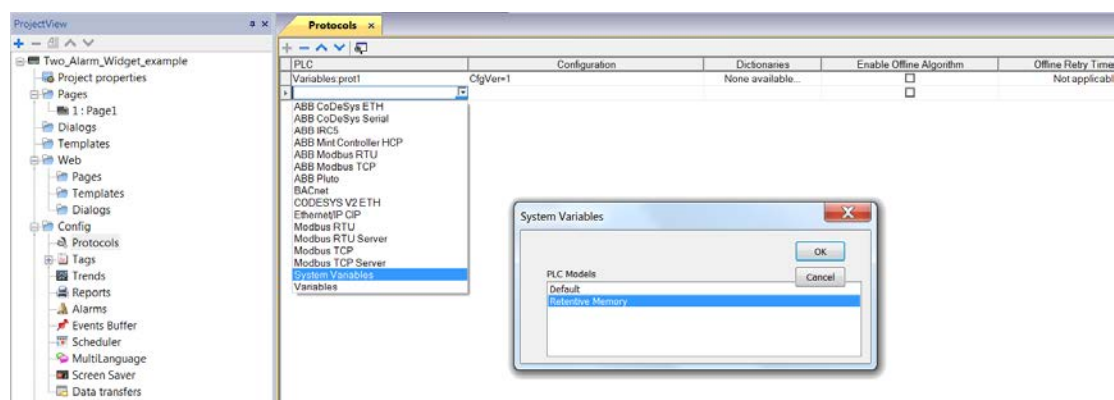
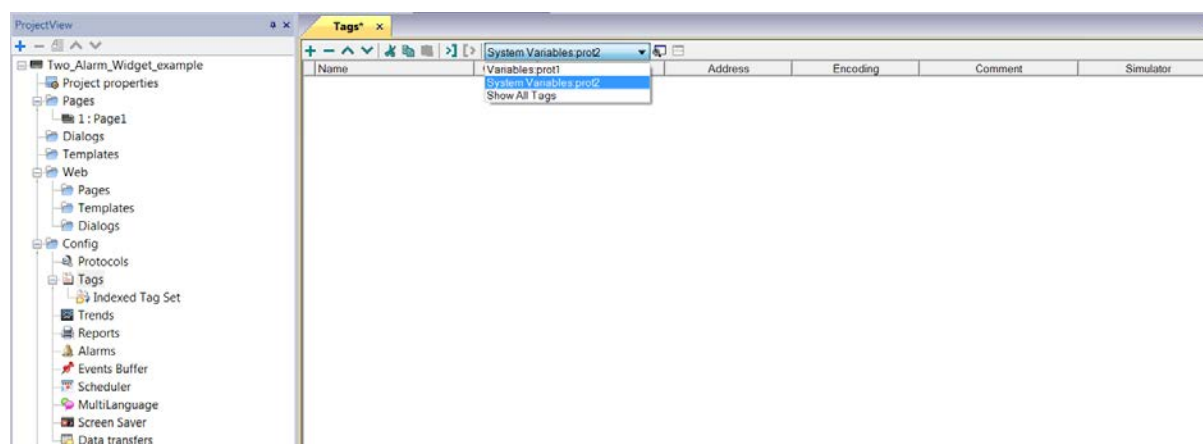




1. INTRODUCTION.....	1
2. REQUIREMENTS	2
3. CREATING A NON-VOLATILE TAG IN PB610 V2.0 (or later).....	3
3.1. As shown in figure 1, below. Double click the Protocol icon in the Config folder to open the Protocol editor.	3
3.2. Click the + icon at the upper-left to create a new protocol, then select System Variables from the drop-down menu in the first column of the newly created protocol (Figure 1, steps 2 and 3).....	3
3.3. When the System Variables dialog opens select Retained Memory and click OK (Figure1, step 4).	3

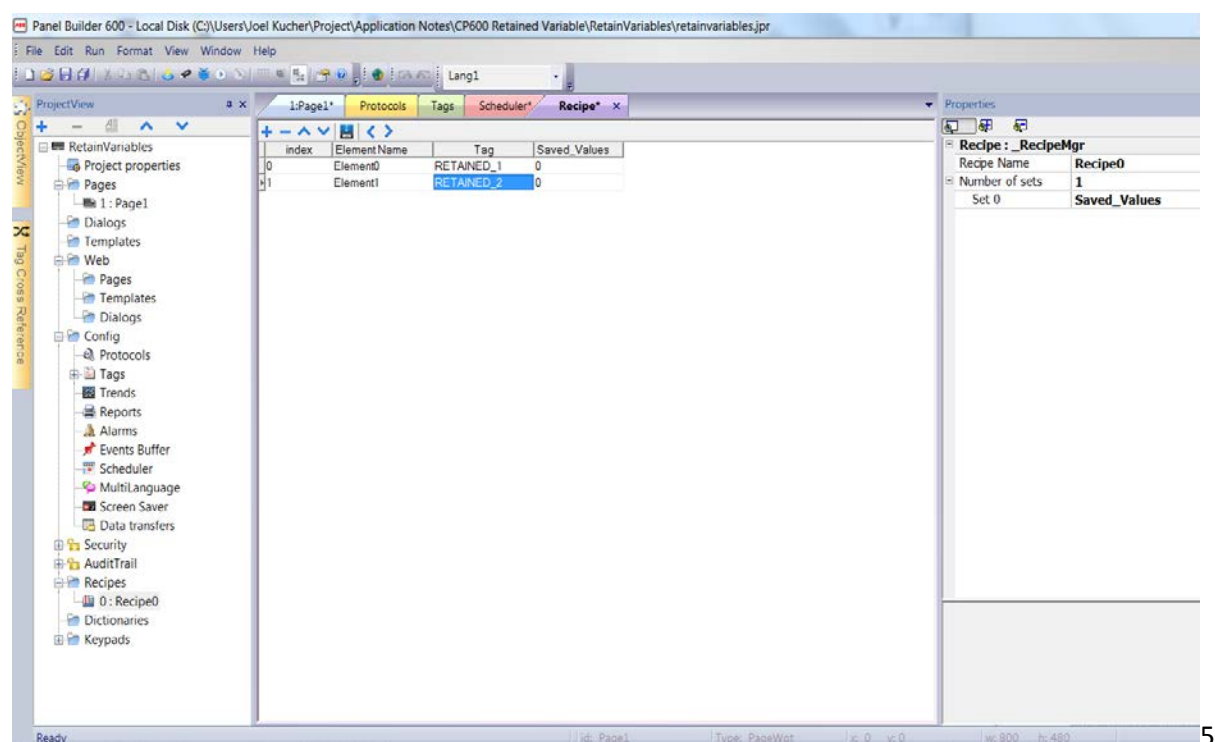


3.4. Double click the Tags icon in the Config folder to open the Tag editor. Select the 'System Variables' protocol that was created in the preceding steps from the drop down menu in the Tags toolbar (Figure 2, step 1 and 2).....	3
3.5. Click the + icon at the upper-left to create a new tag. (Figure 2, step 3).	3



3.6. When the variables dialog opens select the data type and click OK (Figure 3).....	4
3.7. Change the variable Name to something meaningful (Figure 3).....	4
3.8. Repeat steps 3.4-3.7 to create the desired number of tags. Then, skip to step 5 if you wish to create a scheduler to initialize network tags by copying values from the retained variable on power up. Data transfers may also be used for this purpose.	4
4. CREATING A RECIPE (for PB610 v1.95 or earlier).....	5

- 4.1. Up to 32 recipes can be used in a single project and each recipe can contain as many as 32000 elements. So for a relatively few numbers of variables, each can be saved to its own recipe. Multiple variables may be saved to a single recipe as well. In other words, each variable or tag used in the recipe is attached to one element. Each recipe contains one or more elements. To create a recipe, right click on the Recipes folder in the Project View tree and select Add Recipe (figure 1, step 1). Repeat this process to create more than one recipe. Recipes can be renamed if desired. 5
- 4.2. Double click on a recipe to open its tab in the workspace. Next, click the + button at the upper-left for each tag you want to include in the recipe (figure 1, step 2). 5
- 4.3. Set Number of Sets to 1 in the properties pane on the right (figure 1, Step 3). 5
- 4.4. Attach a tag to each row in the recipe column (figure 1, step 4) 5



- 4.5. Select the icon to choose where to save the recipe file, as shown in figure 2. "SD card" is not available for the CP600-eCo series. 6
- 4.6. Save the project. 6
- 4.7. Switch to a project page and place two button controls. Label one "SAVE" and the other "RECALL". 6
- 4.8. Click on the button labeled SAVE and add an OnMouseClicked event. Select UpLoadRecipe() from the action list, then fill in the recipe name and set to be saved. See figure 3. More than one recipe can be uploaded by adding more UpLoadRecipe() actions to the event. 6
- 4.9. Click on the button labeled RECALL and add an OnMouseClicked event. Select DownLoadRecipe() from the action list. Fill in the recipe name and set as in the previous step. 7
- 4.10. A third optional button can be added to clear the retained values. Simply attach the ResetRecipe() action and list any recipe(s) to be cleared in the action properties. 7
5. CREATE A SCHEDULE 7

5.1.	A scheduled action to download the retained values on power-up can be created. To create a schedule, double click the Scheduler object in the Project tree. The scheduler tab will open in the workspace.	7
5.2.	Click + to add a new schedule.	7
5.3.	Click in the schedule column, then click the ellipsis to open the schedule configurator	7
5.4.	Click the Actions Ellipsis and select the UpLoadRecipe() action. In the actions properties select the recipe to be loaded (Figure 4). Create one UpLoadRecipe() action for each recipe.	7
5.5.	Check all three boxes so that the event only executes once at startup and then click OK.	8
5.6.	Save the Project.	8
6.	Download and test the project	8
6.1.	Download the project to the CP600 HMI panel.....	8
6.2.	Change the values of the tags that are to be retained then click the SAVE button.	8
6.3.	Change some of the values and then click the RECALL button. The controls should revert back to the previous (retained) values.....	8
6.4.	Cycle power on the HMI panel.....	8

1. INTRODUCTION

The purpose of this document is to explain how create retained (a.k.a. non-volatile) tags, that is tags whose values are retained when power is removed from the CP600. Panel Builder version 2.0 and later provides a protocol specifically for the creation of non-volatile tags.

Older versions of Panel Builder do not have non-volatile tags. However, recipe tags can be used to store variable values as shown in section 4 of this document. Wherever possible CP600 recipes should not be used to retain data. A PLC, for example is always a better option for retaining data. However, in some instances the CP600 may be the only option for retaining control data. For example, if the panel is connected directly to an ABB variable frequency drive to control speed reference.

WARNING! Using the recipe function to retain values should only be used for operator settings or variables that are not frequently updated. Writing recipe data at a high frequency will burn out the panel's flash memory rendering it useless.

2. REQUIREMENTS

- a) ABB CP600 or CP600-eCo series HMI panel
- b) ABB Automation Builder or Panel Builder software

The following tutorial assumes the reader is familiar with ABB Panel Builder software and aims to show how to create retained variables in an existing project.

3. CREATING A NON-VOLATILE TAG IN PB610 V2.0 (or later)

- 3.1. As shown in figure 1, below. Double click the Protocol icon in the Config folder to open the Protocol editor.
- 3.2. Click the + icon at the upper-left to create a new protocol, then select System Variables from the drop-down menu in the first column of the newly created protocol (Figure 1, steps 2 and 3).
- 3.3. When the System Variables dialog opens select Retained Memory and click OK (Figure1, step 4).

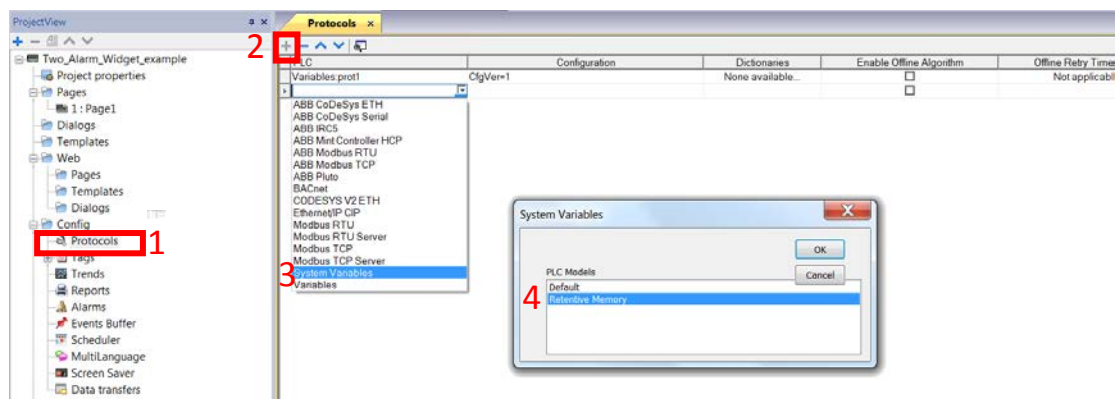


Figure 1 – Creating a System Variable protocol

- 3.4. Double click the Tags icon in the Config folder to open the Tag editor. Select the 'System Variables' protocol that was created in the preceding steps from the drop down menu in the Tags toolbar (Figure 2, step 1 and 2).
- 3.5. Click the + icon at the upper-left to create a new tag. (Figure 2, step 3).

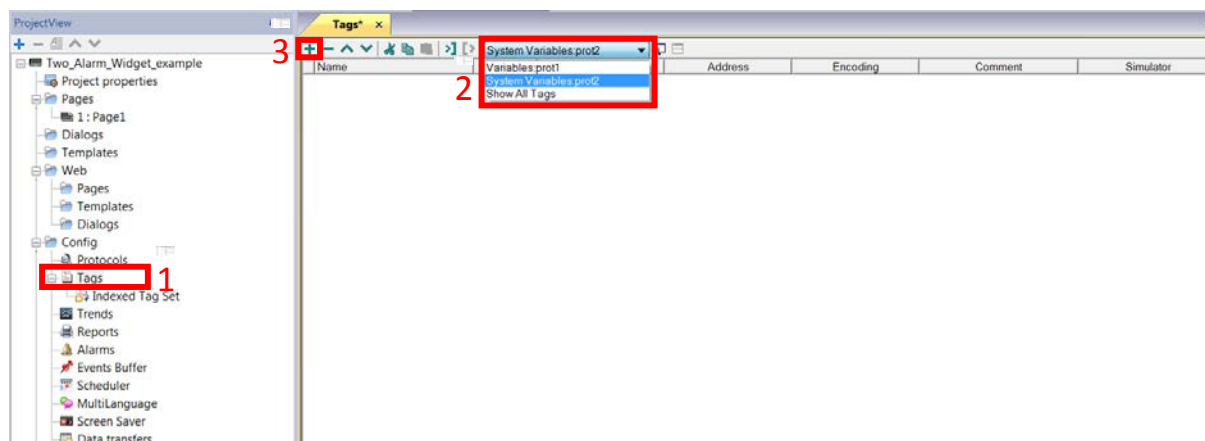


Figure 2 – Creating a new Tag

3.6. When the variables dialog opens select the data type and click OK (Figure 3).

3.7. Change the variable Name to something meaningful (Figure 3).

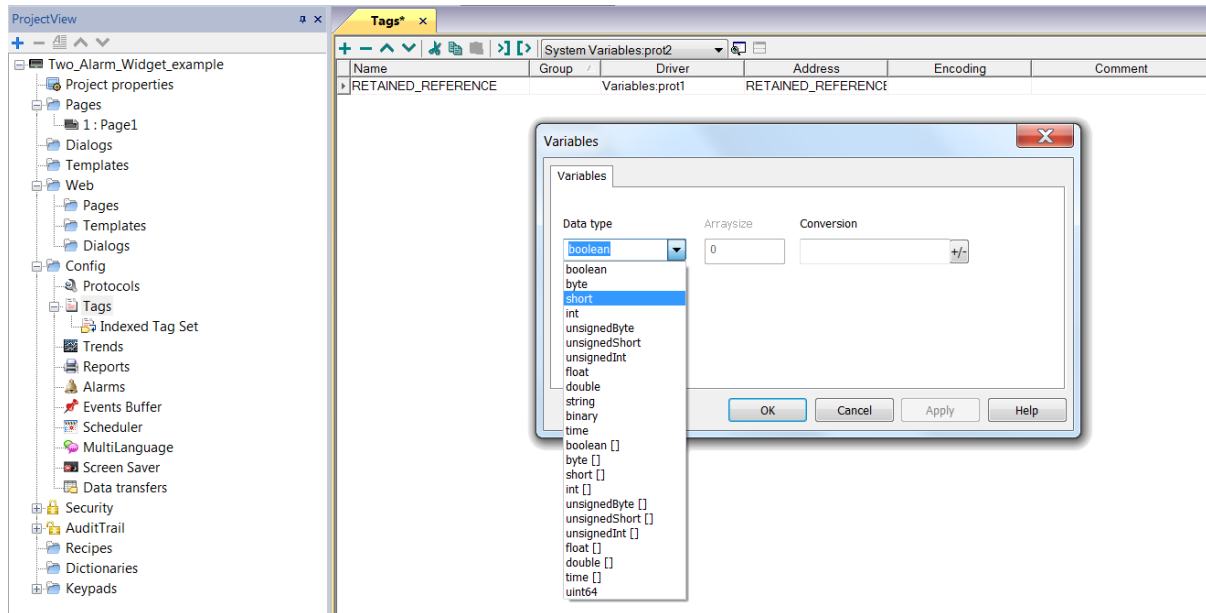


Figure 3- Configuring a tag

3.8. Repeat steps 3.4-3.7 to create the desired number of tags. Then, skip to step 5 if you wish to create a scheduler to initialize network tags by copying values from the retained variable on power up. Data transfers may also be used for this purpose.

Figure 4- Configuring a tag

4. CREATING A RECIPE (for PB610 v1.95 or earlier)

- 4.1. Up to 32 recipes can be used in a single project and each recipe can contain as many as 32000 elements. So for a relatively few numbers of variables, each can be saved to its own recipe. Multiple variables may be saved to a single recipe as well. In other words, each variable or tag used in the recipe is attached to one element. Each recipe contains one or more elements. To create a recipe, right click on the Recipes folder in the Project View tree and select Add Recipe (figure 1, step 1). Repeat this process to create more than one recipe. Recipes can be renamed if desired.
- 4.2. Double click on a recipe to open its tab in the workspace. Next, click the + button at the upper-left for each tag you want to include in the recipe (figure 1, step 2).
- 4.3. Set Number of Sets to 1 in the properties pane on the right (figure 1, Step 3).
- 4.4. Attach a tag to each row in the recipe column (figure 1, step 4)

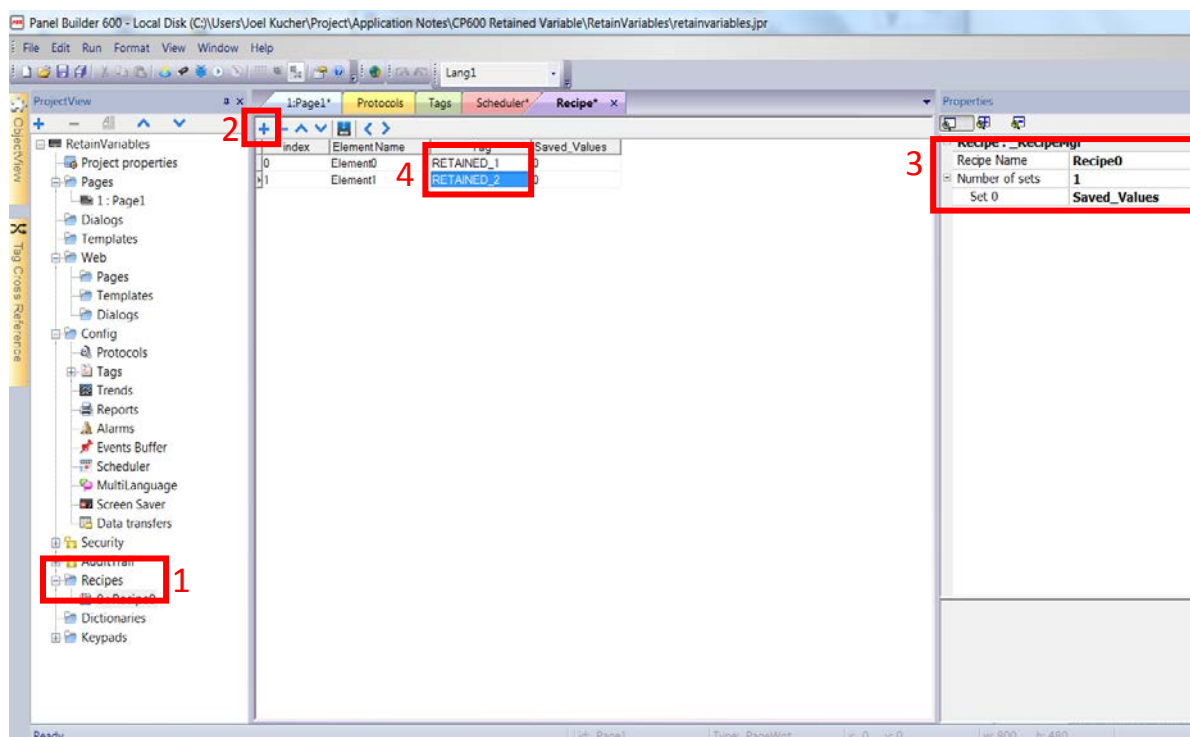


Figure 5- Creating a recipe

- 4.5. Select the icon to choose where to save the recipe file, as shown in figure 2. “SD card” is not available for the CP600-eCo series.

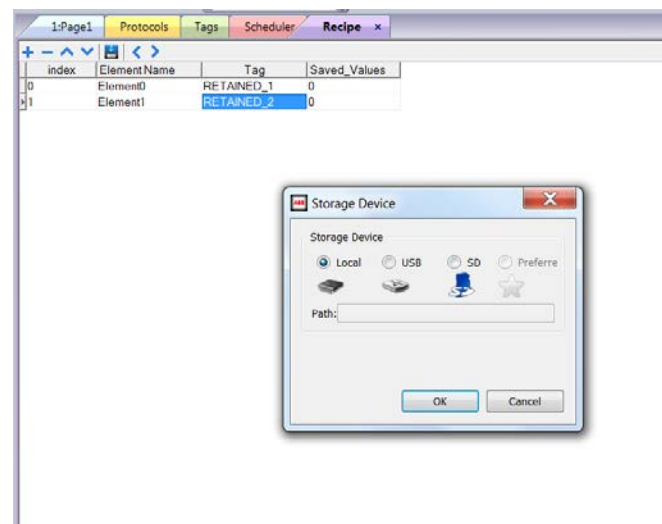


Figure 6- Recipe storage options

- 4.6. Save the project.
- 4.7. Switch to a project page and place two button controls. Label one “SAVE” and the other “RECALL”.
- 4.8. Click on the button labeled SAVE and add an OnMouseClicked event. Select UpLoadRecipe() from the action list, then fill in the recipe name and set to be saved. See figure 3. More than one recipe can be uploaded by adding more UpLoadRecipe() actions to the event.

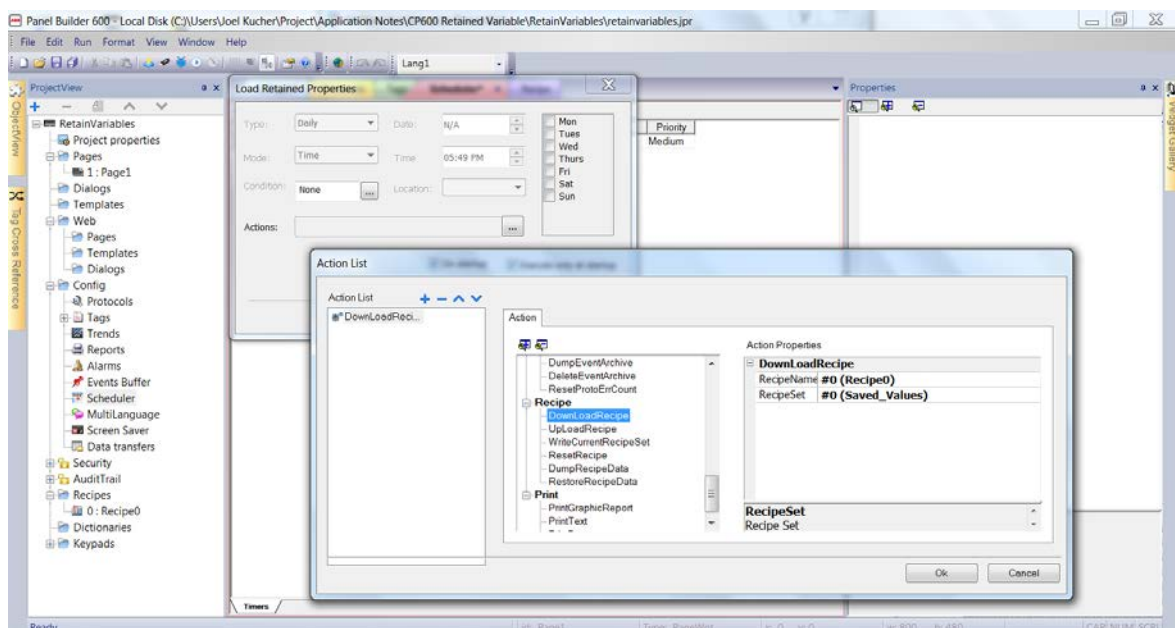


Figure 7- Creating an action to retain values

- 4.9. Click on the button labeled RECALL and add an OnMouseClicked event. Select DownloadRecipe() from the action list. Fill in the recipe name and set as in the previous step.
 - 4.10. A third optional button can be added to clear the retained values. Simply attach the ResetRecipe() action and list any recipe(s) to be cleared in the action properties.
5. CREATE A SCHEDULE
 - 5.1. A scheduled action to download the retained values on power-up can be created. To create a schedule, double click the Scheduler object in the Project tree. The scheduler tab will open in the workspace.
 - 5.2. Click + to add a new schedule.
 - 5.3. Click in the schedule column, then click the ellipsis to open the schedule configurator
 - 5.4. Click the Actions Ellipsis and select the UpLoadRecipe() action. In the actions properties select the recipe to be loaded (Figure 4). Create one UpLoadRecipe() action for each recipe.

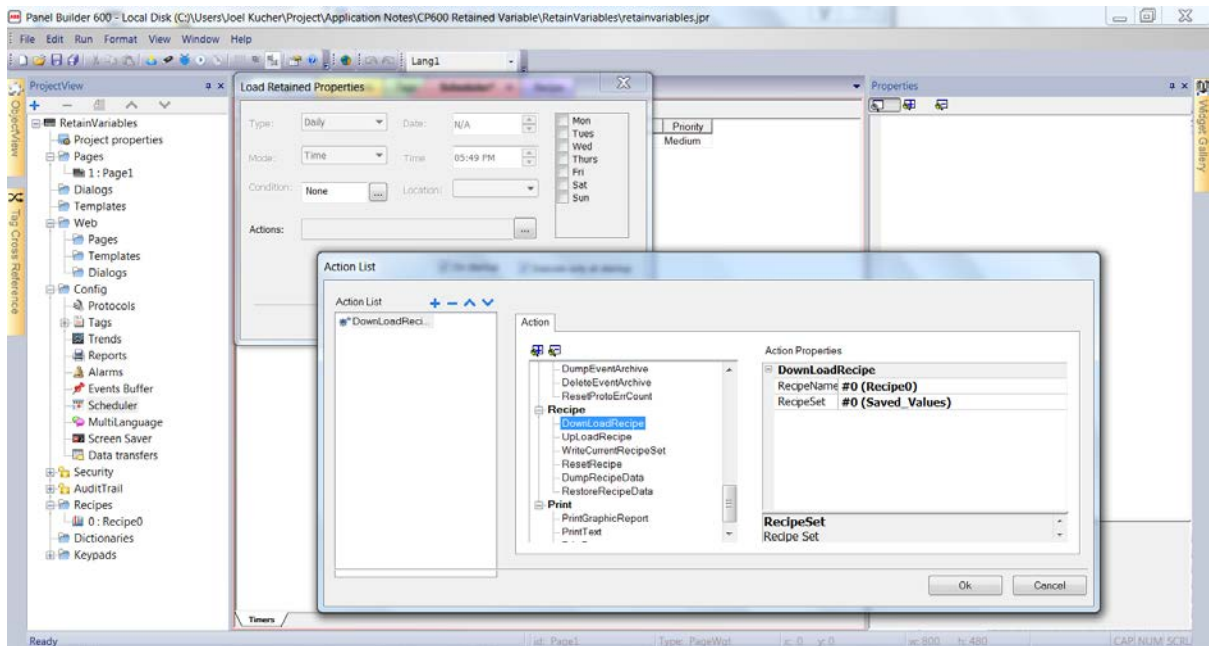


Figure 8- Scheduling an action

5.5. Check all three boxes so that the event only executes once at startup and then click OK.

WARNING!: If all three boxes are not checked the scheduler will execute the action cyclically or never.

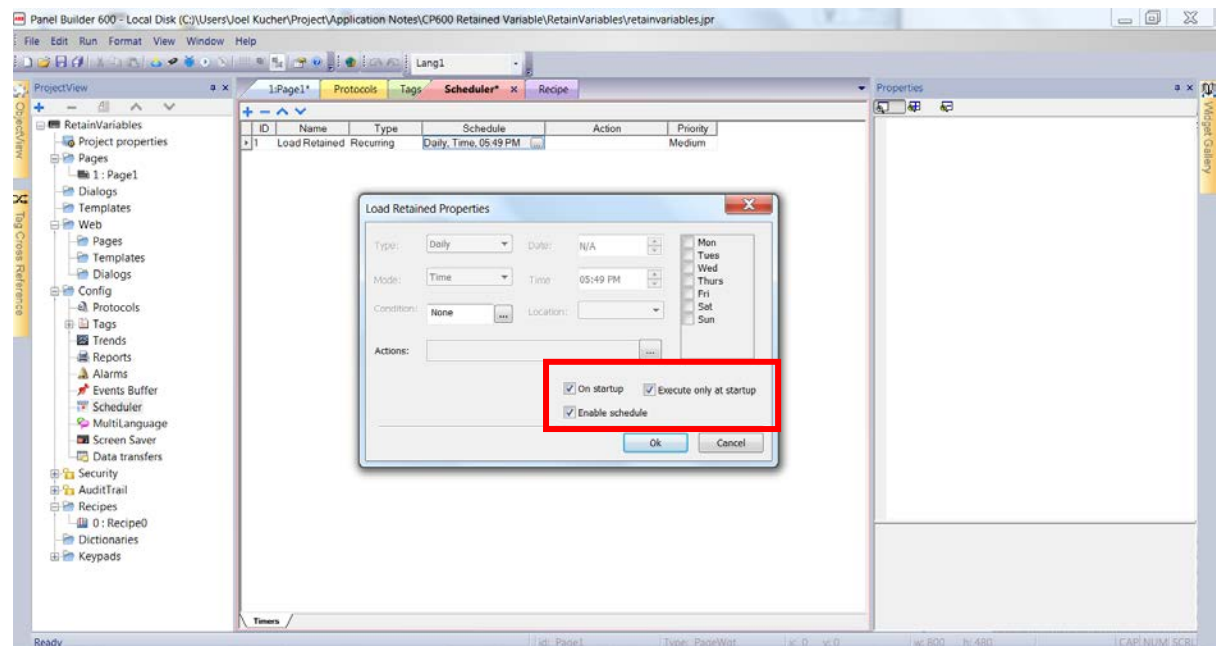


Figure 9- Configure schedule to run once at start-up

5.6. Save the Project.

6. Download and test the project

6.1. Download the project to the CP600 HMI panel

6.2. Change the values of the tags that are to be retained then click the SAVE button.

6.3. Change some of the values and then click the RECALL button. The controls should revert back to the previous (retained) values.

6.4. Cycle power on the HMI panel.

Revision/Version history			
Rev/Ver	Date	Author	Comments
A	20/12/1016	Joel Kucher	Original Document