

## Lab 2: Function Calls

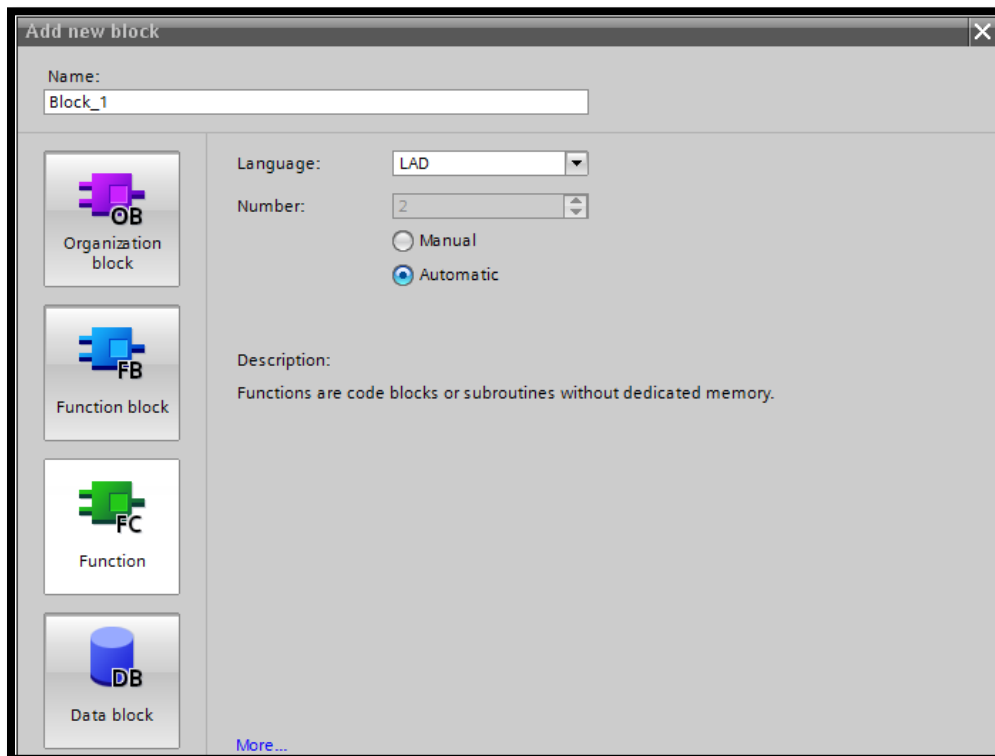
With most of the program finished, we will be branching out to other types of program blocks. Simatic STEP 7 supports four program blocks, as shown below, you are familiar with one of them already, the Organization block (OB). The others are as follow:

**Organization Block (OB):** The main programming block that incorporate networks and components that drive specific tasks. More discussions later.

**Function block (FB):** Create function blocks. More discussions later.

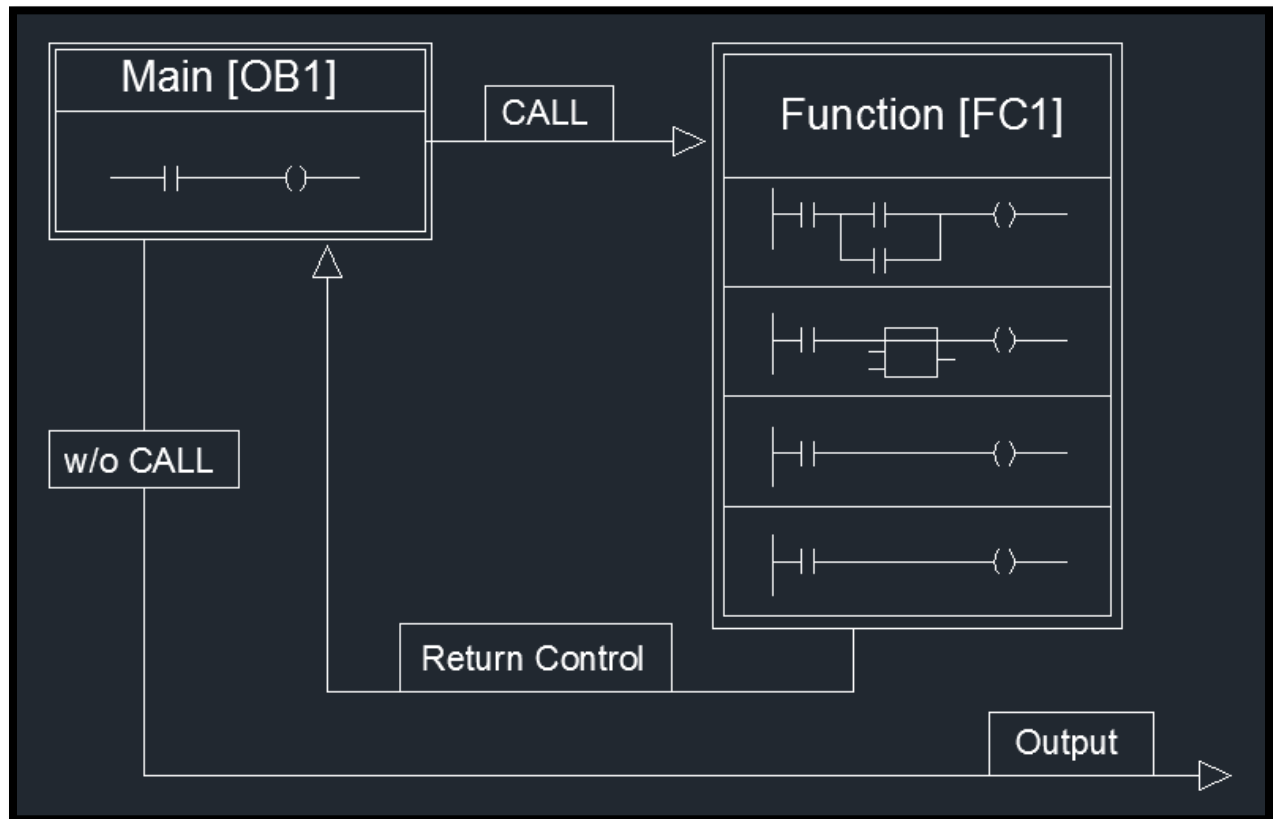
**Function (FC):** A subroutine that perform a specific task, this block is called by the organization block.

**Data block (DB):** An archive where a different set of tags and data are stored; it is used with functions.



**A brief insight on what functions or subroutines are:** In STEP7 programming, a function call has programs written in it, very much like an Organization Block. However, the function will not be executed unless it is called by an organization block. When called, the function will execute all its programs or tasks, then return control to the organization block.

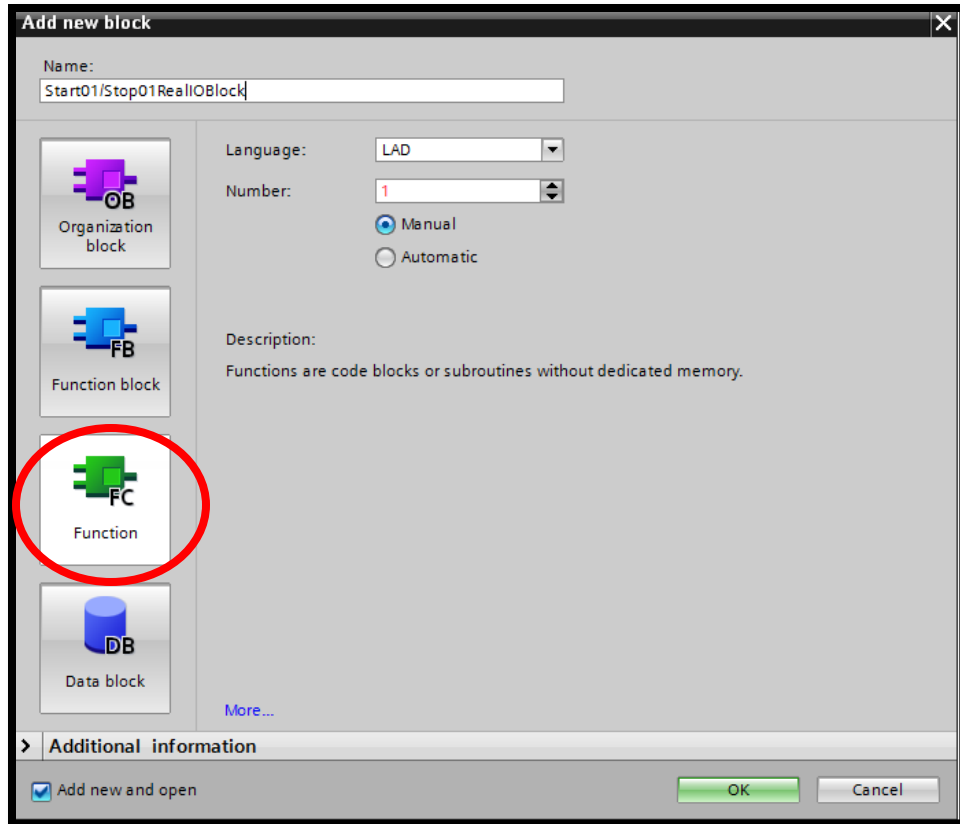
Demonstration can be seen below:



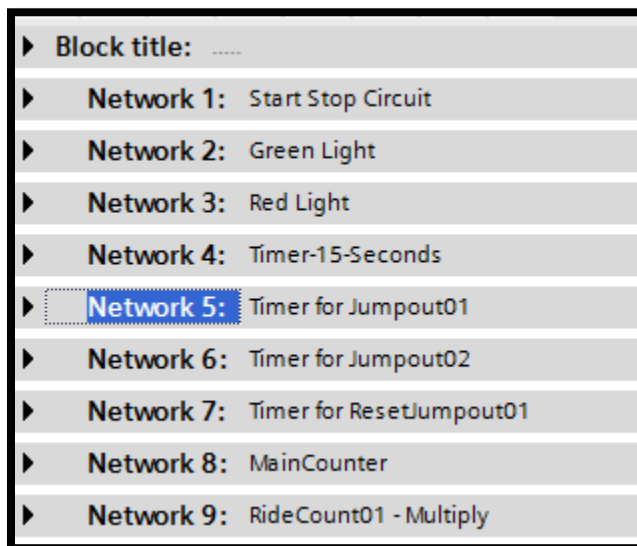
Functions are useful in debugging and organization.

## Creating a function

1. On the left panel, drop down program blocks.
2. Double click add new block.
3. Name it Start01/Stop01RealIOBlock.
4. Click Ok.
5. Repeat these steps but name the function HMIRideControl11 and number it 11.



6. Now that you got your functions created, go double click on Main [OB1].
7. Minimize all your networks, as shown, next hold CTRL and left click all the networks. Right click and cut.



8. Next, double click on the Start01/Stop01RealIOBlock on the left panel, and paste the networks into it.

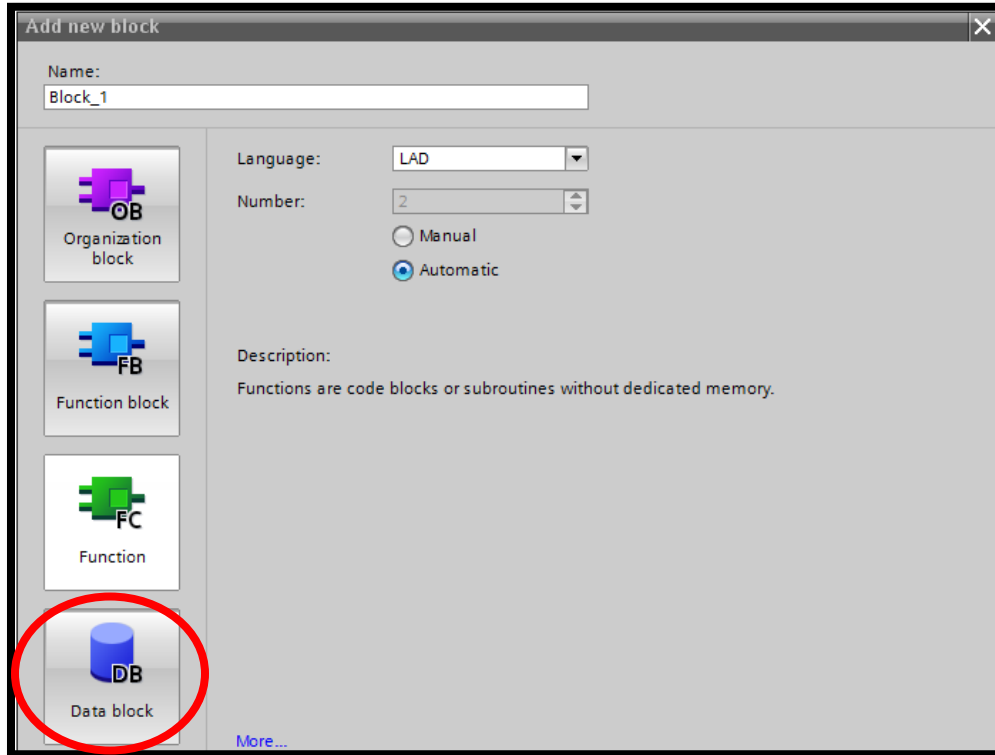
9. Your Main [OB1] should be blank and the Start01/Stop01RealIOBlock should have the networks.
10. Next, drag Start01/Stop01RealIOBlock to network 1 of your Main[OB1]



11. Next, navigate back to Main [OB1] then drag HMIRideControl11 block to network 1.
12. Save, download and test your program.

## Lab 3: Data Block

Last lab we studied the function in programming blocks. This is lab, we will study another program block called the Data Block.



We use the data block to organize tags. All tags are stored in the PLC tags drop down, this can get very confusing if your program is large. As mentioned in the previous lab, Data block works with the function.

1. Create a data block and name it HMIRideControlDB11, select manual and choose number 11. If you have done it correctly, there should be a blank page of tags.

HMIRideControlDB11								
	Name	Data type	Start value	Retain	Accessible f...	Visible in ...	Setpoint	Comment
1	Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
2	<Add new>			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Here we will take time to explore Data Types:

**Bool** – a byte, on or off

**Int** – a number 2 bytes

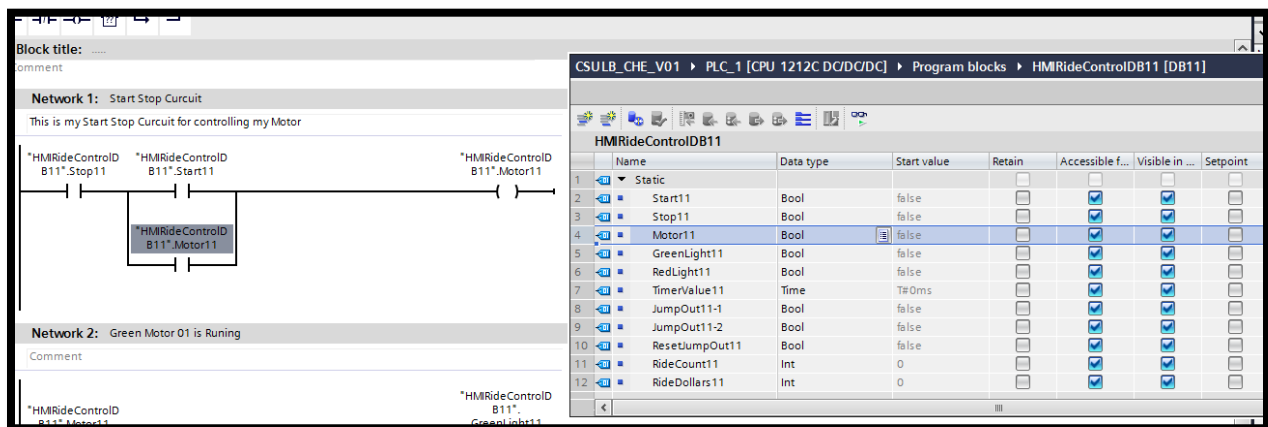
**Dint** – double integer, a word, 4 bytes

**Time** – a special data type that holds time

2. Now fill out your data block as shown.

HMIRideControlDB11								
	Name	Data type	Start value	Retain	Accessible f...	Visible in ...	Setpoint	Comment
1	▼ Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
2	Start11	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	Stop11	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
4	Motor11	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
5	GreenLight11	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
6	RedLight11	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
7	TimerValue11	Time	T#0ms	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
8	JumpOut11-1	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
9	JumpOut11-2	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
10	ResetJumpOut11	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
11	RideCount11	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
12	RideDollars11	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

- Once that's complete, double click Start01/Stop01RealIOBlock and COPY all of its networks.
- Paste the networks into HMIRideControlFC11.
- Now, open HMIRideControlDB11 and float the window, to reveal HMIRideControlFC11 and HMIRideControlDB11.
- Update the all the contacts in HMIRideControlFC11 by dragging the tags from HMIRideControlDB11 to the respected contacts.



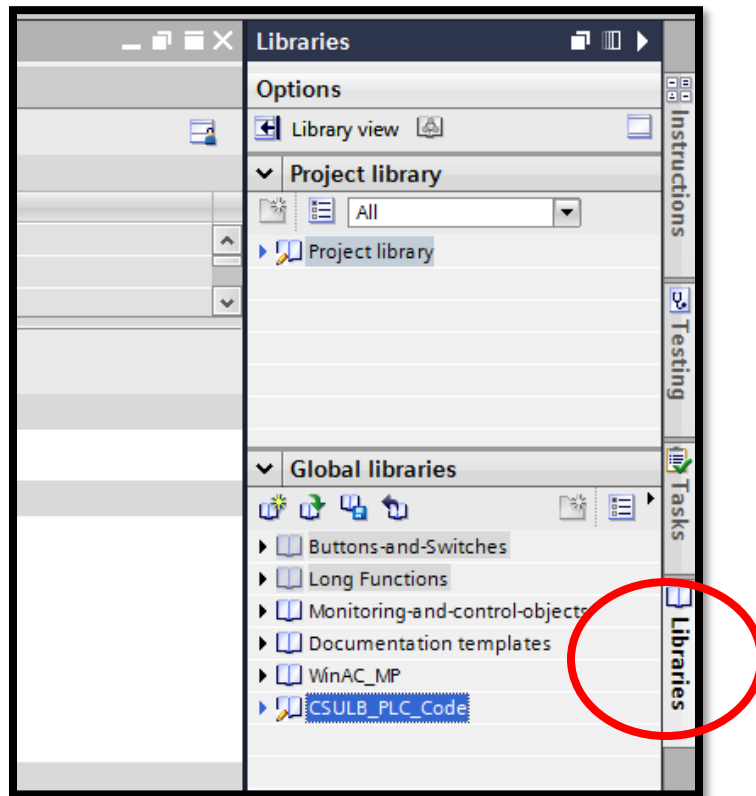
- Save and Download the PLC and monitor.
- Float the DB11 window while the PLC is running. Here, you should be able to see TimerValue11 increasing (in Miliseconds), the RideCount11 incrementing, and the RideDollar11 increasing
- When done, go offline and save.

## Lab 4: Global Libraries

This lab will introduce the useful option of Global Libraries. Global libraries are very useful by allowing you to save programs, such as Data blocks, organization blocks, function, and function blocks. From here, the programs will be available to use when other files are open.

### To get started:

1. Enter the Main [OB1].
2. On the far right, select the Libraries tab and go into Global libraries

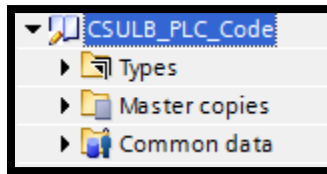


3. On the top of the dropdown menu, select "Create new global library".

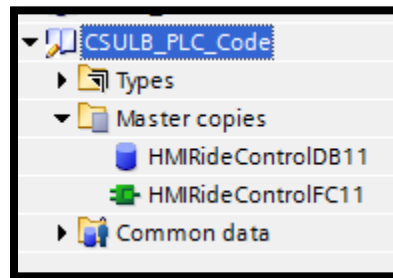


4. When prompt, name the library "CSULB\_PLC\_Code", then click Create.

- Now select the library and dropdown the menu to reveal Master copies.

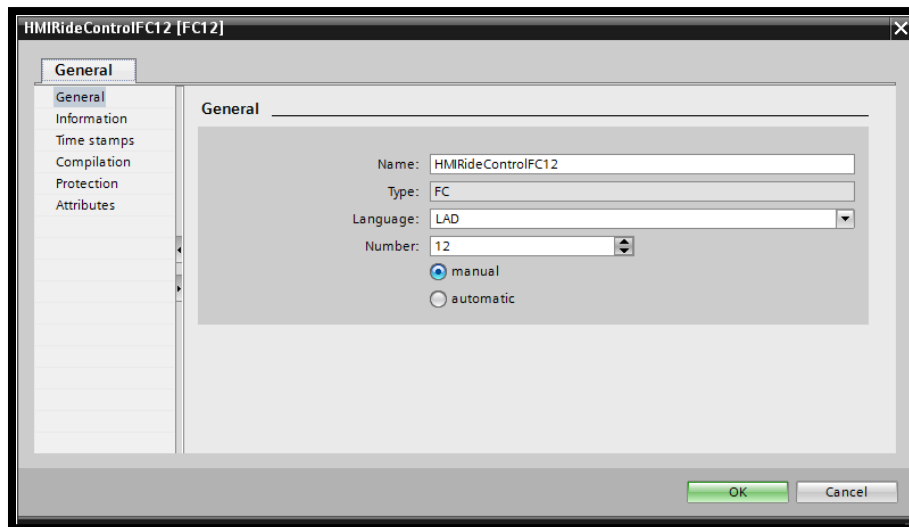


- From here, obtain HMIRideControlFC11 [FC11] and drop it into the master copies. Do this with HMIRideControlDB11 [DB11] also.
- Now you are able to access the FC and DB from any file that you have open.



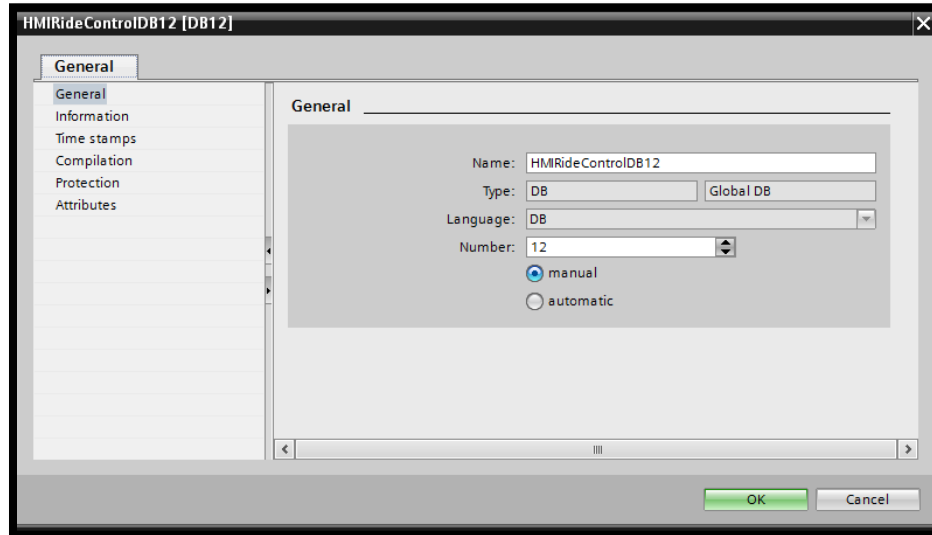
**To use the library, go to the CSULB\_PLC\_Code library:**

- Select Master copies and grab the HMIRideControlFC11.
- Drag the function to the Program Blocks menu on the left and select Rename and Paste Objects.
- You can now see that the new function is named HMIRideControlFC11\_1 [FC11]. This is due to that there are more than one FC with that same name.
- We will fix this by right clicking the HMIRideControlFC11\_1 and selecting Properties. In the popup menu, rename it to HMIRideControlFC12.
- 





6. Select Manual and renumber it to 12. Click OK.
7. Now the HMIRideControlFC12 will not conflict with the existing function.
8. Repeat this process with the HMIRideControlDB11, renaming it to HMIRideControlDB12.



9. Now open Main [OB1] and drag HMIRideControlFC12 into network 3. Save and download to the PLC.

