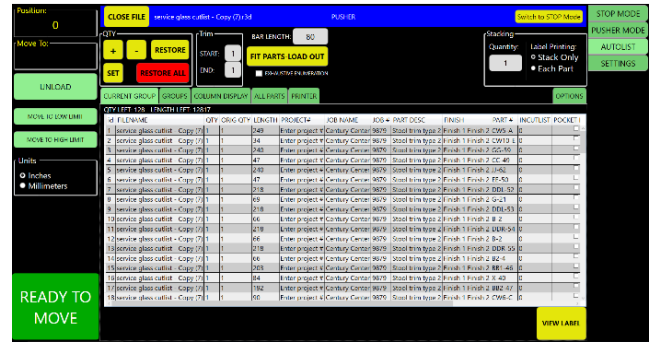




RG3 AutoList Cut List Considerations



RazorGage software requires a comma delimited file in tabular format. There is no requirement for a specific file extension. We get files with extensions like .csv, .txt, .rzg, .dat and more. The only requirement is that the data is arranged in rows with fields separated by either commas or semicolons with each row having the same number of separators. Headings are allowed. We can skip any number of rows at the top of the file but once we get past the skipped rows, the rows must be uniform. It is important to note that values between the separators must not contain a separator as part of the data. For example, it's not uncommon for someone to enter Smith, Inc. as a customer name that gets placed in a comma delimited file. That comma in the customer name will disrupt the reading of the file because now that row will have an extra comma.

Our mapping software, RG3 Mapper, is used to create a mapping definition file called RG3Mapping.db that tells the RG3 software how many rows to skip, where the important fields are, fields by which the data is to be grouped, which part descriptions are to be filtered out, and more. If the file contains headings, the user has the choice to accept the existing headings or create different headings to be displayed at the machine.

The RG3Mapping.db file can either be placed in the folder from which the operator will be opening the files or it can be placed in the folder C:\Cutlists Custom. If the RG3Mapping.db file is placed in C:\Cutlists Custom then any file opened from any folder will have the rules from the RG3Mapping.db file stored in C:\Cutlists Custom applied. If a customer has multiple file formats, as might be the case if they have multiple software suites used for generating cut lists, then a separate RG3Mapping.db file can be created for each format. Then the customer would place the appropriate RG3Mapping.db file in each folder containing the various file formats. If a folder doesn't have an RG3Mapping.db file then the software defaults to the one in C:\Cutlists Custom. When the operator opens a file it is converted to a database (.r3d) and the delimited file is deleted.

When the operator opens a file on the RG3 software at the machine, it first converts the file to an .r3d file, ignoring line items with part descriptions deemed invalid by the RG3Mapping file. It will then sort the line items into groups according to the sorting criteria stored in the RG3Mapping file.

Pre-Optimized Cut Lists

When cut lists have been “pre-optimized” by a third-party software suite it is important that the pre-optimized cut list has a column denoting the “board number”. Some companies call this a “layout number”. For example, the optimizing software decides that parts a, b, c, & d will be applied to board 1, that e, f, g, h, & j will be applied to board 2 and so on. We use that number to group the parts by “board” or “layout” so that the machine runs the parts assigned to certain board just as the optimizing software planned. A column for board length (stock length) is helpful as well because we can designate that field as STOCK LENGTH during mapping and then when an operator selects a group, we can send the pusher to the appropriate load position for the designated stock length. This also helps the operator know what length stock to select in order to ensure that all the parts the optimizing software intended are cut from the selected stock.

When the RG3 software creates a pusher run from a group it will rearrange the parts so that the shortest parts in the group are in the beginning of the run. This is to minimize the likelihood of a very short part ending up at the end of the push run. When the last part on the run is very short, it can make it difficult to clamp. So, while the RazorGage will ensure that the cutting plan from the optimizer software is followed, it does not cut the parts designated for a certain piece of stock in the same order on that stock.

It is also important that certain machine parameters are synchronized with the optimizing software. For example, the pusher on the RazorGage might be adjusted so that, when it is at the low limit, it is 2" away from the saw blade. That means that every stick is going to have at least 2" of waste at the end. The machine also has leading and trailing trim cut parameters. Those define a fixed amount of waste as well. Kerf also defines fixed waste. The optimizing software should have the ability to set these parameters so that it doesn't create a cutting plan that is impossible to recreate on the machine.