Micro800 Programming Basics
Tutorial 3: User Defined Function Block

For Classroom Use Only!

LISTEN. THINK. SOLVE.®

Allen-Bradley • Rockwell Software

Rockwell Automation

Important User Information

This documentation, whether, illustrative, printed, “online” or electronic (hereinafter “Documentation”) is intended for use only as a learning aid when using Rockwell Automation approved demonstration hardware, software and firmware. The Documentation should only be used as a learning tool by qualified professionals.

The variety of uses for the hardware, software and firmware (hereinafter “Products”) described in this Documentation, mandates that those responsible for the application and use of those Products must satisfy themselves that all necessary steps have been taken to ensure that each application and actual use meets all performance and safety requirements, including any applicable laws, regulations, codes and standards in addition to any applicable technical documents.

In no event will Rockwell Automation, Inc., or any of its affiliate or subsidiary companies (hereinafter “Rockwell Automation”) be responsible or liable for any indirect or consequential damages resulting from the use or application of the Products described in this Documentation. Rockwell Automation does not assume responsibility or liability for damages of any kind based on the alleged use of, or reliance on, this Documentation.

No patent liability is assumed by Rockwell Automation with respect to use of information, circuits, equipment, or software described in the Documentation.

Except as specifically agreed in writing as part of a maintenance or support contract, equipment users are responsible for:

- properly using, calibrating, operating, monitoring and maintaining all Products consistent with all Rockwell Automation or third-party provided instructions, warnings, recommendations and documentation;
- ensuring that only properly trained personnel use, operate and maintain the Products at all times;
- staying informed of all Product updates and alerts and implementing all updates and fixes; and
- all other factors affecting the Products that are outside of the direct control of Rockwell Automation.

Reproduction of the contents of the Documentation, in whole or in part, without written permission of Rockwell Automation is prohibited.

Throughout this manual we use the following notes to make you aware of safety considerations:

WARNING

Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.

IMPORTANT

Identifies information that is critical for successful application and understanding of the product.

ATTENTION

Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you:

- identify a hazard
- avoid a hazard
- recognize the consequence

SHOCK HAZARD

Labels may be located on or inside the drive to alert people that dangerous voltage may be present.

BURN HAZARD

Labels may be located on or inside the drive to alert people that surfaces may be dangerous temperature

About this lab

Connected Components Workbench (CCW) is the integrated design environment software package that is used to program, design, and configure your Rockwell Automation Connected Components devices such as, Micro800 programmable logic controllers, PowerFlex drives, SMC soft-starters, and PanelView Component operator interface terminals.

This lab will demonstrate and help guide you on how to use and program a Micro850 controller using the CCW software.

Tools & prerequisites

- Software: Connected Components Workbench Select Version "See Class Website for Version"
- Hardware: Micro850 Programmable Logic Controller, Catalog 2080-LC50-24QBB

Please note:

CCW is an all-encompassing software package for component class controllers (or- small / micro controllers). It contains the application programming environment for the Micro800 Programmable Controllers (PLC), Drives (Variable Frequency Drives or VFD's which use AC voltage, converted to DC, generate a Pulse Width Modulated (PWM) signal to control AC induction Motors) Human-Machine Interface (HMI) displays for control, feedback to an operators panel and some Safety PLC's.

With that- all User Manuals are included in CCW as well as a very extensive Help menus.

At any time that you need help or reference to any item, component or object, simply click on the help pulldown

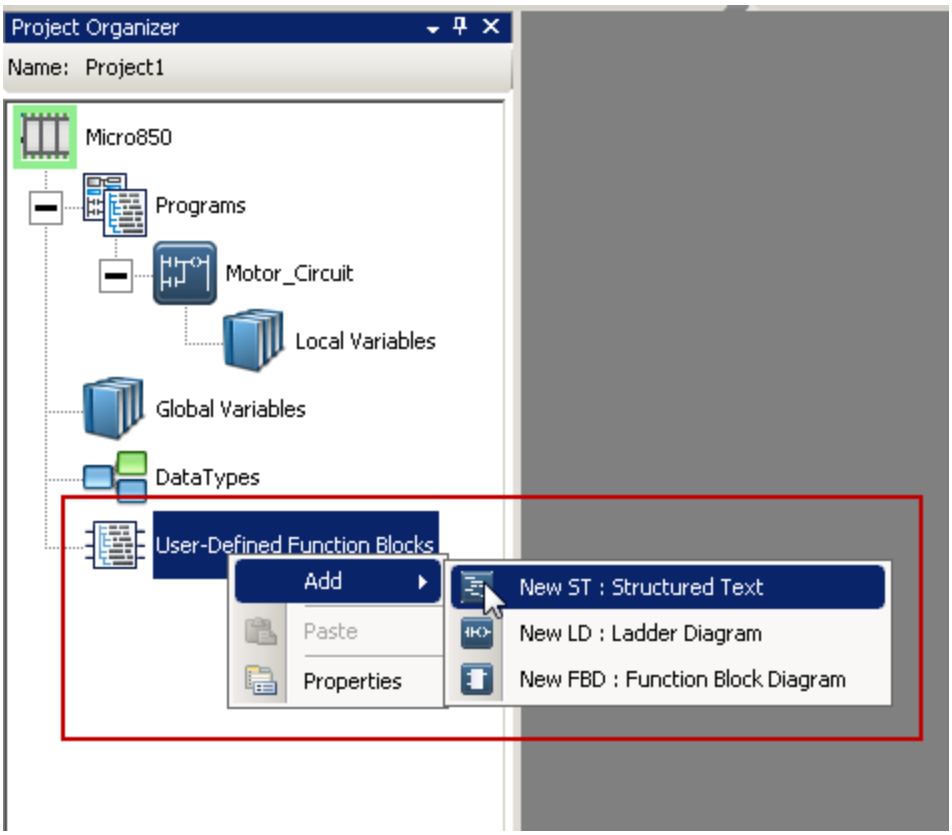
Learn about User Defined Function Blocks

In this section of the lab, you will learn about a User Defined Function Block (UDFB), and how to create one using Structured Text.

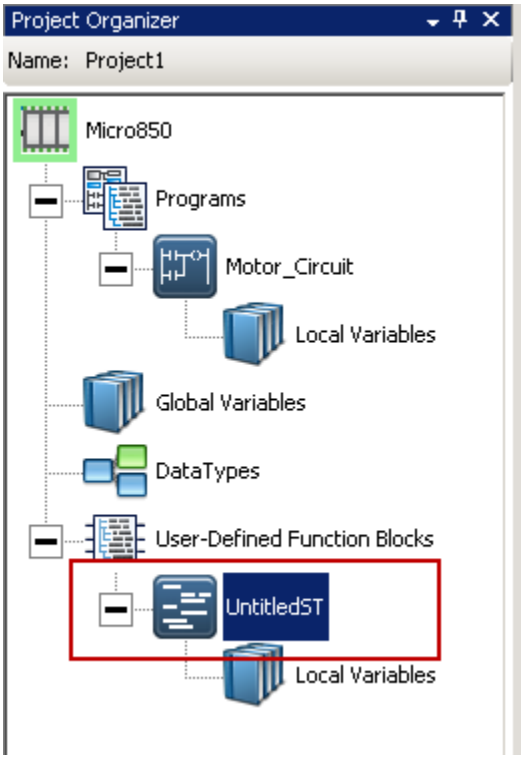
A User-Defined Function Block is a user defined program that can be packaged into an Instruction Block and reused within your Micro800 project. A UDFB can be written in Ladder, Function Block, or Structured Text.

You will be creating a UDFB to calculate the volume of a cylinder based on an inputted radius and height value.

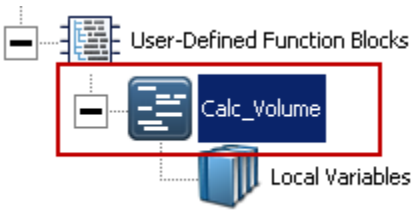
1. In your current project, right click **User-Defined Function Blocks** and select **Add → New ST: Structured Text**.



2. A program called **UntitledST** will be created under User-Defined Function Blocks.

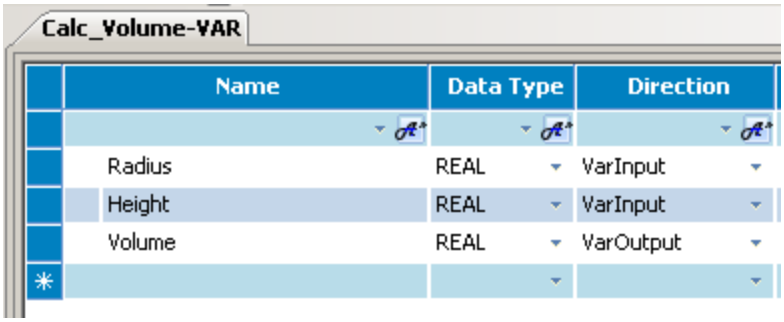


3. Rename this UDFB, **Calc_Volume**.

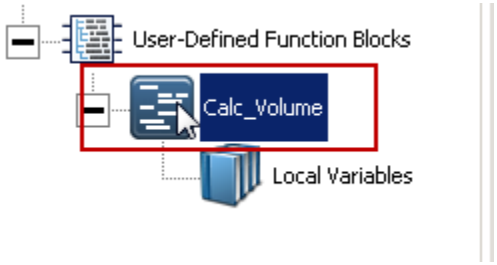


4. Double-click Local Variables under Calc_Volume.

5. Create the following variables. Take careful note to properly configure the **Direction** property. This property defines whether the variable is an Input, Output, or standard Variable.



6. Next, double-click the **Calc_Volume** UDFB to launch the program editor in the main project window.



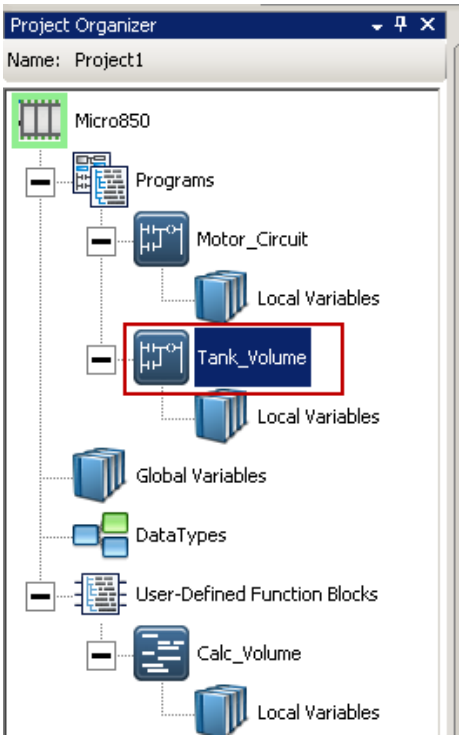
7. Add the following line of code to the program.

   Volume := 3.14 * Radius * Radius * Height;

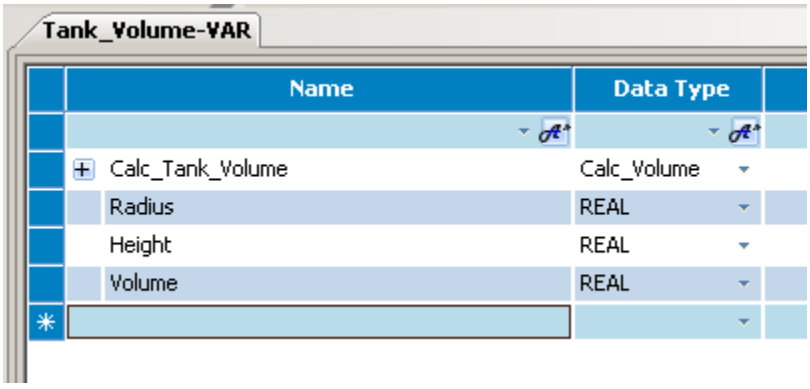
8. You have completed creating your UDFB.

9. Save your program.

10. Next, create a new ladder diagram program called **Tank_Volume**.



11. Open the Local Variables for the Tank_Volume program, and create the following variables. Notice the Data Type for the variable **Calc_Tank_Volume** is the Calc_Volume UDFB you created.

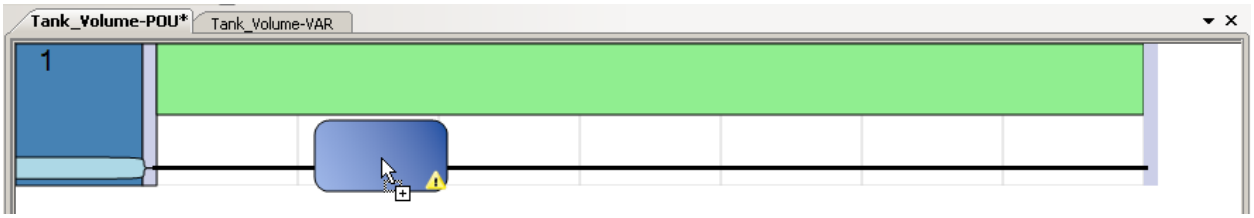


NOTE: Difference Between Real and Integer Values when declared or used to debug a program

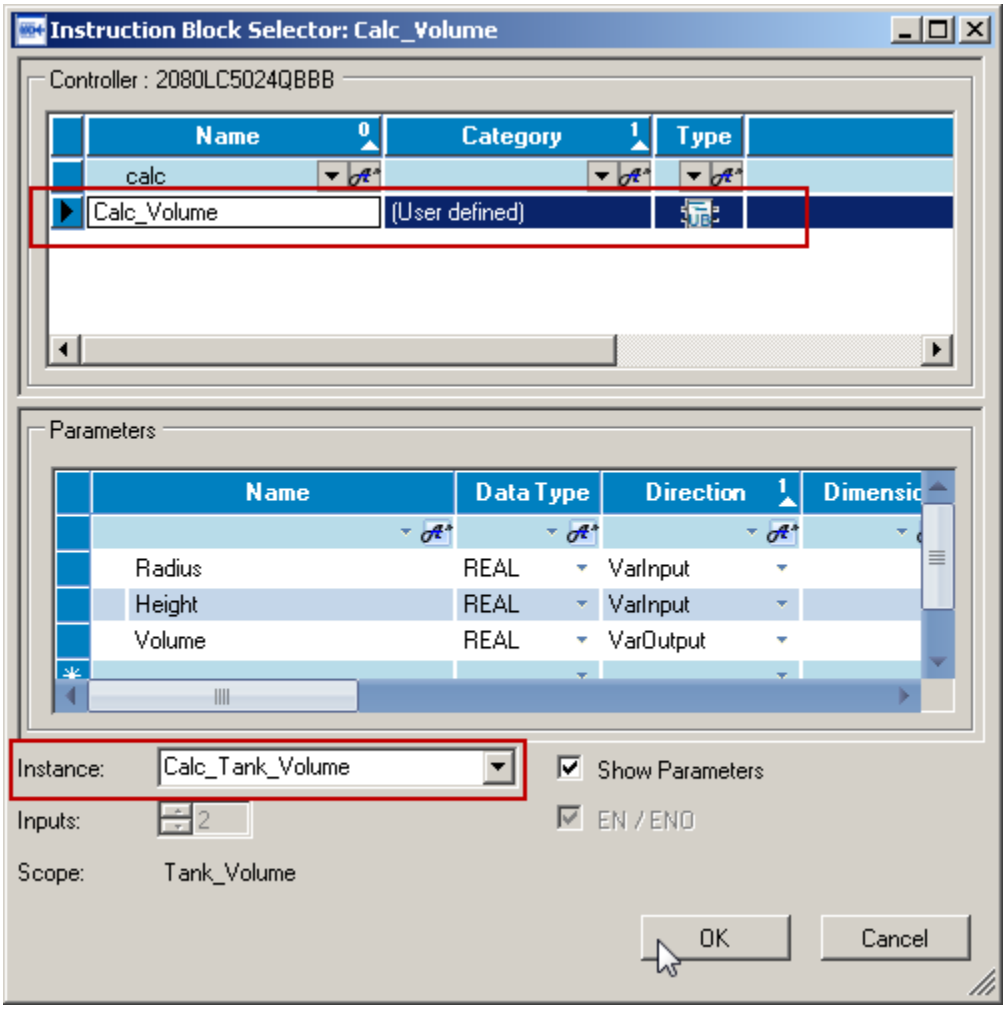
Integer Value: is a whole number without a decimal place (I.E 2)

Real Value: is a number with a decimal place (I.E 2.0). It is important to add the decimal and at minimum one place past the decimal point (even it is a 0 value) or the software will give an error during the program build and will not execute the program.

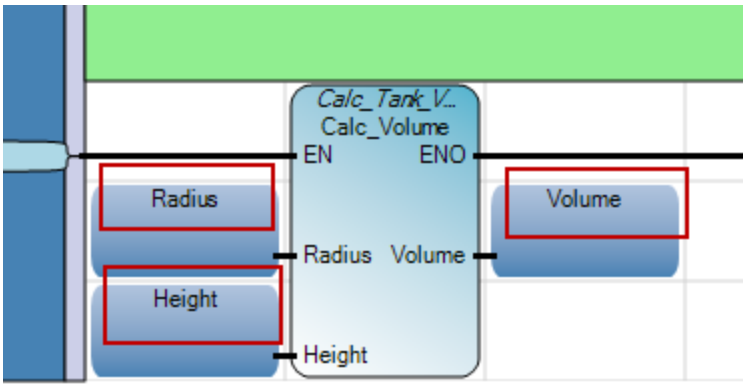
12. Next, open the **Tank_Volume** program, and add a **Block** Instruction to the first rung.



13. Select the **Calc_Volume** UDFB, and specify the Instance **Calc_Tank_Volume**. Then click **OK**.



14. Next, specify the following variables for each parameter of the Block.

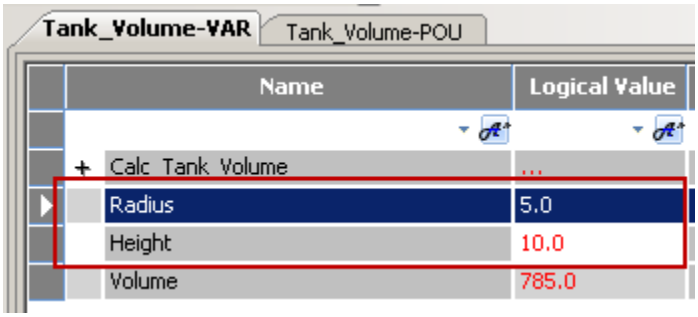


15. Save your project.

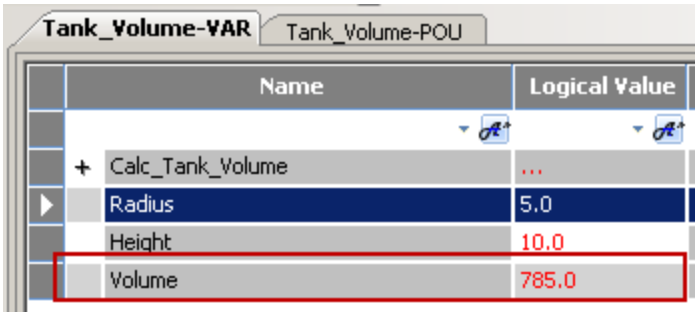
16. Build and download your program to your Micro850 controller.

17. Once your download is complete, press the F5 key to enter Debug Mode.

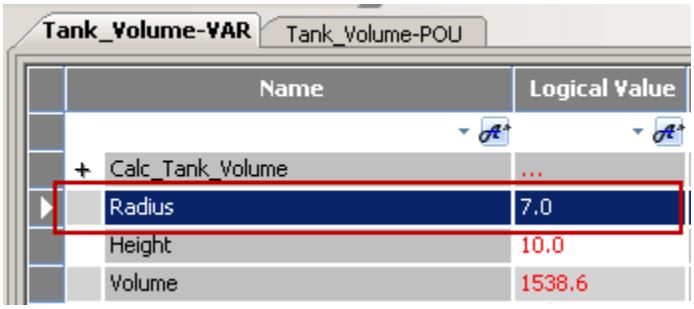
18. Open the Local Variables of your Tank_Volume program, and set the value of Radius to 5, and Height to 10.



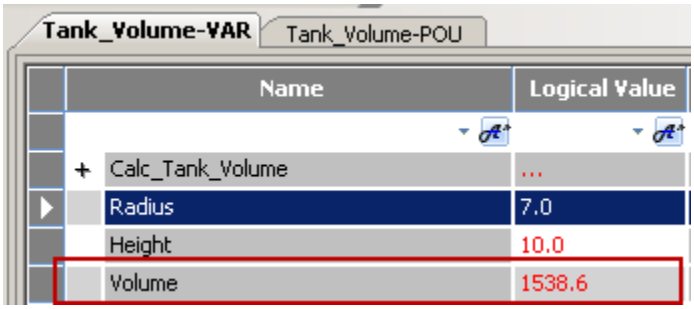
19. The value of Volume should read 785.0.



20. Change the value of Radius to 7.0



21. The value of Volume should read 1538.6



22. Press Shift+F5 to exit Debug Mode.

23. You have completed this section of the lab.

Assignment 3: Warning Overflow

Directions: Complete after working on the tutorial pages 42-49. Create a program that will do the following

1. Storage container Cone: research the volume formula for a cone modify user defined function block
2. Cone Size: Radius 10”; Height: 5”
3. Program: Simulate the filling of the cone with material and showing a warning light if the fill is to high
    a. Program Start with DI4 Direct Contact
    b. Create a Function Block to calculate the volume of a defined cone above
    c. Create a variable for Fill Test Values (These will be inputted manually to test the program)
        i. 500 in^3
        ii. 100 in^3
        iii. 600 in^3

4. Output:
    a. If Fill Test Value is < 250 then both Green Lamps will turn ON
    b. If Fill Test Value is 250 <= x <= 525 then One Green Lamp and One Yellow Lamp will turn ON
    c. If Fill Test Value is > 525 then both Red Lamps will turn ON; wait 3 seconds and then turn OFF the Lights (this is to simulate the shutdown of the system because it is too full).

    HINT: Use Instruction Block Comparison
    NOTE: May use multiple rungs to figure this out.

Submission:
A. Show/Video program working (Show computer screen and PLC Board)

B. Printout a copy of the program