**Automated UVC:**

Even though the overrides are used the variable declaration, pack and unpack needs to be coded manually. There is huge amount of human time and effort is needed while creating the frames from the variables. Python scripting is used to create the script to convert the spreadsheet into the transaction class. The spreadsheet should fill in the following way. Column A: name of the variable, Column B: bit size, Column C: Data type, Column D: pack (Whether variable needs to pack or not) Column E: unpack (Whether variable needs to unpack or not) Column F: Compare (Whether variable needs to compared in scoreboard or not), Column G: rand (Whether variable needs to randomize or not), Column H: FA (Fixed Array) or DA (Dynamic Array), Column I: Factory (Whether variable needs to be registered in factory to use the pack/print/unpack/compare or not) , Column J: This needs to filled only the Column H is DA. Size of the Dynamic array. Possible values are Variable name or fixed value. Column K and L: Constraint details. L is for constraints of dynamic array and for the rest column K is used. If there is more than one constraint is used for a variable, then the constraints are separated by ‘;’. The example of CAN and ethernet spreadsheet is shown in Figure 14 and 15 respectively.

A screenshot of a computer

Description automatically generated

Figure 14: CAN Frame in Spreadsheet

The signal size of the CAN frames are filled in the Column B. CAN frames does not have e\_id and inter\_framr\_gap, so the cell D3, E3, D14 and E14 in the figure 14 are left blank. Only data and eof are mentioned as dynamic array and rest are fixed array. So, the column J8 and J13 has dlc and 7 respectively, which are the size of the dynamic array. The directed values are entered in the column K with integer values and for random values the respective cells in column K are left blank. The directed dynamic data values are entered in column L.

A screenshot of a computer

Description automatically generated

Figure 15: Ethernet Frame in Spreadsheet

Ethernet frames does not have inter\_framr\_gap. So, the cell D8 and E8 in the Figure 15 are left blank. Only eth\_data is mentioned as dynamic array and rest are fixed array. So, the column J6 has length, which are the size of the dynamic array. The directed dynamic data values and their respective constraints are entered in column L.

This spreadsheet is given as an input for python script and the appropriate transaction file can\_txn.sv and eth\_txn.sv is created same as the files in override UVC. To run the python script the command needs to be “python <script\_name.py> <spreadsheet\_name>”. transaction class name is created only from the name of the spreadsheet. So, do not select the same name for both the protocols. The bit size for single bit sized variable and the vector typed variable got created, if the variable is dynamic array, then the symbol needs to be appended. To register in factory Column D, E and F values are used. Different macro is selected for dynamic array while registering in factory. Constraint got coded based on the column J and K. Pack method is coded by the column D and H. Unpack method is done based on the columns E, H and J. The generated basic\_txn is overridden by can\_txn or eth\_txn same as the overrides UVC.