

Table of contents

- **Object 'Recipe Manager'**

Object 'Recipe Manager'

Object 'Recipe Manager'

Tab 'Storage'

The recipe manager provides functions for maintaining user-defined variable lists, known as recipe definitions. The recipe definitions can be stored in recipe files on the PLC.

“Storage Type”

“Textual”: PLC Engineering saves the recipe in a readable Format with the configured columns and delimiters.

“Binary”: PLC Engineering saves the recipe in a non-readable binary format. This format requires less storage space.

Note: You can read binary recipes again only if you have not changed the variable lists.


“File path”

<directory name>\

Example: AllRecipes\

Path on the runtime system

Notes:

- The path has to end with a backslash (“\”).
- The path is usually a relative path on the target system in the directory of the runtime files (PlcLogic).
- Access to paths outside of the directory PlcLogic is not permitted on every controller. An absolute path for Windows systems can be selected by pressing the  button.

Example of the file path in the runtime system: PlcLogic/AllRecipes

PLC Engineering saves a file in this directory for each recipe when downloading to the PLC. The requirement is that you select the *“Recipe management in the PLC”* option.

The files are loaded to the recipe manager each time the application is restarted.

<i>“File extension”</i>	File extension for the recipe file in the format .<file extension> The resulting default name for recipe files is in the form <recipe>.<recipe definition>.<file extension>.
<i>“Separator”</i>	Delimiters between the individual values in the saved file
<i>“Available Columns”</i>	Defines the information that is saved and in which order in the recipe file
<i>“ Selected Columns”</i>	
<i>“Save as Default”</i>	PLC Engineering uses the settings on the tab throughout the entire project for all other recipe managers.

Tab 'General'

<i>“Recipe management in the PLC”</i>	<input checked="" type="checkbox"/> : Has to be selected for the user program or visualization elements to load recipes at runtime. If you transfer recipes to the PLC exclusively via the PLC Engineering program interface, then you can clear this option.
---------------------------------------	---

“Save Recipe”

<i>“Save recipe changes to recipe files automatically”</i>	When <i>“Recipe management in the PLC”</i> is selected, there is the following option for saving the recipe: <input checked="" type="checkbox"/> : We recommend this option because it helps the recipe manager operate "normally". The recipe files on the PLC are updated automatically at runtime whenever a recipe is changed.
--	---

“Load Recipe”

When *“Recipe management in the PLC”* is selected, there are the following two options for downloading from the PLC:

“Download only for exact match of the variable list” : The recipe is only downloaded if the file on the PLC contains all variables from the variable list of the recipe definition of the application and these are sorted in the same order. Additional entries at the end are ignored. If the required match does not exist, then the error status ERR_RECIPE_MISMATCH is set (RecipeManCommands.GetLastError).

“Download variables with matching names” : The recipe values are downloaded only for those variables that have the same name in the recipe definition of the application as in the recipe file on the PLC. If the variable lists differ in composition and sorting, then no error status is set. In this way, recipe files can also be downloaded if variables in the file or in the recipe definition have been deleted.

“Overwrite existing recipes on download” : If recipe files with the same name exist on the controller, then they are overwritten with the configured values from the project when the application is started. If the values from the existing recipe files should be loaded instead, then this option has to be disabled.

Requirement: The *“Storage Type”* is *“Textual”* and the *“Save recipe changes to recipe files automatically”* option is selected.

“Write Recipe”

The following options are available for writing recipe values to the variables on the PLC:

“Limit the variable to min/max when recipe value is out of the range” : If the recipe contains a value that is outside of the value range specified in the definition, then the defined minimum or maximum value is written to the PLC variable instead of this value.

“Do not write to a variable when the recipe value is out of the min/max range” : If the recipe contains a value that is outside of the value range specified in the definition, then no value is written to the PLC variable. It retains its current value.

“Read Recipe”

The following option is available for reading recipe values from the PLC into the recipe in the project:

“Check recipe for changes”

Always use the function block `RecipeManCommands` from `RecipeManagement.library` to read recipes. Never call the method cyclically. This is because each call can be written to the file system, which is time-intensive and burdens the controller. For example, a Raspberry Pi has a limited number of write cycles.

: With each method call, the current PLC variable values are first read into the recipe. Then the system checks whether the values have changed. Only if the values have changed is the recipe saved. This means that the recipe file is overwritten with the current recipes.

The option can be used in order to update the recipe file in the local file system only if recipe values have changed on the PLC. However, it affects performance because it generates additional code for checking.

: With each method call, the current PLC variable values are first read into the recipe. Then the recipe is written to the recipe file in the local file system.

Note: As the file system is written to each call, the controller can be very burdened.

Recipes during online mode

Option *“Save recipe changes to recipe files automatically”* is selected.

Menu commands	Behavior of the recipes defined in the project	Behavior of the defined recipes at runtime
<i>Online → Reset Warm</i>	The recipes of all recipe definitions are downloaded with the values from the current projects.	Dynamically generated recipes remain unchanged.
<i>Online → Reset Cold</i>		
<i>Online → Download</i>		
<i>Online → Reset Origin</i>	The application is removed from the PLC. If a download is done again afterwards, then the recipes are restored as for an online reset warm.	
Shutdown and restart of the PLC	After a restart, the recipes are downloaded again from the automatically created files. This will restore the same state as before shutdown.	
<i>Online → Online Change</i>	The recipe values remain unchanged. In runtime mode, a recipe can be changed only via the function block command <code>RecipeManCommands</code> .	

Menu commands	Behavior of the recipes defined in the project	Behavior of the defined recipes at runtime
<i>Debug → Stop</i> <i>Debug → Start</i>	The recipes remain unchanged when the PLC is stopped or started.	

Option *“Save recipe changes to recipe files automatically”* is not selected.

Actions	Recipes defined in the project	Recipes defined at runtime
<i>Online → Reset Warm</i> <i>Online → Reset Cold</i> <i>Online → Download</i>	The recipes of all recipe definitions are downloaded with the values from the current projects. However, these are set in the memory only. To save recipes to a file, you have to run the <i>“Save Recipe”</i> command explicitly.	Dynamically generated recipes are lost.
<i>Online → Reset Origin</i>	The application is removed from the PLC. When a download is performed afterwards, the recipes are restored.	Dynamically generated recipes are lost.
Shutdown and restart of the PLC	After the restart, the recipes are downloaded again from the automatically created files. This will restore the same state as before shutdown.	
<i>Online → Online Change</i>	The recipe values remain unchanged. In runtime mode, a recipe can be changed only via the function block command <code>RecipeManCommands</code> .	
<i>Debug → Stop</i> <i>Debug → Start</i>	The recipes remain unchanged when the PLC is stopped or started.	

See also

- ↘ *“Changing values with recipes”*
- ↘ *“‘Read and save recipe’ command”*
- ↘ *“Object 'Recipe Definition’”*