Advanced HMI Procedure

Rev: 2

Author: Marcus Laramay Date: 3/27/17

Reviewed and Revised: Benjamin Jacobs, 3.27.2017

General Information:

Advanced HMI (Human Machine Interface) is a project for Microsoft Visual Studio that enables the user to develop an HMI with drag and drop components. It is compatible with the Allen Bradley line of PLC (Programmable Logic Controller). In this procedure it will be explained how to interface the Allen Bradley MicroLogix 1000 PLC with Advanced HMI.

Getting Started:

The first step is obtaining the Advanced HMI project. The project can be found at: <u>https://sourceforge.net/projects/advancedhmi/</u> or by going to the following folder on your computer: <u>\\stuapp1\Instructor\Hartle</u> (Instructor Folder -> Hartle -> AdvancedHMIv399s). Copy the AdvancedHMIv399s folder to your documents folder. Ensure that both Microsoft Visual Studio and RSLogix are installed on your computer.

Programming the PLC:

Before we can create an HMI for the Allen Bradley PLC we need to program the PLC. For this procedure we will use a simple Start/Stop Jog program. Start by opening RSLogix. The program can be found under all programs -> Rockwell Software -> RSLogix Micro English -> RSLogix Micro English.



Create a new file and select "Bul. 1761 MicroLogix 1000" and the driver "AB_DF1-1":

Figure 1. New File

Writing the Program:

- 1. Add four new rungs to the program.
- 2. On the first rung add two Normally Open (NO) contacts in parallel and add a coil. Label one of the NO contacts STOP with the address I:0:0 and label the other NO contact STOP HMI with the address B3:0/15. Label the coil STOP BIT with the address B3:0/0
- 3. On the second rung add two NO contacts in parallel and add a coil. Label one of the NO contacts START with the address I:0/1 and label the other NO contact START HMI with the address B3:0/14. Label the coil START BIT with the address B3:0/1.
- 4. On the third rung add a Normally Closed (NC) contact and label it STOP bit with the address B3:0/0. Then add two NO contacts in parallel. Give the first NO contact the address B3:0/1 and the second NO contact B3:02 with the label HMI output. Finally, add a coil with the address B3:0/2.
- 5. On the fourth rung add NO contact and a coil. Give the NO contact the address B3:0/2 and give the coil the address O:0/0 with the label OUTPUT.

The program should like this:



Figure 2. Advanced HMI Test Program

Save the program as AdvancedHMITest.RSS. Download the program to the PLC using this box:



Figure 3. Downloading Program

- When the Revision Notice box pop's up select OK.
- When it asks you if you are sure you want to proceed with the download select Yes.
- When it asks you if you want to run select Yes.
- When it asks you if you want to go online select No.



Figure 4. RsLinx Shutdown

Left click the arrow in bottom right of taskbar. Then right click on Rslinx and select "Shutdown Rslinx Classic".

Advanced HMI (Only available in room 118):

Open the AdvancedHMIv35.sln Visual Basic project file that is in the AdvancedHMIv399s folder. Select OK for any windows that pop up.

Click sign in and use your SUNY Canton email and password then click "start visual studio".

Open the MainForm.vb file in the Solution Explorer window:



Figure 5. Opening MainForm.vb

To build the AdvancedHMI program go to Build in the main menu and select Build AdvancedHMI:



Figure 6. Building AdvancedHMI

Now we will add some components to the HMI window.

If the toolbox tab is not on the left side of the screen press "Ctrl+Alt+x" to open it.

Go to the AdvancedHMIDrivers Components tab in the toolbox and drag the SerialDF1forSLCMicoCom onto the HMI window:



Figure 7. SerialDF1forPLC5Com driver

Now, under the AdvancedHMIControls Components tab drag two MomentaryButton's onto the HMI Window:



Figure 8. Momentary Push Buttons

Select one of the MomentaryPushbuttons and make the following changes in the properties window:

- Font -> ForeColor: Black
- Font -> Text: Start
- PLC Properties -> PLCAddressClick: B3:0/14
- Misc. -> ButtonColor: Green

Now, Select the other MomentaryPushbutton and make the following changes in the properties window:

- Font -> ForeColor: Black
- Font -> Text: Stop
- PLC Properties -> PLCAddressClick: B3:0/15
- Misc.-> ButtonColor: Red



Figure 9. Properties Window

Now we need a component for the Pushbuttons to output to. In the AdvancedHMIControls Components tab drag a motor onto the HMI Window:



Figure 10. Motor

Select the motor and make the following change in the properties windows:

• PLC Properties -> PLCAddressValue: B3:0/2



Figure 11. HMI Start

Click Start (button with green triangle). The program should start and an HMI window will pop up. When clicking the Start or Stop push buttons it should turn on the PLC's output. You can also wire physical push buttons to the PLC's input's: I:0 for STOP and I:1 for START. These physical push buttons will also control the output and the motor in AdvancedHMI. To stop the HMI close the HMI window.