
0.125 [Conv] [lbs]	0.0125lb
[Conv] [mm]	3185.645mm^3

**Note: Assumes 1.5 tons per cubic yard is stored in* [wt/vol].

[Msnry] Masonry Function

The Enhanced Masonry Function calculates the pieces of masonry (Bricks, Blocks, Tile, Pavers, etc.) based on the below input scenarios. Also note that pieces can also be anything for which you know the area of one "piece", such as a plank, a shingle, a piece of sheathing, etc. The Masonry function has been enhanced to allow you to view multiple piece sizes and edit the list of piece sizes to suit your needs. See the first example below to learn how this is done.

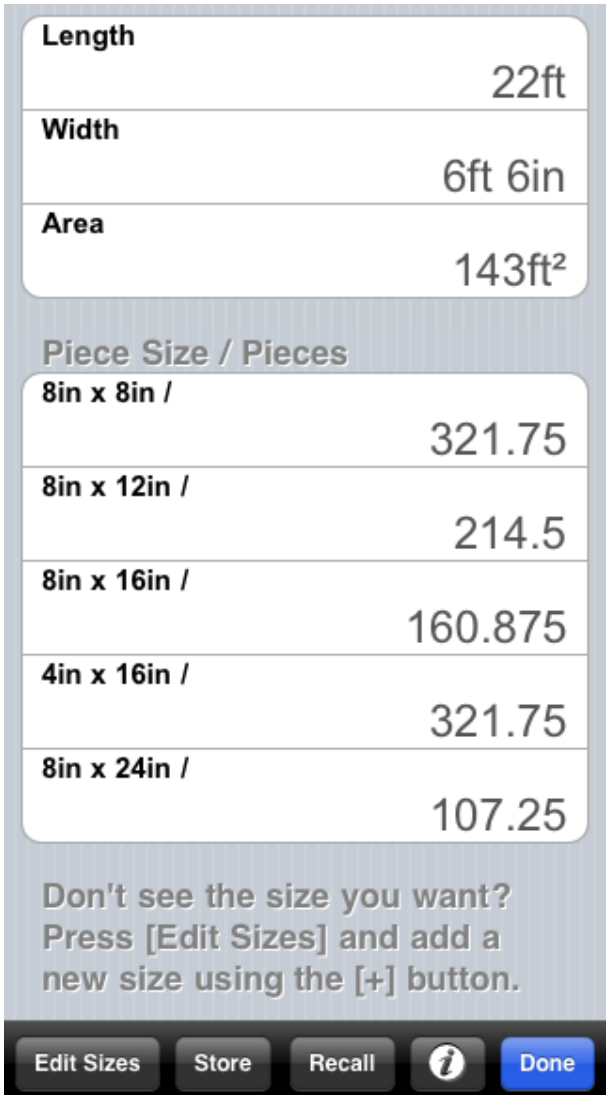
The Enhanced Masonry Function uses one of the following inputs to calculate an estimate how how many pieces of masonry you may need:

1. Stored [Length] and [Height].
2. Stored [Length] and [Width].
3. Input Area.
4. Stored [Length] only.
5. Input Length.
6. Room Mode: Stored [Length], [Width] and [Height].

Examples:

1. Stored Length and Height. How many 6"x6" tile is required for a hallway 22' by 6' 6"? Note that Height could have been used in this example with identical results.

Input	*Display*
clear temporary memory	0
22 [Feet] [Length]	Length 22ft
6 [Feet] 6 [Inch] [Height]	Height 6ft 6in
[Conv] [Msnry]	*See results, below*



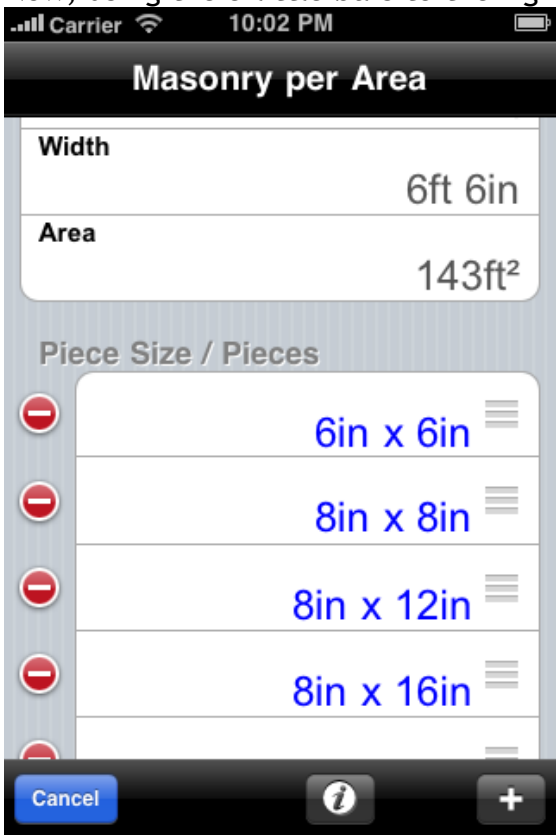
But wait a minute! Where is 6"x6"!?!? Now is your chance to customize the Masonry function for you. Create a 6"x6" tile size using the following steps:

- Tap the [Edit Sizes] button to enter edit mode.

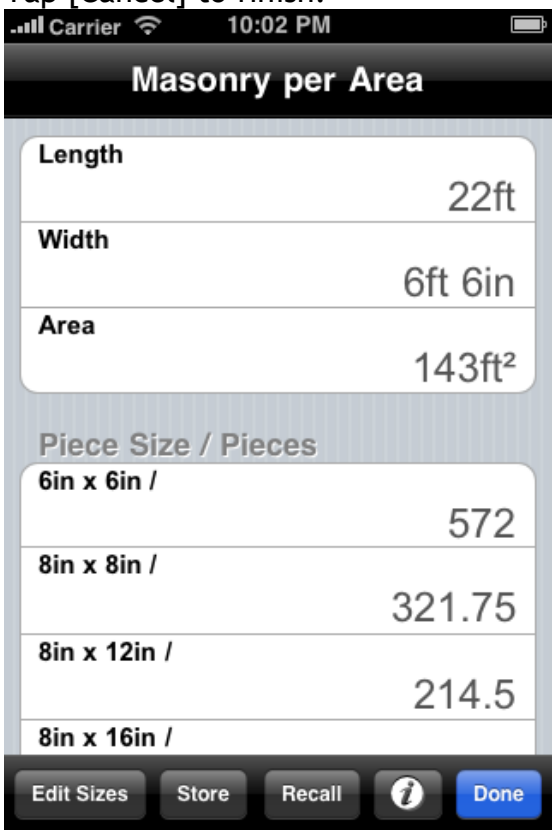
- Tap the [+] button to add a new size

- Tap in `6[Inches]x6[Inches][done]` to enter your new size.

- Now, using the 3 little bars to the right of '6"x6"', drag it to the top

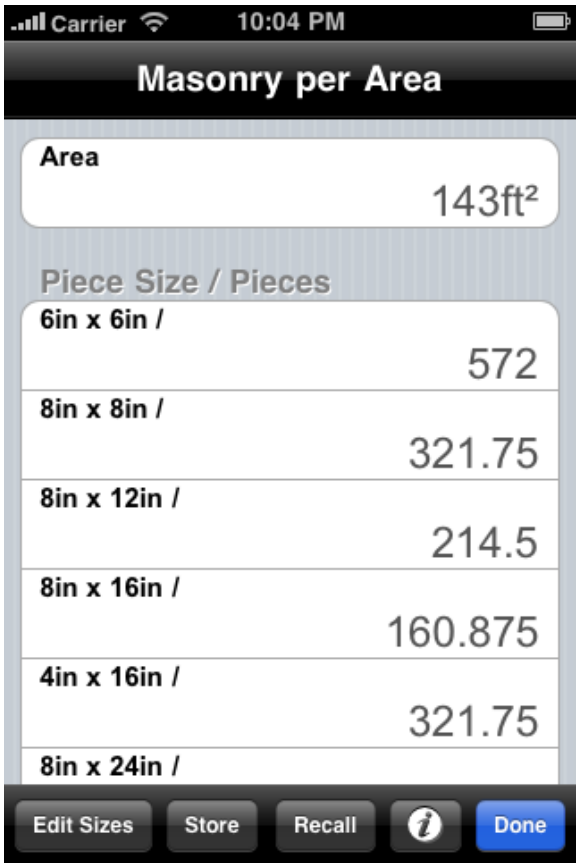


- Tap [Cancel] to finish.



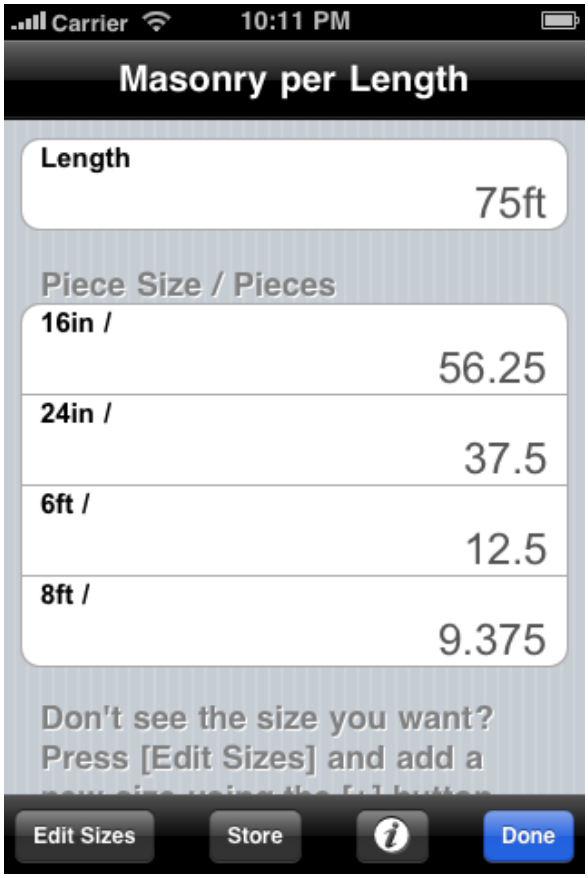
2. Input Area. How much 6”x6” tile for a 22’x6’6” area?

Input	*Display*
clear temporary memory	0
22 [Feet] [x]	22ft x
6 [Feet] 6 [Inch] =	143ft^2
[Conv] [Msnry]	*See results, below*



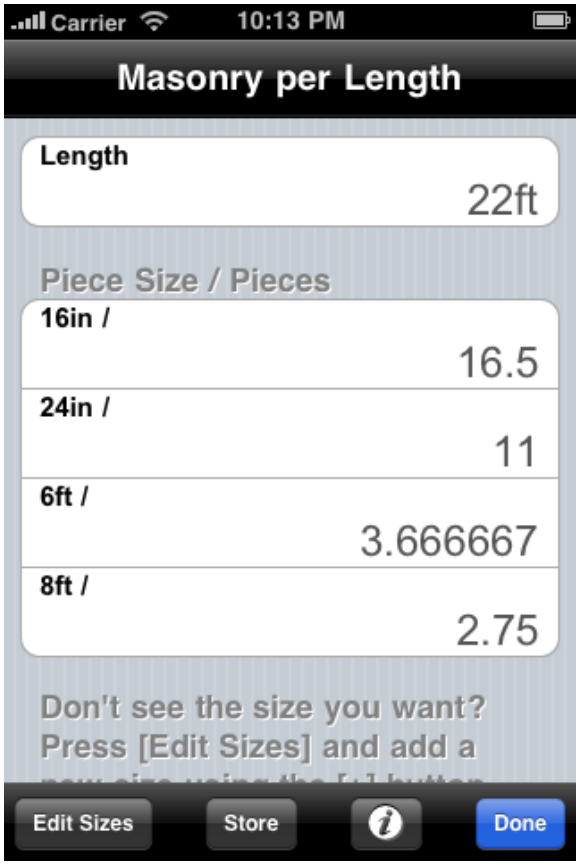
3. Stored Length only. Let's say you have a 75' border you wish to fill wit 16" culture stone. How many pieces will you need?

Input	*Display*
clear temporary memory 	0
22 [Feet] [Length]	Length 22ft
	0
[Conv] [Msnry]	*See results, below*



4. Input Length.

Input	*Display*
clear temporary memory 	0
22 [Feet]	22ft
[Conv] [Msnry]	*See results, below*



[o.c.] On Center Memory

1. Enter an on-center value to be used in other functions*
2. Recall the stored on-center value

Example:

1. Enter an on-center value to be used in other functions*.

Input	*Display*
clear temporary memory 	0
16 [Store] [o.c.]	On-Center Spacing 16in

2. Recall the stored on-center value.

Input	*Display*
clear temporary memory 	0
[Recall] [o.c.]	On-Center Spacing 16in

**Note*: *On-Center is used by the following functions:*

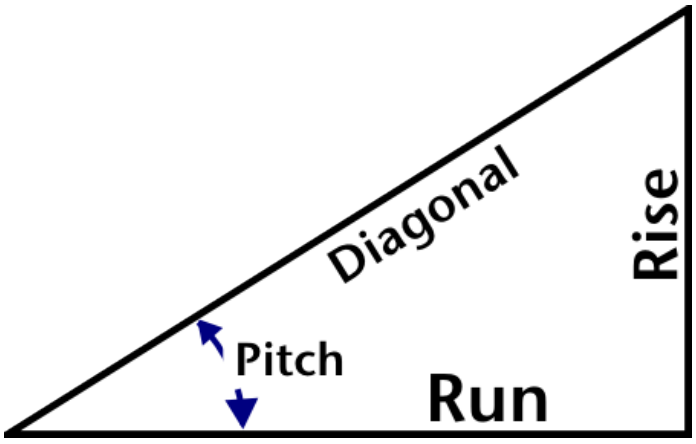
Function	*Calculates*
[qty@oc]**	Number of Vertical Members (studs, posts, poles , columns, or pilings) for a given length
[R/Wall]	Calculates Rake Wall stud lengths
[Jack] and [Ir/Jack]	Regular and Irregular Jack Rafter Lengths

***Note*: *When [Conv] [Prefs] setting "Advanced Function Mode" is set to ON (the default setting for BuildCalc version 2.1), the on-center spacing for the [qty@oc] function is entered using the [qty@oc] function.*

[Pitch] Pitch Function

The pitch key can either:

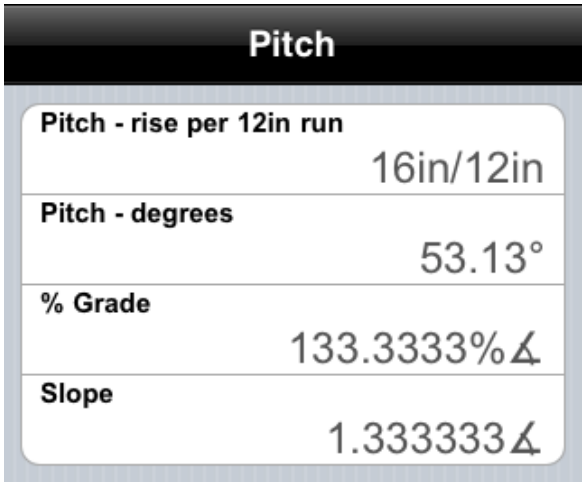
- Calculate a pitch from any two of the following values stored: [Rise], [Run], [Diag]
- Store a pitch value for other calculations, entered in one of the following ways:
  - Unit-less quantity: interpreted as degrees
  - An entered quantity, followed by a [%] key press is interpreted as % Grade
  - A length (ex: inches, cm, etc.) is interpreted as Pitch, where Slope = Length / 12 inches.
- Recall a Pitch value for display



Examples:

1. Calculate Pitch from a given [Rise] and [Run].

*Input**	*Display*
clear temporary memory 	0
8 [Feet] [Rise]	Rise 8ft
6 [Feet] [Run]	Run 6ft
[Pitch]	Pitch 16in
[Pitch]	*See results, below*



2. Enter and Recall a Pitch.

*Input**	*Display*
clear temporary memory © ©	0
8 [Inch] [Pitch]	Pitch 8in
© ©	0
[Pitch]	Pitch 8in

Note: *Calculated pitch values, as in example 1, above, will be cleared from memory upon* © © *or* [ClrAll]. *Entered Pitch values, as in example 2, will be retained until a new value is entered. When a calculated pitch is cleared, the last entered pitch value is put into [Pitch].*

[Polygn] Polygon Function

Given the following inputs:

- Diameter ([Circle]) or [Radius]
- Number of Sides

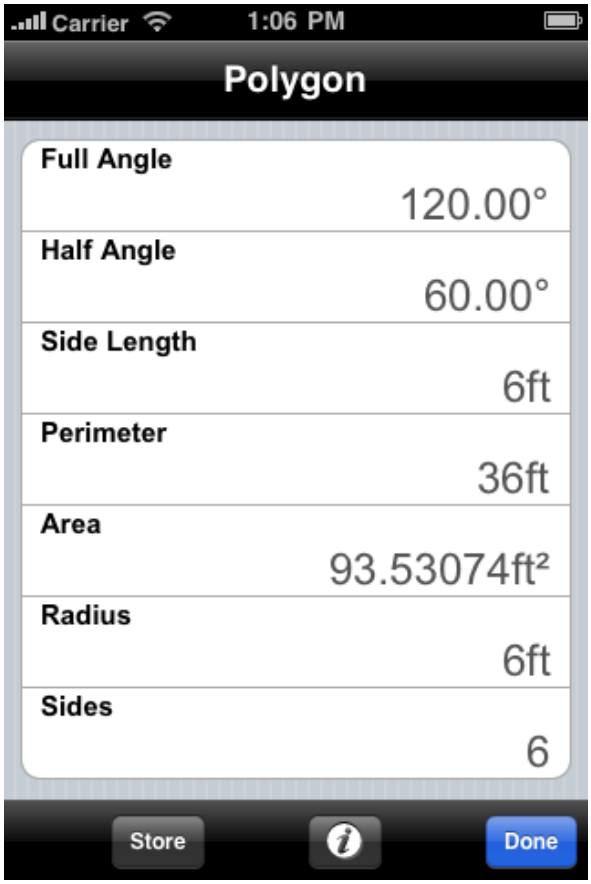
the Polygon function calculates or displays the following descriptive geometry for equilateral polygons:

- Full Corner Angle
- Bisected (Half) Corner Angle
- Length of a side
- Perimeter (sum of the sides)
- Area
- Radius (distance from the center to a corner)

Example:

Calculate the Area of a 6-sided Gazebo floor that is 12ft, corner to corner.

*Input**	*Display*
clear temporary memory 	0
12 [Feet] [Circ]	Diameter 12ft
6 [Conv] [Polygn]	*See below for results*


Carrier
1:06 PM
Polygon
Full Angle
120.00°
Half Angle
60.00°
Side Length
6ft
Perimeter
36ft
Area
93.53074ft²
Radius
6ft
Sides
6
Store
Done

[Prefs] Preferences Storage Function

The comprehensive preference settings for BuildCalc.

Preference Settingss:

Setting	*Description*
Reset	Resets BuildCalc's memory to the factory settings.
Keyboard Clicks	Turns keyboard clicks on/off.
Trig Keys	Switches between the display of the [Length] [Width] [Height] and the [SIN] [COS] [TAN] keys.
Metric Linear Keys	Switches between the display of the [Yards] [Feet] [Inches] and the [m] [cm] [mm] keys.
Advanced Function Mode	Switches between BuildCalc's advanced functionality and the level of functionality available prior to version 2.0.
Fractional Resolution	Sets the resolution for fractional values. Default: $^{1}/_{16}$**in**
Fractional Mode	Switches between displaying fractions to the nearest fraction (**Standard**) or displaying fractions always at the fractional resolution setting (**Constant**). Default: **Standard**
Area Display Format	Determines the units in which area results are displayed. Default: **Standard**
Volume Display Format	Determines the units in which volume results are displayed. Default: **Standard**
Weight Display Format	Determines the units in which weight results are displayed. Default: **Standard**
Pounds Per Ton	Determines how many pounds are assumed to be in an imperial ton. Defaults: 2,000 for North America 2,240 for everywhere else

Setting	*Description*
Thousands Separator	Turns the thousands separator ON/OFF Default: OFF
Quantity Display	Determines the precision of quantity (dimensionless) numbers. Default: **Auto precision**
Meter Linear Display	Changes the number of decimal places meter values are displayed. Default: **standard**
Millimeter Length Display	Changes the number of decimal places millimeter values are displayed. Default: **Auto precision**
Decimal Degree Display	Changes the number of decimal places decimal degree values are displayed. Default: **standard**
Stairwell Headroom Height*	Used by the [Stair] function to calculate staircase opening size. Default: **6ft 8in**. *Note that this setting is not available when "Advanced Function Mode" is set to ON.*
Calculate Rake Wall Studs	Used by the [R/Wall] key to determine which stud length is calculated first, the **Shortest** or the **Longest**. Which ever stud is calculated first will be exactly [o.c.] distance from the end of the wall. Default: **Longest**
Calculate Arched Wall	Used by the [Arc] function to determine if studs are positioned inside of the arch or outside of the arch. Default: **Outside**
Calculate Jack Rafters	Used by the [Jack] key to determine which rafter length is calculated first, the **Shortest** or the **Longest**. Which ever rafter is calculated first will be exactly [o.c.] distance from the end of the hip/valley rafter. Default: **Longest**
Space Minor Jack Rafters	Used to determine if Jack Rafters for Irregular Hip Roofs are positioned with the **On-Center** spacing maintained on both sides, or such that the Jack Rafters **Mate** at the Hip/Valley Rafter. Default: **On-Center**
Protractor Cheek Cut?	By default, Cheek Cut Angles are calculated for setting the bevel of a saw (circular or miter). Switch to on if, in "Results For:

Setting	*Description*
	Protractor" mode, you wish for the Cheek Cut Angles to be for a hand held protractor (on the edge of the board).

[qty@oc] Quantity spaced at On-Center distance Function

For estimating vertical members, such as framing, timbers, posts, poles, columns, pilings, etc.

- Calculates the quantity of vertical members along a length from:
  - User input
  - the value stored in the [Length] key
- Calculates the length that a given number of vertical members will span.

You can specify multiple on-center distances for calculation - which saves you from having to switch between all the different on-center values you use in your day-to-day calculations. See below for a complete example.

Example:

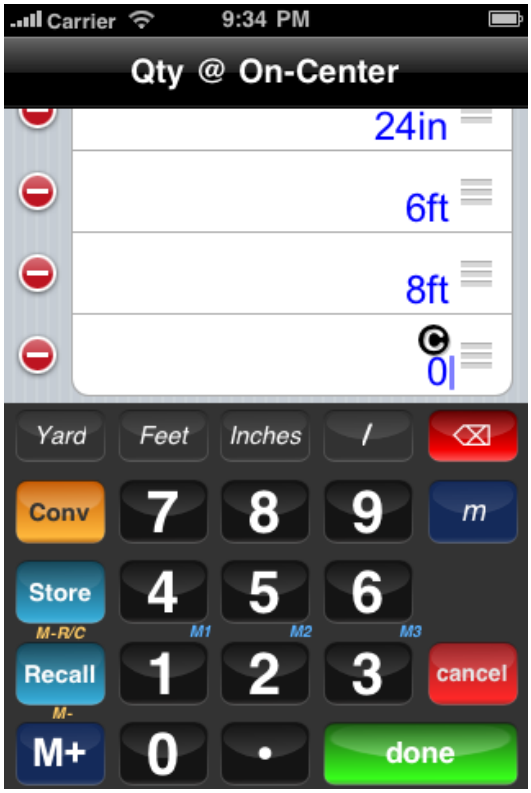
How many joists are required for a floor that spans 22' 8-3/4"?

Input	*Display*
clear temporary memory [C] [C]	0
Enter a fence length 22 [Feet] 8 [Inch] 3 [/] 4	22ft 8- 3/4in
[Conv] [qty@oc]	*See results, below*

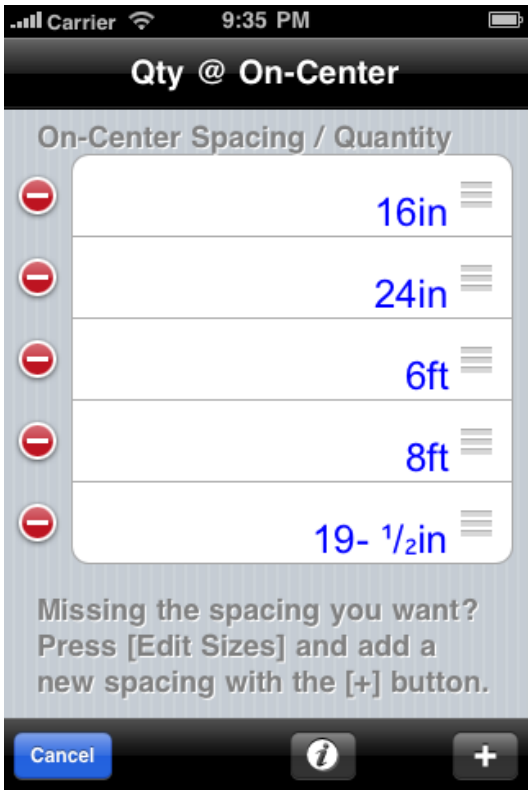
So if you were to space the joists at 16" centers, you would need 19. And for 24" centers, you would need 13. But what if you wanted to space these joists at 19- 1/2" centers? To add a new on-center spacing, try the following:

- Tap the [Edit Sizes] button to enter edit mode.

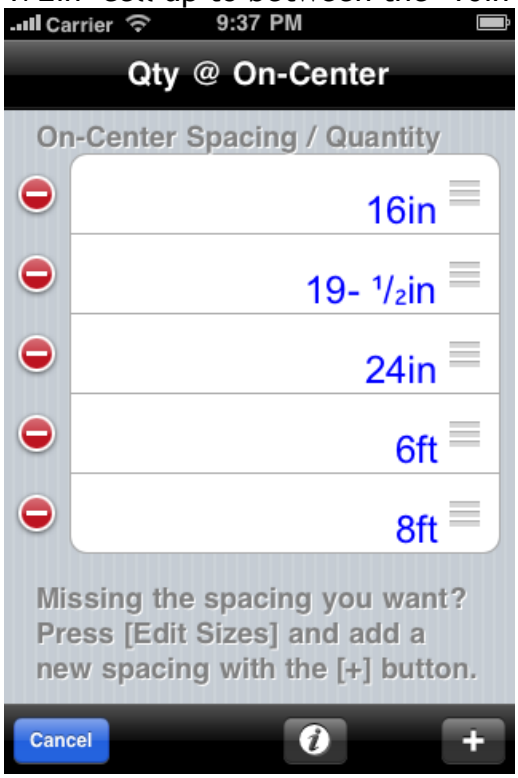
- Tap the [+] button to add a new on-center spacing.



- Enter **`19[Inches]1x/2[done]`** for your new on-center spacing



- Bonus: Using the three little bars to the right of "19-1/2in", drag the "19-1/2in" cell up to between the "16in" and "24in" cells.



- Tap [Cancel] to finish

Now the results show that, for the 22' 8-3/4" span, you will need 15 joists if spaced at 19-1/2" centers.

[R/Wall] Rake Wall Function

The Rake Wall Key can perform the following:

- Calculate the stud lengths for a rake wall as long as two of the following values are stored: [Pitch], [Rise], [Run], and [Diag] (diagonal). **Note:** *This calculation is also dependent upon the following settings:*
  - [Prefs] *Rake Wall Display*
  - On-Center Spacing valued stored at the [o.c.] key.
- Store a base wall value for use in the following functions:
  - [Arc] and [R/Wall]

Examples:

1. Calculate the Rake-Wall layout for a gable with a 8ft Run and a 8in/12in Pitch.

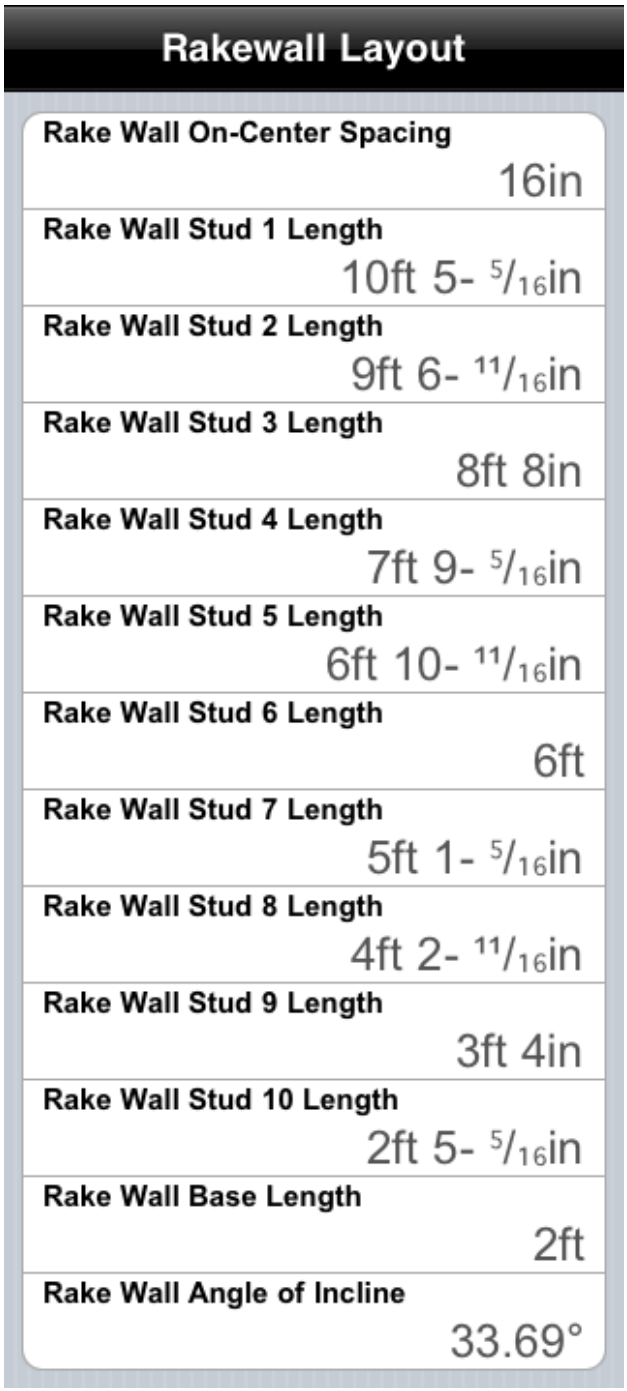
Input	*Display*
clear temporary memory	0
8 [Inch] [Pitch]	Pitch 8in
14 [Feet] [Run]	Run 14ft
[Conv] [R/Wall]	*see results, below*

Rakewall Layout

Rake Wall On-Center Spacing	16in
Rake Wall Stud 1 Length	8ft 5- 5/16in
Rake Wall Stud 2 Length	7ft 6- 11/16in
Rake Wall Stud 3 Length	6ft 8in
Rake Wall Stud 4 Length	5ft 9- 5/16in
Rake Wall Stud 5 Length	4ft 10- 11/16in
Rake Wall Stud 6 Length	4ft
Rake Wall Stud 7 Length	3ft 1- 5/16in
Rake Wall Stud 8 Length	2ft 2- 11/16in
Rake Wall Stud 9 Length	1ft 4in
Rake Wall Stud 10 Length	5- 5/16in
Rake Wall Base Length	0ft
Rake Wall Angle of Incline	33.69°

2. Calculate Rake-Wall layout from a given Run, Pitch and Base Wall.

Input	*Display*
clear temporary memory 	0
8 [Inch] [Pitch]	Pitch 8in
14 [Feet] [Run]	Run 14ft
2 [Feet] [Conv][R/Wall]	*see results, below*



Note: *The above calculations assume:*

- *16in On-Centers value stored at* [o.c.]
- *Largest to Smallest Rake Wall Display* [Prefs] *setting.*

[Radius] Radius Function

Entry or retrieval of a radius value. To be either used by or a result of the following calculations:

Key	*Description*
[Circle]	Circle Geometry Function
[Arc]	Arc Geometry and Layout Function
[ColCon] (Column/one)	Column and Cone Geometry Function

Example:

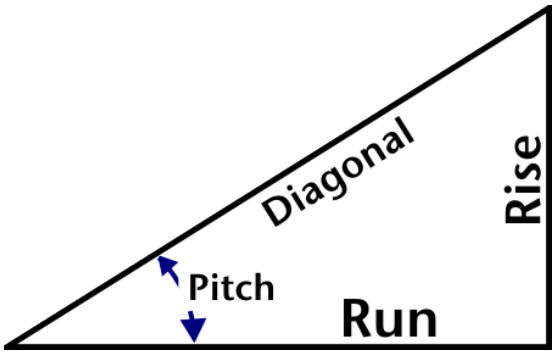
1. Simple conversion between radius and diameter

Input	*Display*
clear temporary memory 	0
6 [Feet] [Conv] [Radius]	Radius 6ft
	0
[Circle]	Diameter 12ft
	0
[Conv] [Radius]	Radius 6ft

[Rise] Rise Function

The Rise key can either:

- Calculate a Rise from any two of the following values stored:
  - [Pitch], [Run], [Diag]
- Store a rise value for other calculations
- Recall a rise value for display
- Calculate the Segment Height (Rise) for an Arc given the following:
  - Diameter ([Circle]) or [Radius]
  - [Arc] length or angle



Examples:

1. Calculate Rise from a given [Pitch] and [Diag] (diagonal).

Input	*Display*
clear temporary memory	0
8 [Inch] [Pitch]	Pitch 8in
22 [Feet] [Diag]	Diag 22ft
[Rise]	Rise 12ft 2- $^{7}/_{16}$in

2. Enter and Recall a Rise.

Input	*Display*
clear temporary memory	0
12 [Feet] [Rise]	Rise 12ft
	0
[Rise]	Rise 12ft

3. Calculated Segment Height (rise) from a diameter and arc angle

Input	*Display*
clear temporary memory 	0
2 [m] [Circle]	Diameter 2m
20.00° [Arc]	Arc 20.00°
[Rise]	Rise 0.015m

Note: *Rise values will be cleared from memory upon*  *or* [ClrAll].

[Roof] Roof Function

The Roof function calculates the following material information for a gable end roof:

- Roof Area
- Number of Shingle Squares
- Floor plan area

In addition, you can also have the Roof Function calculate quantities of custom size material - e.g. sheathing, underlayments and shingling. See the examples below to see how to do this.

To perform this calculation, the Roof Function uses the following stored values:

- Floor Plan Area, provided by either of the following:
  - an Area
  - Floor Plan dimensions: [Length] and [Width]
- Pitch, provided by either of the following:
  - [Pitch]
  - Two of the following: [Rise], [Run],[Diag]

Examples

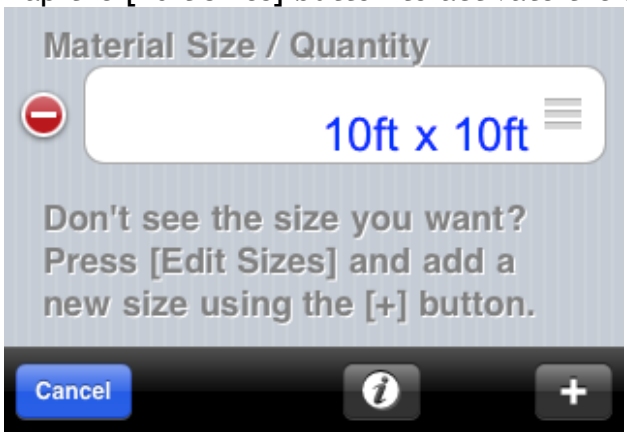
1. [Rise], [Run], [Length] and [Width] given. Let's calculate how much 4'x10' sheathing is needed for a roof that has an 11' rise, a 16' run on a house with a 32' width and a 42' length.

Input	*Display*
clear temporary memory 	0
11 [Feet] [Rise]	Rise 11ft
16 [Feet] [Run]	Run 16ft
32 [Feet] [Width]	Width 32ft

42 [Feet] [Length]	Length 42ft
[Conv] [Roof]	*See results, below*

BuildCalc tells us how many 10'x10' bundles of shingles our roof needs, but there doesn't appear to be any sheathing? No problem, just add it! Here's how:

- Tap the [Edit Sizes] button to activate the size editor.



- Now, tap the [+] button to add a new size

- Tap in **`4[Feet]x10[Feet][done]`** to enter your 4’x10’ sheathing.

- Tap [Cancel] to finish.

And now your results have been updated to include 4’x10’ sheathing.

2: Enter a floor plan area and use the last entered [Pitch]. This example assumes 8" as the last entered pitch.

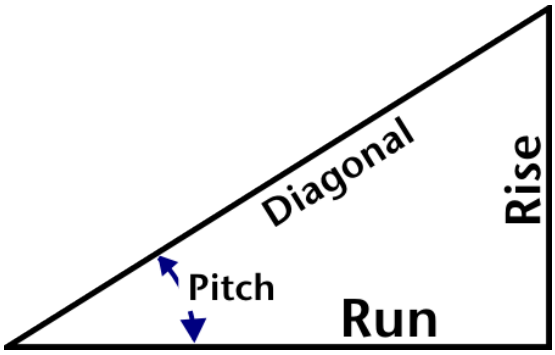
Input	*Display*
1300 [Feet] [Feet]	1300ft^2
[Conv] [Roof]	*See results, below*

Note that if a Material Size (for which you want to calculate an estimated usage) is missing, you can add it using the [Edit Sizes] button followed by the [+] button.

[Run] Run Function

The Run key can either:

- Calculate a Run from any two of the following values stored:
  - [Pitch], [Rise], [Diag]
- Store a run value for other calculations
- Recall a run value for display
- Calculate the Chord Length (Run) for an Arc given the following:
  - Diameter ([Circle]) or [Radius]
  - [Arc] length or angle



Examples:

1. Calculate Run from a given [Pitch] and [Diag] (Diagonal).

Input	*Display*
clear temporary memory [C] [C]	0
8 [Inch] [Pitch]	Pitch 8in
22 [Feet] [Diag]	Diag 22ft
[Run]	Run 18ft 3- $^{11}/_{16}$in

2. Enter and Recall a Run.

Input	*Display*
clear temporary memory 	0
12 [Feet] [Run]	Run 12ft
	0
[Run]	Run 12ft

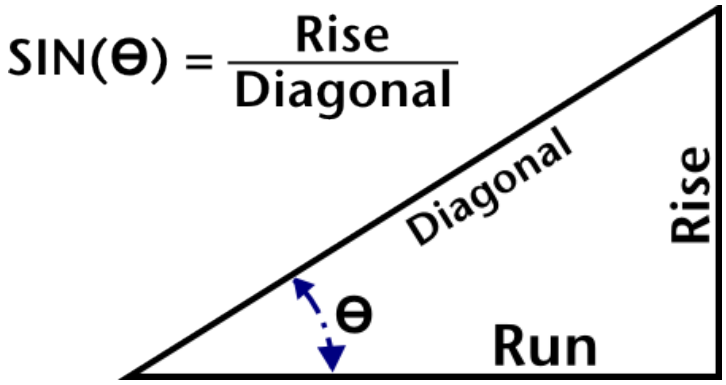
3. Calculated Chord Length (run) from a diameter and arc angle

Input	*Display*
clear temporary memory 	0
2 [m] [Circle]	Diameter 2m
20.00° [Arc]	Arc 20.00°
[Run]	Rise 0.347m

Note: *Run values will be cleared from memory upon*  or [ClrAll].

[SIN] SINE Function

- Calculate the sine for a given angle.
- The SINE of a triangle is the ratio of the rise to the diagonal.



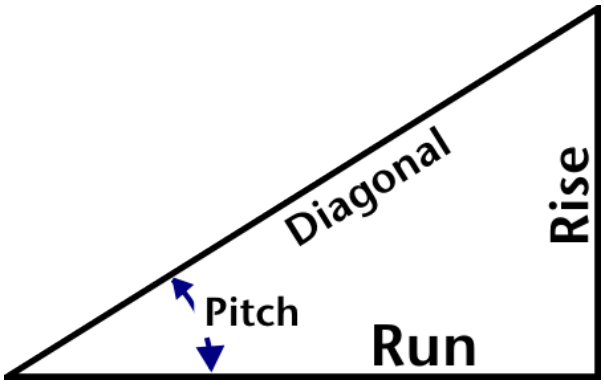
Example:

1. Calculate Sine for 38°

Input	*Display*
clear temporary memory 	0
38 [SIN]	0.615661

[Slope] Slope Function

The Slope key functions identical to the pitch key except that unit-less quantities entered will be interpreted as Slope (rise / run) instead on degrees. See the [Pitch] function for more information.



Example:

Enter a Slope.

Input	*Display*
clear temporary memory C C	0
7 [%] [Conv] [Slope]	% Grade 7
[Pitch]	Slope 7
[Pitch]	Pitch 0- $^{13}/_{16}$in

[/] Solidus (Fractional Entry) Key

Switches from numerator to denominator entry.

Example:

Input	*Display*
clear temporary memory 	0
7 [/] 16 [+]	0- $^{7}/_{16}$in +
3 [/] 4	0- $^{7}/_{16}$in + 0- $^{3}/_{4}$in
[=]	1- $^{3}/_{16}$in

[Stair] Stair Function

The stair function calculates stair layout and geometry based on the following input scenarios:

1. Stored [Rise] and [Run].
2. Stored [Rise] only.
3. Stored [Run] only.
4. Changes to Stairs function Input Parameters

See below for example calculations, notes and definitions.

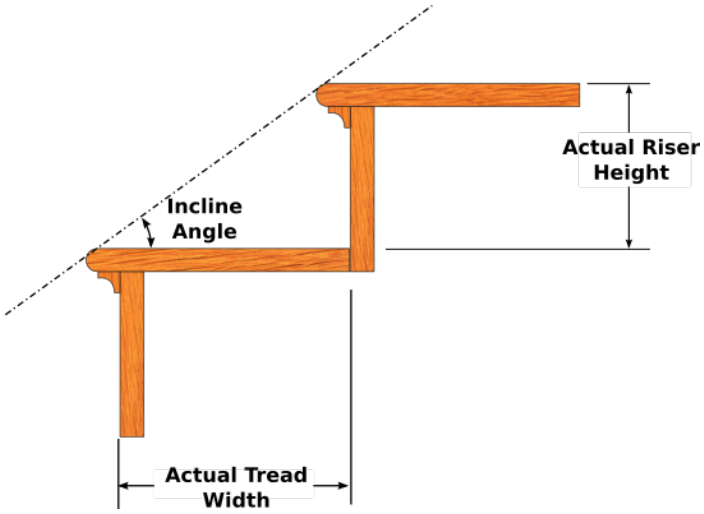
As of version 2.0, BuildCalc's Advanced Stair Function is a bit different than the Stair function in earlier versions. The biggest change is that BuildCalc now gives you **more comprehensive** results via the new Advanced analysis type. This means you can provide more detailed specifications and BuildCalc returns results that require much less guess work. And these results are interactive. You can now make changes to inputs and the results are there without having to start your calculation from the beginning. Just tap on the parameter on the right of any cell in the "Input Parameters" section and then modify that parameter.

BuildCalc now has two Stair calculation modes:

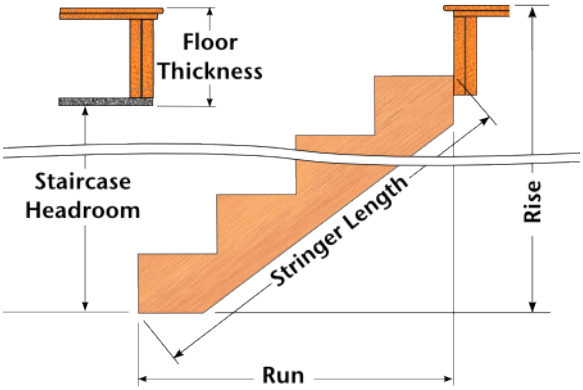
- Simple: Calculate simple stair layout parameters, giving the same results as BuildCalc version 1.3.2. This mode is good for those familiar with laying out stairs using simple layout parameters.
- Advanced: Calculate advanced stair layout parameters, giving detailed results - including dimensioned drawings for the Stringer, the Stringer's installation and the finished staircase - less guess work. This mode is good for those who require a more exact layout as well as for those who are less familiar with stair layouts.

For more information on parameters:

- Tap the on-the-spot help (at the bottom left of each input parameter cell) for more information on input parameters.
- See the drawings and definitions, below.



Tread Detail Parameters



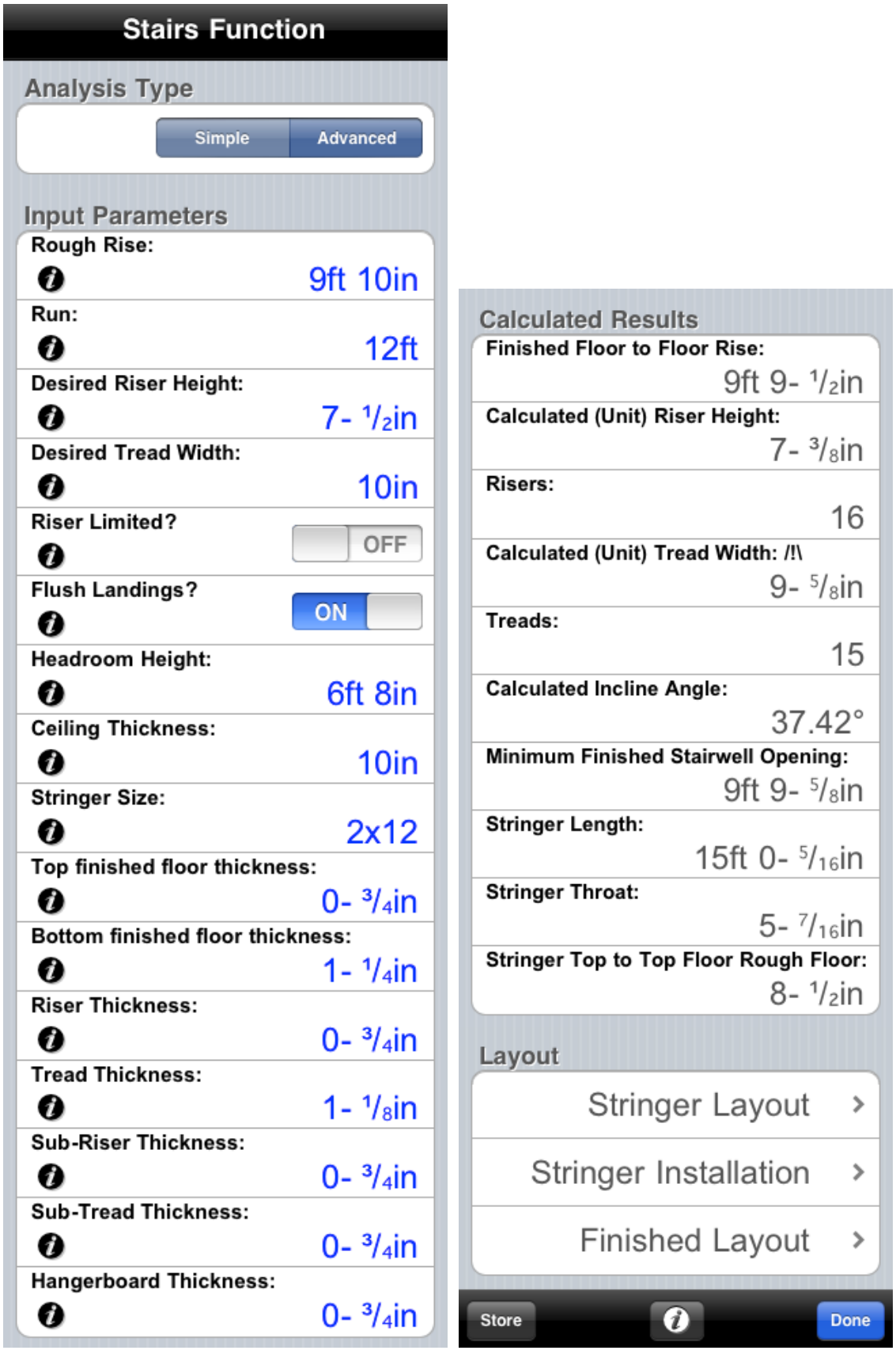
Stair Layout Parameters

Examples:

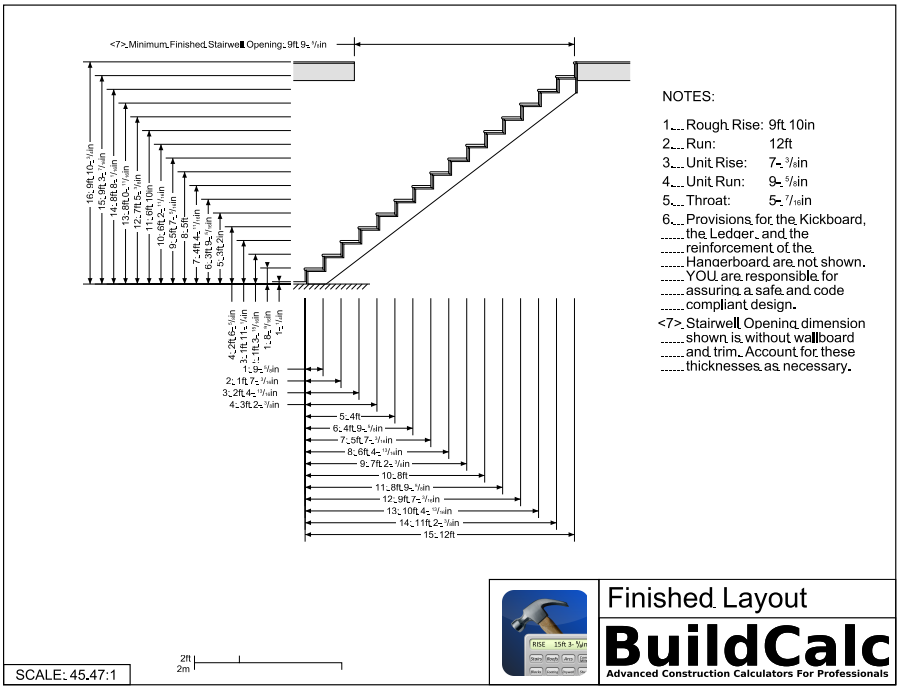
1. Stored [Rise] and [Run].

*Input**	*Display*
clear temporary memory 	0
9 [Feet] 10 [Inch]	9ft 10in
[Rise]	Rise 9ft 10in
12 [Feet] [Run]	Run 12ft
[Stair]	*See results, below*

But wait, where is the dimensioned drawings that BuildCalc promised? Well, the [Simple] Analysis Type emulates the [Stair] results found in BuildCalc prior to version 2.0. To get a more comprehensive stair layout – including dimensioned drawings – switch your Analysis Type to [Advanced].

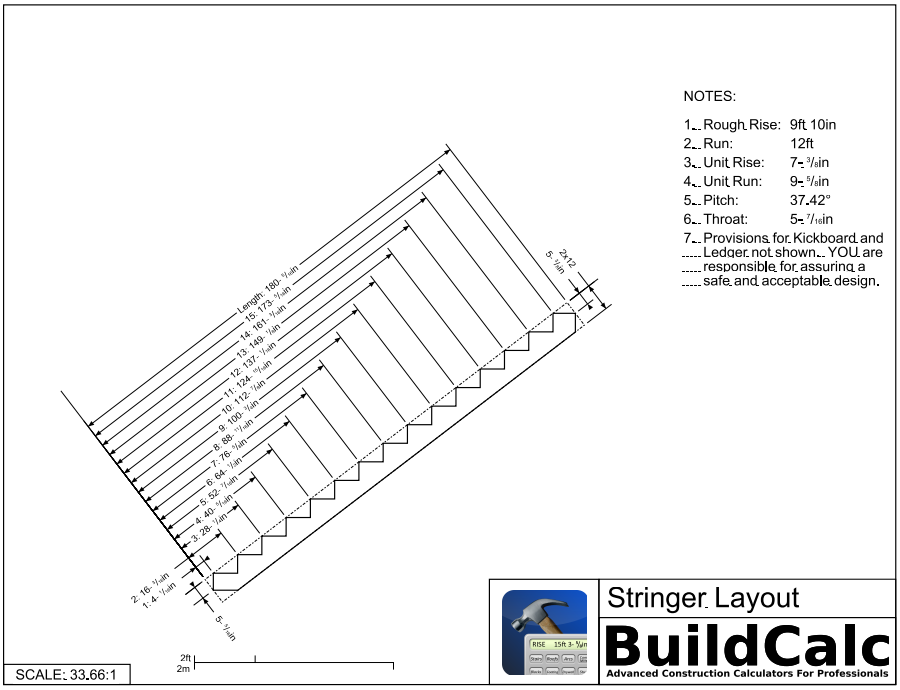

Stairs Function
Analysis Type
Simple
Advanced
Input Parameters
Rough Rise:
9ft 10in
Run:
12ft
Desired Riser Height:
7- 1/2in
Desired Tread Width:
10in
Riser Limited?
OFF
Flush Landings?
ON
Headroom Height:
6ft 8in
Ceiling Thickness:
10in
Stringer Size:
2x12
Top finished floor thickness:
0- 3/4in
Bottom finished floor thickness:
1- 1/4in
Riser Thickness:
0- 3/4in
Tread Thickness:
1- 1/8in
Sub-Riser Thickness:
0- 3/4in
Sub-Tread Thickness:
0- 3/4in
Hangerboard Thickness:
0- 3/4in
Calculated Results
Finished Floor to Floor Rise:
9ft 9- 1/2in
Calculated (Unit) Riser Height:
7- 3/8in
Risers:
16
Calculated (Unit) Tread Width: /!\
9- 5/8in
Treads:
15
Calculated Incline Angle:
37.42°
Minimum Finished Stairwell Opening:
9ft 9- 5/8in
Stringer Length:
15ft 0- 5/16in
Stringer Throat:
5- 7/16in
Stringer Top to Top Floor Rough Floor:
8- 1/2in
Layout
Stringer Layout
Stringer Installation
Finished Layout
Store
Done

Now, to generate your dimensioned drawings, scroll to the bottom and tap on the “Finished Layout” cell.



Having trouble reading the drawing? Tap on the [email] button to send yourself. And, if you want, you can then also print it out.

Is the staircase layout what you want? Well, then [Done] and generate a Stringer layout by tapping on the “Stringer Layout” cell.

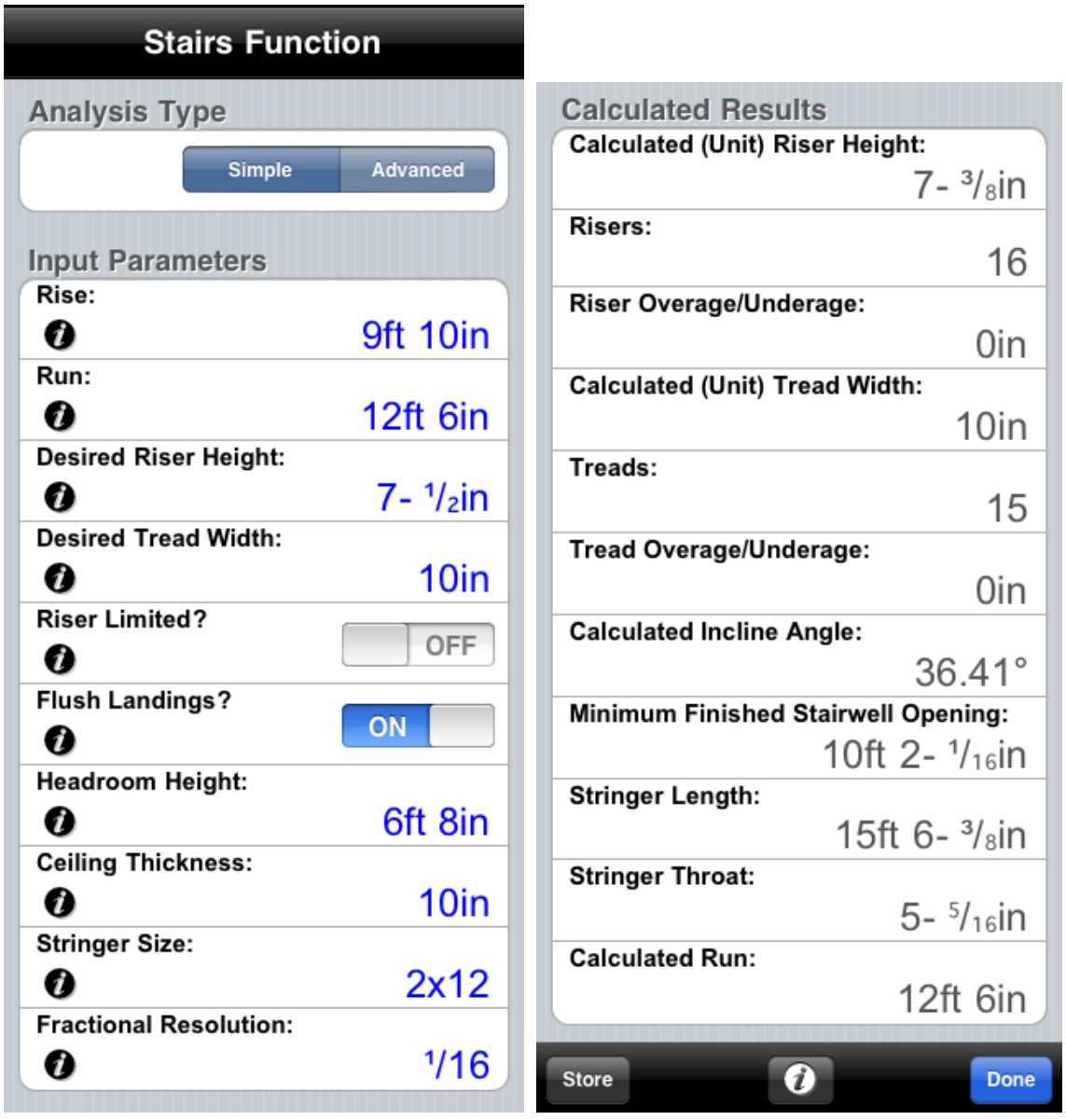


If the layout wasn’t what you wanted, take a look at the input parameters. Don’t understand a parameter? Tap on the ⓘ on-the-spot help for more

information. Still don't see what you need? Then please email us at help@BuildCalc.com or call me at 859-227-1706.

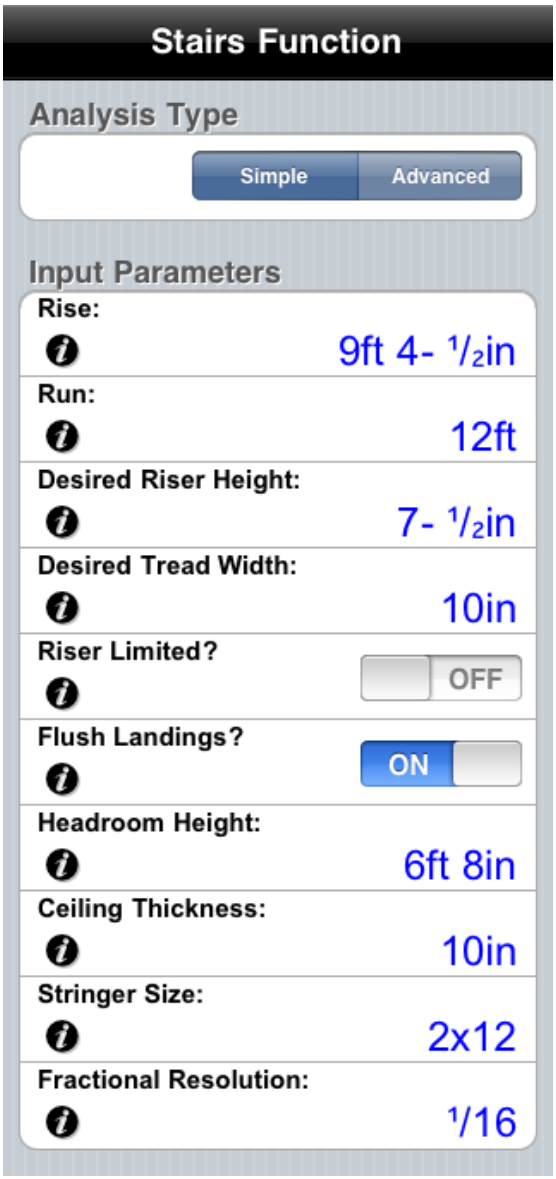
2. Stored [Rise] only. Given a 9'10" rise, calculate a staircase layout – including the staircase run.

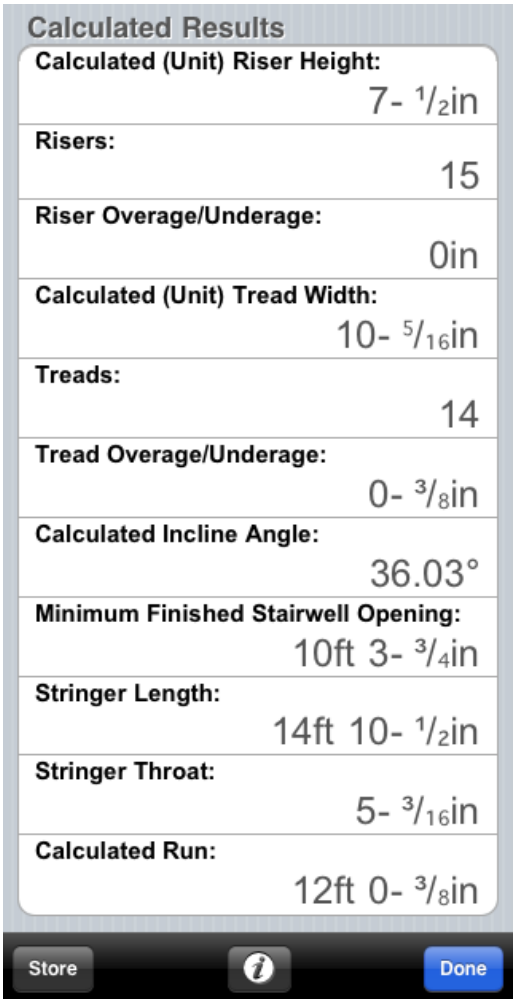
*Input**	*Display*
clear temporary memory 	0
9 [Feet] 10 [Inch]	Rise 9ft 10in
[Stair]	*See results, below*



3. Stored [Run] only. Given a 12 feet run, and 7- ½" desired riser height, what is the staircase run you can fit into this space?

*Input**	*Display*
clear temporary memory C C	0
12 [Feet] [Run]	Run 12ft
[Stair]	*See results, below*





***Note:** Depending on context, the presence of "/!\" means either:

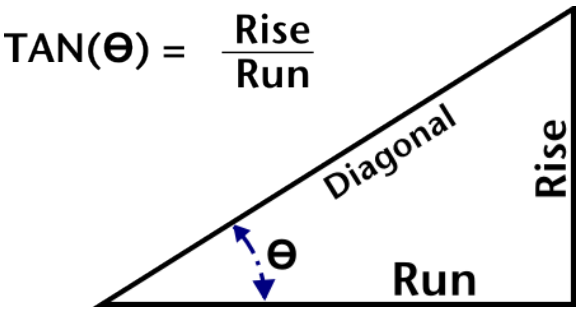
- The Actual Tread Width is less than the Desired Tread Width.
- The Actual Riser Height is greater than the Desired Riser Height.

Definitions:

- **Rise**: The vertical distance, floor to floor.
- **Run**: The horizontal distance from the first riser to the last (the distance the stair case will occupy).
- **Stringer** (a.k.a: carriages, stair horses or stair jacks): The diagonal members that support a staircase.
- **Angle of Incline**: Angle based on the rise (Actual Riser Height) and run (Actual Tread Width) of each stair. Note: This is not the same as the pitch of a stair case (staircase Rise / staircase Run).
- **Stairwell Opening:** The size of the hole in the floor above necessary to assure sufficient Staircase Headroom. Most codes have a minimum staircase headroom of 6' 8".
- **Riser**: The vertical face of a step.
- **Desired Riser Height**: The desired vertical rise for each step.
- **Actual Riser Height**: The calculated vertical riser for each step.
- **Riser Overage / Underage**: The results of step size calculations are in fractional increments (for ease of layout) Because of this, the sum of all the Actual Riser Heights may be a little over or a little under the Rise.
- **Tread:** The horizontal face of a step.
- **Desired Tread Width**: The distance of each tread measured from the face of one riser to the face of the next riser.
- **Actual Tread Width**: The calculated horizontal distance for each tread (measured from riser face to riser face).
- **Tread Overage / Underage**: The results of step size calculations are in fractional increments (for ease of layout) Because of this, the sum of all the Actual Tread Widths may be a little over or a little under the Run.

[TAN] TANGENT Function

- Calculate the tangent for a given angle.
- The tangent of a triangle is the ratio of the rise to the run.



Example:

1. Calculate tangent for 38°

*Input**	*Display*
clear temporary memory C C	0
38 [TAN]	0.781286

[tons] Tons* Function

- Set the units for a quantity to tons.
- Convert a weight to tons.
- Convert a volume to tons (using the density stored in [wt/vol])

NOTE: Switch between long tons (2240lbs) and short tons (2000lbs) using the "Pounds per Ton" setting, found in [Conv] [Prefs].

Examples:

1. Set the units for a quantity to tons.

Input	*Display*
clear temporary memory 	0
3 [Conv] [tons]	3T

2. Convert a weight to tons.

Input	*Display*
clear temporary memory 	0
12 [Conv] [m tons]	12MT
[Conv] [tons]	13.22774T

3. Convert a volume to tons (using the density stored in [wt/vol])

*Input**	*Display*
clear temporary memory [C] [C]	0
11 [Yard] [Yard] [Yard]	$11yd^3$
[Conv] [tons]	16.5T

***Note**:

1. *Assumes 1.5 tons per cubic yard is stored in [wt/vol].*
2. *Assumes short tons. Go to [Conv] [Prefs] to switch between short (2,000 lbs) and long (2240 lbs) tons.*

[Width] Width Function

1. Store and Recall a width for use by the [Roof], [Width], [Height], [Msnry], and [DryWal] functions
2. Calculate Area, Square-up and Perimeter

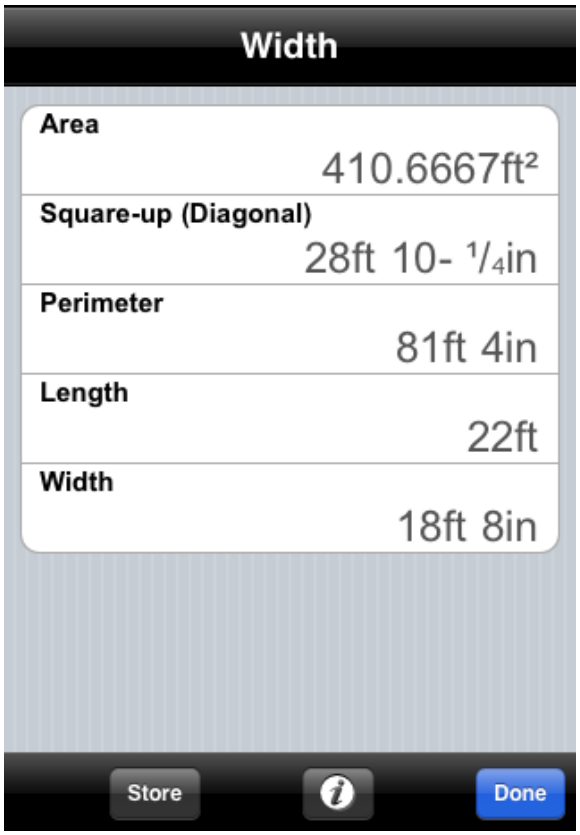
Examples:

1. Store and Recall a width.

Input	*Display*
clear temporary memory [C] [C]	0
[Recall] [Width]	0
10 [Feet] [Width]	Width 10ft
[C]	0
[Width]	Width 10ft

2. Calculate Area, Square-up and Perimeter.

Input	*Display*
clear temporary memory 	0
22 [Feet] [Length]	Length 22ft
18 [Feet] 8 [Inch] [Width]	Width 18ft 8in
[Width]	*See results, below*



[wt/vol] Density Function

1. Enter a density (weight per unit volume) value to be used in other functions*.
2. Recall the stored density value.

Example:

1. Enter a new weight to volume conversion constant of 1600 kg / m^3 and then display it.

Input	*Display*
clear temporary memory C C 1600	1600
[Store] [wt/vol]	1600T/yd^3
[wt/vol]	1600lb/yd^3
[wt/vol]	1600lb/ft^3
[wt/vol]	1600MT/m^3
[wt/vol]	1600kg/m^3
C	0
[Recall] [wt/vol]	1.348444T/yd^3
[wt/vol]	2696.888lb/yd^3
[wt/vol]	99.88474lb/ft^3

[wt/vol]	1.6MT/m^3
[wt/vol]	1600kg/m^3

***Density is used in the following functions:**

Function	*Description*
[Yard]	Conversion between a weight and cubic yards.
[Feet]	Conversion between a weight and cubic feet.
[Inch]	Conversion between a weight and cubic inches.
[m]	Conversion between a weight and cubic meters.
[cm]	Conversion between a weight and cubic centimeters.
[mm]	Conversion between a weight and cubic mm.
[Bd Ft]	Conversion between a weight and board feet.
[lbs]	Conversion between a volume and pounds.
[Tons]	Conversion between a volume and tons.
[kg]	Conversion between a volume and kilograms.
[m tons]	Conversion between a volume and metric tons.

[Yard] Yard Entry and Conversion

1. Set the units for a quantity to yards
2. Convert a length, area or volume to yards
3. Convert a weight to cubic yards

Examples:

1. Set the units for a quantity to yards.

Input	*Display*
clear temporary memory 	0
9 [Yard]	9yd
[Yard]	$9yd^2$
[Yard]	$9yd^3$

2. Converts a length, area or volume to yards.

Input	*Display*
clear temporary memory 	0
9 [m]	9m
[Yard]	9.84252yd

3. Converts a weight to cubic yards (using the density stored in [wt/vol]).

Input	*Display*
clear temporary memory © ©	0
11 [Conv] [tons]	11T
[Yard]	9.7.333333yd^3

**Note: Assumes 1.5 tons per cubic yard is stored in* [wt/vol].