SPREADSHEET BASICS

Setting up estimating and job-cost templates is the first step in learning to use spreadsheets

by Scott Shelley

If I had to choose just one piece of software with which to run my business, it would be the spreadsheet. This is as true today, when I work on a state-of-the-art Pentium machine, as it was when I purchased my first IBM XT clone more than 15 years ago. Over the years, I've used spreadsheets for the usual purposes — estimating, invoicing, and job-costing — and the unusual — figuring cuts for rake walls or hip roofs. I have even heard of a contractor who used a spreadsheet to lay out a tile pattern for a shower.

Most new computers come with software, such as Microsoft Works or Claris Works, that includes a spreadsheet. If you spend a few hours learning how to use a spreadsheet, you'll begin to find ways to put it to work in your business, without having to spend another cent on software.

In this article, I'll use a couple of simple spreadsheets I set up when I first started using a computer to explain how spreadsheets work. But those are just the tip of the iceberg. Once you feel comfortable with these simple examples, you'll find that's it's easy to set up more complex spreadsheets to fit your business.

Rows and Columns

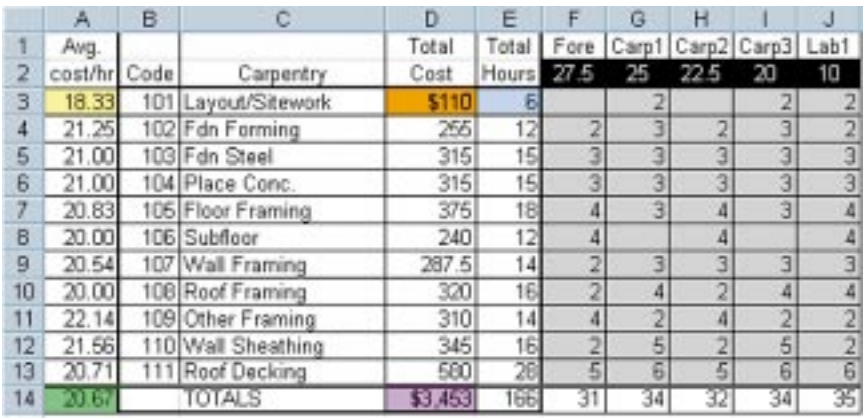
If you have ever used a columnar pad to estimate or do your bookkeeping, then perhaps without realizing it, you've used a manual version of a spreadsheet. Spreadsheet software builds on the row-and-column structure of a paper columnar pad, adding automated shortcuts and other features that make the job of entering and manipulating data easier.

The biggest advantage a spreadsheet has over a paper columnar pad is that a spreadsheet will do all the math for you. Even if you use a calculator to estimate, it's tedious to add up all your labor, material, and subcontractor costs across each row, then total the columns. If you're interrupted midstream, it's easy to forget where you left off. And changes — like a call from the architect telling you to use $6 per square foot for tile instead of $5 — mean you have to start all over again. After all that, your client may still get this strange smile on his face during your presentation, causing you to wonder if your price is $20,000 too low because you made some math error.

A spreadsheet takes the guesswork and the tedium out of all of these situations. A spreadsheet can link individual entries, or "cells," in the rows and columns so that when the value of one cell changes, the values in all linked cells change, too. The process is accurate and it all happens in the blink of an eye.

Estimating

When I first started using a computer in my business, I did almost everything on a spreadsheet. Early on, the most pressing issue was to be able to estimate and feel confident that I had not made any math errors. At the time, I was using my best judgment as to how long each component would take to build, since almost everything we did was custom and I had not yet done enough job-costing to come up with square-foot



	A	B	C	D	E	F	G	H	I	J
1	Avg.			Total	Total	Fore	Carp1	Carp2	Carp3	Lab1
2	cost/hr	Code	Carpentry	Cost	Hours	27.5	25	22.5	20	10
3	18.33	101	Layout/Sitework	$110	6		2		2	2
4	21.25	102	Fdn Forming	255	12	2	3	2	3	2
5	21.00	103	Fdn Steel	315	15	3	3	3	3	3
6	21.00	104	Place Conc.	315	15	3	3	3	3	3
7	20.83	105	Floor Framing	375	18	4	3	4	3	4
8	20.00	106	Subfloor	240	12	4		4		4
9	20.54	107	Wall Framing	287.5	14	2	3	3	3	3
10	20.00	108	Roof Framing	320	16	2	4	2	4	4
11	22.14	109	Other Framing	310	14	4	2	4	2	2
12	21.56	110	Wall Sheathing	345	16	2	5	2	5	2
13	20.71	111	Roof Decking	580	28	5	6	5	6	6
14	20.67		TOTALS	$3,453	166	31	34	32	34	35

Figure 1. The rows in this labor-estimating template correspond to specific tasks, which are coded and listed in columns B and C. Hours for each employee are entered in columns F through J. The formulas that fill columns A, D, and E (colored cells) calculate totals.

or linear-foot costs for labor. So I set up a "template" with a list of labor items I used regularly to estimate labor for the job (see Figure 1). I used the template as a worksheet to do my estimating and to track my labor costs once the job was underway.

The layout of this template is simple. Columns B and C hold task codes and descriptions, and match those on my company's timesheets. Each of the columns F through J correspond to one of my employees, and the headings of these columns (cells F1 through J1) hold job descriptions, such as "Foreman," "Laborer," or one of several skill levels of "Carpenter." Immediately below (cells F2 through J2) is the hourly wage for each job description. The cells in the other three columns (A, D, & E) hold formulas that perform math calculations on the numbers I enter in each row; the cells in row 14 hold formulas that total the columns.

Adding numbers in rows. When I do a labor estimate, I follow the list of tasks, row by row, entering the number of hours I think each employee will spend on each one into the cells under each job description (columns F through J). To find the total hours estimated for a given task, column E (labeled "Total Hours") holds a formula that adds the hours, row by row, from the job description columns. In cell E3, for example, the formula looks like this:

```
=SUM(F3:J3)
```

If you key this formula in directly, you have to be careful to include the correct symbols in the proper position. You can save time and reduce the chances of making a typo by using one of two other ways to enter the formula. One method is to click on the function button, *fx* which calls up a list of available functions (Figure 2). After choosing SUM from the list, a dialogue box opens and walks you through the formula. During this process, you can use the keyboard or mouse to select the cells you want to add.

Because the SUM function is used so often in most spreadsheets, there's a second, easier way to enter the formula: Just click on the SUM button Σ on the toolbar at the top of the screen. This button takes you directly to the dialogue box for the SUM function.

Adding numbers in columns. The SUM function works with columns of numbers, too. In my labor estimating template, for example, the formula in cell D14 looks like this:

```
=SUM(D3:D13)
```

Like the other formulas in row 14, this formula adds together all of the values in the column above it.

Absolute references. So far, we've looked at formulas that operate on cells based on their position relative to each other. For example, the SUM formula in cell E3 operates on the five cells immediately to the right (F3:J3). If we were to copy this formula into the cell just below (E4), the spreadsheet would automatically change the references from F3:J3 to F4:J4, because these cells are immediately to the right of E4.

Sometimes, however, I need to use an "absolute reference" — a formula that operates on the same cell no matter where on the spreadsheet the formula is located. The formulas in column D of my template use absolute references to calculate the labor costs for each employee and add them together to get the total cost of each task. In cell D3, for instance, the formula first multiplies the number of hours in cell F3 times the hourly rate in cell F2; it does the same for each of the other four cells in the row and adds them all together. One cell down in D4, however, the formula again refers to cell F2, even though cell F2 is not located in the row directly above.

If this formula used relative references, it would look like this:

```
=(F3*F2)+(G3*G2)+(H3*H2)+(I3*I2)+(J3*J2)
```

Using absolute references, it looks like this:

```
=(F3*$F$2)+(G3*$G$2)+(H3*$H$2)+(I3*$I$2)+(J3*$J$2)
```

The dollar signs indicate that the formula should reference this cell no matter where the formula is located on the spreadsheet.

Naming cells. You have to be careful when copying formulas that use absolute references from one cell to another. Most spreadsheets assume that all references are relative, so you have to manually go into each copied formula and add the dollar signs. An easy way to work around this problem is to give the absolute reference cells a name. In my template, for example, the cells that hold wage amounts are named "Foreman," "Carpenter 1," Carpenter 2," and so on. Using these names, the formula in cell D3 looks like this:

```
=(F3*Foreman)+(G3*Carpenter1)+(H3*Carpenter2)+(I3*Carpenter3)+(J3*Laborer)
```

When I copy it down one cell to D4, the names of the absolute references stay the same:

```
=(F4*Foreman)+(G4*Carpenter1)+(H4*Carpenter2)+(I4*Carpenter3)+(J4*Laborer1)
```

If-Then Statements

In my template, column A (labeled "Avg. cost/hr") holds a formula that divides total cost by total hours. In its simplest form, the formula would look like this:

```
=D3/E3
```

The problem with this formula, and with others that use division, is that any time column E is empty or contains a zero, the formula automatically considers this a "divide-by-zero" error. Instead of calculating a number, the formula puts the message #DIV/0! (or ERR! in some versions) in column A. When clients see this — and I know plenty of contractors who have shown spreadsheets to their clients with ERR! all over the place — they begin to wonder about the accuracy of the rest of the numbers.

Fortunately, there's a simple fix using an "if-then" statement. (In Excel, this is called the IF function.) The formula in cell **A3** looks like this:

```
=IF(E3=0,"0",D3/E3)
```

The formula argument (the part within the parentheses) has three parts, separated by commas: a "logical test," a "value if true," and a "value if false" (Figure 3). In plain language, the if-then statement first "tests" to see if the value of cell E3 is zero; if it is, the formula will display "0" (or whatever message you type between the quotes); if the value is not zero, then the formula divides D3 by E3 and displays the result. Using this IF function, when the value in Column E is zero, "0" appears in column A instead of "ERR!."

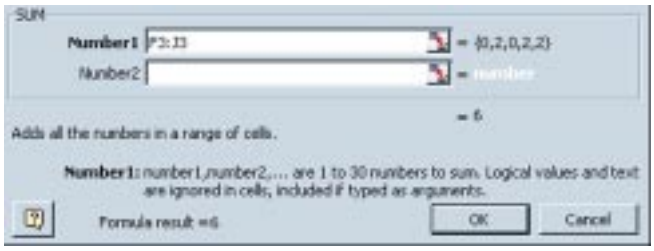


Figure 2. The easiest way to enter a formula is to pick from a list, such as this one in Excel (left). Clicking on the SUM function, for example, opens a dialogue box (below), which walks you through the formula.

Job-Costing

To estimate accurately, you have to do some sort of job-costing. If you are not using actual costs to update your estimating system, then you are missing a big opportunity to become a more confident estimator. In my company, I want to know on a day-to-day basis where I stand on each phase of every job. If we have gone over budget on the foundation, for example, I want to know as soon as possible so I can work with my people in the field to make up the overrun someplace else in the job.

You don't need accounting software to job-cost; instead, you can set up a spreadsheet that keeps track of all of a job's invoices, bills, paychecks, and so on, along with a code to indicate which phase of work these costs belong to. For the first four years I was in business, this was the way I kept track of job expenses. The spreadsheet I used was accurate, and it allowed

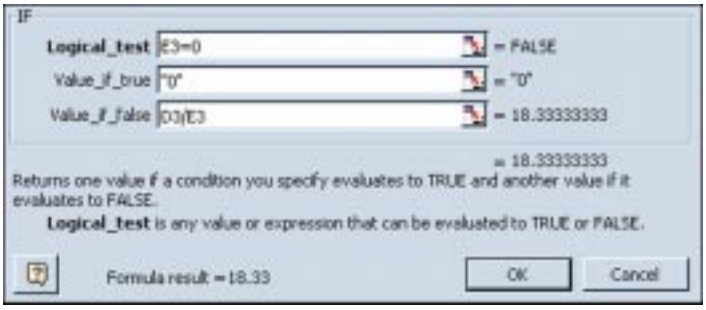


Figure 3. When a cell is left blank, a spreadsheets interprets it as a zero. In formulas that use division, this can cause divide-by-zero error messages. To solve the problem, use an if-then statement. The built-in IF function has three parts, separated by commas: a logical test, a value if true, and a value if false. In this example, if the argument E3=0 is true, the cell holding the formula will display a zero (or any other text typed between the quote marks). If the argument is false, the cell will display the result of D3 divided by E3. Without this if-then formula, when cell E3 is zero, the spreadsheet will display an error message.

	A	B	C	D	E	F	G	H
1				Cost	Calc'd %	Calc'd	Est'd %	Est'd
2	Code	Category	Budget	to Date	Complete	Over/Under	Complete	Over/Under
3	1	Sitework	700	653.18	93%	46.82	100	46.82
4	2	Foundation	244	238.27	98%	5.73	95	(1.08)
5	3	Carpentry	345	356.12	103%	(11.12)	100	(11.12)
6	4	Lumber	567	562.68	99%	4.32	100	4.32
7	5	hardware	567	523.23	92%	43.77	100	43.77
8	6	Windows & Skylights	5677	5589.74	98%	87.26	95	(119.68)
9	7	Doors & Locksets	544	575.46	106%	(31.46)	100	(31.46)
10	8	Electrical Base	3777	3500	93%	277.00	93	280.66
11	9	Electrical Fixtures	2656	2875.62	108%	(219.62)	100	(219.62)
12	10	Plumbing Base	56	56	100%	0.00	100	0.00
13	11	Plumbing Fixtures	456	460.21	101%	(4.21)	100	(4.21)
14		TOTALS	15589	15390.51	99%	198.49	98	(1.70)
15								
16		=IF(G3=0,"",IF(G3=100,F3,F3-((D3/(G3/100))-C3)))						
17								

Figure 4. In this job-cost spreadsheet, column E calculates percent complete by dividing the cost-to-date (column D) by the budget (column C). But this calculation doesn't bear any relationship to actual progress on the job.

The solution is two additional columns: one (G) holds a percent complete estimate; the other (H) uses an if-then statement (yellow cell) to determine whether the job will be over or under budget. The formula combines two if-then statements, one inside the other. The first handles the divide-by-zero problem; the second calculates dollar amounts.

me to have as many categories as I liked. An abbreviated version is shown in Figure 4.

The simplest way to use this job-cost spreadsheet is to use only columns A through E. The "Code" and "Category" columns are the same as those I use for estimates, and I enter amounts into the "Budget" column by hand from my estimate. As invoices and timesheets come into the office, I enter total amounts in column D. Column E holds a formula that calculates the percent complete for each category — it simply divides the Cost-to-Date by the Budget. Column F tells me how much money is left in the budget for each category — it subtracts Cost-to-Date from the Budget.

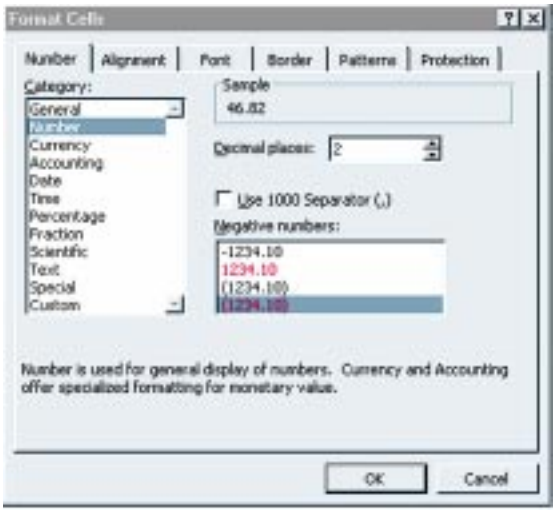


Figure 5. Negative numbers are easier to see if they are displayed and printed in red. This setting can be changed on the format menu, which can also be used to display dollar signs, percentage signs, and time or date notations, among other information.

These five columns work well as a quick comparison of estimated costs to actual costs. But the calculated percentage in column E is based on dollars budgeted and dollars spent — it doesn't bear any relationship to actual progress of the work on the site. For example, in row 5, the spreadsheet tells me that I've spent about $11 more for carpentry than I had budgeted. This is no big deal if all of the carpentry work is done. But if there's still some carpentry work to be finished, I could overrun my carpentry budget by more than $11.

Nested if-then statements. Columns G and H tie the spreadsheet to what's actually happening on the job site. I enter the numbers in column G based on site visits and discussions with the crew about how far each category of work has progressed on the job. Column H holds a formula that uses these estimated percent complete numbers to predict whether we'll come in under budget or run over.

The formula is intimidating at first glance, because it "nests" one if-then statement inside another:

```
=IF(G3=0,"",IF(G3=100,F3,F3-((D3/(G3/100))-C3)))
```

Everything between the first and second "IFs" is necessary to handle the divide-by-zero problem. The seemingly complex series of equations after the second "IF" actually perform a couple of simple calculations. The first part `G3=100,F3` looks at the percentage I've entered and if it's 100 (meaning that phase of the job is complete), it uses the number from column F to show how much we went over or under budget. If the percentage I've entered in column G is less than 100, however, that means work in that category is still going on. The second part of this formula `F3-((D3/(G3/100))-C3` predicts how much money will be left for each category if work proceeds at its current pace.

By looking at the dollar amounts in column H, I can tell whether any category is running over budget. To make this easier to see, I've set the "Format" menu on the spreadsheet toolbar (Figure 5) to display negative numbers in red. As long as my estimate of percent complete is fairly accurate, I can use this spreadsheet at any stage of construction to evaluate progress and make adjustments to bring the job back in line with my budget.

Scott Shelley *owns Scott C. Shelley Construction, a remodeling company in Petaluma, Calif.*