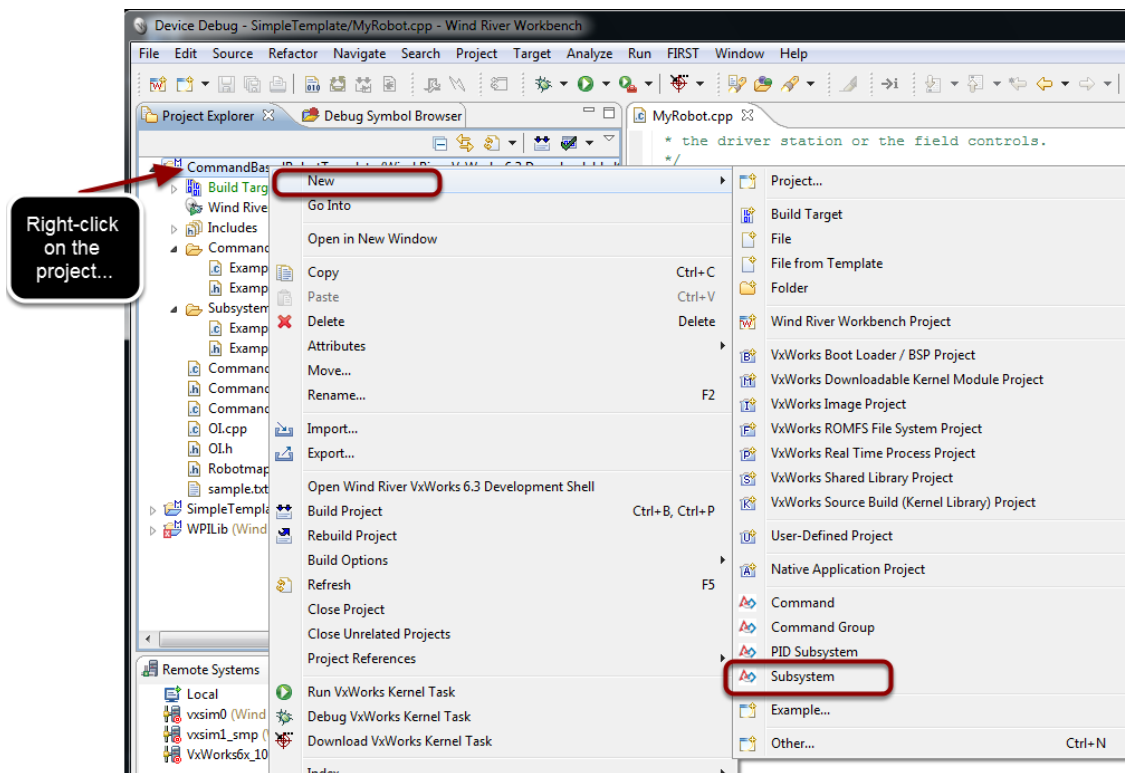


Adding Commands and Subsystems to the project - C++

Adding Commands and Subsystems to the project - C++

Commands and Subsystems each are created as classes in C++. The plugin has built-in templates for both Commands and Subsystems to make it easier for you to add them to your program.

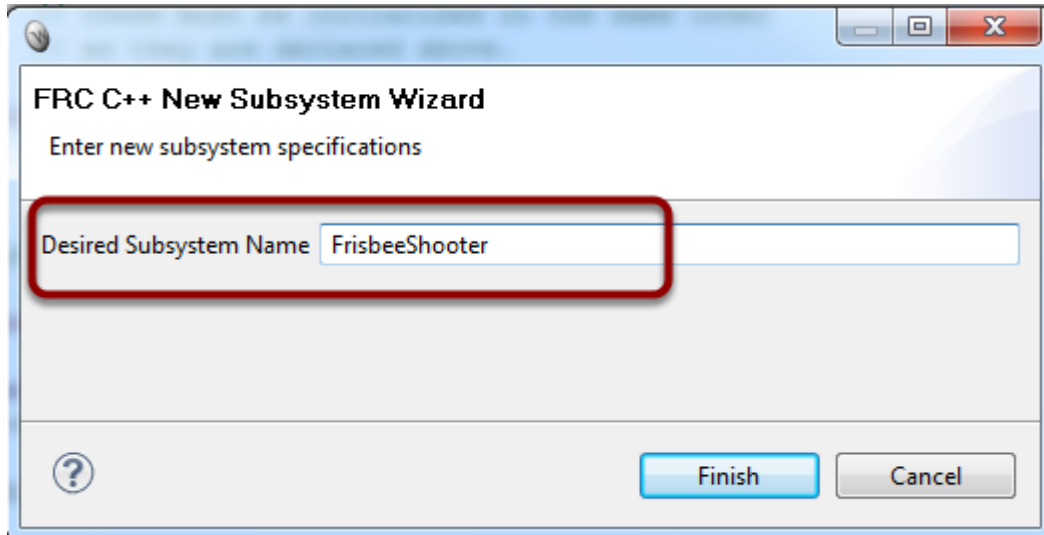
Adding subsystems to the project



To add a subsystem, right-click on the project name and select "New" then "Subsystem" in the drop down menu.

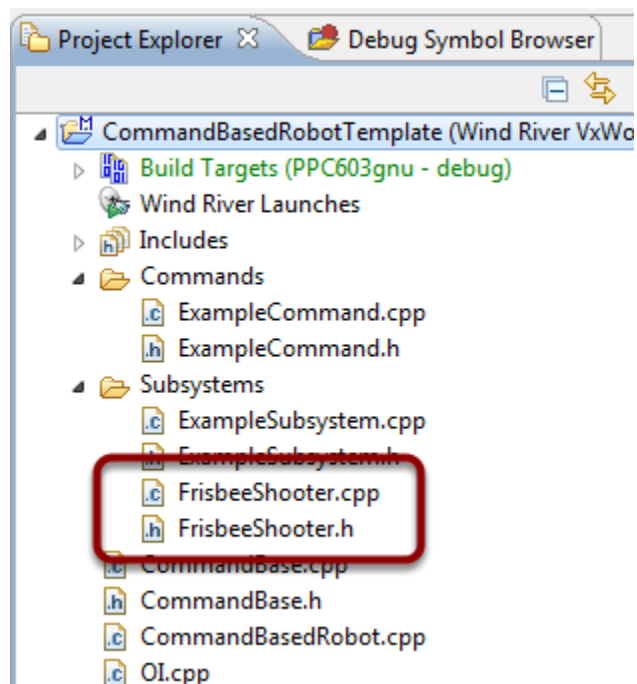
Adding Commands and Subsystems to the project - C++

Naming the subsystem



Fill in a name for the subsystem. This will become the resultant class name for the subsystem so the name has to be a valid C++ class name.

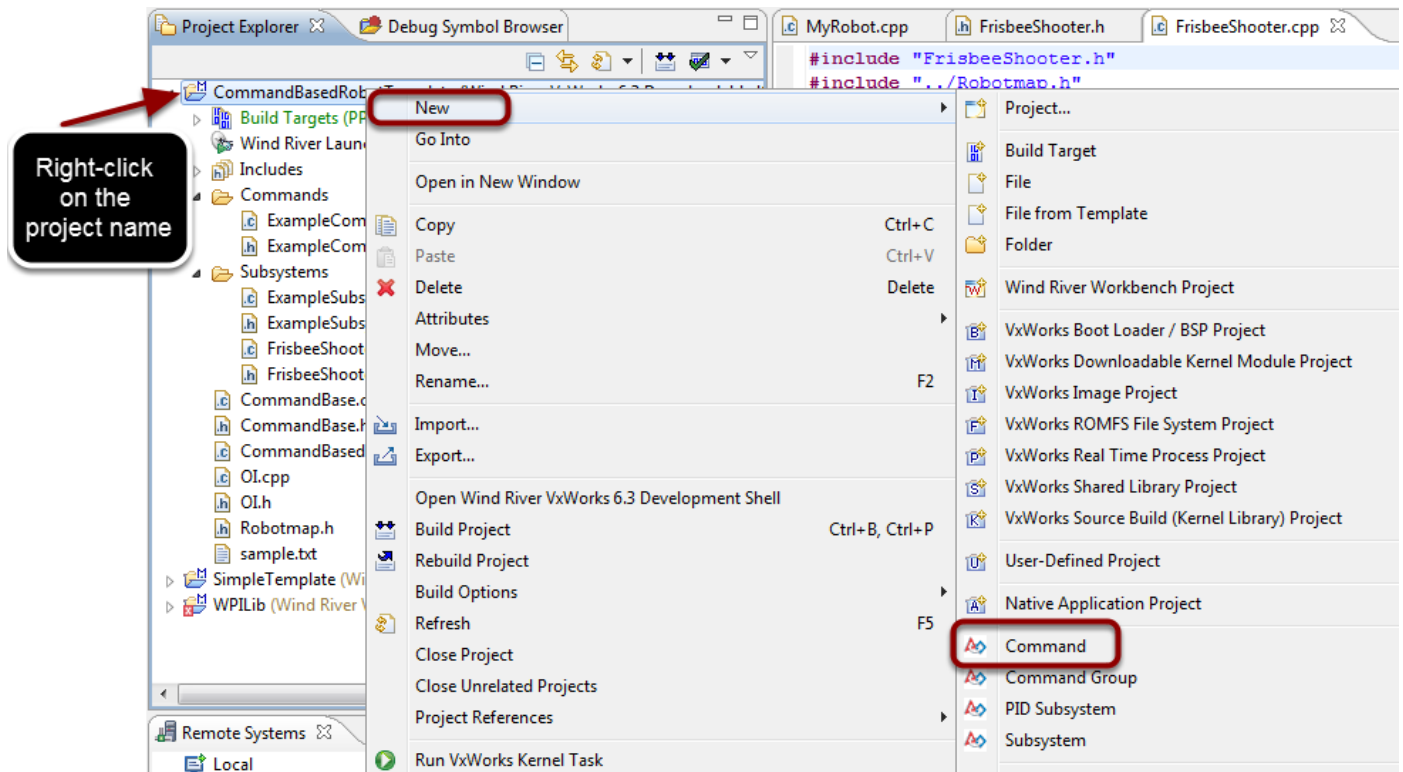
Subsystem created in project



Adding Commands and Subsystems to the project - C++

You can see the new subsystem created in the Subsystems folder in the project. To learn more about creating subsystems, see the [Simple Subsystems](#) article.

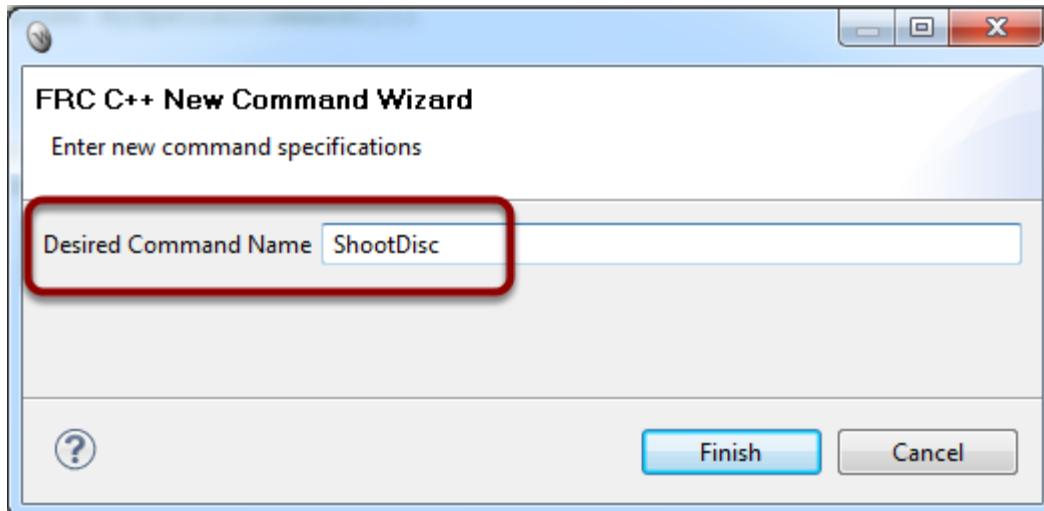
Adding a command to the project



A command can be created for the project using steps similar to creating a subsystem. First right-click on the project name in the Project Explorer and select "New Command".

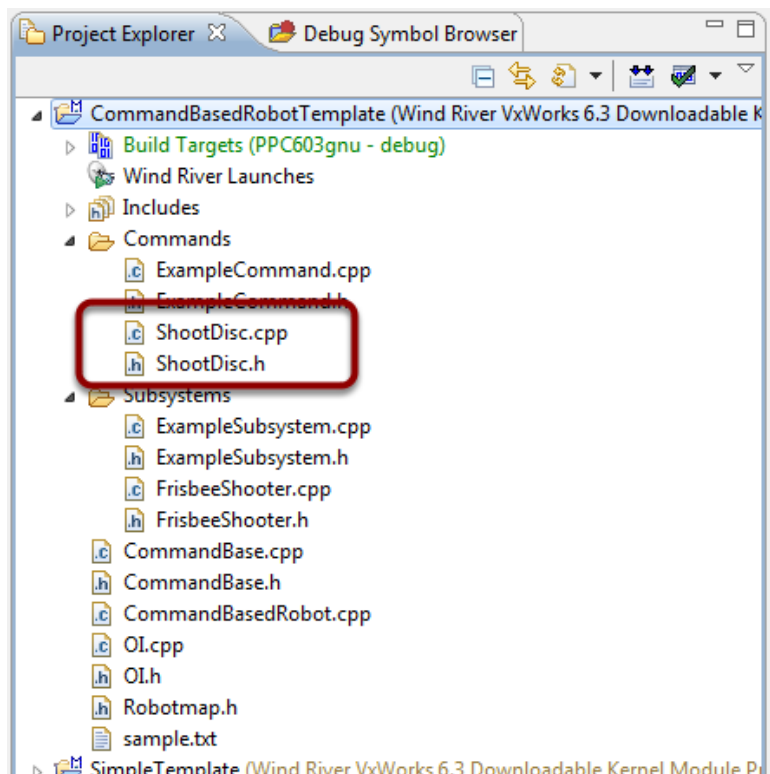
Adding Commands and Subsystems to the project - C++

Set the command name



Enter the Command name into the "Desired Command Name" field in the dialog box. This will be the class name for the Command so it must be a valid C++ name.

Command created in the project



Adding Commands and Subsystems to the project - C++

Liquid error: undefined method `account' for nil:NilClass