AN-289

C-Bus Integration with Protege GX and Protege WX

Application Note

Last Published: 04-Oct-21 4:53 PM

Contents

Introduction

The Protege system provides onboard support for the C-Bus communication protocol, allowing you to integrate with any new or existing C-Bus system. Control C-Bus Groups using any output or input in the Protege system, or trigger access and security functions from C-Bus units.

This application note provides instructions on integrating both Protege GX and Protege WX with a C-Bus system for building automation services.

Prerequisites

C-Bus Integration with Protege GX requires:

- An operational Protege GX system, version 4.0.128 or higher.
- A Protege GX controller, firmware version 2.08.825 or higher.
- C-Bus software and hardware (as specified below).

C-Bus integration in Protege WX requires:

- A Protege WX controller, version 2.10.39 or later.
- Access to the Advanced Mode features of Protege WX.
- C-Bus software and hardware (as specified below).

The legacy integration with Protege SE required an additional PRT-COMM unit to be installed. This is not required for Protege DIN rail modules.

C-Bus Requirements

Hardware

The C-Bus system communicates with the Protege network over ethernet. Therefore, this integration requires the use of a C-Bus Network Interface (CNI), product code 5500CN2. This must be connected to the same network as the Protege controller, either directly or via an ethernet switch/hub.

The diagram below indicates how the C-Bus network should be connected with the Protege network:
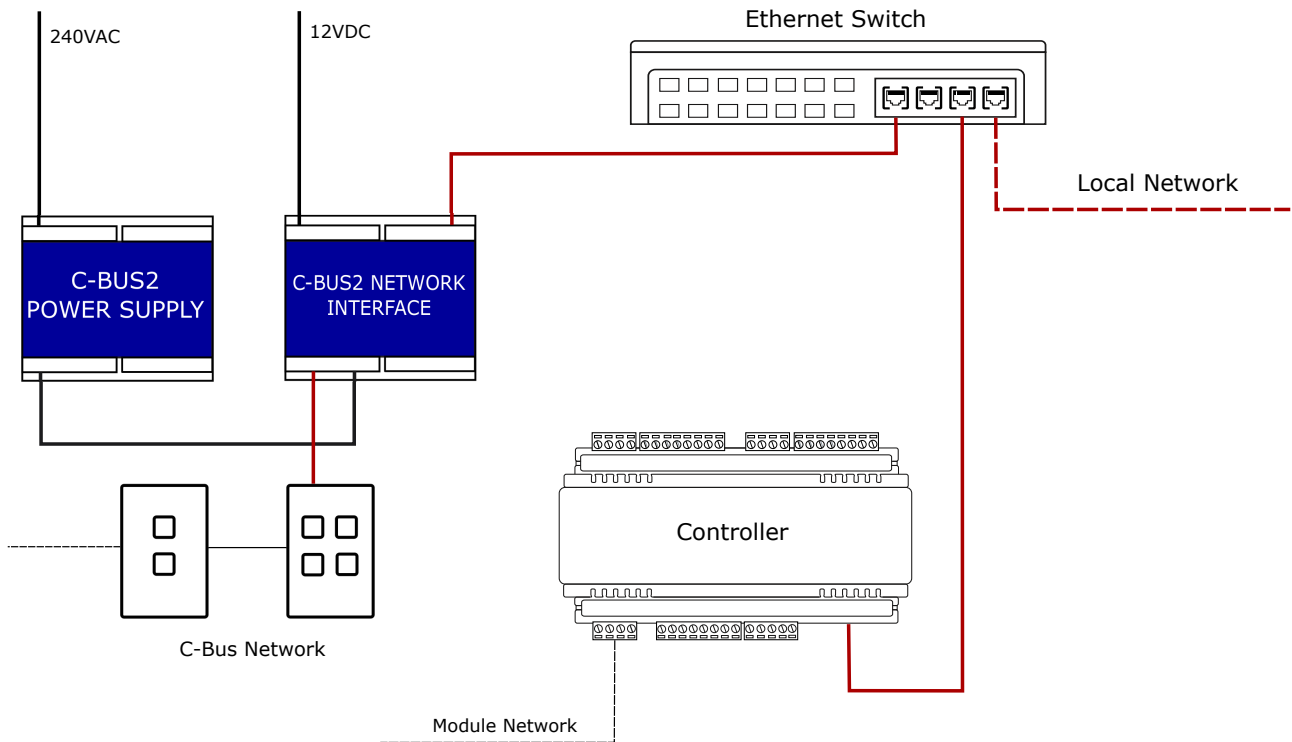
The C-Bus hardware must be set up and configured according to the installation documents supplied by the vendor.

Software

This integration requires the full C-Bus Toolkit software installation, including the C-Gate service. You can download and install the full distribution package from the Clipsal website. This also installs the Control Systems IP Utility, which is useful for detecting the IP address of connected modules.

The instructions in this document are based on C-Bus Toolkit version 1.15.6. The integration has also been tested and validated with earlier versions of the toolkit software.

Setting up the Integration

The process for setting up the integration is split into two parts:

- Configuring the Clipsal hardware via the C-Bus Toolkit application software
- Configuring Protege GX or Protege WX for C-Bus integration.

Important: The C-Bus Toolkit and Protege C-Bus Service use the same port. This means that only one can be running at any time. If the C-Bus network is open, the Protege C-Bus Service will fail to start. If the Protege C-Bus Service is running, the C-Bus Toolkit will fail to open the network.

Configuring the C-Bus Network

Protege can be integrated with both new and existing C-Bus networks. The following procedure describes setting up a basic C-Bus network from scratch. However, even if you are working with an existing C-Bus Network, it may be useful to read this section to better understand the requirements for integrating with Protege.

Creating a new C-Bus Project

If you are testing or working with a new C-Bus installation, it is necessary to first create a C-Bus project for use with the Protege integration.

1. Ensure that the CNI is connected correctly to the ethernet network and that it is pingable on a known IP address. If you do not know the IP address, you can use the Control Systems IP Utility included with the full C-Bus Toolkit download to view or change the CNI's address and port.

   Make a note of the CNI IP address and port, as these are also required in the Protege programming.

2. Open the C-Bus Toolkit software. In the main menu, select **Projects > Add Project**.

3. In the Add Project window, enter a **Project Name**. Click **OK**.

4. You will be prompted to add a network to the project. Select **Yes**.

5. C-Bus will request the name and details of the network interface:
   - **Name**
   - **Type**: CNI.
   - **Address**: IP address of the CNI.
   - **Port**: Port used by the CNI. By default this is 10001.

6. Expand the project in the tree sidebar, then select the newly created network. Click the **Open Network** button. If the CNI has been configured correctly, this should open the port for communication and automatically detect all connected modules. This may take some time if you are opening the network for the first time.

   If you are attempting the make changes to the C-Bus system and the network will not open, check that the Protege C-Bus Service is not running. The C-Bus network and Protege C-Bus Service cannot be open at the same time.

7. Once the network is running, navigate to **Units** in the sidebar. The detected units will be displayed in the Network window (lower half of the screen).

   If the toolkit has detected the CNI but no other modules, open the CNI unit by double clicking on it. Click **Advanced**, then select the **Global** tab and check both the **Enable C-Bus Clock** and **Enable Burden** options. Click **OK** to save and close, then click the **Rescan** button to retrieve the connected units.

8. Highlight all detected units and click the **Add/Transfer to DB** button. This creates records for these units in the C-Bus database (upper half of the screen), allowing you to program them.

Configuring C-Bus Groups

Each individual input and output in a C-Bus system must be assigned to an application (such as Lighting) and group in order to be programmed. When new units are imported into the database, each input and output connected to them is automatically assigned to an application and group. These can be modified and programmed as outlined below.

The Protege integration can send data to or from particular groups in the C-Bus programming. Therefore, the inputs and outputs assigned to each group will be treated as a single item by Protege.

The following instructions cover very basic principles of programming in C-Bus, as required for testing the integration with Protege. Any live C-Bus site should be programmed by an experienced professional.

1. On the **Units** page, double click on any unit in the **Database** window. This opens a programming window for the unit.

2. Each part of the unit will have been automatically assigned to a group and application. In the simple view, each part can be assigned to a different group within the same application using the relevant dropdown menus. To add a group, use the green **[+]** button.

3. Assign a **Function** to each part of the unit (e.g. On/Off, Dimmer, Timer, etc.).

4. To change the application of an entire unit, click the **Advanced** button. On the **Unit Identification** tab, you can change the **Application**.

5. Click **Apply**. Select both **Save to Database** and **Save to Physical Unit**.

6. If any changes have been made in the database but not saved to the physical unit, highlight the unit(s) in the **Units in Database** window and click **Transfer to Network**.

7. Complete any other required programming. Groups can be viewed and configured under the relevant **Application** in the tree sidebar.

8. Once all groups have been configured as required, open the **Project** page from the sidebar. Click the **Document Project** button. This generates an HTML page detailing the address of each application and group, which will be required for the Protege integration.

   The address of an application or group can also be found by right clicking on the record in the tree navigation sidebar, and selecting **Edit**.

Ensure that you test the groups by activating any inputs. For example, if you press a lighting input, all lights assigned to the same group should be activated together.

Closing the C-Bus Network

The C-Bus Toolkit software and the Protege C-Bus Service use the same port to communicate with the C-Bus network. Therefore the C-Bus Network must be closed before the integration with Protege will function.

1. Navigate to the relevant **Project** from the sidebar.

2. Click on the **Close All Networks** icon.

3. Exit out of the C-Bus Toolkit software.

Configuring Protege

There are two main aspects to the Protege integration with C-Bus: a C-Bus service which communicates information between C-Bus and Protege, and automations which handle specific functions.

Creating the C-Bus Service

1. Navigate to **Programming | Services**. Select the **Controller** that is hosting the integration and **Add** a new service for the C-Bus integration.

2. Set the **Service Type** to C-Bus.

3. In the **General** tab, set the following:

    -   **Port Number**: TCP/IP (this option is not configurable in Protege WX)
    -   **CNI IP Address**
    -   **CNI Port**
    -   **Communication Failure Output (Group)**: This output or output group will be triggered when the service communication fails.

4. To log C-Bus communication activity for troubleshooting, enable the **Log C-Bus PCI Failure Message**, **Log C-Bus ACK Message**, and **Log C-Bus Data Activity** options. Troubleshooting messages will appear in the event log.

    The **Enable Text Output** option converts communications from Protege to a human readable form, and skips initialization of the CNI. This allows for debugging by connecting a monitoring device in place of the CNI. However, the integration will not function with this option enabled.

5. Click **Save**.

6. Now you can start the service. For Protege GX: Right click on the C-Bus Service record and click **Start**. For Protege WX: Navigate to **Monitoring | Services**, click on the **Controls** button, and select **Start Service**.

    Note: The C-Bus service will not start unless the C-Bus network is closed. The C-Bus network will not open unless the Protege C-Bus service is stopped.

Creating Automations

In Protege systems, automations are configured to control devices that operate regularly, such as lighting or HVAC systems. For the purposes of this integration, each automation represents a single-direction connection between a Protege output and a C-Bus group. Using different automation configurations, it is possible to achieve the following:

-   Controlling Protege output (or output group) from a C-Bus group.
-   Controlling a C-Bus group from a Protege output (or output group).
-   Controlling a C-Bus Group from a Protege input type.

Each automation function has different configuration requirements (see further below). To create an automation:

1. Navigate to the **Automation | Automation** menu and add a new automation.

2. Enter the **C-Bus Application Code**. This is the address of the C-Bus application that this automation is to communicate with. You can find the relevant application code in the HTML documentation file generated from the C-Bus Toolkit (see previous page).

3. Enter the **C-Bus Group Code**. This is the address of the C-Bus group that this automation is to communicate with. You can find the relevant group code in the HTML documentation file generated from the C-Bus Toolkit (see previous page).

4. Set the **C-Bus Service** to the service created above.

5. Set an **Automation Output** or **Automation Output Group** (if required). This output or output group either controls the linked C-Bus group, or is controlled by it.

6. The following settings are available in the **Options** tab. Which options must be selected will depend on the type of automation being configured (see below).

    -   **Display Inverted Status**: This is a legacy option that has no effect.
    -   **Enable C-Bus Automation Functions**: When this option is enabled the automation controls or is controlled by the assigned C-Bus group.

        This option is required for the automation to begin communicating with C-Bus.

- **C-Bus Automation Output**: When this option is enabled this automation will control the assigned C-Bus group. When this option is disabled the automation will be controlled by the assigned C-Bus group.
- **Use Output Status in C-Bus Function**: When this option is enabled the automation will change or transmit the status of its automation output or output group rather than the status of the automation itself.
- **C-Bus Operates on Rising Edge**: C-Bus processing activates on the rising edge of a change in the output/input state, i.e. changing from OFF to ON. If the option is disabled the C-Bus processing ignores these changes.
- **C-Bus Operates on Falling Edge**: C-Bus processing activates on the falling edge of a change in the output/input state, i.e. changing from ON to OFF. If the option is disabled the C-Bus processing ignores these changes.

C-Bus Group controls a Protege Output

It is possible for a Protege output to follow the state of a C-Bus group. In this case, C-Bus will send Protege a signal with the application and group codes, causing the automation to activate the set **Automation Output**. This has wide applicability: a C-Bus light button might control any physical output in the system (such as a door lock), or activate a virtual output to drive further logic.

To control an output from a C-Bus group, navigate to **Automation | Automation** and create a new automation with the following settings:

- **General** tab:
  - **Automation Output Time**: The length of time in seconds that you wish the output or output group to be activated for.
  - **Automation Output (Group)**: The output or output group that will be controlled by the C-Bus function. This can be any physical output in the system, or a virtual output that will drive further logic.
  - **C-Bus Application Code**: The address of the C-Bus application.
  - **C-Bus Group Code**: The address of the C-Bus group.
  - **C-Bus Service**: The configured C-Bus service (not configurable in Protege WX).
- **Options** tab:
  - **Enable C-Bus Automation Functions**: Enabled.
  - **C-Bus Automation Output**: Disabled.
  - **Use Output Status in C-Bus Function**: Enabled.
  - **C-Bus Operates on Rising/Falling Edge**: As required. Enabling both options will allow C-Bus to switch the output both on and off.

When these have been programmed, test by manually activating the C-Bus group. When the group is activated, the output should turn on. When the group is deactivated, the output should turn off.

Protege Output controls a C-Bus Group

As well as using C-Bus switches to control Protege outputs, it is possible to do the reverse and control C-Bus groups with Protege outputs. In this case, when the output is switched Protege will send C-Bus a message containing the application and group code, causing C-Bus to change the state of that group.

To control a C-Bus group using an output (whether real or virtual), navigate to **Automation | Automation** and create a new automation with the following settings:

- **General** tab:
  - **Automation Output (Group)**: The output or output group that will control the C-Bus group.

    The **Automation Output Time** setting has no effect in this case. Any activation times must be set in the C-Bus system.

  - **C-Bus Application Code**: The address of the C-Bus application.
  - **C-Bus Group Code**: The address of the C-Bus group.
  - **C-Bus Service**: The configured C-Bus service (not configurable in Protege WX).

- **Options** tab:
  - **Enable C-Bus Automation Functions**: Enabled.
  - **C-Bus Automation Output**: Enabled.
  - **Use Output Status in C-Bus Function**: Enabled.
  - **C-Bus Operates on Rising/Falling Edge**: As required. Enabling both options will allow the Protege output to switch the group both on and off.

When this has been programmed, test by activating the Protege output (for example, by right clicking on the output record and clicking **Activate**). When the output is activated, the group should activate/turn on. When the output is deactivated, the group should deactivate/turn off.

Protege Input controls a C-Bus Group

Finally, it is possible to control a C-Bus group directly using a Protege input. When the input goes into alarm or is restored, a signal will be sent to the C-Bus system containing the application and group code. This allows you to use any Protege input as if it were a C-Bus switch, without rewiring the C-Bus system.

The programming is slightly more involved than linking an output to an automation. The following records are required:

- An automation that is not controlled by an output.
- An automation/control area that is always armed but does not report alarms.
- An input type that instructs inputs to control the automation.
- An input with the area and input type assigned.

Note that changes in automation state do not appear on status lists or event logs. If precise events for automation activation are required, it may be more useful to follow the programming steps below but assign a **Control Output** rather than a control automation; this (virtual) output can then be used as an automation output as above.

Creating the Automation

Navigate to **Automation | Automation** and create a new automation with the following settings:

- **General** tab:
  - **Automation Output (Group)**: None.
  - **C-Bus Application Code**: The address of the C-Bus application.
  - **C-Bus Group Code**: The address of the C-Bus group.
  - **C-Bus Service**: The configured C-Bus service (not configurable in Protege WX).
- **Options** tab:
  - **Enable C-Bus Automation Functions**: Enabled.
  - **C-Bus Automation Output**: Enabled.
  - **Use Output Status in C-Bus Function**: Disabled.
  - **C-Bus Operates on Rising/Falling Edge**: As required. Enabling both options will allow the Protege input to switch the group both on and off.

Creating the Automation Area

If you do not already have an automation or control area, navigate to **Programming | Areas** and create a new area with the following settings:

- **Configuration** tab:
  - **Rearm Time**: 1 minute.
  - All other timings should be set to 0.
- **Outputs** tab: All outputs should be left as <not set>.
- **Options (1)** tab:
  - Enable **Re-Arm Enabled** and **Use Unattended Brute Force Arming**. This allows the area to rearm following the **Rearm Time** even if there are inputs open.

- All other options should be disabled.
- **Options (2)** tab:
  - **Enable Force Arming**: Enabled.
  - **Clear Alarm Memory after Arming**: Enabled.
  - All other options should be disabled.

Creating the Input Type

Navigate to **Programming | Input Types** and create an input type that will allow an input to control automation. A separate input type must be created for each automation that is to be controlled; however, a single input type may be assigned to multiple inputs, allowing 'many to one' control of a C-Bus group.

- **General** tab:
  - **Control Automation**: Set the automation that will be controlled by any input assigned with this input type.
- **Options (1)** tab:
  - **Generate Alarms**: Enabled.
  - **Generate 24HR Alarms**: Enabled.
  - All other options should be disabled.
- **Options (2)** tab: All options disabled.
- **Options (3)** tab: The **Automation Options** section allows you to set how inputs assigned with this Input Type will control the automation.
  - **Activate Automation on Alarm**: When an input is OPENED, the automation will switch ON.
  - **Activate Automation on Restore**: When an input is CLOSED, the automation will switch ON.
  - **Deactivate Automation on Alarm**: When an input is OPENED, the automation will switch OFF.
  - **Deactivate Automation on Restore**: When an input is CLOSED, the automation will switch OFF.
  - **Toggle Automation State**: When an input is opened, the automation state will change. For example, pressing a button once will turn the automation ON, pressing it again will turn it OFF.
- **Options (4)**: All options disabled.

Configuring the Input(s)

Navigate to **Programming | Inputs** and select the input(s) that you wish to configure. Set the following:

- **Areas and Input Types** tab:
  - **Area**: Automation/Control area created above.
  - **Input Type**: Input type created above.

The **Control Automation** option in the input programming is not used.

Finally, arm the Automation/Control Area by right clicking on its record and clicking **Arm**.

Test by opening and closing the input. Depending on the settings in the input type, you should be able to activate and deactivate the C-Bus Group.

Bi-directional Control

It is possible for the service to communicate in both directions: for example, the same Protege output may both control and be controlled by a single C-Bus group. To achieve this, it is necessary to create two automation records with the same **Automation Output** - one for sending commands to C-Bus, and one for receiving commands from C-Bus.

Programming Scenario: Arming/Disarming an Area

The following programming shows how a C-Bus application can control the arming and disarming of an area so that it follows the status of a C-Bus group, such as a particular key on a C-Bus switch unit. In this scenario, there are two C-Bus buttons that communicate with Protege: one arms the area when activated, and the other disarms it. These buttons may also have any other function within the C-Bus system, such as lighting or heating control.

This programming utilizes virtual outputs and Area Control programmable functions. It can be used to arm and disarm a single area, or an area group.

Creating Virtual Outputs

For this programming, it is necessary to create two virtual outputs (per area): one to command the area to arm, the other to command it to disarm.

1. If you do not already have virtual outputs available, navigate to **Expanders | Output Expanders** and create a virtual output expander:
   - From the toolbar, select the **Controller** that will control these outputs, then click **Add**.
   - Ensure that the **Virtual Module** option is enabled.
   - Set the **Physical Address** to a value above existing physical expanders (e.g. 32). Click **Save**.
   - Disable **Add Trouble Inputs** and click **Add Now**.
   - In **Programming | Outputs**, rename the resulting outputs so that they include the term Virtual in their names.
2. Select two of the virtual outputs and give them descriptive names. For this example, we will use:
   - C-Bus Area Arming Command
   - C-Bus Area Disarming Command

Creating the Automations

This scenario includes two C-Bus groups (two separate buttons) that will be used to control the area. Therefore, two automations are required, each of which controls a single virtual output.

1. In **Automation | Automation**, create an automation with the following settings:
   - **General** tab:
     - **Name**: C-Bus Arming Button (or similar)
     - **Automation Output Time**: 0
     - **Automation Output**: C-Bus Area Arming Command
     - **C-Bus Application Code**: C-Bus address for the application the arming button is assigned to.
     - **C-Bus Group Code**: C-Bus address for the group the arming button is assigned to.
   - In the **Options** tab, enable the following:
     - **Enable C-Bus Automation Functions**
     - **Use Output Status in C-Bus Function**
     - **C-Bus Operates on Rising Edge**
     - **C-Bus Operates on Falling Edge**
2. Create a second automation with the following settings:
   - **General** tab:
     - **Name**: C-Bus Disarming Button (or similar)
     - **Automation Output Time**: 0

- **Automation Output**: C-Bus Area Disarming Command
- **C-Bus Application Code**: C-Bus address for the application the disarming button is assigned to.
- **C-Bus Group Code**: C-Bus address for the group the disarming button is assigned to.
- In the **Options** tab, enable the following:
  - **Enable C-Bus Automation Functions**
  - **Use Output Status in C-Bus Function**
  - **C-Bus Operates on Rising Edge**
  - **C-Bus Operates on Falling Edge**

At this stage, you should be able to observe the outputs being controlled in the event log or status page. Pressing either C-Bus button will turn the corresponding output ON, and releasing the button will turn it OFF.

Note: Setting an **Automation Output Time** will not work for this application, as the programmable functions require the outputs to be switched on rather than 'on timed'.

Creating Programmable Functions for Area Arming/Disarming

It is necessary to create two programmable functions: the first will arm the area when it receives the arming command, and the second will disarm the area when it receives the disarming command.

1. Navigate to **Automation | Programmable Functions** and select a **Controller** in the toolbar.

2. Add a new programmable function with the name Area Arming Following C-Bus.

3. Set the **Type** to area control.

4. In the **Area Control** tab, set the following:
   - **Area Function**: 2 - Area Arms on Output Turning On
   - **Output to Check**: C-Bus Area Arming Command
   - **Area (Group) to Control**: Required area or area group.

   This function will arm the controlled area only when the assigned output transitions from an OFF to an ON state. As the function is only evaluated at the ON edge, it means that when the output turns off, the area is not required to disarm.
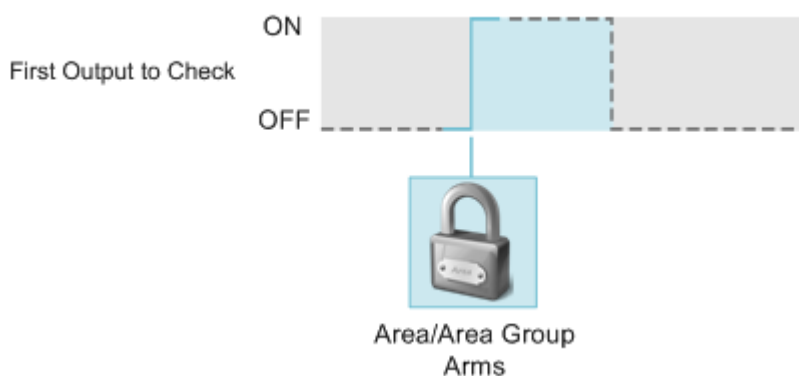
5. Add another programmable function as before, with the name Area Disarming Following C-Bus.

6. Set the **Type** to Area Control.

7. In the **Area Control** tab, set the following:
   - **Area Function**: 3 - Area Disarms on Output Turning On
   - **Output to Check**: C-Bus Area Disarming Command
   - **Area (Group) to Control**: Required area or area group (same as above)

This function will disarm the controlled area only when the assigned output transitions from an OFF to an ON state. As the function is only evaluated at the ON edge, when the output turns off the area is not required to arm.
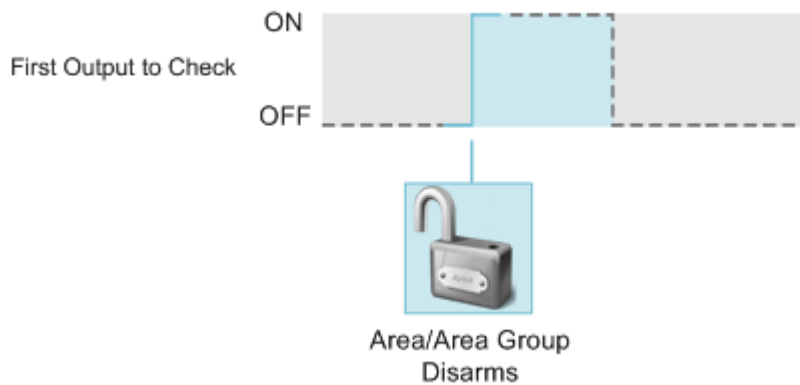
8. Right click on each programmable function and click **Start**. Allow the programming to be downloaded to the controller.

Test the Programming

Once the programmable functions are running successfully, the operation can be tested using an event log, a status page in Protege GX, or the **Monitoring | Areas** page in Protege WX.

- When the arming button is pressed, the C-Bus Area Arming Command output will switch on. The area should then go into exit delay and then arm.
- When the arming button is released, the C-Bus Area Arming Command output will switch off. The area state should not change.
- When the disarming button is pressed, the C-Bus Area Disarming Command output will switch on. The area should disarm.
- When the disarming button is released, the C-Bus Area Disarming Command output will switch off. The area state should not change.