

## Stock Beta Calculator

[\*http://spreadsheetml.com/finance/stockbetacalculatorspreadsheet.shtml\*](http://spreadsheetml.com/finance/stockbetacalculatorspreadsheet.shtml)

Copyright (c) 2008-2009, ConnectCode Pte Ltd.

All Rights Reserved.

ConnectCode accepts no responsibility for any adverse affect that may result from undertaking our training. No statements in this document should be construed or thought to represent investment advice of any type since the sole purpose of the explanation is to illustrate the technique.

Microsoft and Microsoft Excel are registered trademarks of Microsoft Corporation. All other product names are trademarks, registered trademarks, or service marks of their respective owners

# Table of Contents

<b>1.</b>	<b>Stock Beta</b> .....	<b>1-1</b>
<b>2.</b>	<b>Stock Beta Calculator</b> .....	<b>2-2</b>
	2.1 Inputs .....	2-2
	2.2 Outputs.....	2-2
<b>3.</b>	<b>Advance Stock Beta Calculator</b> .....	<b>3-3</b>
<b>4.</b>	<b>Customizing the Stock Beta Calculator</b> .....	<b>4-4</b>
	4.1 StockBetaInternal Worksheet .....	4-4
	4.2 How is the data downloaded?.....	4-4
	4.2.1 GetStock subroutine .....	4-4
	4.2.2 DownloadData subroutine.....	4-6

## ConnectCode's Financial Modeling Templates

*Have you thought about how many times you use or reuse your financial models? Everyday, day after day, model after model and project after project. We definitely have. That is why we build all our financial templates to be reusable, customizable and easy to understand. We also test our templates with different scenarios vigorously, so that you know you can be assured of their accuracy and quality and that you can save significant amount of time by reusing them. We have also provided comprehensive documentation on the templates so that you do not need to guess or figure out how we implemented the models.*

*All our template models are only in black and white color. We believe this is how a professional financial template should look like and also that this is the easiest way for you to understand and use the templates. All the input fields are marked with the '\*' symbol for you to identify them easily.*

*Whether you are a financial analyst, investment banker or accounting personnel. Or whether you are a student aspiring to join the finance world or an entrepreneur needing to understand finance, we hope that you will find this package useful as we have spent our best effort and a lot of time in developing them.*

ConnectCode

# 1. Stock Beta

Stock Beta is the measure of the risk of an individual stock. Basically, it measures the volatility of a stock against a broader or more general market. It is a commonly used indicator by financial and investment analysts. The Capital Asset Pricing Model (CAPM) also uses the Beta by defining the relationship of the expected rate of return as a function of the risk free interest rate, the investment's Beta, and the expected market risk premium.

## **CAPM**

Expected rate of return = Risk free rate + **Beta** \* (Market Risk Premium)

To interpret and understand the numbers from the Beta is simple and straight forward. The Beta of the general and broader market portfolio is always assumed to be 1. A stock Beta is calculated to be relative to the Beta of the broader market. Thus when a stock has a Beta that is greater than 1, it is considered to be more risky and more volatile than the broader market while a stock with a Beta of less than 1 is considered to be less risky and less volatile than the broader market. Finally a stock with a Beta equal to 1 is considered neutral and as volatile as the broader market.

The following formula is used for calculating the value of Beta.

Beta = Covariance(Rate of Return of Stock, Rate of Return of Market) / Variance of Market

Covariance is a measure of how two variables change together or is related and Variance is a statistical measure of the dispersion of values from the mean. The rest of this document will illustrate on how to calculate the Beta of an individual stock against the broader S&P 500 portfolio. The interesting part is we will be downloading live data from <http://finance.yahoo.com> to perform the calculation.

## 2. Stock Beta Calculator

This Stock Beta Calculator spreadsheet allows you to calculate Beta of U.S stocks very easily. First, it provides the formulas for calculating the Beta. Second, and more importantly, it calculates the Beta by automatically downloading stock quotes and the S&P 500 data from <http://finance.yahoo.com>. The spreadsheet is shown below:

	A	B	C	D	E	F	G	H	I
1	Stock Beta (Using S&P 500 as market portfolio)								
2	Copyright © 2009, ConnectCode. All rights reserved.								
3	This spreadsheet uses Monthly stock quotes and S&P500 data downloaded automatically								
4	from <a href="http://finance.yahoo.com">http://finance.yahoo.com</a> to calculate the Stock Beta.								
5									
6	Stock Symbol*								DELL
7	Start Date (MM/DD/YYYY)*								6/1/1998
8	End Date (MM/DD/YYYY)*								6/1/2001
9									
10	Covariance(Stock,Market)								0.005362
11	Variance(Market)								0.002461
12	Stock Beta								2.179139
13									
14									Calculate
15									
16									

### 2.1 Inputs

- Stock Symbol - The Stock Symbol used by Yahoo Finance. For example, "YHOO" is the stock symbol for Yahoo. "MSFT" is the stock symbol for Microsoft. Check out <http://finance.yahoo.com> for the list of Stock Symbols supported.
- Start Date - The start date in MM/DD/YYYY format.
- End Date - The end date in MM/DD/YYYY format.

### 2.2 Outputs

After clicking on the Calculate button, an Excel VBA macro will be launched to download the Monthly Stock Quotes from the Start Date to the End Date of the specified Stock Symbol. The Monthly Returns in column H and column P are then tabulated in the StockBetaInternal spreadsheet. Before any calculation, please make sure you are connected to the internet.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	S&P								DELL							
2	Date	Open	High	Low	Close	Volume	Adj Close	Returns	Date	Open	High	Low	Close	Volume	Adj Close	Returns
3	6/1/2001	1255.82	1265.34	1246.88	1260.67	2030000000	1260.67	0.00386	6/1/2001	24.71	25.1	24.25	24.61	63822000	24.61	0.010263
4	5/1/2001	1249.46	1315.93	1232	1255.82	1170568100	1255.82	0.00509	5/1/2001	25.56	27.39	23.49	24.36	34090900	24.36	-0.07165
5	4/2/2001	1160.33	1269.3	1091.99	1249.46	1333839500	1249.46	0.07681	4/2/2001	25.81	31.32	21.31	26.24	42909900	26.24	0.021409
6	3/1/2001	1239.94	1267.42	1081.19	1160.33	1322155000	1160.33	-0.0642	3/1/2001	21.06	28	20	25.69	40080400	25.69	0.174668
7	2/1/2001	1366.01	1376.38	1215.44	1239.94	1203668400	1239.94	-0.0923	2/1/2001	26.37	27.5	19.94	21.87	32893000	21.87	-0.16271
8	1/2/2001	1320.28	1383.37	1274.62	1366.01	1386909500	1366.01	0.03464	1/2/2001	17.5	28.81	16.94	26.12	44193700	26.12	0.497706
9	12/1/2000	1314.95	1389.05	1254.07	1320.28	1232315000	1320.28	0.00405	12/1/2000	19.94	22.37	16.25	17.44	45610600	17.44	-0.09403

The Stock Beta is calculated as the formula below.

$$\text{Stock Beta} = \text{Beta} = \frac{\text{Covariance}(\text{Rate of Return of Stock}, \text{Rate of Return of Market})}{\text{Variance of Market}}$$

### 3. Advance Stock Beta Calculator

This spreadsheet builds on the Stock Beta Calculator described above. It adds the following capabilities.

- The calculation of Daily, Weekly and Monthly returns. If you recall, the Stock Beta Calculator only supports Monthly returns.
- The calculation of the Beta for up to 5 stocks. If you recall, the Stock Beta Calculator only supports the calculation of Beta for one stock at one time.

	A	B	C	D	E	F	G	H	I
1	<b>Stock Beta (Using S&amp;P 500 as market portfolio)</b>								
2									
3	<a href="#">Copyright © 2009, ConnectCode. All rights reserved.</a>								
4									
5	<i>This spreadsheet uses stock quotes and S&amp;P500 data downloaded automatically from</i>								
6	<i>http://finance.yahoo.com to calculate the Stock Beta.</i>								
7									
8	Start Date (MM/DD/YYYY)*				1/1/2001				
9	End Date (MM/DD/YYYY)*				6/1/2001				
10	Stock Quotes Frequency*				Monthly	<input type="button" value="v"/>			
11									
12					Stock 1	Stock 2	Stock 3	Stock 4	Stock 5
13	Stock Symbols*				DELL	YHOO	NONE	NONE	NONE
14									
15	Covariance(Stock,Market)				0.000945	0.01402			
16	Variance(Market)				0.003516	0.003516			
17	Stock Beta				0.268778	3.987291			
18					<input type="button" value="Calculate"/>				
19									

## 4. Customizing the Stock Beta Calculator

This section describes the VBA source code used to construct the Stock Beta calculator. With the understanding of the basic source code it will be easy to customize the Stock Beta calculator or reuse and expand it in other financial models. The Advance Stock Beta Calculator will not be described. However by looking at the source code, you will find that it is similar to the basic Stock Beta Calculator.

### 4.1 StockBetaInternal Worksheet

This worksheet is used internally by the calculator. Column A to column G contains the downloaded S&P 500 market data. The columns contain data that include Date, Open, High, Low, Close, Volume and Adj Close.

1	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
2	S&P								DELL							
3	Date	Open	High	Low	Close	Volume	Adj Close	Returns	Date	Open	High	Low	Close	Volume	Adj Close	Returns
4	6/1/2001	1255.82	1265.34	1246.88	1260.67	2030000000	1260.67	0.00386	6/1/2001	24.71	25.1	24.25	24.61	63822000	24.61	0.010263
5	5/1/2001	1249.46	1315.93	1232	1255.82	1170568100	1255.82	0.00509	5/1/2001	25.56	27.39	23.49	24.36	34090900	24.36	-0.07165
6	4/2/2001	1160.33	1269.3	1091.99	1249.46	1333839500	1249.46	0.07681	4/2/2001	25.81	31.32	21.31	26.24	42909900	26.24	0.021409
7	3/1/2001	1239.94	1267.42	1081.19	1160.33	1322155000	1160.33	-0.0642	3/1/2001	21.06	28	20	25.69	40080400	25.69	0.174668
8	2/1/2001	1366.01	1376.38	1215.44	1239.94	1203668400	1239.94	-0.0923	2/1/2001	26.37	27.5	19.94	21.87	32893000	21.87	-0.16271
9	1/2/2001	1320.28	1383.37	1274.62	1366.01	1386909500	1366.01	0.03464	1/2/2001	17.5	28.81	16.94	26.12	44193700	26.12	0.497706
10	12/1/2000	1314.95	1389.05	1254.07	1320.28	1232315000	1320.28	0.00405	12/1/2000	19.94	22.37	16.25	17.44	45610600	17.44	-0.09403

The Returns column is calculated by the VBA macro which we will describe below. It uses the latest Adj Close and the previous Adj Close to calculate the periodic rate of return. The formula used is shown below:

$$\text{Returns} = (\text{Latest Adj Close} - \text{Previous Adj Close}) / \text{Previous Adj Close}$$

The Returns column is tabulated for use in the calculation of the Covariance and Variance output.

Column I to column O contain similar information to the S&P 500 data except that the data is for the individual stock specified in the 'Stock Beta' worksheet. Column P is also calculated using the Returns formula above.

### 4.2 How is the data downloaded?

The Excel spreadsheet uses the macro DownloadData to automatically populate the data in the 'StockBetaInternal' worksheet. If you goto Developer->Visual Basic and open up the Microsoft Visual Basic Editor. After that, double click on the 'VBA Project (FreeStockBetaCalculator.xls)' and open up Module->Module1. This module contains all the source code for automatically downloading the data.

#### 4.2.1 GetStock subroutine

The VBA code for the GetStock subroutine is listed below. This function downloads data from <http://finance.yahoo.com> by specifying a Stock Symbol, Start Date and End Date. The last "desti" parameter specifies the location to place the downloaded data.

```
Sub GetStock(ByVal stockSymbol As String, ByVal startDate As Date, ByVal endDate As Date,
ByVal desti As String)
```

```
    Dim noErrorFound As Integer
    Dim DownloadURL As String
    Dim StartMonth, StartDay, StartYear, EndMonth, EndDay, EndYear As String
    StartMonth = Format(Month(startDate) - 1, "00")
    StartDay = Format(Day(startDate), "00")
    StartYear = Format(Year(startDate), "00")
```

```
    EndMonth = Format(Month(endDate) - 1, "00")
```

```

EndDay = Format(Day(EndDate), "00")
EndYear = Format(Year(EndDate), "00")
DownloadURL = "URL;http://table.finance.yahoo.com/table.csv?s=" + stockSymbol + "&a=" +
StartMonth + "&b=" + StartDay + "&c=" + StartYear + "&d=" + EndMonth + "&e=" + EndDay +
"&f=" + EndYear + "&g=m&ignore=.csv"

On Error GoTo ErrHandler:
With ActiveSheet.QueryTables.Add(Connection:=DownloadURL, Destination:=Range(desti))
    .FieldNames = True
    .RowNumbers = False
    .FillAdjacentFormulas = False
    .PreserveFormatting = True
    .RefreshOnFileOpen = False
    .BackgroundQuery = True
    .RefreshStyle = xlInsertDeleteCells
    .SavePassword = False
    .SaveData = True
    .AdjustColumnWidth = True
    .RefreshPeriod = 0
    .WebSelectionType = xlSpecifiedTables
    .WebFormatting = xlWebFormattingNone
    .WebTables = "20"
    .WebPreFormattedTextToColumns = True
    .WebConsecutiveDelimitersAsOne = True
    .WebSingleBlockTextImport = False
    .WebDisableDateRecognition = False
    .WebDisableRedirections = False
    .Refresh BackgroundQuery:=False
End With
noErrorFound = 1

ErrHandler:
If noErrorFound = 0 Then
    MsgBox ("Stock " + stockSymbol + " cannot be found.")
End If
Resume Next

End Sub

```

In the whole block of code above, the most important line is the following.

```
With ActiveSheet.QueryTables.Add(Connection:=DownloadURL, Destination:=Range(desti))
```

It specifies we will be downloading data from DownloadURL and placing the result into the cell specified in 'desti'. The DownloadURL is constructed based on the parameters explained below.

```
http://table.finance.yahoo.com/table.csv?s=YHOO&a=01&b=01&c=2007&d=08&e=05&f=2008&g
=m&ignore=.csv
```

- "s=YHOO" means to download the stock prices of Yahoo. YHOO is the stock symbol of Yahoo.
- "a=01&b=01&c=2007" specifies the start date in Month, Day, Year. You may have noticed that the month is subtracted with 1, which is the format required by Yahoo.
- "d=08&e=05&f=2008" specifies the end date in Month, Day, Year. You may have noticed that the month is subtracted with 1, which is the format required by Yahoo.
- "g=m" specifies to download monthly data. Change the "m" to "d" for daily data and "w" for weekly data.

## 4.2.2 DownloadData subroutine

This is the subroutine that is called by the Calculate button in the 'Stock Beta' worksheet.

### 4.2.2.1 Calling the GetStock subroutine

The source code for the DownloadData subroutine is shown below. The sections highlighted in Red show the part where DownloadData calls the GetStock subroutine. The first subroutine call gets the S&P 500 data by passing the "^GSPC" symbol. The second call to the GetStock subroutine uses the stock symbol specified in the 'Stock Beta' worksheet to get the individual stock data.

```
Sub DownloadData()  
    Dim numRows As Integer  
    numRows = DateDiff("m", Worksheets("Stock Beta").Range("$I$9"), Worksheets("Stock Beta").Range("$I$10")) + 2  
  
    Worksheets("StockBetaInternal").Select  
    Worksheets("StockBetaInternal").UsedRange.Clear  
    Cells(1, 1) = "S&P"  
    Cells(2, 8) = "Returns"  
    Call GetStock("^GSPC", Worksheets("Stock Beta").Range("$I$9"), Worksheets("Stock Beta").Range("$I$10"), "$A$2")  
    ActiveWindow.SmallScroll Down:=-12  
    Columns("A:A").TextToColumns Destination:=Range("A1"), DataType:=xlDelimited, _  
        TextQualifier:=xlDoubleQuote, ConsecutiveDelimiter:=False, Tab:=True, _  
        Semicolon:=False, Comma:=True, Space:=False, Other:=False, FieldInfo _  
        :=Array(Array(1, 1), Array(2, 1), Array(3, 1), Array(4, 1), Array(5, 1), Array(6, 1), _  
        Array(7, 1))  
  
    Cells(1, 9) = Worksheets("Stock Beta").Range("$I$8")  
    Cells(2, 16) = "Returns"  
    Call GetStock(Worksheets("Stock Beta").Range("$I$8"), Worksheets("Stock Beta").Range("$I$9"), Worksheets("Stock Beta").  
    ActiveWindow.SmallScroll Down:=-12  
    Columns("I:I").TextToColumns Destination:=Range("I1"), DataType:=xlDelimited, _  
        TextQualifier:=xlDoubleQuote, ConsecutiveDelimiter:=False, Tab:=True, _  
        Semicolon:=False, Comma:=True, Space:=False, Other:=False, FieldInfo _  
        :=Array(Array(1, 1), Array(2, 1), Array(3, 1), Array(4, 1), Array(5, 1), Array(6, 1), _  
        Array(7, 1))  
    Columns("A:O").EntireColumn.AutoFit  
  
    Cells(3, 8) = "=IF(G4="" & Chr(34) & Chr(34) & "," & Chr(34) & Chr(34) & ", (G3-G4)/G4)"  
    Worksheets("StockBetaInternal").Range("H3").Select  
    Selection.AutoFill Destination:=Range("H3:H" & numRows), Type:=xlFillDefault  
  
    Cells(3, 16) = "=IF(O4="" & Chr(34) & Chr(34) & "," & Chr(34) & Chr(34) & ", (O3-O4)/O4)"  
    Worksheets("StockBetaInternal").Range("P3").Select  
    Selection.AutoFill Destination:=Range("P3:P" & numRows), Type:=xlFillDefault  
  
    Worksheets("Stock Beta").Select  
End Sub
```

#### 4.2.2.2 Formatting the data

The next part of the code formats the downloaded data. The initial downloaded data will be placed in one single column of the spreadsheet. The **TextToColumns** function splits this column into multiple columns.

For example, initially the Date, Open, High, Low, Close, Volume and Adj Close will all be downloaded into a single column of Excel. The TextToColumns function will split the Date into 1 column, Open into 1 column, High into 1 column and so on.

```
Sub DownloadData()
    Dim numRows As Integer
    numRows = DateDiff("m", Worksheets("Stock Beta").Range("$I$9"), Worksheets("Stock Beta").Range("$I$10")) + 2

    Worksheets("StockBetaInternal").Select
    Worksheets("StockBetaInternal").UsedRange.Clear
    Cells(1, 1) = "S&P"
    Cells(2, 8) = "Returns"
    Call GetStock("^GSPC", Worksheets("Stock Beta").Range("$I$9"), Worksheets("Stock Beta").Range("$I$10"), "$A$2")
    ActiveWindow.SmallScroll Down:=-12
    Columns("A:A").TextToColumns Destination:=Range("A1"), DataType:=xlDelimited, _
        TextQualifier:=xlDoubleQuote, ConsecutiveDelimiter:=False, Tab:=True, _
        Semicolon:=False, Comma:=True, Space:=False, Other:=False, FieldInfo _
        :=Array(Array(1, 1), Array(2, 1), Array(3, 1), Array(4, 1), Array(5, 1), Array(6, 1), _
        Array(7, 1))

    Cells(1, 9) = Worksheets("Stock Beta").Range("$I$8")
    Cells(2, 16) = "Returns"
    Call GetStock(Worksheets("Stock Beta").Range("$I$8"), Worksheets("Stock Beta").Range("$I$9"), Worksheets("Stock Beta").
    ActiveWindow.SmallScroll Down:=-12
    Columns("I:I").TextToColumns Destination:=Range("I1"), DataType:=xlDelimited, _
        TextQualifier:=xlDoubleQuote, ConsecutiveDelimiter:=False, Tab:=True, _
        Semicolon:=False, Comma:=True, Space:=False, Other:=False, FieldInfo _
        :=Array(Array(1, 1), Array(2, 1), Array(3, 1), Array(4, 1), Array(5, 1), Array(6, 1), _
        Array(7, 1))
    Columns("A:O").EntireColumn.AutoFit

    Cells(3, 8) = "=IF(G4=" & Chr(34) & Chr(34) & ", " & Chr(34) & Chr(34) & ", (G3-G4)/G4)"
    Worksheets("StockBetaInternal").Range("H3").Select
    Selection.AutoFill Destination:=Range("H3:H" & numRows), Type:=xlFillDefault

    Cells(3, 16) = "=IF(O4=" & Chr(34) & Chr(34) & ", " & Chr(34) & Chr(34) & ", (O3-O4)/O4)"
    Worksheets("StockBetaInternal").Range("P3").Select
    Selection.AutoFill Destination:=Range("P3:P" & numRows), Type:=xlFillDefault

    Worksheets("Stock Beta").Select
End Sub
```

### 4.2.2.3 Calculation of the Returns

The final part of the code tabulates the Returns column of the S&P 500 data and the Returns of the Stock Quotes.

```
Sub DownloadData()
    Dim numRows As Integer
    numRows = DateDiff("m", Worksheets("Stock Beta").Range("$I$9"), Worksheets("Stock Beta").Range("$I$10")) + 2

    Worksheets("StockBetaInternal").Select
    Worksheets("StockBetaInternal").UsedRange.Clear
    Cells(1, 1) = "S&P"
    Cells(2, 8) = "Returns"
    Call GetStock("^GSPC", Worksheets("Stock Beta").Range("$I$9"), Worksheets("Stock Beta").Range("$I$10"), "$A$2")
    ActiveWindow.SmallScroll Down:=-12
    Columns("A:A").TextToColumns Destination:=Range("A1"), DataType:=xlDelimited, _
        TextQualifier:=xlDoubleQuote, ConsecutiveDelimiter:=False, Tab:=True, _
        Semicolon:=False, Comma:=True, Space:=False, Other:=False, FieldInfo _
        :=Array(Array(1, 1), Array(2, 1), Array(3, 1), Array(4, 1), Array(5, 1), Array(6, 1), _
        Array(7, 1))

    Cells(1, 9) = Worksheets("Stock Beta").Range("$I$8")
    Cells(2, 16) = "Returns"
    Call GetStock(Worksheets("Stock Beta").Range("$I$8"), Worksheets("Stock Beta").Range("$I$9"), Worksheets("Stock Beta").
    ActiveWindow.SmallScroll Down:=-12
    Columns("I:I").TextToColumns Destination:=Range("I1"), DataType:=xlDelimited, _
        TextQualifier:=xlDoubleQuote, ConsecutiveDelimiter:=False, Tab:=True, _
        Semicolon:=False, Comma:=True, Space:=False, Other:=False, FieldInfo _
        :=Array(Array(1, 1), Array(2, 1), Array(3, 1), Array(4, 1), Array(5, 1), Array(6, 1), _
        Array(7, 1))
    Columns("A:O").EntireColumn.AutoFit

    Cells(3, 8) = "=IF(G4="" & Chr(34) & Chr(34) & "," & Chr(34) & Chr(34) & ",(G3-G4)/G4)"
    Worksheets("StockBetaInternal").Range("H3").Select
    Selection.AutoFill Destination:=Range("H3:H" & numRows), Type:=xlFillDefault

    Cells(3, 16) = "=IF(O4="" & Chr(34) & Chr(34) & "," & Chr(34) & Chr(34) & ",(O3-O4)/O4)"
    Worksheets("StockBetaInternal").Range("P3").Select
    Selection.AutoFill Destination:=Range("P3:P" & numRows), Type:=xlFillDefault

    Worksheets("Stock Beta").Select
End Sub
```

The code starts by setting "Cells(3,8)" to the following

=IF(G4="" , "" , ((G3-G4)/G4))

'Chr(34)' is equivalent to double quotes in VBA. After setting 'Cells(3,8)'. The code then uses the **AutoFill** function to automatically fill the rest of the rows that require the Returns calculation. Using the AutoFill function is like using Copy and Paste in Excel. It has the advantage of letting Excel automatically update (increment) the formula for you. The table below illustrates what happens to each cell populated by the AutoFill function. When AutoFill populates the cell in Row 4, it updates the formula by changing (incrementing) the "G3" and "G4" portion.

<b>Row 3</b> =IF(G4="" , "" , ((G3-G4)/G4))
<b>Row 4</b> =IF(G5="" , "" , ((G4-G5)/G5))
<b>Row 5</b> =IF(G6="" , "" , ((G5-G6)/G6))
.
.
.