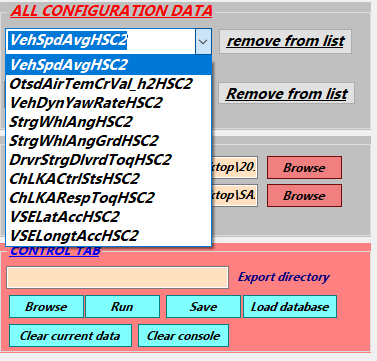
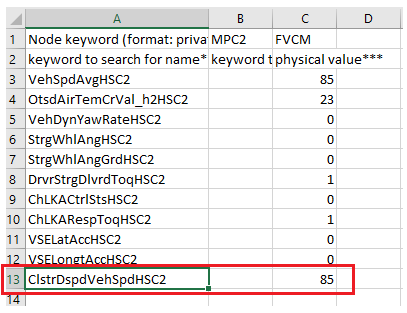
**FUNCTION GENERATION TOOL GUIDELINE R1.2**

|  |  |  |  |
| --- | --- | --- | --- |
| **Releases** | **Date** | **Author** | **Description** |
| **Function\_tool\_R1.1** | **23/10/2020** | **Ha Tien Tai (RBVH-EDA23)** | * **First release** |
| **Function\_tool\_R1.2** | **03/11/2020** | **Ha Tien Tai (RBVH-EDA23)** | * **Synchronize precondition signal list with db\_keyoword** * **Added high-low level for function disabled** * **Add none security option in extended session** * **Enable private can signal for TJA function** |

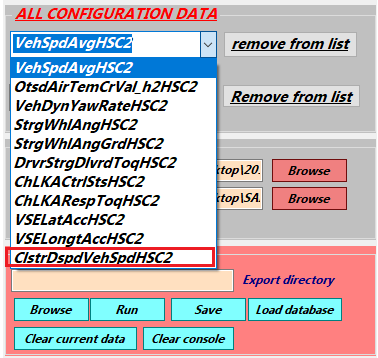
* **Note: This guideline only mentions about update feature. To get the overall step of how to use the tool, please refer to the first gen guideline**
* **Automatically synchronize precondition signal list when updating db\_keyword**
* Now you can update the precondition signal list in db\_keyword file without worrying about losing the one which already exist
* These are the step to do it



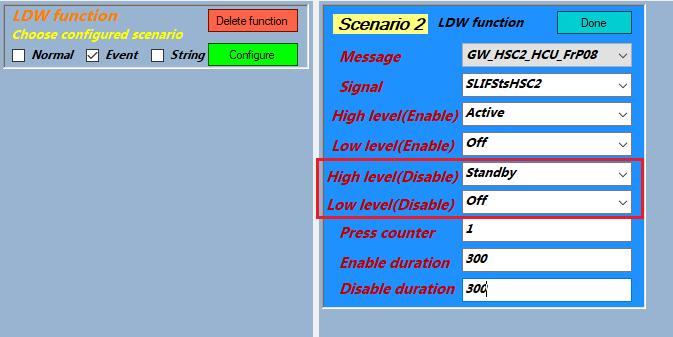
* The precondition list you already had. Add new signal in db\_keyword file



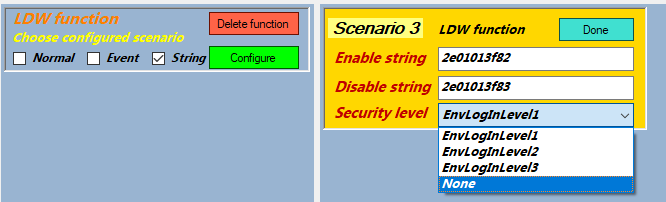
* Save and close it. It will be updated in the list



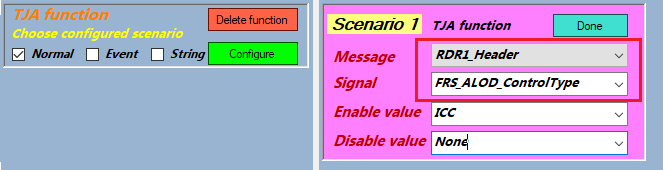
* **Added seperated high-low level for function disabled**
* For some function, the high-low level for enabling and disabling is different. Therefore, every function goes with event scenario will have separate high-low level input textbox



* **Added none-security option when testing function in extended session**
* In some variant, writing DID string to ECU doesn’t require security unlock
* So the None Security option has been added



* **Update some private can signal for TJA function**
* TJA function is the unique case when come to testing. The function requires some signals from radar node to be activated. That why the signal list for this function specifically also includes signal from private can



* FRS\_ALOD\_ControlType is the signal from private can