Tutorial

Introduction

This document shows you the basic steps and techniques used to program an ICON system. After understanding this tutorial you may refer to the application notes and example programs to help assist you in writing your own application. Programming a new application consists of:

1. Getting Started
2. Writing Your Program
3. Saving Your Program
4. Moving Around Your Program
5. Deleting Variables
6. Debugging the Programmed
7. Creating Human Machine Interface (HMI) Screens
8. Copying and Moving Program Statements and Modules
9. Enhancing the Program HMI
10. Deploying Your System

This demonstration will create a simple program that causes a variable to have its value ramp from 0 to 100 by adding 10 to the variable every 1 second. When the value reaches 100 the value is reset to 0 and is ramped again. A simple HMI screen is created to show these values.

Before continuing, make sure you have connected your PC to the ICON and have established communication as described in the "ICON Installation" manual. Then connect in programming mode.

Note: Refer to the following manuals under the "Manuals" link in addition to this tutorial:

- ICON Installation
- Configuring ICON System Parameters
- Programming ICON
- Program Instruction List
- Program Debugging
- Creating an HMI (Human Machine Interface)
- HMI Objects List
- ICON File and Transfer Operations

Getting Started

Make sure you have selected the "Program View" tab.

1. In the white programming area you will see a program listed down the left side.
2. Position you mouse within this white programming area and right click.
3. A list will appear with a number of entries. Click on the "New Program" entry.
4. Then click the "OK" button.
5. Any currently listed program will be erased and a new five line skeleton program will be displayed:

Module Loop_1
Module Loop_2
Module Loop_3
Module Loop_4
Module End

This is a minimal program and you can't delete these five lines. Notice that the "Module Loop_1" instruction is highlighted. This is referred to as the "selected" instruction. You may select "Module Loop_2" by clicking on it.

In this document, keys are referred to by <>. So <PgUp> would indicate pressing the "PgUp" key. In addition to clicking on an instruction to select, you may also use the <Up arrow> and <Down arrow> keys. And, after your program gets larger the <PgUp> and <PgDn> keys become useful as well to move through larger blocks of code. A later section in this document titled "Moving Around Your Program" describes other ways to move to, and select an instruction.

When you right click to get the operations list, there is a shortcut listed on the right. You may type the shortcut key instead of doing a right click and clicking on the desired entry. So, to create a new program, you could also type <ctrl><n>. This means hold the <ctrl> key down and press n.You may prefer to use these shortcut keys in the future for much faster operation.

Next set the loop executing rate for each loop:

1. Click the "System Parameter Config" drop down list above the programming white space area and slightly off center to the right.
2. Click the "Password/system" entry.
3. For this demo set "Loop 1 time" to "1000". This sets the loop to execute every 1000 mS or once per second. You can set times for the other loops but for the demo set the others to "0" to disable their execution.
4. Set the "Site title" to "Demo".
5. Then click the OK button.

Writing Your Program

To enter the first program instruction:

1. Click to select instruction "Module Loop_1" if it is not already highlighted. (The insert instruction function will insert immediately after the highlighted instruction.)
2. Right click in the program area and click on "Insert".
3. Use the drop down list to select the "Module" instruction.
4. Enter a caption of "My demo module".
5. Then click the OK button.

Notice the new instruction becomes the top of the list and is selected. You may use the <Up arrow> or <PgUp> key to position to the previous instruction if you wish just to verify that that instruction is still there.

If you want to change your caption after you have entered the instruction:

1. Click to select the "Module My demo module" instruction.
2. Right click in the program area and click on "Recaption".
3. You may now type in a new caption. Note you may not change the captions for the five basic instructions (Loop_1-Loop 4 and End).
4. Click OK.

Now insert an Expression instruction:

1. Click to select instruction "Module My demo module" if it is not already highlighted.
2. Right click in the program area and click on "Insert".
3. Use the drop down list to select the "Expression" instruction.
4. Notice this instruction has a field for "Number of expressions". Enter 2 and click OK.
5. Enter a caption of "Ramp the display variable".
6. Then click the OK button.

This instruction requires configuration:

1. Either double click the newly inserted instruction or click to highlight the instruction and press <Enter>.
2. Notice you have two rows with a "Result", "Status", "Expression" and "Coment" column.
3. The result of the expression evaluation will be put in the result variable. For our demo, enter "ramp" for the first "Result" row entry variable name
4. For the first expression enter "ramp+" . (Yes this is entered as a syntax error on purpose.)
5. Click the "Update" button. This causes the ICON to compile the instruction.
6. Click the "Refresh" button and notice that the status is 6. This indicates a syntax error at column 6.
7. Now correct the expression to read "ramp+10".
8. Click the "Update" button to recompile the instruction.
9. Click the "Refresh" button. Notice this time that the status is 0 which indicates no syntax errors.

At this time the expression is being executed every second. The programming for this system is "Live", meaning as soon as you have inserted and configured an instruction, it starts executing.

Now for the second expression, right click inside the "Result" field in the second expression configuration row. Notice you get a list of all variables created and within scope for this module. Specifically notice that the variable "ramp" is in the list. Click to select this variable. For the expression enter "0 'if' (ramp>100)". As before, click the Update button to compile this statement and then the Refresh button to update the status value. Again, a status of 0 indicates a proper compile. This expression will set the Result variable value (ramp) to 0 only if the expression to the right of the 'if' is true. So when the value of ramp gets set to 110 in the first expression, this second expression resets the ramp variable value to 0.

Notice the () around ramp>100. This is required because expression evaluation is from left to right. So to evaluate all tokens to the right of the 'if' you must enter as (ramp>100).

Saving Your Program

It is very important that you save your work as you program. This is done with the "Save" function. This function saves not only your program, but also all the "System Parameter Config" settings and your HMI screens.

1. Right click in the program area and click on "Save".
2. Enter "Demo1" for program name. (Program names are case sensitive.)
3. Click OK.

Your program has now been saved to flash disk on the ICON under name "Demo1.icn". (Note, all ICON program names have .icn appended automatically.) Now reload:

1. Right click in the program area and click on "Load".
2. Enter "Demo1" for program name.
3. Click OK.

A load deletes the current program and all variables, configurations and HMI screens. Then the program named in flash memory is loaded. As new variables are created their values are set to 0. Therefore, on the first pass, as each variable is encountered for the first time, its value will be 0.

Every time you save your program, a backup is created. In our example this will be Demo1.icn.bkup. If for some reason you can't load Demo1.icn, click the "List Icon Files" button under the load function and click on "Demo1.icn.bkup". The next time you do a save Demo1.icn.bkup is renamed to Demo1.icn.bkup2. This continues after each save until you have five files, Demo1.icn, Demo1.icn.bkup, Demo1.icn.bkup2, Demo1.icn.bkup3 and Demo1.icn.bkup4. After this another save will delete Demo1.icn.bkup4 and all other files will be renamed appropriately and your latest version will be contained in Demo1.icn. Therefore at any given time you will have a saved image of your program from the last save and 4 earlier backups.

With the above procedure you will be well backed up on the ICON itself. However, you should also backup on your own local computer's disk (and from there to possibly another backup say on floppy disk or CD ROM) for archive purposes. Refer to the procedure in the "Download File From ICON to Local Computer" section in the "ICON File and Transfer Operations" manual.

Moving Around Your Program

As your programs get large, often times you need to get to another area of your program and the Up and Down Arrow and Page Up and Page Down keys are inefficient. Note, you can move through your program one module at a time by holding the shift key down while you press the up or down arrow keys. You may also position to an instruction with the following right click functions:

- Loop Position
- Instruction Position
- Variable Position
- Caption Position

You may move directly to the start of any loop by:

1. Right click in the program area and click on "Loop Position".
2. Enter the loop number of 1, 2, 3 or 4 depending on the loop you wish to move to.
3. Click OK.

You may move directly to the next instruction of a given type by:

1. Right click in the program area and click on "Instruction Position".
2. Enter the name of an instruction such as "Expression". You must properly spell the name in full but the match is **case insensitive**. You may click to select for an "Up" (towards the beginning of the program) or "Down" (search toward the end).
3. Click OK.

You may move directly to the next instruction that contains a specified variable name by:

1. Right click in the program area and click on "Variable position".
2. Enter the variable name, or right click in the variable name field and click the variable of interest.
3. Click OK.

The search will begin from the current instruction and search forward (or backward or backward as selected) until a variable is found that contains the text you entered. The search is case sensitive but you don't need to match the entire name. For instance, If you were currently on instruction "Module Loop_2", you could enter "ram" to search backwards to your "Expression Ramp the display variable" entry. It would have matched on variable name "ramp".

You may move directly to the next instruction with a specified caption by:

1. Right click in the program area and click on "Caption Position".
2. Enter all or part of an instruction's caption.
3. Click OK.

The search will begin from the current instruction and search forward (or backward as selected) until a caption is found that contains the text you entered. The search is case sensitive but you don't need to match the entire caption. For instance, If you were currently on instruction "Module Loop_2", you could enter "Ramp the" to search backwards to your "Expression Ramp the display variable" entry.

Deleting Variables

As you are programming, every time you enter a new program variable name, that variable is automatically created. Sometimes you may create a variable and then not want to use it. To remove a variable name:

1. Remove all references to that variable in your program statements and HMI objects.
2. Save your program to flash disk.
3. Load your program from flash disk.

If after you reload and the variable name still appears in your variable list, you have not removed every reference to that name. You may look for program statements with a particular variable name as described in the previous section.

Debugging the Program
To see the instruction in "action", click the "Debug" tab. Now right click in the program area and click on "Inspect variables". After a several second pause, the variable value will start updating. You are now viewing the value in real-time. After each pass through the program loop the ramp variable is incremented by 10. Note, sometimes it appears that it increments by 20. This is not the fault of the program, but instead the "Inspect variable" function checks the value asynchronously from the execution of the program and sometimes misses a loop pass. To avoid this problem you can stop the program every time it executes this instruction:

1. First highlight the Expression instruction, then right click and select "Set break".
2. Shortly thereafter the "Program status" will change from "Run" in green to "Stop" in red.
3. Now right click and pick "Inspect variables". Notice now the value is frozen.
4. Now click the cancel button to exit the view and right click and select "Continue".
5. Again the "Program status" will indicate "Run" and the "Stop".
6. Now when you inspect the variable it will be 10 greater.

The program status indicates stop in red to indicate that the program is not executing. The debug mode is the only place where you can stop the program. Whenever you pick any other tab or lose connection, if the program is stopped, it will be restarted and your break point will be deleted. To remove the break point:

1. Right click and select "Delete break".
2. Right click and select "Continue" or "Run".

You may resume program execution with a Continue or a Run. Continue simply restarts execution with the next instruction. Run will start with the first instruction in loop 1. Note that neither Run nor Continue reset program variables to 0. The only way to do that is to save your program and then load it as described before. You may also stop your program with the "Stop" selection. In this case the program will stop wherever it is when you click "Stop".
One of the best ways to debug is with the single step function:

1. Stop the program with a break point or the stop function.
2. Click on any instruction to start single stepping. (Click on the instruction before the first one of interest.)
3. Right click and click on "Step" to step through the next instruction.
4. After each step you can select "Inspect variables" to see their values.
5. Once "Inspect variables" is up there is a step button at the bottom of that screen.
6. "Module" instructions have no variables to display but you may step through them anyway.
7. When you hit the end of loop 1 the step function continues into loop 2. (This is not normal operation during run and only happens in step.)
8. When the end instruction is reached, stepping stops.
9. You may select a new instruction and begin stepping from this different location at any time.

For example, you may click to select the "Module My demo module" and click step. Now view the variable values. Now click to select the "Module My demo module" again and click step and view the variable values again. In this manner you can repeat executing an instruction or small group of instructions over many times to verify operation.

The "Inspect variable" function also allows you to change a value. Simply enter a new value in the desired value field and click the Update button. You can change values whether the program is running or not. If the program is running, however, it may overwrite your value with one of it's own.
If the variable is a member of a subscripted array, pressing expand will display the rest of the values in the array starting with the original index. The function will display a maximum of 128 elements.

Creating an HMI

You now have a program with a variable named ramp that has values that ramp 0, 10, 20, ... 100, 0, 10, etc. Now we will create a simple HMI screen to display the value in several forms and allow you to change the value.

1. Click to select the instruction "Module My demo module".
2. Click on the "HMI Edit" tab.
3. Right click and select "Detach Window". Notice that you now have a floating frame that you can position and resize.
4. Within your HMI window frame, right click and select "Window Properties".
5. Set "Width" to 640 and "Height" to 480.
6. Enter "Demo" for "HMI Frame Window" title.
7. Now pick your background window color (See below) and click OK.

There are two ways to pick color. The first is to drag the small black circle in the color wheel around until you get the desired color. You can use the slider underneath the color wheel to lighten or darken your choice. The second method is to type a number into the "Color ID" field. This is useful if you already have a color you like from a previous screen. Pick CCFFFF for a nice light blue. If you check the "Web Colors" box it will limit your choices to values that are better displayed as shades of gray (for non color displays) or displays with low color resolution.

Position your window in the center of your screen. Now we will enter our first HMI object:

1. To enable the editing of objects you must first "unlock" the editing. Right click and select "Lock HMI". This will remove the check mark and allow editing of the HMI. Later, if you want to prevent editing functions that allow objects to be moved, resized, copied or deleted you may click this entry again to lock these editing functions. You may also click the padlock at the top of the screen to enable and disable the lock function.
2. Within your HMI window frame, right click and select "Insert".
3. Select the "Frame" object in the drop down list, enter a caption of "Demo" and click OK. Your object will be inserted in the upper left corner.
4. Use drag and drop to position the frame slightly to the right. (Note, you can also use your arrow keys to move it around.)
5. Right click and select "Resize".
6. Grab the lower right corner of the frame with your mouse and drag down and to your right to make it bigger. Drag almost to the lower right of your window. (Note, you can also use your arrow keys to resize. Usually the arrow keys work better than the mouse.)
7. Right click and select "Resize" again. Now the resize mode is turned off.
8. Double click within the frame object to bring up the frame parameters.

9. Enter a "Security Level" of 255. The security level determines which users can display which HMI objects. My way is to assign all objects that everyone should be able to see a value of 255 and objects that only privileged users can see a 1. The range is 1-255.
10. Click the "Attributes" "Configure" button. All HMI objects have this screen.
11. The "HMI Object Caption" has already been set to "Demo" you could change it here if you wish.
12. Set the "Width" to 430 and the "Height" to 270 to match my instructions.
13. Click to select "Black Border".
14. Enter a "Color ID" of EECC99 for a tan color.
15. Leave the "Font Size" at 12 and "Font Color" at black.
16. Click OK to exit the Attributes screen.
17. Click OK again to exit configuring the frame parameters.
18. Drag the frame until its centered in your window.

You should now have a light blue window with the title "Demo" and inside a tan frame with the caption "Demo". The HMI defaults to a 10 pixel grid. In other words, when you drag and resize, it snaps to the nearest (invisible) grid line which is every 10 pixels. You can override this by explicitly setting your "Width" and "Height" parameters. You can also set the snap grid to a different value (including 1) as follows:

1. Right click and click on "Grid Snap Size".
2. Set the drop down list to pick a new grid spacing in the range of 1 to 20 and click "OK".
3. This is a temporary change and is not persistent between editing sessions.
4. To create a persistent change, click the "Program View" tab.
5. Click the "System Parameter Config" drop down list above the programming white space area and slightly off center to the right.
6. Pick the "HMI edit/Serial ports" selection.
7. Enter your grid value and click OK.

Now we will add a variable display object:

1. Return to the "HMI Edit" tab and detach the window.
2. Within your HMI window frame, right click and select "Insert".
3. Select the "Variable" object in the drop down list, enter a caption of "Ramp" and click OK.
4. Drag this object onto the upper left of your frame.

Now you have two HMI objects. Notice you can select an HMI object by clicking on it. The selected object has a pulsing yellow border. Once an object is selected you can drag it, resize it or double click (or press <Enter>) to configure its parameters. To configure the variable object:

1. Double click the variable object.
2. Click the "Attributes" "Configure" button.
3. Click to select "3-D Raised Border" and click OK.
4. Set "Security" to 255.
5. Set "Write Security" to 1. (This will allow you to change the value. Setting a 0 would disable changing the value.)
6. Right click in the "Variable" field and select the ramp variable.

When you execute your HMI, the screen associated with Module Loop_1 is the first screen to display. So we need to create an HMI screen for this module with a button to access our demo screen:

1. Click the "Program View" tab.
2. A dialog box will come up informing you that you have repositioned your previous HMI window. Click OK to save the new position.
3. Click to select Module Loop_1.
4. Click the "HMI Edit" tab.
5. Type <D> to detach.
6. Right click and pick "Window Properties".
7. Enter "Main Menu" for the HMI Frame Title and click OK.
8. Right click and pick "Insert".
9. Select the "Button" object.
10. Enter "Demo" for your caption and click OK. (This button is going to select our "Demo" HMI window.)
11. Double click the object.
12. Under "Attributes" press the "Configure" button.
13. Select "3-D Raised Border" and click OK.
14. Set security level to 255.
15. Select "Hide" for the "Window Control" parameter.
16. "Select Loop" should be set for "Loop 1".
17. Click the "HMI Window" button for "Select HMI". You are presented with a list of HMI window names. Pick "Demo" and click OK.
18. Click OK to exit editing the button object.
19. Click the "HMI Run" tab to view your completed HMI screens.

First your "Main Menu" screen comes up. When you click the "Demo" button your demo screen comes up. Notice the value is updating. Since we forgot to put an exit button on, we can't return to the main menu. For now:

1. Right click in the HMI screen.
2. Click "Select HMI Window".
3. Click to select "Main Menu" and click OK.

To add the exit button:

1. Click to select the "Program View" tab.
2. Select the "Module My demo module" and click the "HMI Edit" tab.
3. Repeat the procedure to add a button to take you back to the main menu. Set the button caption to "Exit".
4. Drag the button to the right of the variable display object.
5. You may resize the button to a smaller size if you wish so it looks better proportioned. Width=50 and Height=25 is nice.
6. Configure this button as previously described but select "Main Menu" for the "Select HMI" parameter.
7. Click the "HMI Run" tab.
8. Now you should be able to go back and forth between HMI screens with your buttons.

Copy and delete HMI objects and drag a magnetized frame:

1. Click to select the HMI variable object.
2. Right click and select "Copy". A copy of the HMI object is created to the right.
3. Drag so it is to the right of the first object.
4. Do another copy.
5. To delete this object, right click and select "Delete". (Just to demonstrate that you can delete an object.)
6. Right click and select "Magnetize".
7. Drag the frame. Notice all items on the frame drag with it!
8. Right click and select "Magnetize" again to turn magnetize off.
9. Now when you drag the frame the other objects do not drag with it.
10. Double click the second copy of the variable object to configure.
11. Click the "Attributes" button and set the border to "3-D Recessed Border" and click OK.
12. Set "Write security" to 0.
13. Set "Min value" to 0 and "Max value" to 100.
14. Set type to "Meter" and click OK.
15. Click the "Variable Object Options" button for "Style".
16. Click "Value" for "Display type".
17. Now resize so the meter looks better.
18. Select the "HMI Run" tab and watch the meter move!
19. Click the first variable object which will cause a data entry field to appear.
20. Enter a value of 5 and click OK.
21. Notice the value ramps 5, 15, 25 ... until it reaches 95, then it wraps back to 0 and increments as before.

Now we will add a strip chart:

1. Click to select the "Program View" tab.
2. Select the "Module My demo module", click the "HMI Edit" tab and detach your window.
3. Resize your frame bottom so that the lower edge is just below the biggest variable (meter) object.
4. Set magnetize enabled, select the frame and drag up so you have lots of room underneath the frame for our strip chart.
5. Insert a "Graph_vars" HMI object and enter a caption of "Ramp".
6. Drag down below the frame and position on the left side.
7. Resize so the lower right frame is in the lower right corner of your window. Width=630 and Height=250 is good.
8. Set security to 255.
9. Set the parameter "Interval" to 5M/1S. (This indicates that the graph is 5 minutes long and updated every 1 second.)
10. Right click in the "Variable 1" field and select ramp.
11. Enter "Ramp" for "Caption 1" and click OK.
12. Select the "HMI Run" tab and watch the strip chart graph!
13. As before, click on the first variable object and enter a new value such as 25 and watch the graph take off from there.

Copying and Moving Program Statements and Blocks

Now lets copy the Expression instruction as an example of how you can quickly duplicate an instruction:

1. Select the "Program View" tab.
2. Click to select your Expression instruction.
3. Right click and select "Block start". Notice at the bottom that the "Start Block" field displays "Expression Ramp the display variable".
4. Right click and select "Block end". Notice at the bottom that the "End Block" field displays "Expression Ramp the display variable".
5. Right click and select "Copy".

Notice you now have two copies of the same instruction. Change the caption of the new instruction to "Ramp the display variable 2". Now change your configuration so the ramp variable name is changed to ramp2.

You can also copy several instructions:

1. Click to select your first Expression instruction.
2. Right click and select "Block start". Notice at the bottom that the "Start Block" field displays "Expression Ramp the display variable".
3. Click to select your second Expression instruction.
4. Right click and select "Block end". Notice at the bottom that the "End Block" field displays "Expression Ramp the display variable 2".
5. Click to select "Module Loop_2".
6. Right click and select "Copy".

You will see that a copy of the two expression statements has now been copied after the Loop_2 Module. To Move these statements from Module Loop_2 to Module Loop_3:

1. Click to select your first Expression instruction in loop 2.
2. Right click and select "Block start".
3. Click to select your second Expression instruction in loop 2.
4. Right click and select "Block end".
5. Click to select "Module Loop_3".
6. Right click and select "Move".

You will see that the two instructions have been moved from loop 2 to loop 3. To delete these two instructions:

1. Click to select your first Expression instruction in loop 3.
2. Right click and select "Block start".
3. Click to select your second Expression instruction in loop 3
4. Right click and select "Block end".
5. Right click and select "Delete".
6. Click OK for confirmation in the dialog box.

You will see that the two instructions have been deleted. You can copy an entire module as follows:

1. Click to select the instruction "Module My demo module" in loop 1.
2. Right click and select "Block start".
3. Right click and select "Block end". Notice at the bottom that the "End Block" field displays "Expression Ramp the display variable 2". Even though you did a block end

function on the Module instruction, it automatically included all instructions in the module.

4. Click to select "Module Loop_2".
5. Right click and select "Copy".

Notice that a copy of the entire module was made in loop 2. When you copy modules, all instructions in the module and the HMI for that module are copied. To verify, click to select the new "Module My demo module" in loop 2. Click on the "HMI Edit" tab to see that your previously created HMI has also been copied over. You delete the HMI as follows:

1. Select the "Program View" tab.
2. Click to select the instruction "Module My demo module" in loop 2.
3. Right click and select "Delete HMI Objects".

Click on the "HMI Edit" tab to see that your previously created HMI has been delete. To copy just the HMI objects from your first module to the second:

1. Select the "Program View" tab.
2. Click to select the instruction "Module My demo module" in loop 1.
3. Right click and select "Block start".
4. Click to select the instruction "Module My demo module" in loop 2.
5. Right click and select "Transfer HMI Objects".

You can now verify that the HMI objects have been transferred back.

Enhancing the Program and HMI

Now we will add some code so a user from the HMI at HMI run time can set the high and low values for the ramp function. First add a Limit instruction:

1. Click to select instruction "Expression Ramp the display variable".
2. Right click in the program area and click on "Insert".
3. Use the drop down list to select the "Limit" instruction.
4. Enter a caption of "Set maximum/minimum ramp value".
5. Then click the OK button.
6. Double click to configure the instruction.
7. Configure as follows:

Input:	ramp
Result:	reset
Limit:	Null
High:	1
Went high:	1
Window:	0
Went window:	0
Low:	0
Went low:	0
High limit:	50

High clear:	10
Low clear:	0.
Low limit:	0

The variable reset will be 0 untill the value of ramp goes above 50. Then it will be set to 1. When the value of ramp goes below 10 the value of reset will be set to 0. We will now modify our expression instruction to make the ramp variable ramp up and down between the High limit and the High limit clear values.

1. Click to select instruction "Exression Ramp the display variable" .
2. Double click to configure the instruction.
3. Change the first expression from "ramp+10" to "ramp+10 'if' (reset==0".
4. Change the second expression from "0 'if' (ramp>100)" to "ramp-10 'if' (reset==1)"
5. Click "Update" to compile and "Refresh" to make sure there are no syntax errors.
6. Now Select the "HMI Run" tab and watch the your variable objects move between 10 and 50!

Now we will add an HMI object that will allow a user at HMI run time to change the limit values in the Limit instruction. This is an example of how you can make certain instruction parameters accessible to a user after deployment.

1. Slect the HMI edit tab.
2. Within your HMI window frame, right click and select "Insert".
3. Select the "Instruction" object in the drop down list, enter a caption of "Change Limits" and click OK. Your object will be inserted in the upper left corner.
4. Use drag and drop to position the object in an appropriate place.
5. Configure this object for a raised 3-D border.
6. Then click the "Instructions" button.
7. Use the drop down list under "Selected Instruction" to display a list of instructions in the module. Click to select your newly entered Limit instruction.
8. Click the "Selected Override Template" button.
9. Enter "Set maximum/minimum ramp value" for "Override Template Name".
10. Enter "Enter Maximum Ramp Value" for the "High limit" field.
11. Enter "Enter Minimum Ramp Value" for the "High clear" field.
12. Leave all the rest of the fields blank.

Now run your HMI again and click your newly entered instruction object. Notice you can change the values for just those two fields. Enter a value of 80 for "Set maximum/minimum ramp value" and 40 for "Enter Minimum Ramp Value" and watch the values vary between these two limits.

<u>Deploying Your System</u>

After you have completed a project you should perform the following steps:

1. From the "Programming tab" drop down list select "Password/system".
2. For "Watchdog" select "Enabled". (If the ICON should fail, this will cause it to automatically restart.)

3. Click the "Table" button. A dialog box with "Unencrypted passwords are about to be sent" will be displayed. Click OK.
4. Now resize the table for the number of passwords you desire. Normally you need at least three entries as described below. The first row is the programming password. The rest of the rows are for executing the HMI user screens only.
5. Enter a password, the Low and High security levels and encryption keys for each row. (More about this below.)
6. Click OK to close the table view and click OK again to close the "Password/system" window.
7. From the "Programming tab" drop down list select "Connection". Make sure your "IP/netmask", "Port number", "Broadcast", and "Gateway" fields are set properly. Click OK.
8. Save your program under name icon. (This is the file automatically executed at power on.)
9. Set switch 3 on the ICON to "on" if you don't want to allow programming mode.
10. From the "Programming tab" drop down list select "Password/system".
11. For "Watchdog" select "Reset".
12. Click OK. Now your ICON system will reboot.
13. After about one minute you can reconnect. If you changed your IP address or port number you must change your web browser URL.

For secure passwords use at least 8 characters. You may use any printable characters except |, ~, { and }. For the encryption keys you may use any numbers 0 to 9 or letters A, B, C, D, E or F. (This an 8 digit hex number.) If you pick a password of nothing (you delete all characters in the password entry field (which displays as "none")) and an encryption key of 00000000 then you can log on without entering any password or encryption key. This is obviously very insecure.

When you created each HMI object on the HMI screens you assigned a security level. If you use 1 for all objects that can change values or parameters and 2 or higher for display objects, you should have at least 3 entries in your password table. The first entry is for programming mode. The second entry will have your password for the users that can change parameters. Pick a Low security of 1 and a High security of 255. The third entry will have your password for the users that can only view items on HMI screens. Pick a Low security of 2 and a High security of 255.

Try logging onto your system with a password different than the programming password. Notice you go right to the HMI screens.