CIRCUIT CELLAR ONLINE

LESSONS FROM THE TRENCHES

George Martin

And The List Goes On…

Putting Together Parts Lists

With promises of fame, George tries to entice you into submitting an entry to Trinity's Fire Fighting Home Robot Contest. But before you build that bot, you're going to have to buy the parts. It may seem like a daunting task at times, but with everything you do, you need the parts to do it. The simplest way to go about it is to create a list and a purchase order. From there, you can start cashing those winning checks.

Let's imagine for a moment that you've just finished watching the championship round of BattleBots and decided to get off the couch and build a robot for the next round of competition. In the back of your mind you know you should be building one for Trinity's annual Fire Fighting Home Robot Contest (and I stress "should be"), but the fame, glory, and TV hype has gotten the best of you. You have some great design and control ideas stored up in the recesses of your mind, and they're just begging to come out. You're sure that after your design is seen on TV, the orders will start rolling in. And of course, there's the added bonus that you'll be famous. Well, let me give you another piece of the puzzle to help you manage the astronomical growth ahead.

At some point in your career as a designer (either work or play), you are going to have to buy parts. The simplest way to get that done is to have a list of the parts you require. So this month, let me take you from a simple spreadsheet bill of material (BOM) solution to a fairly complicated manufacturing requirements planning (MRP) solution.

PHASE 1—IN THE BEGINNING

It all begins with a small project—a project so small that you could probably keep all the parts requirements in your head and even remember the price and supplier names and phone numbers. If you're using a CAD system for schematics and artwork, that system will produce a BOM. The typical solution at this phase of the project is to use a spreadsheet, which has columns provided for:

- item number
- part name
- part number
- description
- quantity required
- reference designators
- manufacturer
- manufacturer part number
- distributor
- distributor part number

This basic information lets you manage the parts list and purchasing efforts. And, as you see fit, you can add other columns. For example, I usually add the part shape information (i.e., part surface mount 0805 or through-hole 0.5″ lead spacing). In the distributor's name column, I also put the catalog number and page from which am ordering so I can easily resurrect the source information when necessary.

Add a column for the purchase order (PO) number, and you can see which parts are on order. If you enter the total number of assemblies to be built at the top of the spreadsheet, you can calculate the total number of parts required and the total cost. You can also account for scrap and add other requirements such as assembly and test labor or shipping costs on this spreadsheet. In the end, the result is a fairly useful itemized list.

I have even taken this approach one step further and expanded the column for the quantity required. Whenever I

have several assemblies that make up one unit and they have common parts, I have a column for each assembly and then add up the requirements in a total column. So you can see that the spreadsheet method is easily extendible, and seeing that most people have access to spreadsheet software, it is a common approach to constructing a parts list.

PHASE 2—CREATING A DATABASE

As the size and complexity of the project develops, the next problem you'll encounter is controlling the amount of dollars committed. You might get a great deal when pricing the component, but eventually you'll have to pay for those parts. And, you'll probably have to pay for them before you assemble, test, and ship the final product. Of course, you can schedule parts deliveries over a duration of time in order to spread out the cash requirements, but your accounting manager (perhaps your spouse) will still want to know how much money is needed and when.

The controlling document for this mess is a purchase order (PO) where you can list the parts ordered, delivery dates, shipping information, billing information, tax information, and any other pertaining information.

The step that ties the PO to the parts list can be done using the software you have on hand and is probably best done using a database program. I built a simple system using Borland's Paradox database software. However, this software is old and DOS-based, and on top of that, I'm running out of computers that can run the code. So, I've switched to a new system, but I'll go over that later.

I built a simple database consisting of three files—Vendor, PO, and Parts Requirements. The Vendor file contains the ID (a quick and easy name and a key to link the files), name, address, and telephone and fax numbers of the vendor. The PO file consists of the PO number and date, vendor ID, whether or not the part is taxable, terms, and special instructions. And finally, the Parts Requirements file contains the part name, number, and description, the PO number, item number, quantity, delivery date, project, and manufacturer.

Note that the key fields are used to link the files. Read your database textbook if you've forgotten how it works. If this is a new concept for you, get an introductory text. You can easily make a nightmare of a database, so tread carefully.

MANIPULATING THE FILES

The Parts Requirements file is used as the main file for entering and manipulating information. I usually take the spreadsheet data and enter a build quantity, which supplies me with the total requirements. Then, I import the data into the Parts Requirements file. I now have open requirements. One of the vendors I can use is MY INVENTORY. And I do just that, take parts from inventory.

When you need to issue a PO, make an entry in the PO file taking the next number and enter the vendor and any other pertaining information in that PO file. Then in the Parts Requirements file, enter the PO price and project information. After all parts are linked for a PO, print the PO and send it out. As you can see, you now have a database that can show you items not yet ordered, orders at each vendor, the total project orders amount, and fairly sophisticated reports such as the cost or parts due each week. This way you can keep track of whether or not you are over a particular vendor's credit limit and how much cash is needed each week to pay for the parts.

As with the spreadsheet solution, one real benefit is that whenever you need to make a second build, all the information is in the database and it's just a matter of making adjustments by pressing the right keys.

I've also done this Phase 2 implementation using BASIC with Parts List, Open Orders, Kit, and Inventory files. You get a little more control, but there's a lot more effort involved. If your background is in software and you've constructed a database in the past, then it'll be easy for you to create your own custom software. I wouldn't normally recommend this approach, but it's a learning process that will allow you to tune the software to any emerging requirements as your projects grow. Sooner or later, however, the requirements of production will outstrip your ability to support the software.

PHASE 3—AN ENTRY-LEVEL MRP PROGRAM

The MRP program is usually driven from the parts list. These programs typically have all that I've talked about and more. Two of the key features that an MRP program includes are indented bill of materials and master items.

An indented bill of materials is a parts list that groups parts into assemblies or build steps. These groups are to hardware construction what subroutines are to software construction. I once worked with a group of people on a printed circuit test set that had over 2000 line items in total but only about 1200 unique items. The indented bill of materials software saved our hide. We entered each assembly and could produce a total for how many 1K resistors or ¼ to 20 ½ pan head screws we needed. We could also see if parts from different assemblies were or could be made with one common part. The power of the indented bill of materials is obvious. It helps you divide and conquer.

Master items is also an indispensable feature. Let's step back and discuss what you're purchasing. Classical thinking identifies purchased parts as either specification or source-controlled. Source-controlled parts are items such as a microprocessor. If you're using an Intel 80C196KC, you are going to use whatever Intel ships you. You could, of course, reject parts that don't function properly, but the bottom line is that you are controlling the source of the part.

Specification-controlled items are typically specified by a drawing you have created. Only parts that conform to that drawing will be accepted. Printed circuit boards or custom sheet metal front panels are standard spec-controlled items.

There are other classifications such

as artwork or processes for parts, but they are refinements of the basics of source- and spec-controlled.

The benefit of master items is that, when you create the part number, you also add all (and I repeat, all) the supporting information. Information such as whether it's source- or spec-controlled, where to purchase, pricing, and price breaks can all be contained in the database for your master items. Also, different projects with the same requirements should all use the same master items program. Using master items also gives you the benefit that, after the item is entered into the database, the documentation process is complete. This becomes a project milestone that you can use to identify the parts lists as complete.

ADDED BENEFITS

But wait, there's more. Using these systems, you get to identify sub-assemblies, build and test those assemblies, and put them into inventory. That way, when you need to build a complete machine, you can pull the sub-assemblies. Also, tasks like developing kits and generating labels for the kits and inventory become automatic. The particular software may not give you all that a custom solution can offer, however, the purchased solutions typically use a production database so you are still able to get in and design your own reports and operations.

One of the reasons for implementing this level of software is to meet the requirements of regulatory agencies. FCC and UL stipulations require you to control the manufacturing process and build each unit identical to the ones you submitted for test approval. An MRP program will go a long way in meeting these requirements.

There are advertisements for two vendors for this type of software in *Circuit Cellar* each month, Xelos and Parts and Vendors. Both have free downloads that run in a demo mode. I am using Parts and Vendors V.4. It's much better than my old DOS solutions. It doesn't have all the features I build into my programs, but I do have access to the underling database, so if there is something I require, I can have it.

I'm not an expert in this area, so I don't feel comfortable recommending either of the products. However, I will say that I'm satisfied with my P&V V.4. Trilogy has a news group that answers even the dumbest of questions in a day or so.

One area you shouldn't forget to consider is that, every couple of years or so, you'll want to upgrade the software and there is some cost involved.

PHASE 4—LOOKING FORWARD

Well, Phase 4 is one area that I haven't figured out for myself yet, but I'll bet it has something to do with the Internet. You can trust that when I do work it out, I'll have something to say about it, and you'll be on the top of my list to share it with.

George Martin began his career in the aerospace industry in 1969. After five years at a real job, he set out on his own and cofounded a design and manufacturing firm. Typical systems that George designs include servo-motion control, graphical input and output, data acquisition, and remote control. George is a charter member of the Ciarcia Design Works Team and most recently, he's been working on the people-tracking system for Bill Gates' new house. You can reach him at george.martin@worldnet.att.net

SOFTWARE

Parts and Vendors V.4
Trilogy Design
(800) 280-5176
(530) 273-1985
Fax: (530) 477-9106
www.trilogydesign.com

Xelos
(866) 615-9290
www.xelosproducts.com