

# Spreadsheet Example Formulas and Usage

## Related Documentation

---

- [Overview of Spreadsheet Studies](#)
  - [Using Spreadsheet Studies](#)
  - [Spreadsheet Systems, Alerts and Automated Trading](#)
  - [Spreadsheet Study Inputs](#)
  - [Referencing Other Charts in Spreadsheet Study Formulas](#)
  - [Spreadsheet Studies Special Tasks](#)
  - [Sharing Your Spreadsheet Study With Another User](#)
  - [Working with Spreadsheets](#)
  - [Spreadsheet Functions](#)
  - **Spreadsheet Example Formulas and Usage**
- 

## On This Page

---

- [Introduction](#)
- [Common Formulas and Usage](#)
  - [Determining Slope or Direction](#)
    - [Formula that Matches the Coloring of Up and Down Price Bars](#)
    - [Formula that Matches the Slope Coloring of a Study Subgraph Line](#)
  - [Moving Averages](#)
    - [Simple Moving Average](#)
    - [Exponential Moving Average](#)
    - [Simple Moving Average - Skip Zeros](#)
  - [Using Date Time](#)
    - [Formula Equivalents of Serial Date Time Values](#)
    - [Rounding Date Time Values](#)
- [Trading Formulas](#)
  - [Disable Trading From Spreadsheet Study for Trading](#)
    - [Disable Autotrading Outside of Regular Trading Hours](#)
    - [Disable Trading Outside of Regular Trading Hours and for 2 Hours Midday](#)
  - [Cancel Unfilled Working Order n Minutes After Entry](#)
  - [Calculate Daily Net Closed Profit/Loss After Commissions](#)
- [Persistent and Incrementing Variables](#)
  - [Formula to Count Bars Since An Event](#)

- [Cancel an Unfilled Order After n Bars](#)
- [Cancel Unfilled Orders After n Bars - Reset Counter on New Order in Same Direction](#)
- [Buy on First Signal and Ignore Multiple Signals](#)
- [Count Events Since First Occurrence](#)
- [Data Manipulation](#)
  - [Removing Floating Point Imprecision](#)
  - [Spreadsheet Study Equivalents of Simple Alerts and Main Price Graph Identifiers](#)
  - [Other Examples](#)
    - [Return Last Two Zig Zag Reversal Values](#)
    - [Using the FORECAST Function](#)

---

## Introduction

---

This page provides some common formulas to be used with the [Spreadsheet Studies](#) to accomplish specific tasks and explains the particular usage in those cases.

The examples on this page are formatted for entry into the **Spreadsheet Study**, the **Spreadsheet Study for Trading**, or the **Spreadsheet System/Alert**. The concepts in the formulas can be applied to the **Spreadsheet Formula** using the [Spreadsheet Study Equivalents of Simple Alerts and Main Price Graph Identifiers](#) section. For example, if the entry in the **Spreadsheet Study** is `=B3 - E3` (Open - Last/Close), then this would be entered in the **Spreadsheet Formula** as `=O - C` or alternatively it could be entered as `=ID0.SG1 - ID0.SG4` where ID0 is the main price graph.

---

## Common Formulas and Usage

---

### Determining Slope or Direction

---

#### Formula that Matches the Coloring of Up and Down Price Bars

This example formula would be entered in any available Formula Column at row 3. Although the formula is coded to rely on it being entered in the T column.

It returns a **1** if price bar is **up-colored** or a **-1** if price bar is **down-colored**.

`=IF(OR(E3>B3,AND(B3>E4,B3=E3)),1,IF(OR(E3<B3,AND(B3<E4,B3=E3)),-1,T4))`

#### Formula that Matches the Slope Coloring of a Study Subgraph Line

This example formula would be entered in any available Formula Column at row 3. Although the formula is coded to rely on it being entered in the Z column.

It returns a **1** if the line is sloped **up**, or if the current segment is flat and the previous segment was up.

It returns a **-1** if the line is sloped **down**, or if the current segment is flat and the previous segment was down.

```
=IF(OR(ID1.SG1@3>ID1.SG1@4,AND(ID1.SG1@5>=ID1.SG1@4,ID1.SG1@3>ID1.SG1@4)),1,IF(1,Z4))
```

**ID1** needs to be changed to the particular study ID number it will actually reference. **SG1** needs to be changed to the particular Subgraph number within the particular study it will reference. For more information, refer to [References to Study Subgraph Columns](#).

## Moving Averages

### Simple Moving Average

Any of the following formulas will return a Simple Moving Average of the Last price, where the Length is in cell H2. A formula would be entered into row 3 of any Formula Column.

```
=AVERAGE(OFFSET(E3,0,0,$H$2,1))
```

```
=AVERAGE(OFFSET(ID0.SG4@3,0,0,$H$2,1))
```

```
=SUM(OFFSET(E3,0,0,$H$2,1))/ $H$2
```

```
=AVERAGE(E3:INDEX(E3:OFFSET(E3,$J$30-1,0),$H$2,1))
```

### Exponential Moving Average

This formula gives an Exponential Moving Average of the Last/Close price. The Length is defined in cell **H2**. The formula would be entered in cell P3, a Formula Column.

```
=(E3 - P4) * (2 / (1 + $H$2)) + P4
```

### Simple Moving Average - Skip Zeros

This set of formulas gives a Simple Moving Average skipping over any zero values. The Length is defined in cell **H2**. The input data is in column **P**, and the output is in column **S**.

- Cell Q3: 

```
=P3 > 0
```
- Cell R3: 

```
=SUM(OFFSET(Q3, 0, 0, $H$2, 1))
```
- Cell S3: 

```
=SUM(OFFSET(P3, 0, 0, $H$2, 1)) / R3
```

## Using Date Time

The following sections contain information on how to work with Date and Time values in the Spreadsheets.

Internally, Sierra Chart Spreadsheets store the Date and Time in the same manner as Microsoft Excel. As such, the Date is stored as the number of integer days since January 1, 1900. The hours are then the decimal portion of the number and represent the fractional amount of the day. For example, the Date and Time for August 18, 2018 at 16:32:00 would be 43330.68889. This also means that a value of 1 represents 1 full day.

## Formula Equivalents of Serial Date Time Values

1 millisecond: `=1 / 86400000`

1 second: `=1 / 86400`

1 minute: `=1 / 1440`

1 hour: `=1 / 24`

1 day: `=1`

Simple Alert: Bar at Specific Time: `BARTIME = FRACTIME(A3)`

Simple Alert: Bar at Specific Date: `BARDATE = INT(A3)`

Simple Alert: Bar at Specific Date and Time: `BARDATETIME = A3`

## Rounding Date Time Values

This section gives examples of how to round the Date and Time to achieve a desired result.

Round to the Nearest Millisecond to remove floating point imprecision to the nearest millisecond: `=MROUND(A3, 1 / 86400000)`

Round to the Nearest Second to remove floating point imprecision to the nearest second: `=MROUND(A3, 1 / 86400)`

Round Down to the Nearest Second to remove any millisecond increments: `=FLOOR(A3, 1 / 86400)`

Round Down to Midnight of the referenced date: `=FLOOR(A3, 1)`

## Trading Formulas

### Disable Trading From Spreadsheet Study for Trading

The following sections give examples of formulas to be used to control when trading can occur with the **Spreadsheet System for Trading** study.

### Disable Autotrading Outside of Regular Trading Hours

The following formula disables the **Spreadsheet System for Trading** study from creating new open positions outside of the specified times. The **Spreadsheet System for Trading** can still exit from Open Positions during all hours with this formula.

This formula should be entered into cell **J28**.

```
=AND(J8 = 0, OR(FRACTIME(J41) < TIMEVALUE("09:30:00"), FRACTIME(J41) > TIMEVALUE("16:00:00")))
```

In this example, it will allow the **Spreadsheet System for Trading** to open new Positions between the hours of 09:30:00 and 16:00:00. Change these values in the formula as needed.

## Disable Trading Outside of Regular Trading Hours and for 2 Hours Midday

The following formula disables the **Spreadsheet System for Trading** study from creating new open positions outside the specified times along with disabling a period in the middle of the day from 11:30:00 to 13:30:00. The **Spreadsheet System for Trading** can still exit from Open Positions during all hours with this formula.

This formula should be entered into cell **J28**.

```
=AND(J8 = 0, OR(OR(FRACTIME(J41) < TIMEVALUE("09:30:00"), FRACTIME(J41) > TIMEVALUE("16:00:00")), AND(FRACTIME(J41) > TIMEVALUE("11:30:00"), FRACTIME(J41) < TIMEVALUE("13:30:00"))))
```

In this example, it will allow the **Spreadsheet System for Trading** to open new positions between the hours of 09:30:00 and 11:30:00 and again between 13:30:00 and 16:00:00. Change these values in the formula as needed.

## Cancel Unfilled Working Order n Minutes After Entry

This section lists a set of formulas to be entered in the **Spreadsheet System for Trading** that will automatically cancel an unfilled order some number of minutes after the close of the entry signal bar.

Adjust the value in Cell H4 as needed to change the number of minutes to wait between the Close of the Entry Bar and Cancelling the Order.

- **Cell Y3:** `=IF(AND (K5 = 0, K4), FRACTIME(A3), Y4)`

This formula returns the closing timestamp of a Buy Entry signal in Column K.

- **Cell Z3:** `=IF(AND(M5 = 0, M4), FRACTIME(A3), Z4)`

This formula returns the closing timestamp of a Sell Entry signal in Column M.

- **Cell H4:** `3`

This is the number of minutes to wait before canceling an unfilled working order.

- **Cell J27:** `=OR(AND(J4 > 0, FRACTIME(J41) - Y3 >= H4 / 1440), AND(J4 < 0, FRACTIME(J41) - Z3 >= H4 / 1440))`

This formula will cancel an unfilled working order 3 minutes after the close of the

entry bar. If bars are not time-based, it can occur intrabar.

## Calculate Daily Net Closed Profit/Loss After Commissions

---

The following formulas give ways to calculate the net Profit or Loss and how to display that information when using the **Spreadsheet System for Trading**.

Calculate the net Profit or Loss including Commissions, not including any Open Positions:

- **Cell H2:** Enter the round-trip commission cost.
- **Cell H45:** `=J45 - J53 / 2 * H2`

Alternatively, the following formulas will calculate the net Profit or Loss including Commissions, and including any Open Positions, once they are closed.

- **Cell H2:** Enter the round-trip commission cost.
- **Cell H45:** `=J10 + J45 - (ABS(J8) + J53) / 2 * H2`

If desired, the results of these formulas can be displayed on the Chart by using the **Text Display for Study** study using these settings:

- Spreadsheet System for Trading **Cell R3:** `=IF(ROW()=3, $H$45, 0)`
- Text Display for Study **Study and Subgraph to Display:** Spreadsheet System for Trading - **R (SG8)**

## Persistent and Incrementing Variables

---

The following sections describe ways to create Persistent Variables or Incrementing Variables that can then be used for specific purposes.

### Formula to Count Bars Since An Event

---

The following formula will count the number of bars since a Buy Entry event occurred (cell K3). This will also stop counting if a Sell Entry event occurs (cell M3). This could be used to cancel an unfilled Order at some number of bars.

The formula below would be entered in cell P3.

`=IF(M3, 0, IF(OR(K3, P4 > 0), P4 + 1, P4))`

The above formula can also be reversed so that it counts bars since a Sell Entry event (M3) and stops counting if a Buy Entry event occurs (cell K3). The formula below would be entered in cell Q3.

`=IF(K3, 0, IF(OR(M3, Q4 > 0), Q4 + 1, Q4))`

### Cancel an Unfilled Order After n Bars

---

The following formulas will cancel an unfilled order after some number of bars have passed. This

builds on the formulas from **Count Bars Since an Event** above, which are repeated here.

These formulas only reset the count when the opposing event occurs (a Sell event for a Buy Position, or a Buy even for a Sell Position).

The number of bars (n) is a value stored in cell H4.

These formulas check at the **close** of a bar because row 4 is referenced in columns P and Q.

- **Cell P3:** `=IF(M3, 0, IF(OR(K3, P4 > 0), P4 + 1, P4))`  
This counts the bars since a Buy Entry event occurred (see **Formula to Count Bars Since An Event** above).
- **Cell Q3:** `=IF(K3, 0, IF(OR(M3, Q4 > 0), Q4 + 1, Q4))`  
This counts the bars since a Sell Entry event occurred (see **Formula to Count Bars Since An Event** above).
- **Cell J27:** `=OR(AND(P4 >= H4 + 1, J4 > 0), AND(Q4 >= H4 + 1, J4 < 0))`  
This cancels the unfilled orders after n bars.

## Cancel Unfilled Orders After n Bars - Reset Counter on New Order in Same Direction

---

The following formulas will cancel unfilled orders after some number of bars have passed. This builds on the formulas from **Cancel an Unfilled Order After n Bars** above, which are repeated here. In this case, however, the counting of the bars will reset if a new Order is placed.

The number of bars (n) is a value entered in cell H4.

These formulas check at the **close** of a bar because row 4 is referenced in columns P and Q.

- **Cell P3:** `=IF(BuyZeroingEvent, 0, IF(OR(K3, P4 > 0), P4 + 1, P4))`  
This counts the bars (in cell P4) since a Buy Entry event occurred (see **Formula to Count Bars Since An Event** above). The term **BuyZeroingEvent** should be replaced by an appropriate formula that will reset the count to 0.
- **Cell Q3:** `=IF(SellZeroingEvent, 0, IF(OR(M3, Q4 > 0), Q4 + 1, Q4))`  
This counts the bars (in cell Q4) since a Sell Entry event occurred (see **Formula to Count Bars Since An Event** above). The term **SellZeroingEvent** should be replaced by an appropriate formula that will reset the count to 0.
- **Cell J27:** `=OR(AND(P4 >= H4 + 1, J4 > 0), AND(Q4 >= H4 + 1, J4 < 0))`  
This cancels the unfilled orders after n bars.

Alternatively, the following formulas can be used to simply reset the count to 1 on a new order in the same direction, but does not reset otherwise.

The number of bars (n) is a value entered in cell H4.

- **Cell P3:** `=IF(K3, 1, P4 + 1)`  
This counts the bars in P4 since a Buy Entry event occurred.
- **Cell Q3:** `=IF(M3, 1, Q4 + 1)`  
This counts the bars in Q4 since a Sell Entry event occurred.

- **Cell J27:** `=OR(AND(P4 >= H4 + 1, J4 > 0), AND(Q4 >= H4 + 1, J4 < 0))`

This cancels the unfilled orders after n bars.

## Buy on First Signal and Ignore Multiple Signals

---

The following set of formulas creates a Buy order per a set of Buy Conditions and then ignores additional Triggers until the corresponding Sell Order is placed.

- **Cell P3:** `=IF(BuyConditions, 1, IF(SellConditions, 0, P4))`  
In the above formula **BuyConditions** would be replaced with a desired Buy formula, and the **SellConditions** would be replaced with a desired Sell formula.
- **Cell K3:** `=AND(P4 = 0, P3)`
- **Cell M3:** `=AND(P4, P3 = 0)`

## Count Events Since First Occurrence

---

The following set of formulas will count the number of events that occur since a first event triggered the start of the count.

In this example the Spreadsheet System for Trading is used and the number of Buy Entry events is counted since a Sell Entry event occurred. Additionally, the number of Sell Entry events is counted since a Buy Entry event occurred.

There is also a limit set in cell H4 that will prevent any Buy or Sell trades from occurring if the count is larger than this value. This can be used to limit the number of open positions

- **Cell P3:** `=IF(M3, 0, IF(K3, P4 + 1, P4))`  
This counts the number of Buy Events from cell K3 and is reset if a Sell Event is detected in cell M3.
- **Cell Q3:** `=IF(K3, 0, IF(M3, Q4 + 1, Q4))`  
This counts the number of Sell Events in cell K3 and is reset if a Buy Event is detected in cell K3.
- **Cell K3:** `=AND(BuyEntryConditions, P4 < $H$4)`  
This defines the Buy Entry conditions, but limits the ability to enter the position if the counter in cell P4 is not less than the value in H4. In the above formula **BuyEntryConditions** would be replaced with a desired Buy Entry formula.
- **Cell M3:** `=AND(SellEntryConditions, Q4 < $H$4)`  
This defines the Sell Entry conditions, but limits the ability to enter the position if the counter in cell Q4 is not less than the value in H4. In the above formula **SellEntryConditions** would be replaced with a desired Sell Entry formula.
- **Cell H4:** Enter a value to limit the number of Buy or Sell events.

## Data Manipulation

---

The following sections give formulas and information on how to manipulate the data that is being used in the Spreadsheet Studies.

### Removing Floating Point Imprecision



## REMOVING FLOATING POINT IMPRECISION

The following formulas can be used to remove floating point imprecision from various items in the Spreadsheet Studies.

1. Remove Floating Point Imprecision of the Tick Size in cell J21. Place this formula in cell H21:
  - `=1 / ROUND(1 / J21, 0)`
2. Remove Floating Point Imprecision of a Calculated Value to a Multiple of the Tick Size (see above formula):
  - `=MROUND(CalculatedValue,$H$21)`  
In this case the CalculatedValue could be any formula or another cell.
3. Remove Floating Point Imprecision of the OHLC Values to a Multiple of the Tick Size (see above formula):
  - `=MROUND(B3, $H$21)`
  - `=MROUND(C3, $H$21)`
  - `=MROUND(D3, $H$21)`
  - `=MROUND(E3, $H$21)`

## Spreadsheet Study Equivalents of Simple Alerts and Main Price Graph Identifiers

The following table gives different ways to reference data, or to calculate values.

Common Name	Main Price Graph Identifier	Spreadsheet Equivalent	Alt
Date of the Bar	BARDATE	<code>=INT(A3)</code>	
Time of the Bar	BARTIME	<code>=FRACTIME(A3)</code>	
Date and Time of the Bar	BARDATETIME	<code>=A3</code>	
Bar Open	O	<code>=B3</code> or <code>=ID0.SG1@3</code>	

Bar High	H	=C3 =ID0.SG2@3	or
Bar Low	L	=D3 =ID0.SG3@3	or
Bar Close/Last	C	=E3 ID0.SG4@3	or
Bar Volume	V	=F3 =ID0.SG5@3	or
Open Interest (Historical Only)	OI	G3 =ID0.SG6@3	or
Number of Trades (Intraday Only)	NT	=G3 =ID0.SG6@3	or
Average of Open, High, Low, and Close	OHLC	=AVERAGE(B3, C3, D3, E3)	
Average of High, Low, and Close	HLC	=AVERAGE(C3, D3, E3)	
Average of High and Low	HL	=AVERAGE(C3, D3)	

Bid Volume	BV		Ad Vo stu ref SG
Ask Volume	AV		Ad Vo stu ref SG
Open Value for Renko	ID0.SG22		Ad Re Vis Op Va stu ref SG
Close Value for Renko	ID0.SG23		Ad Re Vis Op Va stu ref SG
Daily High	DAILYHIGH	=J17	
Daily Low	DAILYLOW	=J18	

Last Trade	LASTTRADE		
Last Trade Size	LASTTRADESIZE		
Bid Price	BID	=J12	
Ask Price	ASK	=J14	
Previous Day Close	PREVIOUSDAYCLOSE		Ad Da stu ref SG 'Re Da to
Tick Size	TICKSIZE	=J21	
Volume At Price	VAP		Se Vol Pri to the Sp Stu
Bid Volume at Price	BVAP		

Ask Volume at Price	AVAP		
---------------------------	------	--	--

## Other Examples

This section contains examples of formulas to perform tasks that do not fit into the other categories.

### Return Last Two Zig Zag Reversal Values

The following set of formulas can be used with the Spreadsheet Study in order to obtain the values of the three most recent Zig Zag reversals. The concept can be continued on to obtain however many Zig Zag reversal points are desired.

This example assumes that the Zig Zag study is ID1 and that the Zig Zag Input Option for **Additional Output for Spreadsheets** is set to **Yes**.

- **Cell O3:** `=IF(ID1.SG3@3 > 0, ID1.SG1@3, O4)`  
This sets cell O3 to the Price of the most recent Zig Zag reversal point.
- **Cell P3:** `=IF(ID1.SG3@3 > 0, O4, P4)`  
This sets cell P3 to the Price of the second most recent Zig Zag reversal point.
- **Cell Q3:** `=IF(ID1.SG3@3 > 0, P4, Q4)`  
This sets cell Q3 to the Price of the third most recent Zig Zag reversal point.

### Using the FORECAST Function

The spreadsheet **FORECAST** function returns the Y-Value for a give X-Value based on a simple linear regression of a set of X and Y values. This can be used to predict (forecast) a value in the future based on existing information.

A linear regression takes the existing data and does a best fit straight line to the data. This line can then be extrapolated to determine where the data would be expected to be at some point in the future.

This is only a prediction of the future data based on the existing data. It is not possible to know exactly what will happen as future bars are added to the chart.

The following example uses the Open data for the most recent 60 bars in order to predict the Open value of the 5th bar forward in time.

- **Cell K3:** `=FORECAST(A3 + 5 * (A3 - A4), B3:B62, A3:A62)`

The first parameter of the FORECAST() function is the X value for which the predicted value is

desired. In this case, we want 5 bars in the future. Therefore, we use the difference in time between bars and multiply that by 5. This allows for the timeframe of the chart to change without having to update the spreadsheet. For more information on how time is handled in Sierra Chart Spreadsheets, refer to [Serial Date Time Values](#).

The second parameter of the FORECAST() function is the range of Y values (or outcomes). These are the values for which you want to get the forecasted value. In this case, the Open data (Column B) for the previous 60 bars is used.

The third parameter of the FORECAST() function is the range of X values. These are inputs to the equation that results in the set of Y values. The time values will often be used for this parameter, since the charts are time-based, but it is not required. In this case, the time (Column A) for the previous 60 bars is used. It is important that the number of rows being referenced for both the X and the Y ranges is the same. If there are not the same number of values for both of these parameters then the spreadsheet will return a **#NUM!** error.

---

\*Last modified Wednesday, 22nd February, 2023.