QuickBuilder QuickStart Guide

Doc. No. 951-530030-005

© 2008 Control Technology

25 South Street
Hopkinton, MA 01748

Phone: 508.425.9595
Fax: 508.435.2373

Wednesday, January 09, 2008

Control Technology

Table of Contents

1 Chapter 1: Overview 5

2 Chapter 2: QuickBuilder Installation and Setup 6

- Installing the QuickBuilder Software 6
- Computer and Controller Setup 6
- Starting WebMON 10
- Mounting a RAMdisk 14

3 Chapter 3: Your first QuickBuilder project 15

- Let's get started 18
- Configuring the Model 5300 controller in QuickBuilder 19
  - Adding a controller 19
  - Configuring the hardware: rack and IO 20
  - Configuring the hardware: M3-18A card 23
  - Naming the inputs 25
- A sample program 27
  - Creating a StartTask 27
  - Adding comments 28
  - Adding the next step in the program: Punch 30
  - Entering QS4 code: Step Editor and Code Windows 33
  - Finishing your program 37
  - Saving your program 41
  - Commission 42
  - Maintenance 48

4 Appendix A - Detailed Overview of the QuickBuilder Automation Suite Environment 52

- Configuration 53
  - Project Manager 53
  - Property Inspector 54
  - IO Configuration 55
- Program 56
- Program Libraries 58
- Test & Debug tools 59
- Maintenance 60
- QuickBuilder benefits 61

5 Appendix B - QuickStep 2 Code 62

6 Appendix C - One Step Program 64

Index 65

© 2008 Control Technology

QuickBuilder QuickStart Guide

Copyright © 2007 Control Technology Corp. All Rights Reserved.

Control Technology Corp.
25 South Street
Hopkinton, MA 01748
Phone: 508.435.9595 • Fax 508.435.2373

Document No. 951-530030-005

⚠ **WARNING:** Use of CTC Controllers and software is to be done only by experienced and qualified personnel who are responsible for the application and use of control equipment like the CTC controllers. These individuals must satisfy themselves that all necessary steps have been taken to assure that each application and use meets all performance and safety requirements, including any applicable laws, regulations, codes and/or standards. The information in this document is given as a general guide and all examples are for illustrative purposes only and are not intended for use in the actual application of CTC product. CTC products are not designed, sold, or marketed for use in any particular application or installation; this responsibility resides solely with the user. CTC does not assume any responsibility or liability, intellectual or otherwise for the use of CTC products.

The information in this document is subject to change without notice. The software described in this document is provided under license agreement and may be used and copied only in accordance with the terms of the license agreement. The information, drawings, and illustrations contained herein are the property of Control Technology Corporation. No part of this manual may be reproduced or distributed by any means, electronic or mechanical, for any purpose other than the purchaser's personal use, without the express written consent of Control Technology Corporation. Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

See www.ctc-control.com for the availability of firmware and software updates or contact CTC Technical Support.

Conventions used in this guide:

When you need to type some text it will appear like this:

Type: **`StrtInput`**

where the characters you need to type are shown in a larger Bold Courier font.

Notes for QuickStep 2 users will appear in highlighted boxes as shown below. These are added to assist our existing customers in transitioning to QuickBuilder; new customers can skip over these boxes.

```
A note for QuickStep 2 users:
```

1 Chapter 1: Overview

This document is a step-by-step guide to building your first QuickBuilder project. It will take you through installing the software, connecting to the controller, and creating and downloading a simple test program into CTC's 5300 series Blue Fusion automation controller.

More detailed information on QuickBuilder and its usage is available in *Document No. 951-530020: QuickBuilder Reference Guide*.

```
A note for QuickStep 2 users:

If you are familiar with programming with QuickStep2.x, you will
find that QuickBuilder allows for much more flexibility in
programming style.

You can still program in a very similar fashion to QuickStep2.x,
but QuickBuilder also allows you to break programs into reusable
and flexible tasks, events and functions that encourage good
programming practices.

In this document we will show two different ways to program the
same simple program. The first way will be structured closely to
the way a QuickStep2.x application is programmed. The second
will be a more streamlined program that will combine the code
from several traditional Quickstep steps into one 'step'.

You can reference Appendix B to see the equivalent code in
QuickStep2.x, if you're familiar with that version of QuickStep.
```

2 Chapter 2: QuickBuilder Installation and Setup

2.1 Installing the QuickBuilder Software

Using all default options:

1) Install Microsoft .Net Framework Runtime using this link:

http://www.microsoft.com/downloads/details.aspx?FamilyID=0856eacb-4362-4b0d-8edd-aab15c5e04f5&displaylang=en

2) Install QuickBuilder using the QuickBuilder Setup on the QuickBuilder CD.

3) Install QuickBuilder Support on the QuickBuilder CD.

4) Install the Java Runtime Environment (required for WebMON) using this link:

http://java.sun.com/j2se/1.4.2/download.html

⚠ **Note:**

QuickBuilder requires specific Administrative rights in order to authorize the application. Because of Vista's new security mechanism called User Account Control (UAC), the definition of an Administrator and what rights an Administrator is granted differs slightly compared with Windows XP.

Therefore, additional steps are required when authorizing QuickBuilder under Vista.

The user needs to find the location of the installed application which is usually at:

C:\Program Files\Control Technology Corporation\QuickBuilder

The user needs to right-click the application file named "QBApp" and choose "Run as Administrator."

Once the application is launched with full rights and privileges, it can be authorized.

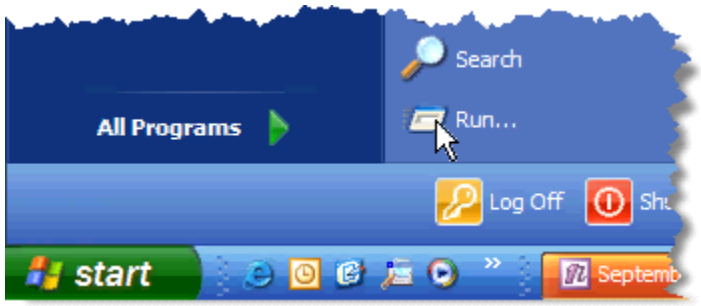
The user can run QuickBuilder the "normal way" (i.e. using the Start Menu or the desktop shortcut) after authorization.

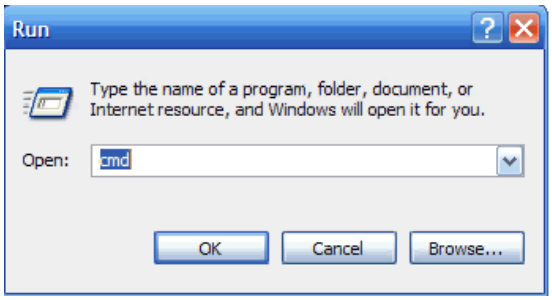
2.2 Computer and Controller Setup

The first thing we will do is check compatibility of the IP Address of your computer and your controller.

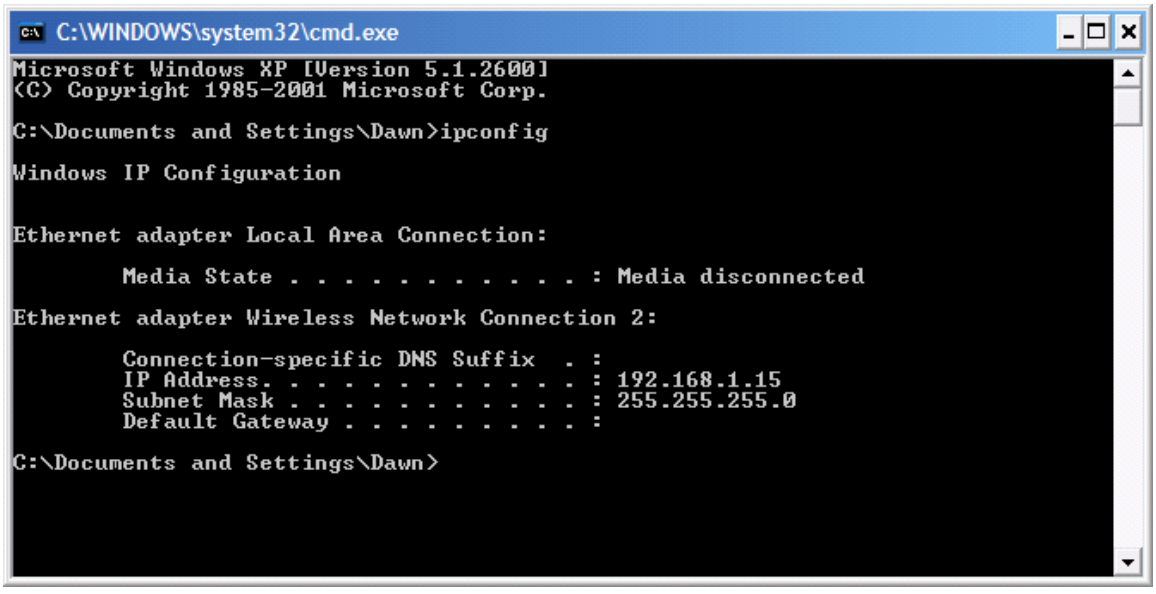
If you are already familiar with setting up your Computer and Controller for communications, you can set it up however you want and jump to the Starting WebMON section. The default IP Address for a 5300 controller is 192.168.1.53.

1. Connect one of the controller's Ethernet ports directly to your computer using an Ethernet cable or connect the controller to the same network your computer is on via an Ethernet switch or hub. (The 5300 uses intelligent ports, so you can use either a standard Ethernet cable, or a crossover cable).
2. Click on the start menu on your computer.
3. Click **Run**.
4. Type: **cmd**
5. Click OK.





Type **IPConfig** in the cmd window. You will see something similar to this:

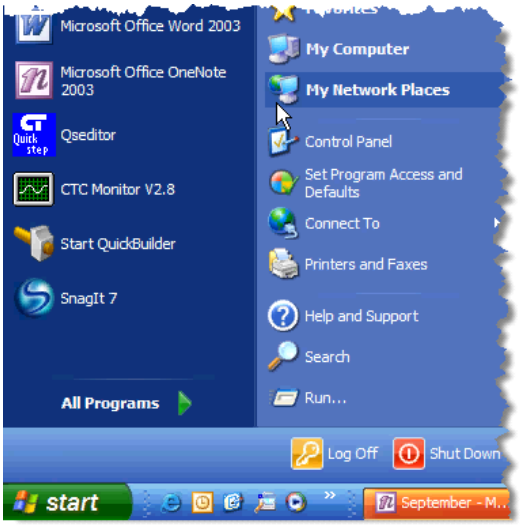


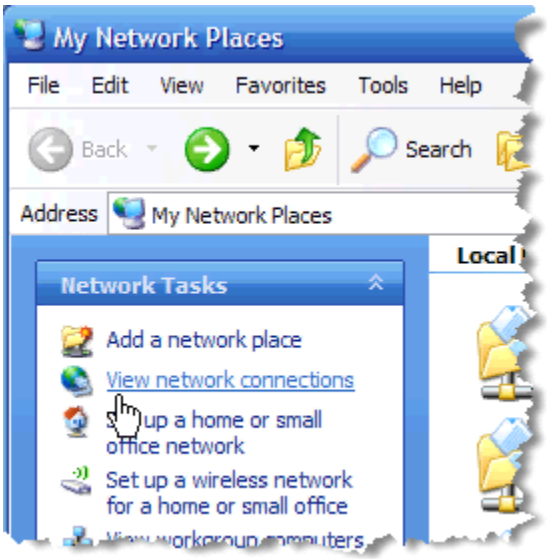
If your IP Address is 192.168.1.XXX (as shown above), you are all set and don't have to change anything. Skip to the Starting WebMon section of this manual.

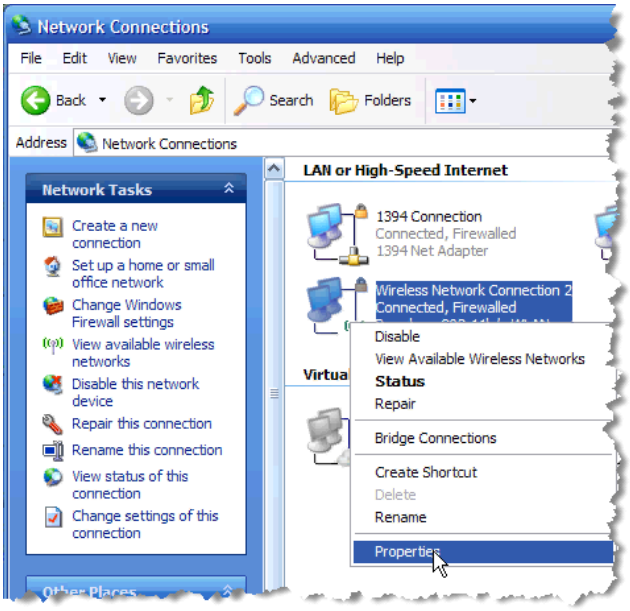
⚠ **Note:** We will show you how to set up your computer using Windows XP. The steps will be similar but will vary slightly using other versions of Windows. Please see your IT contact if you need help with your version of Windows.

Otherwise:

1. Click on the start menu on your computer.
2. Click **My Network Places**.
3. Click **View network connections**.

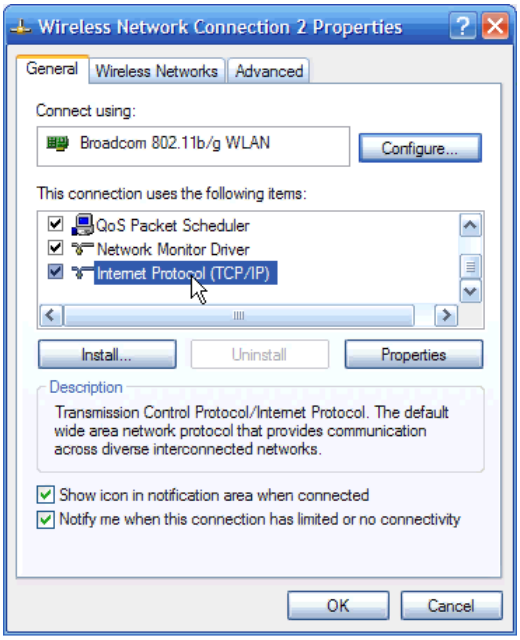






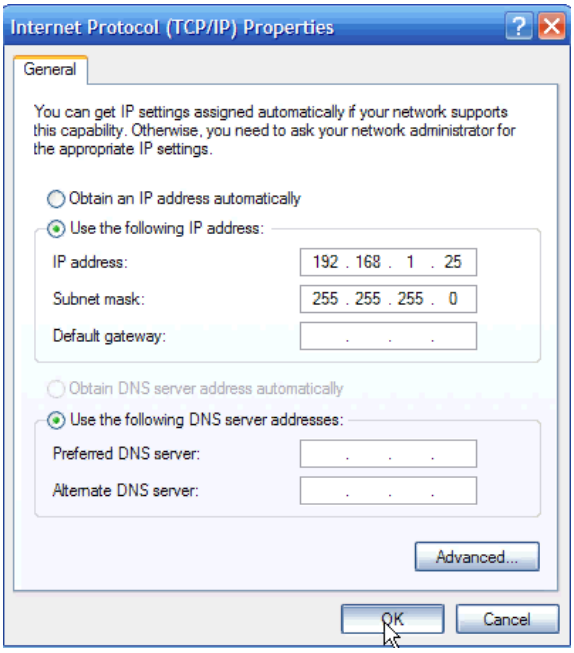
4. Right click on the network connection you have the Blue Fusion 5300 connected to.
5. Select **Properties**.

⚠ **Note:** In this example the controller is connected to the network and a wireless computer is being used. Be sure to use the correct LAN connection.

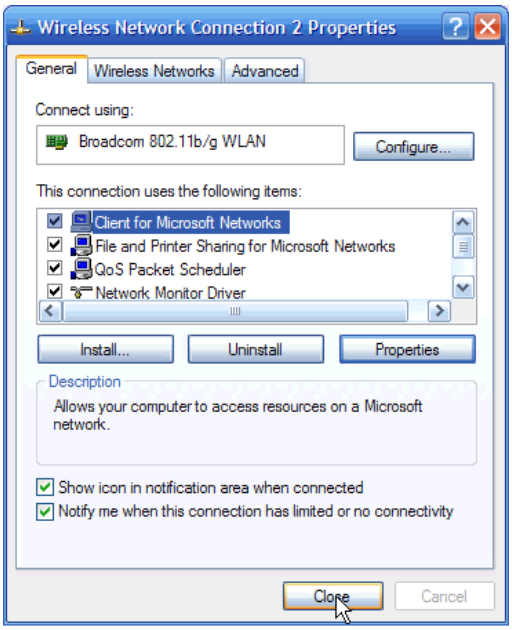


6. Click on Internet Protocol in the Network Properties Window.
7. Click OK.

⚠ **Note:** You will see a window like the one below. *Be sure to make a note of your current settings so you can go back to those settings later.*



8. Use the following IP address:
9. Enter the IP Address and Subnet mask as shown to the right.
10. Click OK.



11. Click on **Close** in the Network Properties Window.

The IP Address on your computer will now be compatible with the Controller and you can now use WebMON to set up your Model 5300 controller.

2.3 Starting WebMON

WebMON is CTC's web based configuration and diagnostic utility that is factory installed into all Blue Fusion controllers. It is served up by the controller's web server to your PC's web browser, so there's no need to load application software onto your PC. WebMON allows you to perform many different tasks such as configuring your controller's hardware and setting security and email features. It can also be used to remotely control and monitor your project. However, in this Quick Start guide we will only be showing you how to use WebMON to configure your Ethernet settings and how to set the real time clock. For detailed information on all of WebMON's features, visit the CTC website at

http://www.ctc-control.com/customer/techinfo/docs/5200_951/951-520012.pdf

To start WebMON, do the following:

1. Open Internet Explorer.
2. Type: http://192.168.1.53 in the address field. The following web page will be displayed:

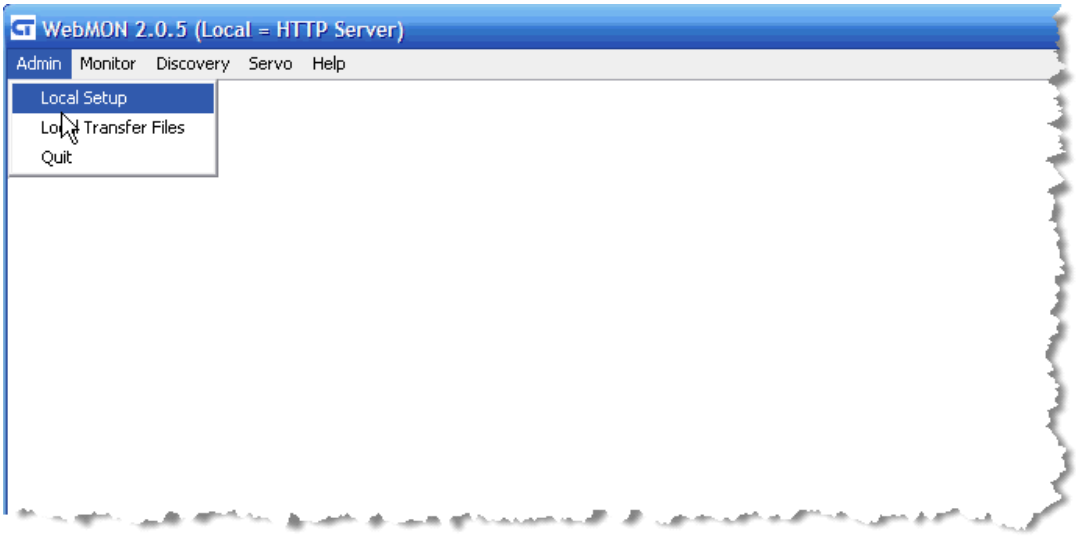


The **Feature highlights**, **Controller specifications**, **Dimensions** and **Configuring your 5300** links will take you to the appropriate CTC web page if your computer has an Internet connection.

After a short delay you should see the following screen pop-up:

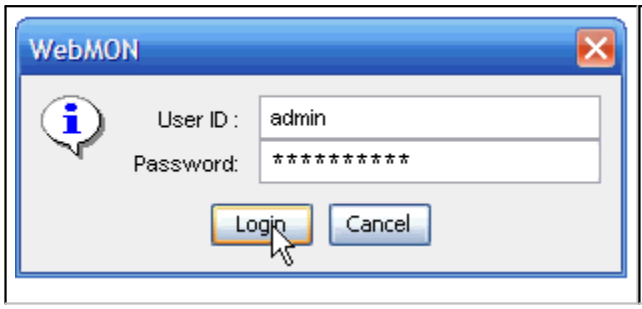


Click **Yes**. After another delay you should see the following screen pop-up:



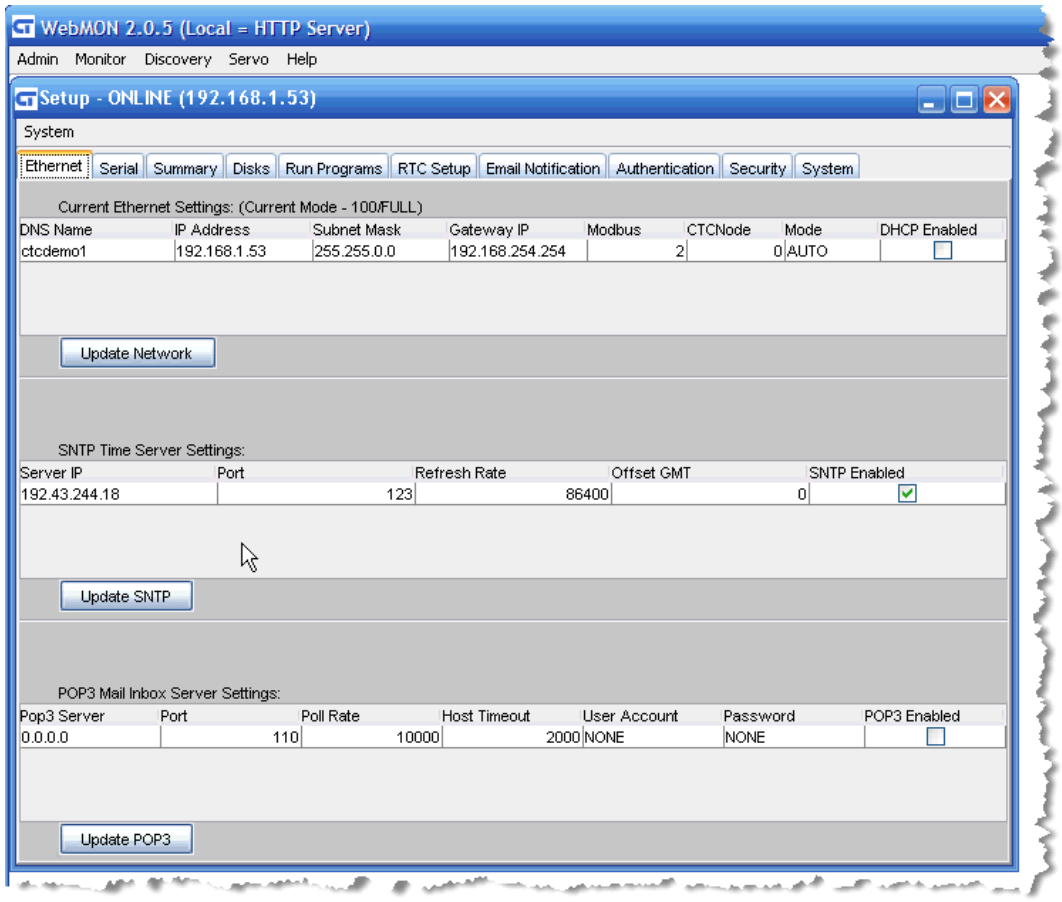
Now:

1. Click on **Admin**.
2. Click on **Local Setup**.



Now:

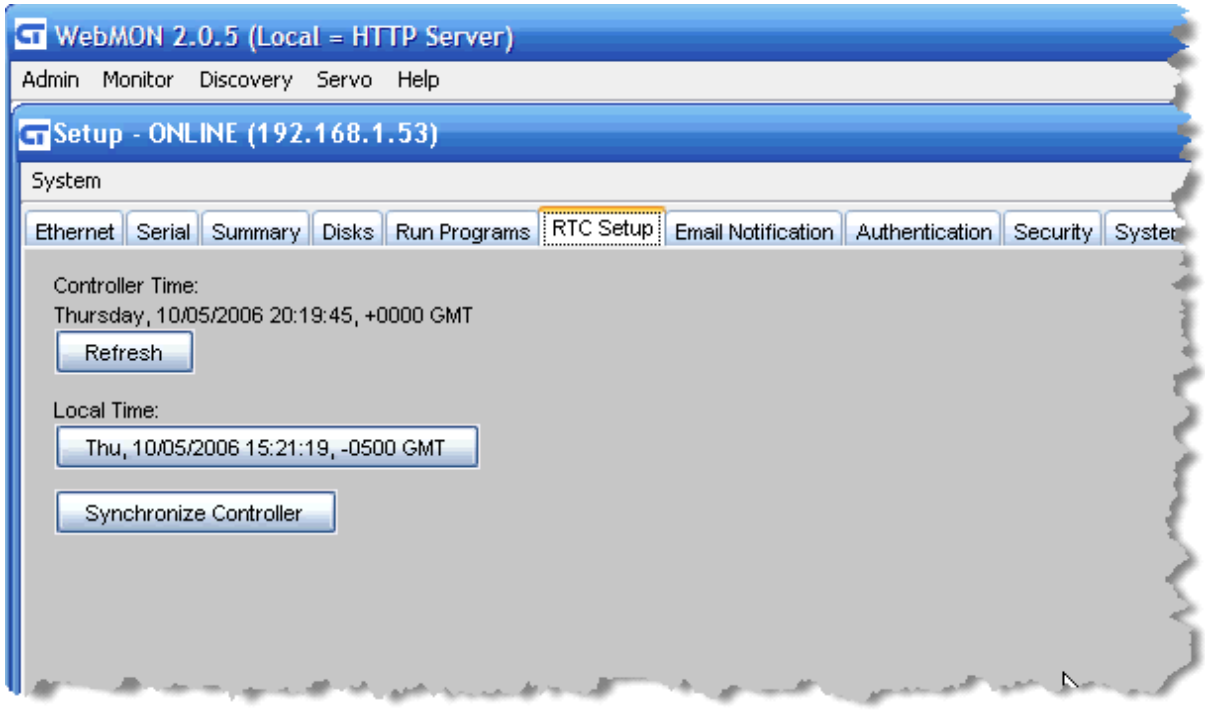
1. Type **admin** as the UserID.
2. Type **BlueFusion** as the Password.
3. Click Login.



You will see the window shown above. This is where you can change the DNS Name, IP Address, Subnet Mask, Gateway IP and other communication settings of the controller. Simply enter in the information that needs to be changed and Click on **Update Network**.

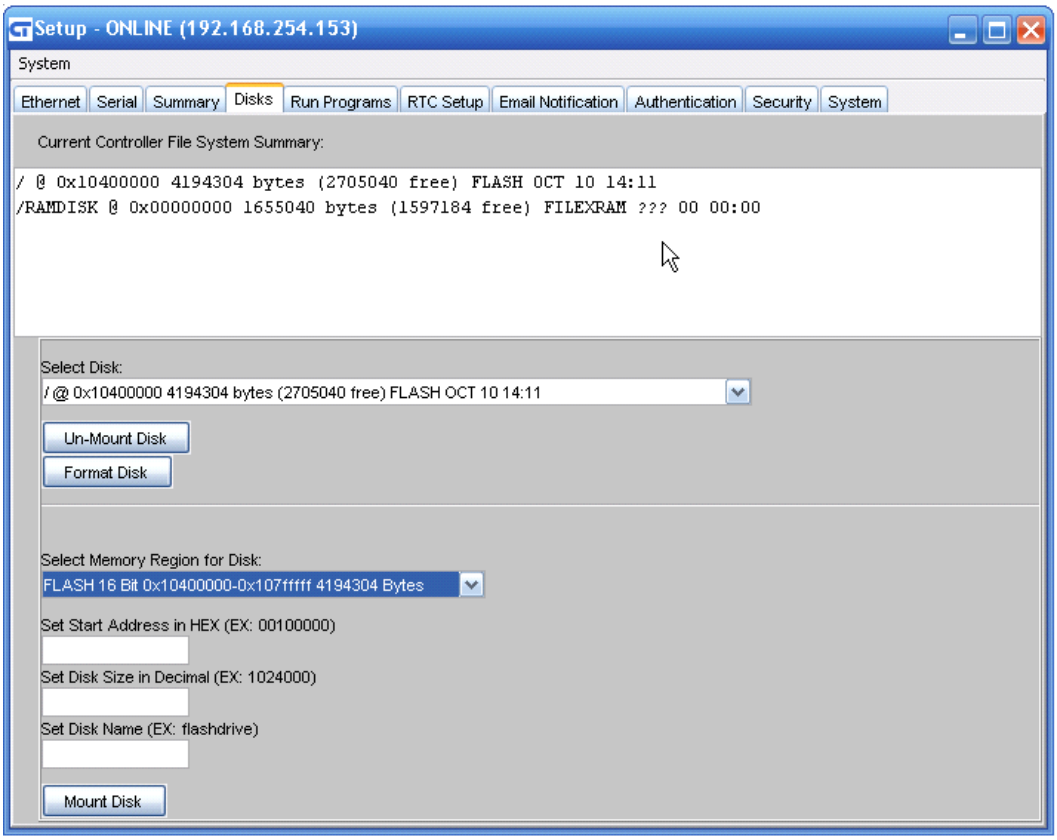
Note: Any changes you make to the Network Settings will be stored to flash memory and take effect on the next Power Cycle.

Now: Click on the **RTC Setup** tab. The controller time will be displayed. You can update this by using the **Refresh** button. The controller time can be synchronized to the Local Time shown above by clicking on the **Synchronize Controller** button.



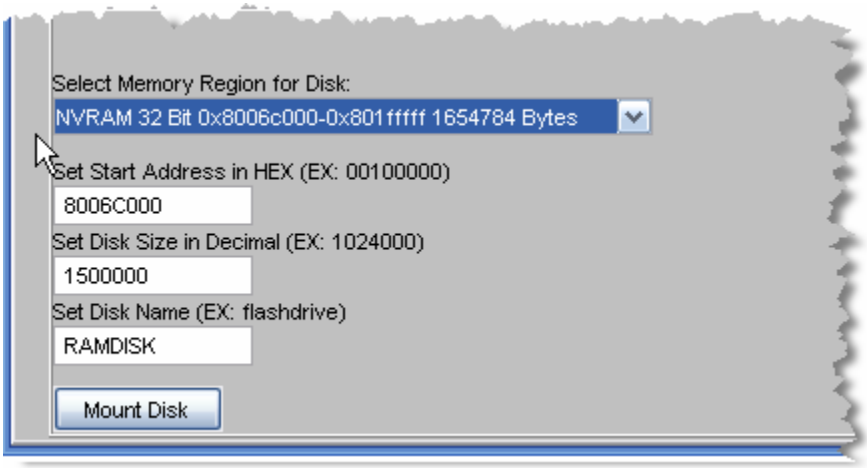
2.4 Mounting a RAMdisk

Click on the **Disks** tab.



You should see a screen similar to the one above.

⚠ **Note:** If you do not see a /RAMDISK as shown above, your unit needs to be configured with one. You will need to fill in the **Set Start Address**, **Set Disk Size in Decimal**, and **Set Disk Name** as shown below and click on **Mount Disk**. **Be sure to select NVRAM for the Memory Region selection box.**



3 Chapter 3: Your first QuickBuilder project

The Part Stamping Project:

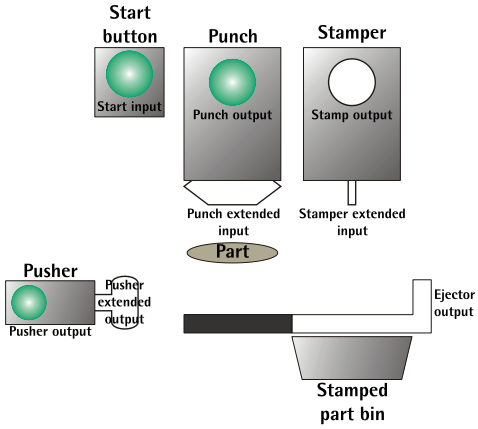
For your first project we will be taking you through the steps necessary to automate a part stamping machine. The machine is designed to have an operator manually load a part, then press the start button to automatically stamp the part and move it into the stamped part bin.

The QuickBuilder Process:

Even though this is a very simple project, we will still take you through all four stages of a QuickBuilder project. Those stages are:

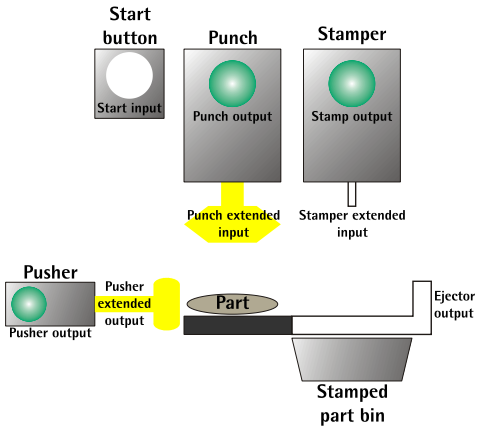
1. **CONFIGURE**:
   a. Select the controller hardware to be used
   b. Define and assign the resources
2. **PROGRAM**:
   a. Create the logic program
3. **COMMISSION**
   a. Download the project to the controller
   b. Test and debug the application
4. **MAINTAIN**:
   a. Auto generate the project documentation file

The sequence of operations for the stamping machine are as follows:



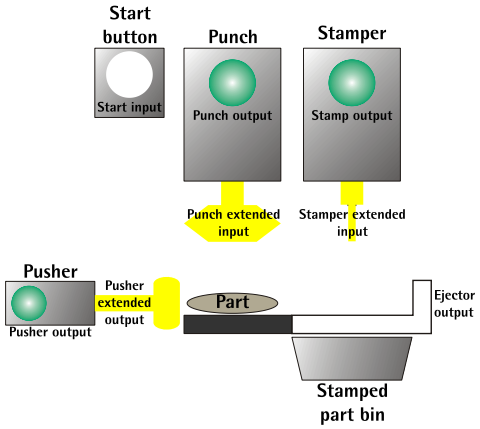
Steps 1 and 2

1. Wait for a StartInput (Input 1) to be on.
2. Turn on a Punch Output (Output1) and a Pusher Output (Output2).



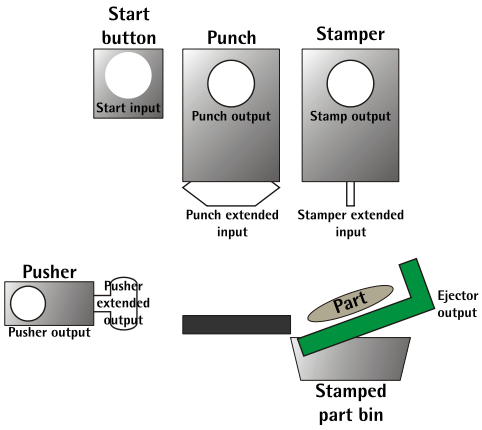
Steps 3 and 4

3. Wait for a PunchExtended Input (In1) and PusherExtended Input (In2) to be on.
4. Turn on a Stamp Output (Output 3)



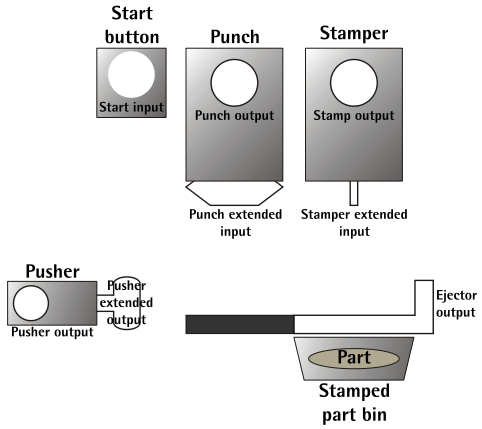
Step 5

5. Wait for a StampExtended Input (In3) to be on.



Step 6

6. Turn off the first 3 outputs (Punch, Pusher, and Stamp) and turn on the Ejector Output (Out4).

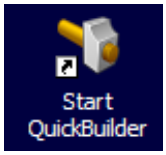


Steps 7 and 8

7. Delay 1 second.
8. Turn off The Ejector Output, go back to Step 1 to wait for the Start input again.

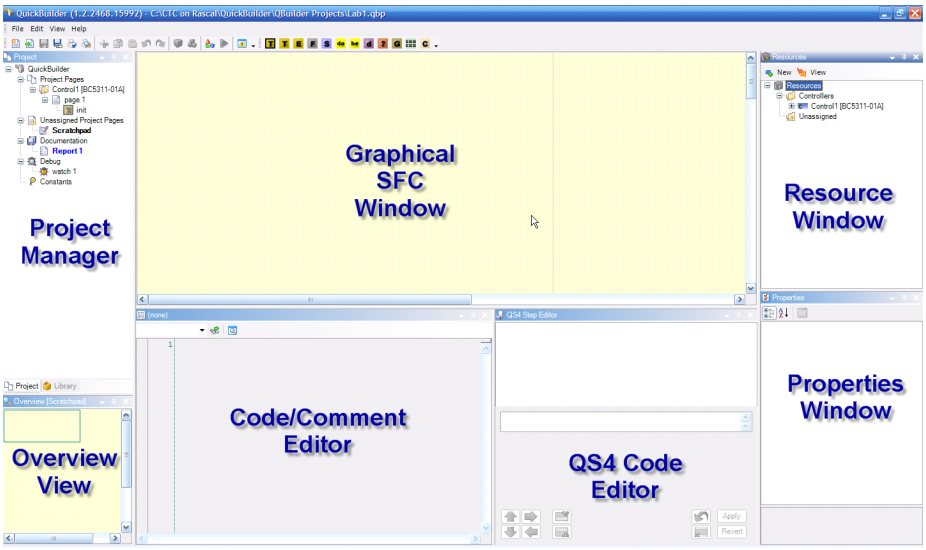
We will show two different ways to program this same simple program. The first way will be structured closely to the way a QuickStep program needed to be programmed. The second will be a more streamlined program that will combine the code from several typical Quickstep steps into one ‘step’.

3.1 Let's get started



To start QuickBuilder, double-click on the Start QuickBuilder icon.

The first time you open your QuickBuilder, your project will look like this:



In a nutshell, the different windows labeled above have the following functions:

Project Manager: Holds the logic, documentation and debug views for the project

Graphical SFC (Sequential Function Chart) Window: This is where you use graphical elements like steps and decision blocks to organize the tasks in your project.

Resource Window: Allows the user to define their physical configuration (controllers, racks, I/O)

Properties Window: Allows you to set the property settings of the Resources in the Resource manager.

Code/Comment Editor: Used to directly type and edit QuickStep4 (QS4) code and comments.

QS4 Code Editor: Used to enter and edit QuickStep4 (QS4) code using the familiar "fill-in-the-blanks" method.

Overview View: Shows you an overview of your page and allows you to zoom.

Note: See the Appendix A for more specific information on QuickBuilder Windows listed above.

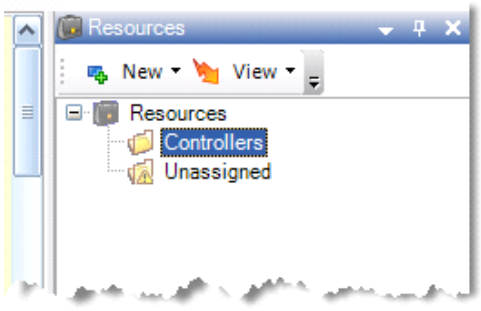
3.2 Configuring the Model 5300 controller in QuickBuilder

We will now configure our 5300 controller. In this example, we will be using a controller with the following configuration:

- BC5311-01A CPU (Single bay, GEN 3 CPU, 4 MB Flash)
- M3-10A Digital I/O module (16 Digital Inputs, 16 Digital Outputs, VS VDC Sinking)
- M3-18A Digital Switch/LED module

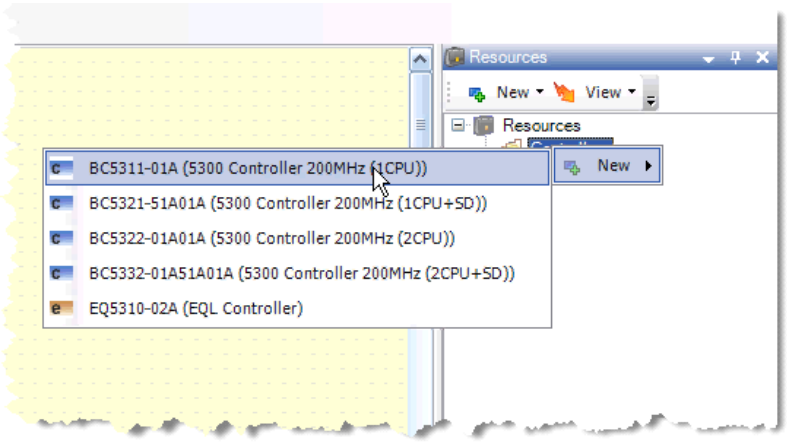
3.2.1 Adding a controller

1. Right click on **Controllers** in the Resources window.

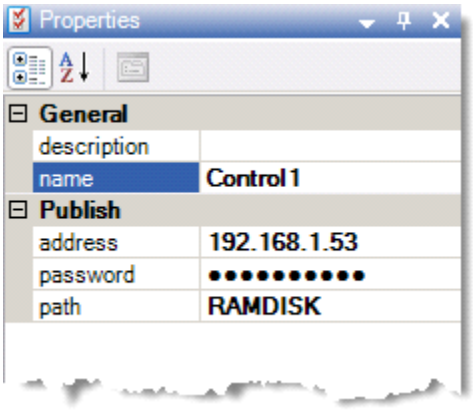


2. Place cursor over **New**.

3. Click on your controller configuration.



In the Properties Window shown below:

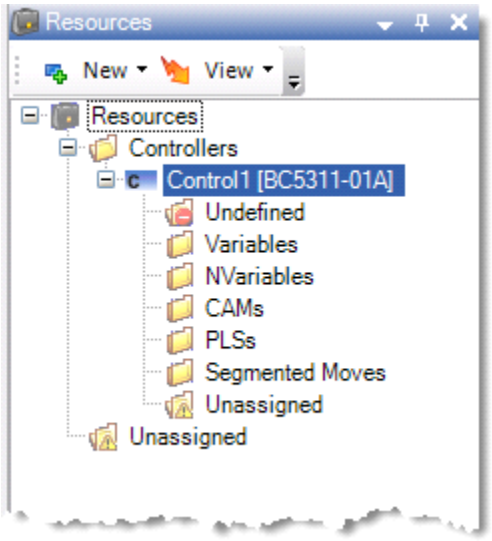


1. Set name to **Control1**.
2. Set up the IP address.
3. Set the path to **RAMDISK**.

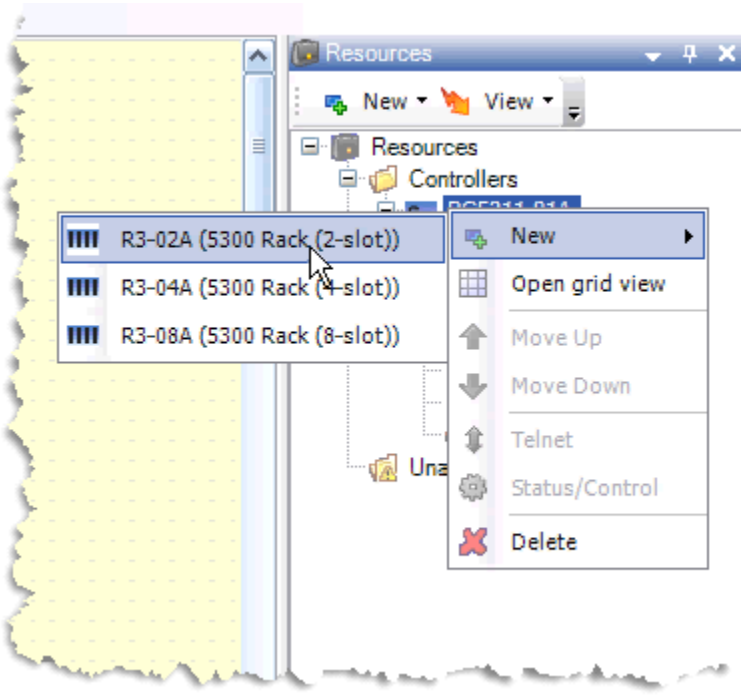
Notes

1. The name here distinguishes the controller in your project and does not affect your DNS name setting set through WebMON.
2. Note: The address will have to be set up to what your controller's address is. The default IP setting from the factory is 192.168.1.53.

3.2.2 Configuring the hardware: rack and IO

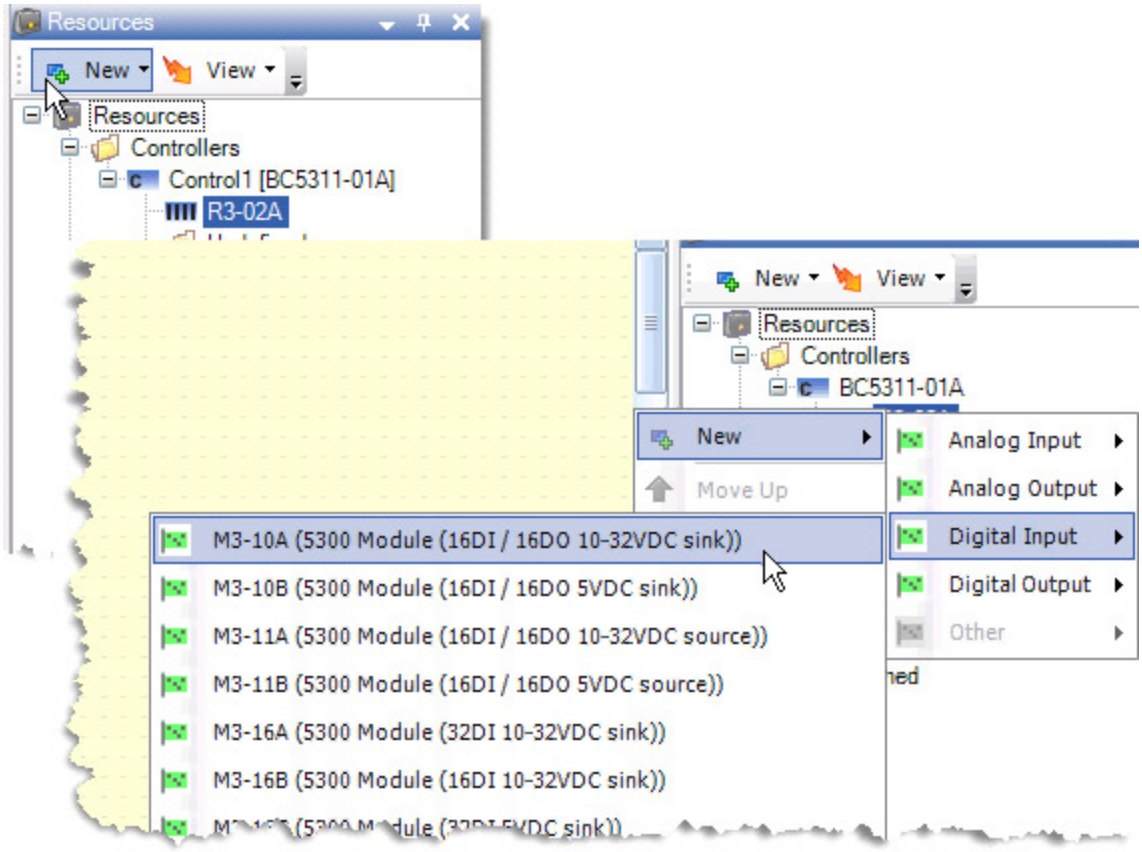


1. Right click **Control1**.



2. Select **New**.
3. Select the rack option that corresponds to the number of slots (2, 4, or 8) in your rack.

4. Right click on the rack slot option. In this example, the rack slot option is R3-02A (2 slot rack).



5. Place cursor on Digital Input.
6. Click on the M3-10A Digital Input/Output card.

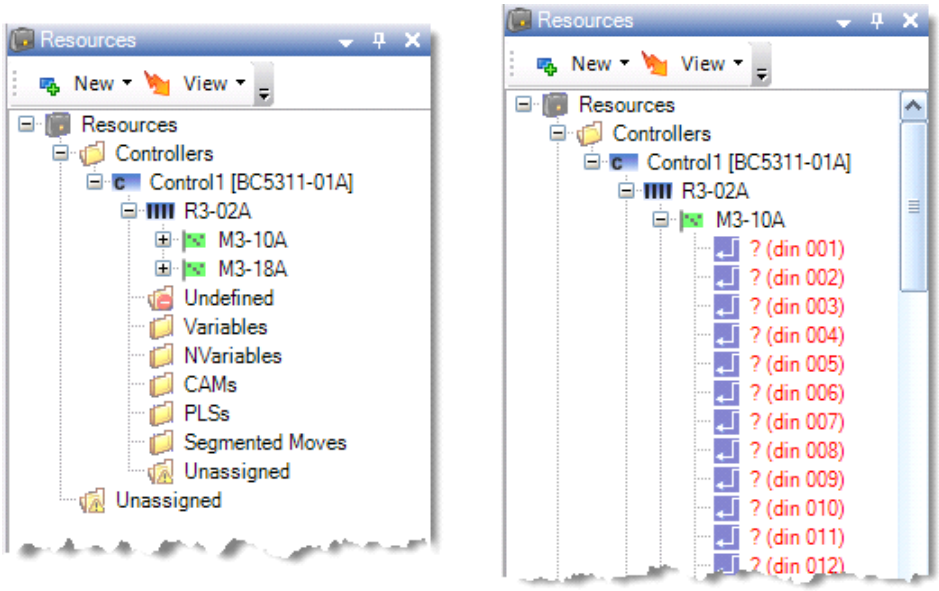
Note: The M3-10 is found under both Digital Inputs and Digital Outputs since it has 16 inputs and 16 outputs.

3.2.3 Configuring the hardware: M3-18A card

Add another card as shown previously but select an M3-18A Digital In/Out card instead of an M3-10A.

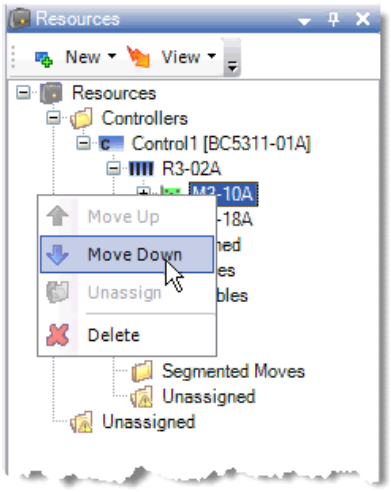
Notes:

1. Your Resource Window should look like the image on the left below. If you click on the plus to the left of the card it will show you a list of the I/O for that card like shown in the image on the right. The I/O resources are currently red because they have not been used in the program yet. All unnamed resources are given a default name of **?**

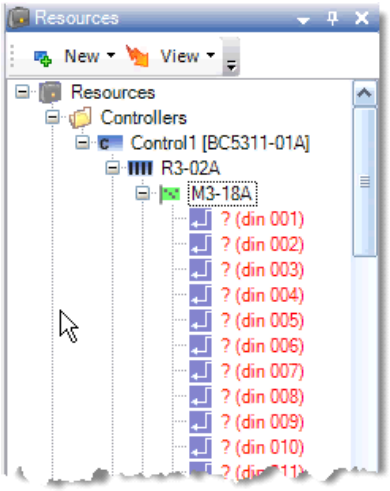


2. If you place the cards in the wrong order you can simply right click on a card and use the move up or move down options to move the cards to the correct order. The top card in the Resource window list should be the leftmost card in your controller.
3. The cards can be moved at any time even after your program is written and QuickBulder will re-assign the correct I/O numbering to the cards as placed in the rack from left to right automatically. In example above, the M3-10A card is the leftmost card in the controller; it contains digital inputs 001 – 016 and digital outputs 001 – 016. The M3-18A card is in the next slot to the right of the M3-10A card; it contains digital inputs 017 – 032 and digital outputs 017 – 032

At this point, the M3-10A card appears above the M3-18A card in the Resource Manager list. Move it down by doing the following:



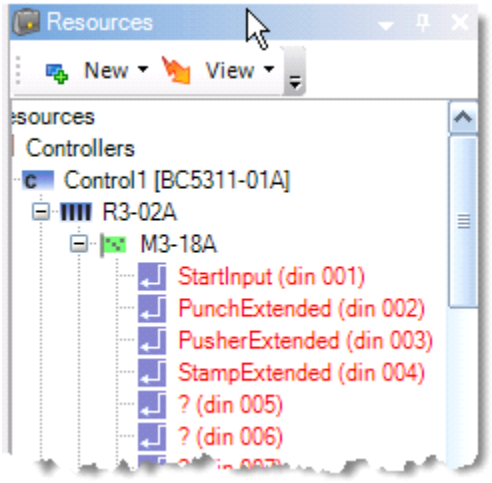
1. Right click on M3-10A.
2. Click on **Move Down**. The M3-18A card will now have digital inputs and outputs 1 through 16 and the M3-10A card will have digital inputs and outputs 17 through 32:



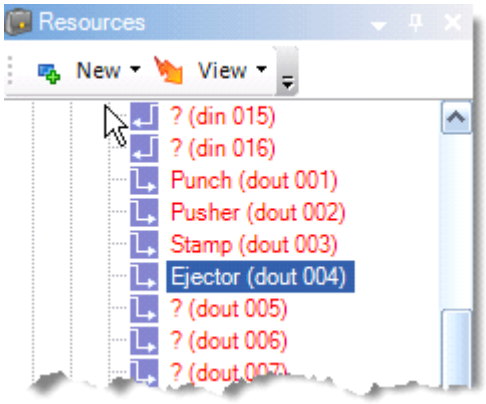
3.2.4 Naming the inputs

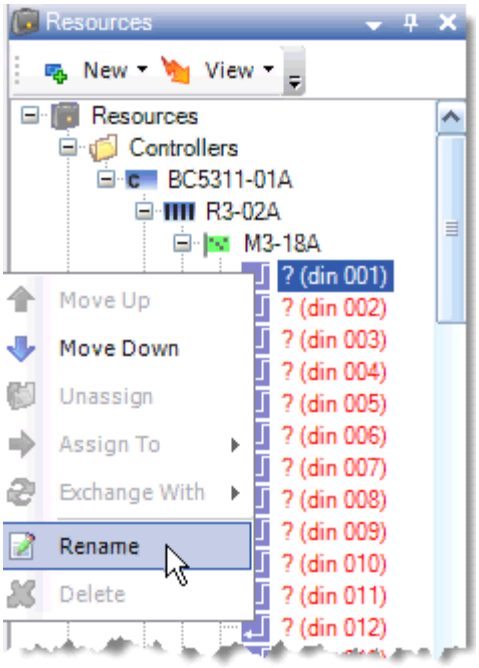
Use the method from the previous section to rename the input and output resources to the names shown below so we can begin to write a very simple program.

Inputs:

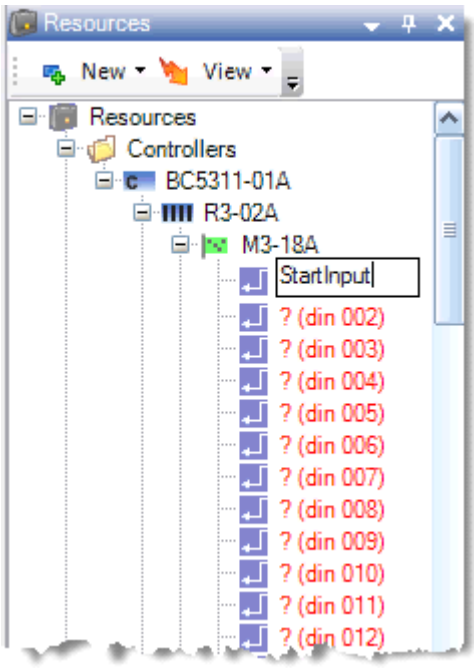


Outputs:





1. Right click on ? (din001)
2. Select **Rename**.



3. Type **StartInput**.

Notes:

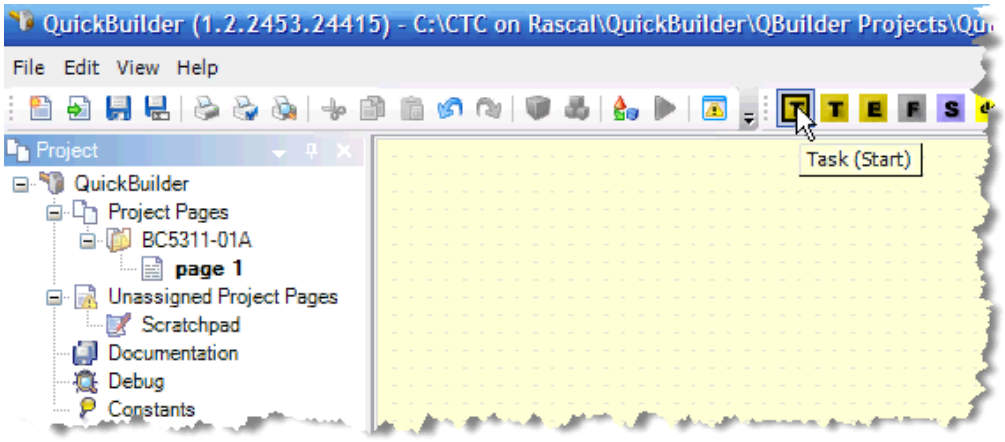
1. Naming conventions are case sensitive. Special characters and spaces are not allowed. Using numbers is valid but names must begin with an alpha character.
2. You have the option to move input and output resources up and down also. This allows for easy program changes when wiring changes occur after the program is already written.

3.3 A sample program

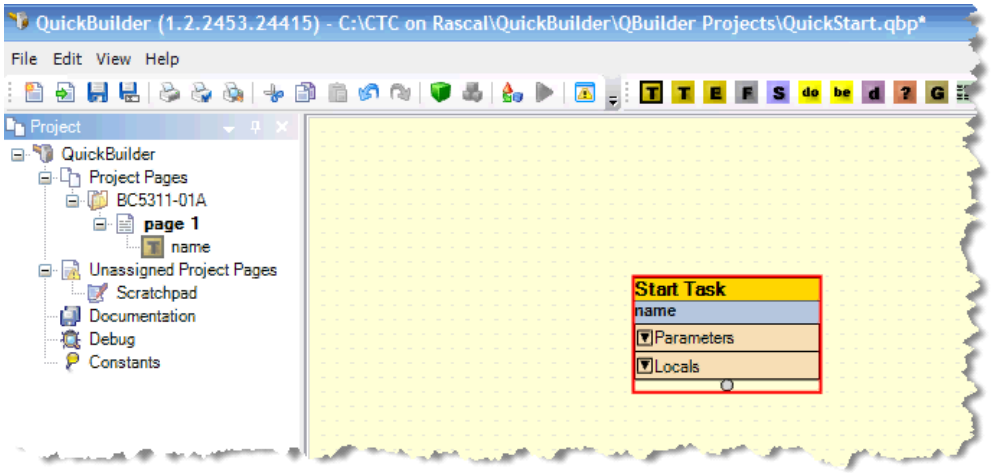
We will now program the logic for our Model 5300 controller. With QuickBuilder you use a two-step approach to building your logic program. First you insert the graphical elements of a task, such as steps and decision blocks, into the Graphical SFC Window. Then you select the graphical element and add the appropriate Quickstep4 (QS4) code into it.

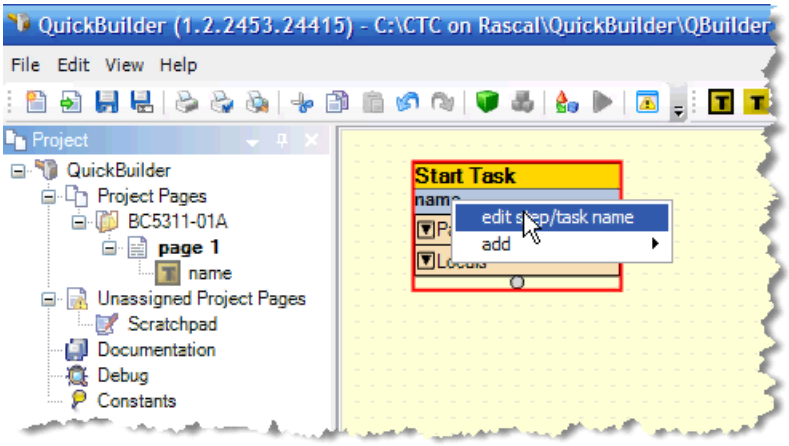
3.3.1 Creating a StartTask

1. Click on the Task (Start) icon to add a StartTask. All programs must have one and only one StartTask component.

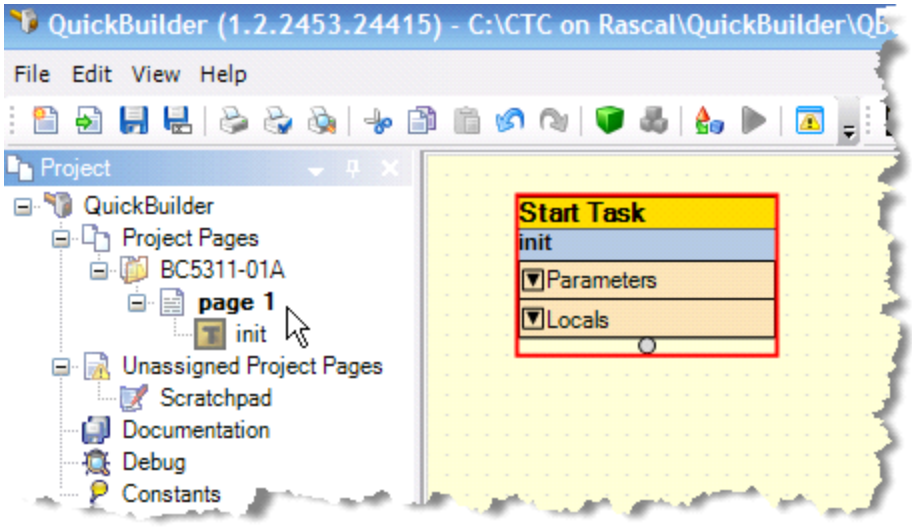


The Start Task graphical component is automatically added to the middle of the graphical page as shown below. The object was also automatically added to the Project Window list under page 1.





2. Drag and Drop the Start Task object to the upper left corner of the page as shown.
3. Right click in the blue box of the Start Task graphical component.
4. Select edit step/task name.
5. Type **init**.



3.3.2 Adding comments

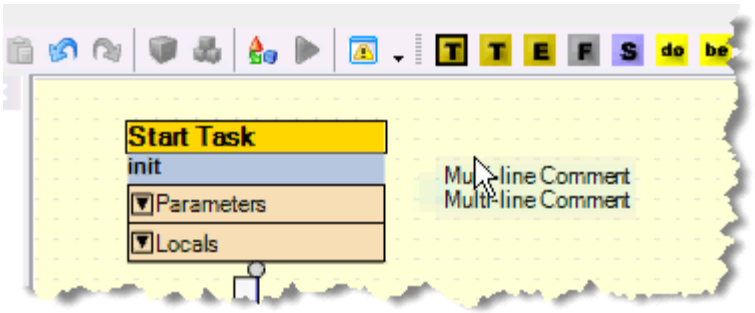
Commenting with the Comment icon

Normally you could initialize your program with the Start Task, but our simple program will not require this. We will add a multi-line comment with a brief description of the program.

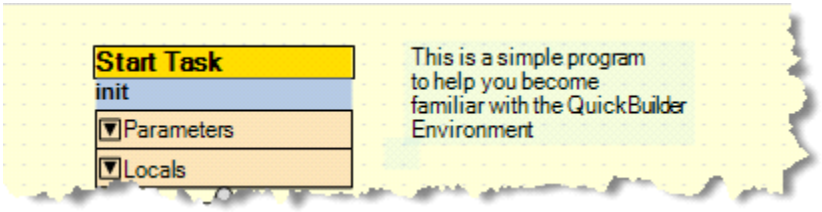
1. Click on the Comment icon as shown below:



2. Drag and drop the Multi line Comment closer to the Start Task as shown below.



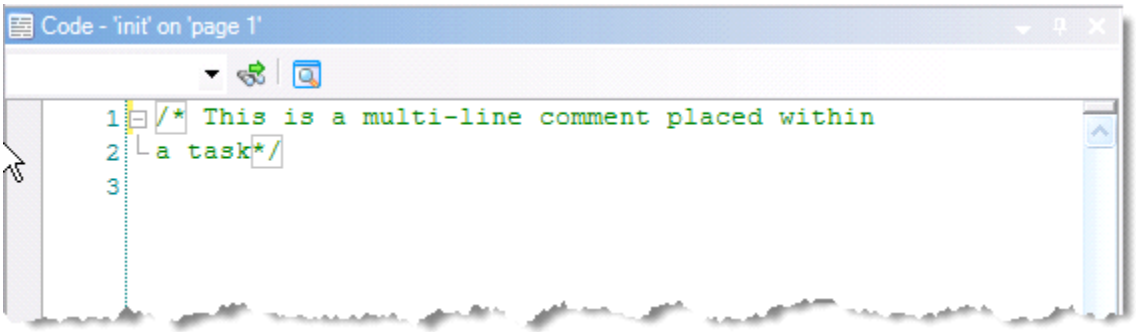
3. Click on the Multi line Comment.
4. Type a message of your choice.



Commenting with the Code/Comment Editor

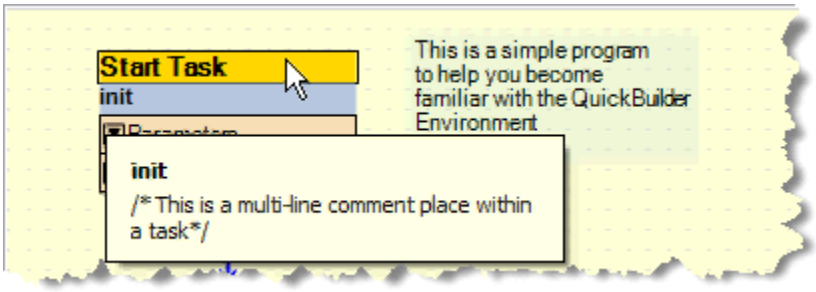
You can also add comments to your steps. We will add a multi-line comment to the StartTask.

1. Click on the Start Task **init**.
2. Click on the Code/Comment Editor window directly below the Graphical SFC window.
3. Type:
   /* This is a multi-line comment placed within a task*/



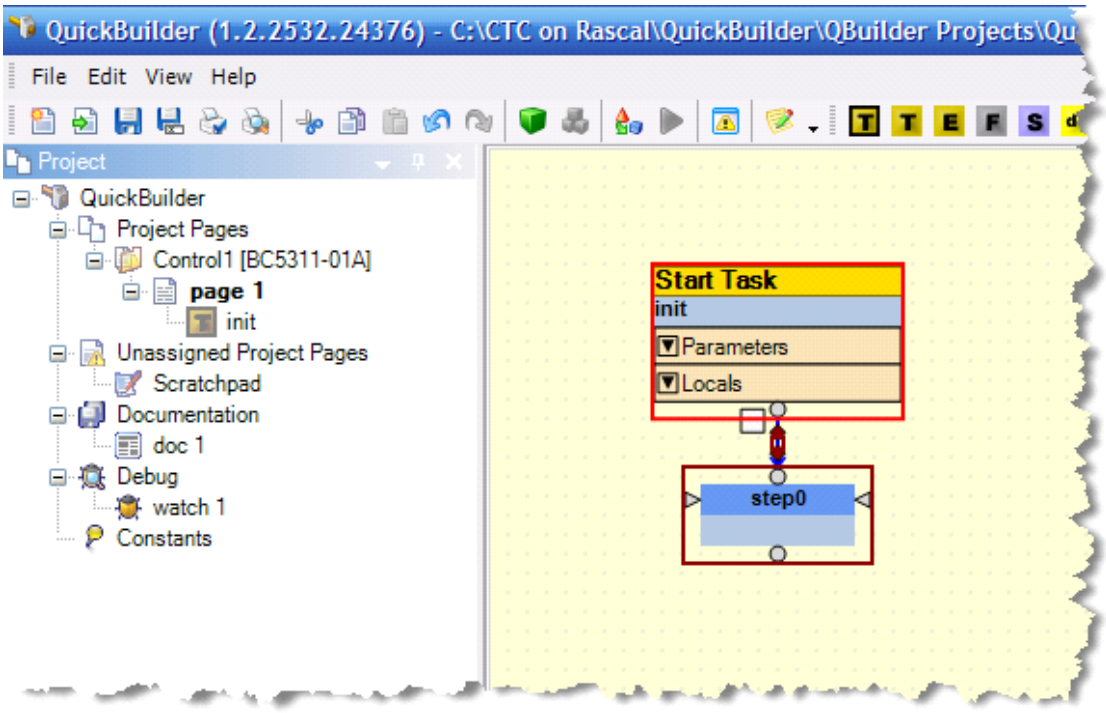
This comment will now show up when you place your cursor over the Start Task as shown below.

Note: The first 10 lines of code for any given step are always shown when your cursor is directly over it.

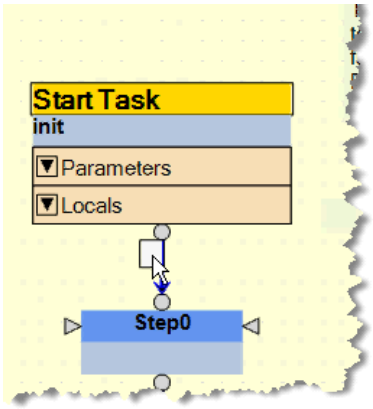


3.3.3 Adding the next step in the program: Punch

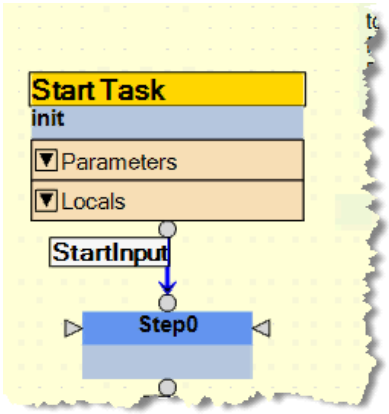
1. Click on the Step S icon while the Start Task is still highlighted to automatically add a connected step. The small white box between the Start Task and step0 allows you to enter a transition condition that must be met before the program process or program flow is allowed to continue.



2. Click on the white box (transition condition).

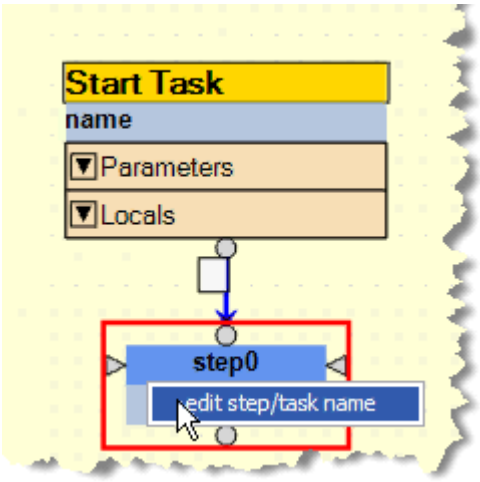


3. Type **StartInput**.

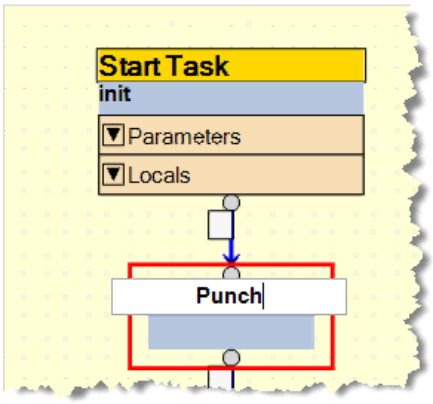


Entering the condition StartInput (Digital Input 1) will cause the process to wait for the StartInput to be on (the true value). If you wanted to wait for it to be off you would use !StartInput. The ! means not true (or in the case of an input, off).

Now you are ready to configure the next step.



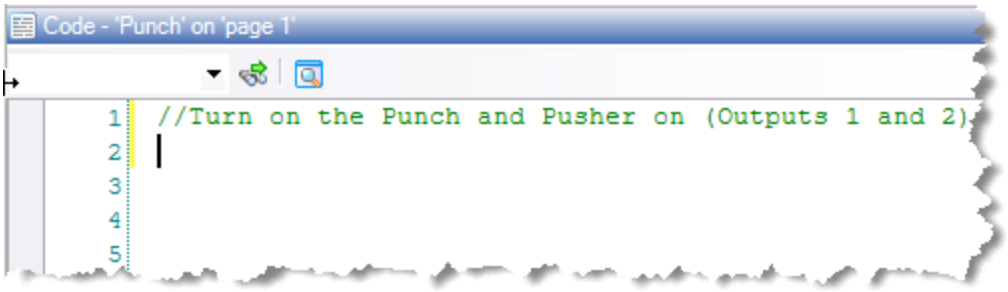
4. Right click on the step0 graphical component.
5. Select edit step/task name.



6. Type **Punch**.

It is always a good idea to comment your steps whether it is with a single or multi-line comment. We will now add a single line comment.

7. Click on the Code/Comment Editor
8. Type **//Turn on the Punch and Pusher (Outputs 1 and 2)**



3.3.4 Entering QS4 code: Step Editor and Code Windows

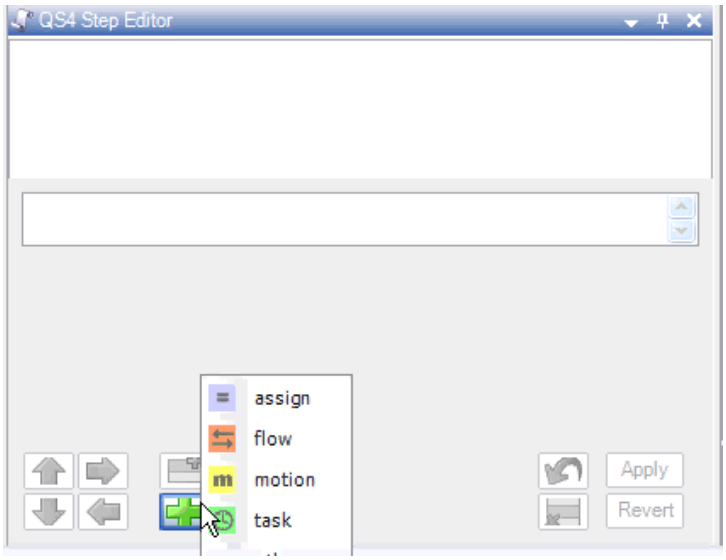
After you enter your initial comment you can begin adding statements for the step to execute. There are two ways to do this:

- if you are familiar with QuickBuilder and the QS4 syntax, you can simply type your statements directly into the code editor window. The code editor will automatically color your text and flag any errors as you type;
- or if you are a beginner, you can use the QS4 Step editor window to the right of the code editor and it will guide you through the process of creating and entering a command.

The QB4 Step Editor Window

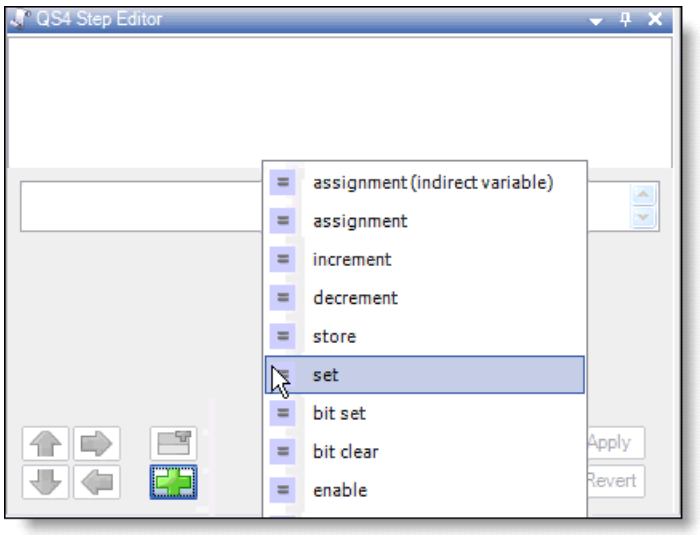
Since you're not familiar with the language yet, we will use the QB4 Step Editor Window to assist in writing the code.

1. Click on the Insert Statement icon  in the QS4 Step Editor.
2. Click on **assign**:



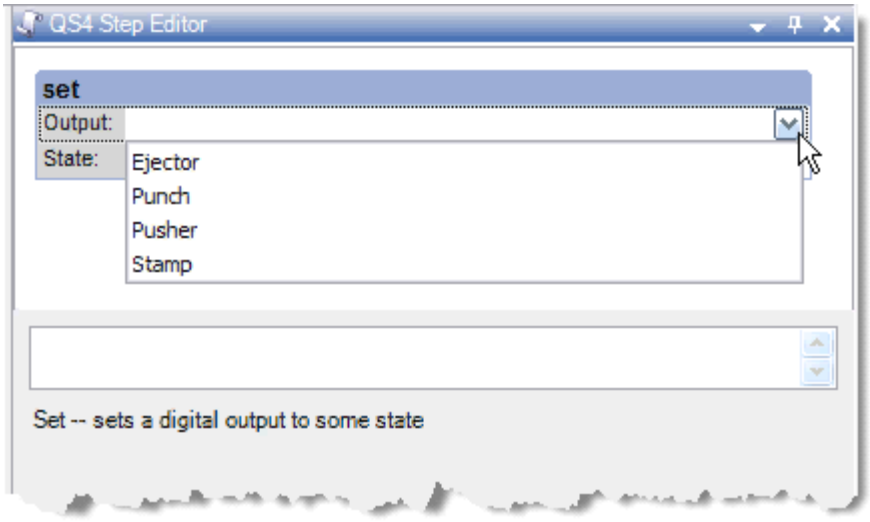
The following commands in the assign group will appear as shown below.

3. Click on **set**:

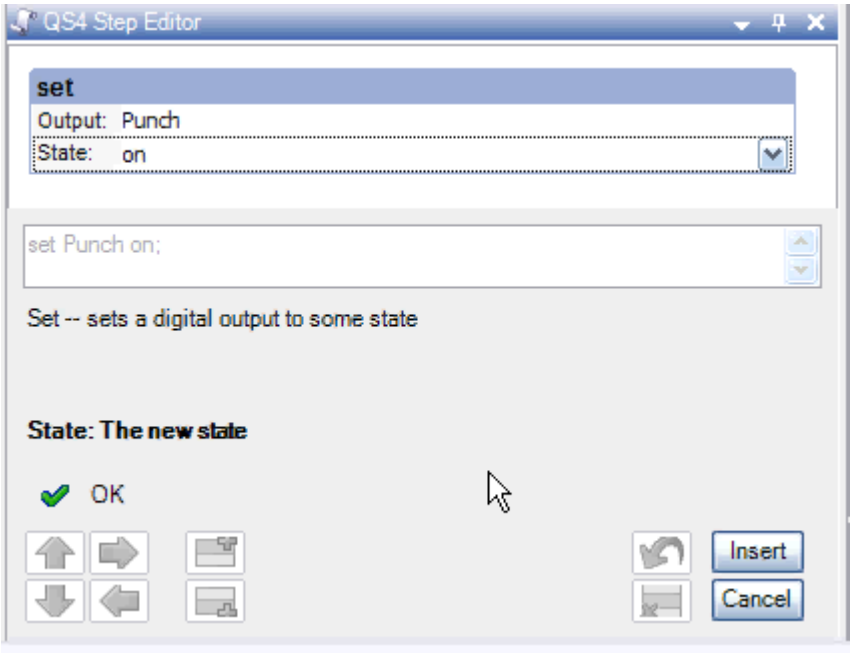


The QS4 Step Editor gives the definition of the set command.

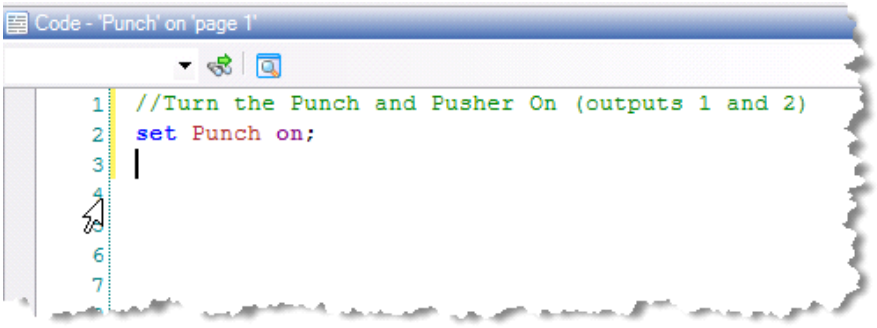
4. Click in the white box next to **Output**:
5. Click on the list box at the right side and select **Punch**.



6. Select **on** for the State.
7. Click on **Insert** located in the lower left corner of the Step Editor.



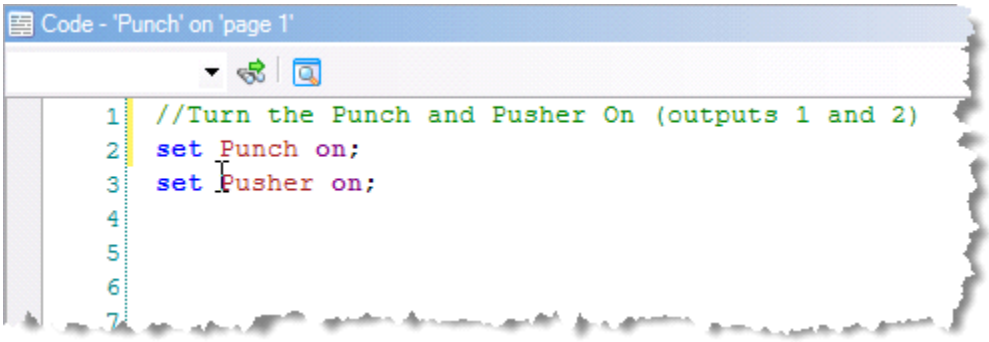
You will see the following code has been inserted the code window. This will turn the Punch (Digital Output 1) on.



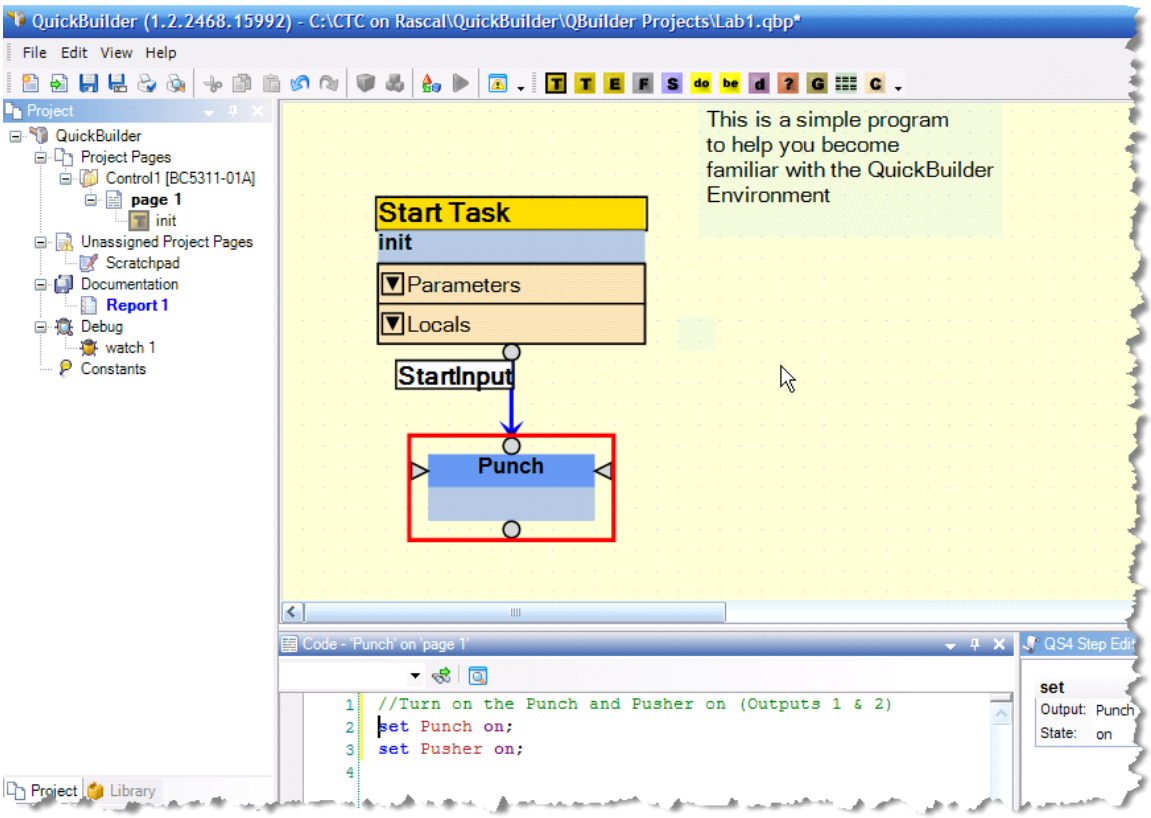
The QB4 Code window

QuickBuilder allows you to use the step editor to enter the code for your step or you can type the code directly into the code window. You can mix and match your approach to entering code at any time.

1. Continuing from the example above, type **set Pusher on;** on line 3.



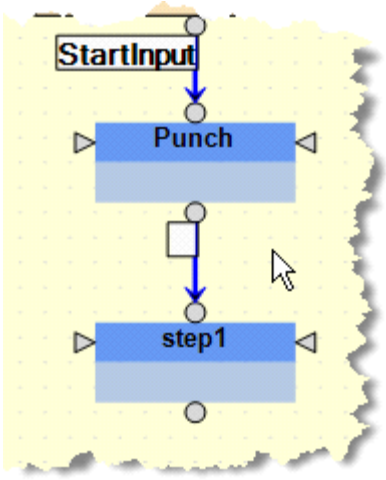
Your project should now look like this:



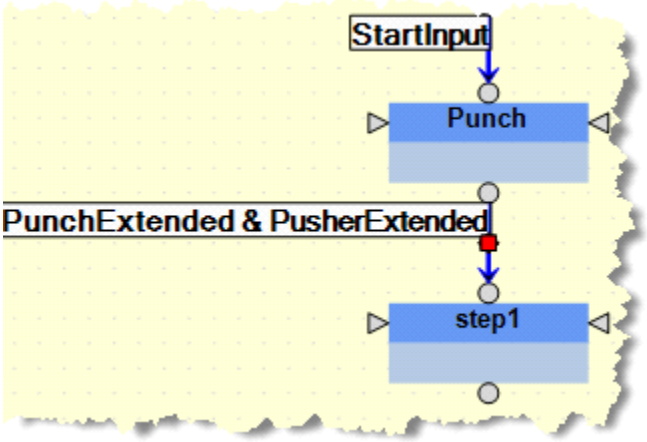
Note: StartInput, Punch, and Pusher are no longer red in the Resources Window since they are being used, because black resource names are used somewhere in the project while red names are not being used.

3.3.5 Finishing your program

StampOn step

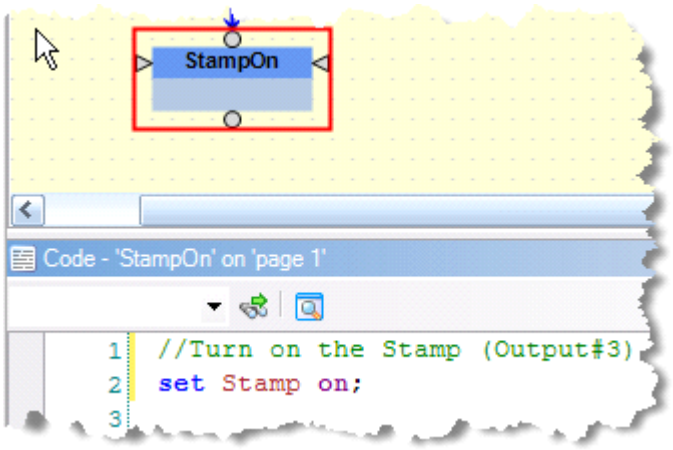
1. Continuing with the sample program, Click on the S icon while the **Punch** step is still highlighted to add a step.
   .
2. Click on the transition condition for **Punch**.
3. 

4. Type **PunchExtended & PusherExtended**



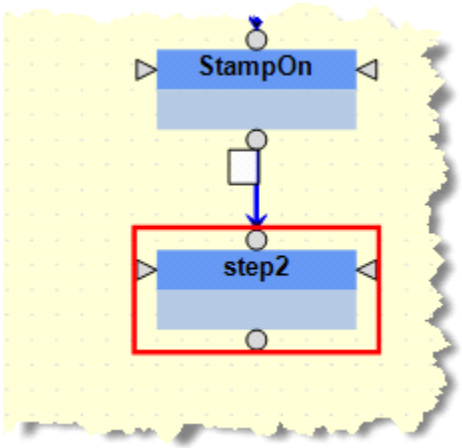
This will keep the process from continuing on until we see that the **PunchExtended** and **PusherExtended** inputs are both on. Once they are both on, the process will continue to the next step.

5. Right click on step1.
6. Select **edit step/task name**.
7. Type **StampOn**.
8. Click on the **StampOn** Code Editor.
9. Type **//Turn on the Stamp (Output #3)** on line 1.
10. Type **set Stamp on;** in line 2 of the Code Window.

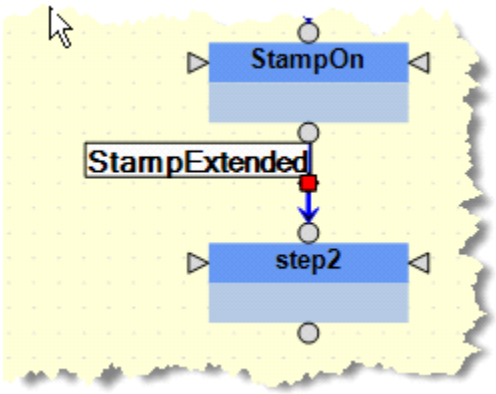


EjectPart step

1. Click on the S icon while the **StampOn** step is still highlighted to add a step.

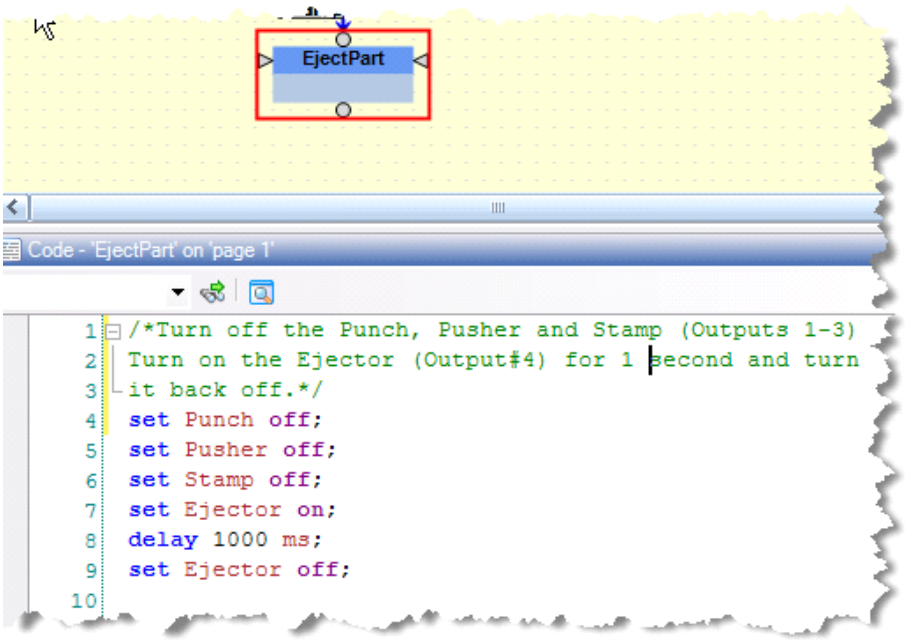


2. Click on the transition condition for **StampOn**.
3. Type **StampExtended**. This will keep the process from continuing on until we see that the **StampExtended** input is on.



4. Right click on step2.
5. Select edit step/task name.

6. Type **EjectPart**.

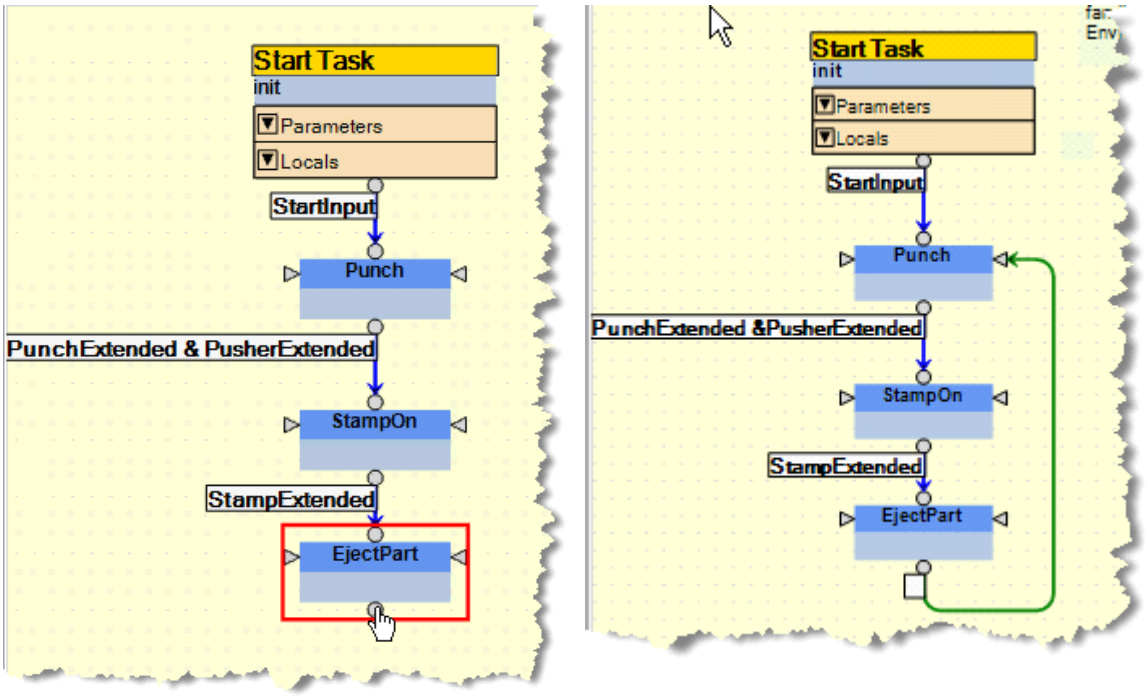


7. Type the following on the following lines in the **EjectPart** Code Window.

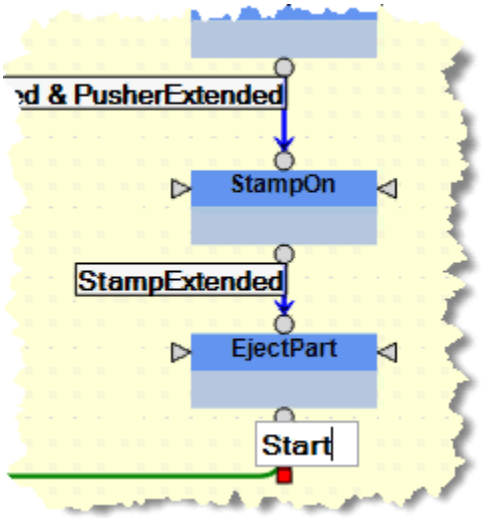
```
1 /*Turn off the Punch, Pusher and Stamp (Outputs 1-3)
2 Turn on the Ejector (Output#4) for 1 second and turn
3 it back off. */
4 set Punch off;
5 set Pusher off;
6 set Stamp off;
7 set Ejector on;
8 delay 1000 ms;
9 set Ejector off;
```

Adding a loop to the program

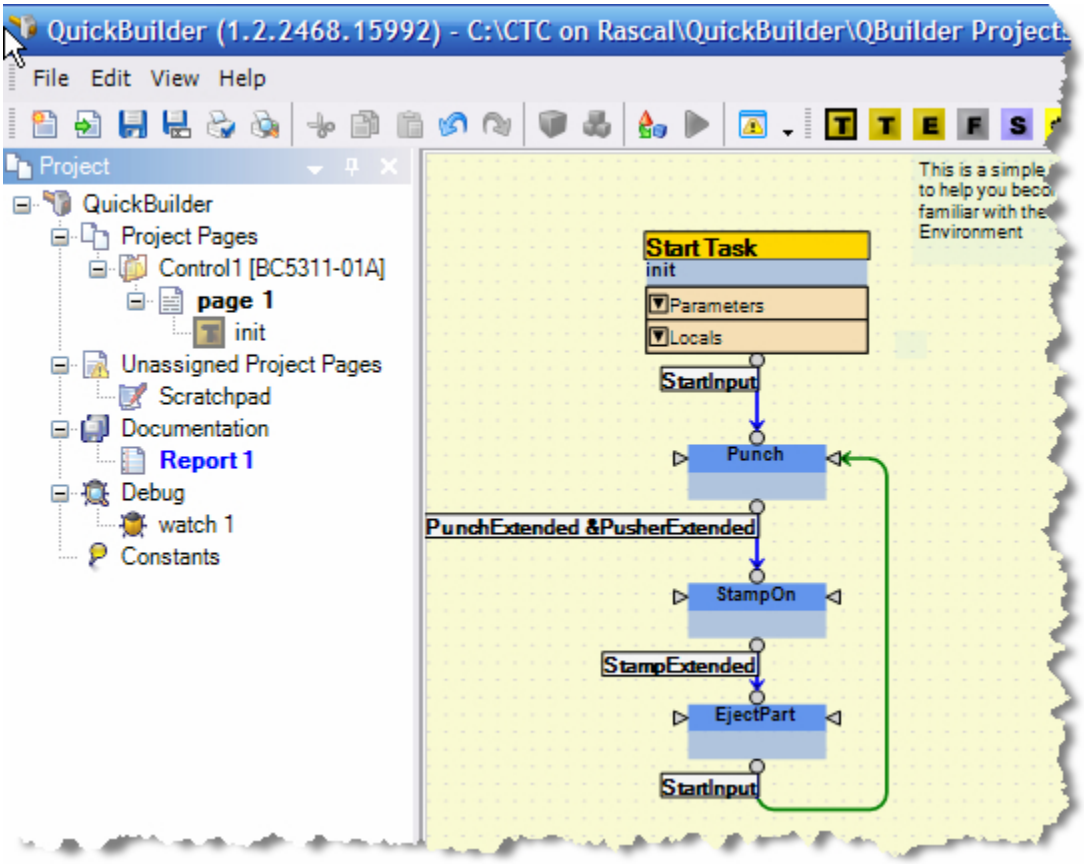
The last step will be to be to add the code to repeat the process after the **StartInput** is on again. We will first draw a line between the **EjectPart** step and the **Punch** step.



1. As shown above, click on the bottom circle on the **EjectPart** step.
2. Click on the right hand arrow on the **Punch** step.
3. Click on the **EjectPart** transition condition.
4. Type: **StartInput**.



Your project is complete and will now look like this:



3.3.6 Saving your program

Our program for this application is now complete, and we will now save the project file.

1. Click on **File** (from pull down menu).
2. Click on **Save Project As**.
3. Select the directory you want to save your file to.
4. Type: **QuickStart** as the project name.
5. Click on **Save**.

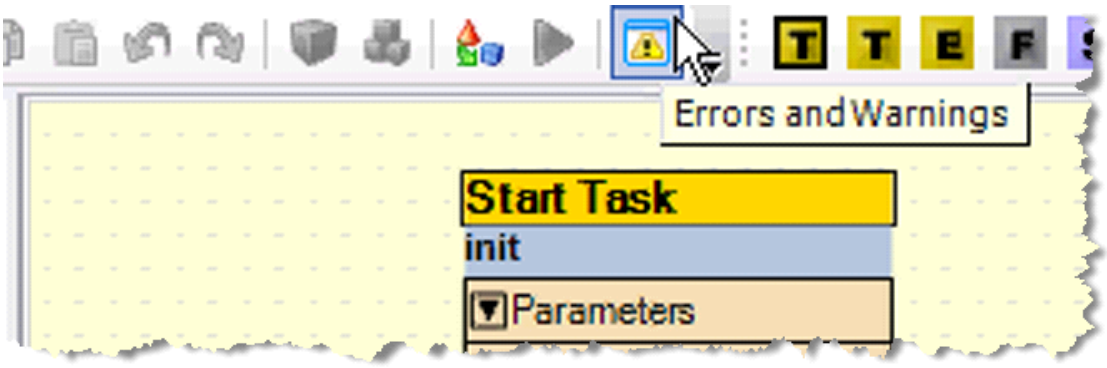
Note: To see another approach to creating a project that uses only one step to accomplish what was done in three steps in the QuickStart project, see Appendix C.

3.3.7 Commission

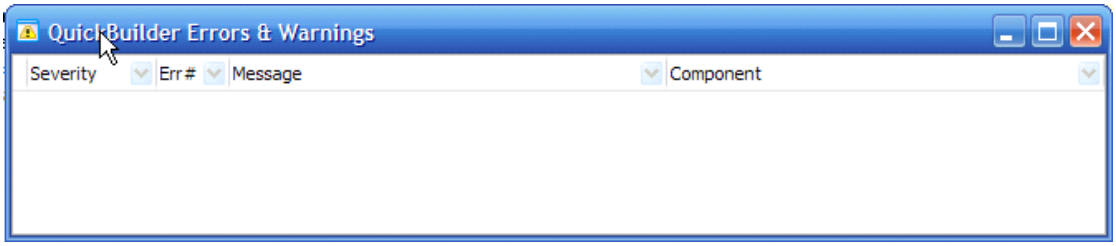
Checking for errors

We will now begin the commissioning process to load and test our project in the Model 5300 controller. First we will check for errors in the code.

Click on the Errors and Warnings icon shown below.



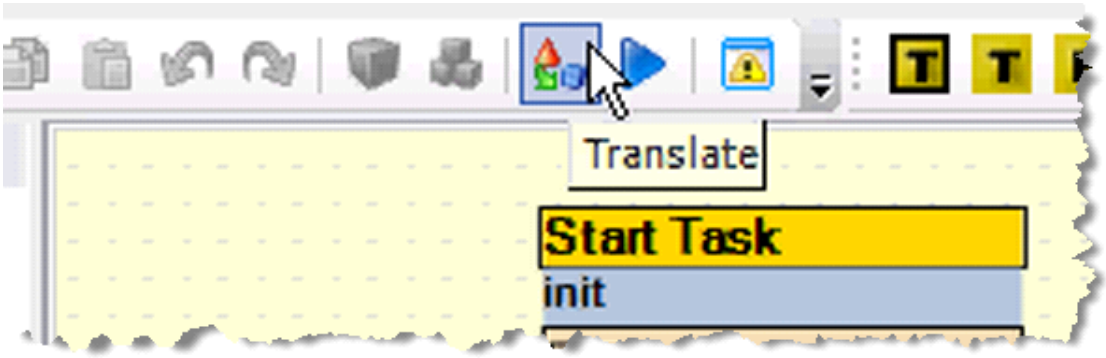
This will bring up the following window. If you followed the directions so far, your Error/Warning Window will be empty as shown below. *If you have any errors, double-click on the error and QuickBuilder will bring it up in the code window. You can then compare your code to this document and make the necessary corrections.*



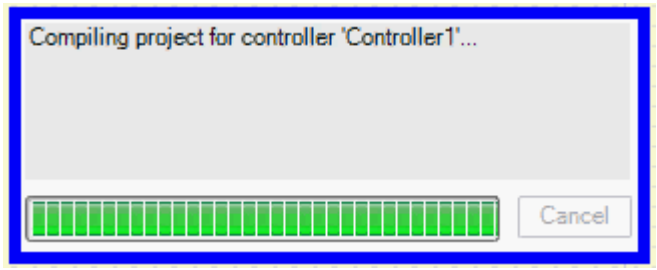
Close the Errors & Warnings Window (Click on the X in the upper right corner). We will now translate the code to prepare it for download.

Translating the code

Click on the Translate icon shown below.



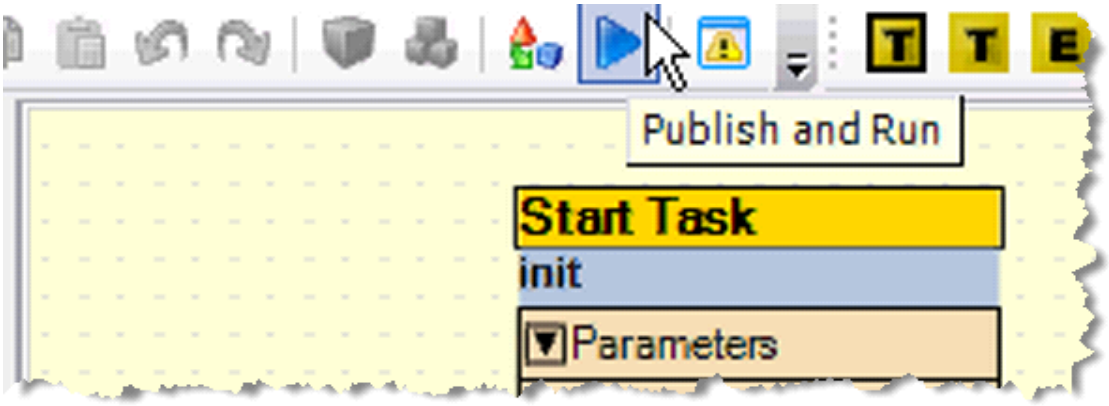
The following message will flash on your screen and then disappear when done.



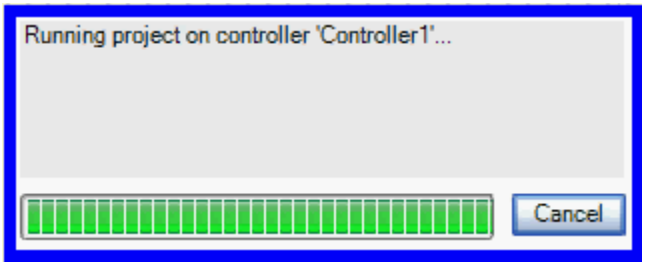
Publishing and running the code

We will now Publish and Run (or download) the code.

Click on the Publish and Run icon shown below.



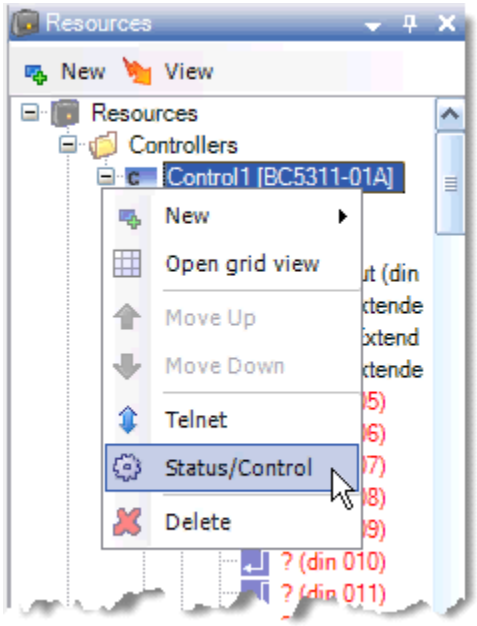
The following message will flash on your screen and then disappear when done.



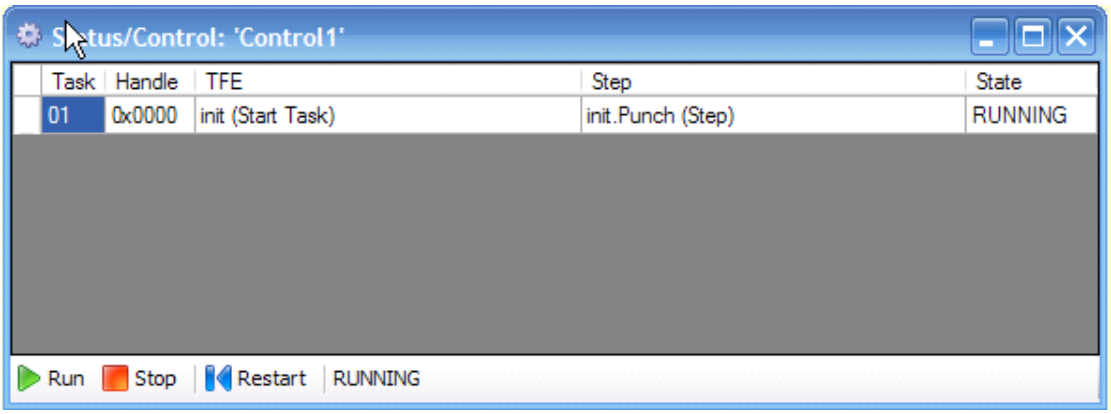
Real-time debugging: Status/Control and Watch windows

When the program has been published to your controller, you can watch the progress of your program using the Status/Control Window to help you verify the correct operation of your program.

1. Access the Status/Control Window by right clicking Control1 in the Resource Window.
2. Click Status/Control.

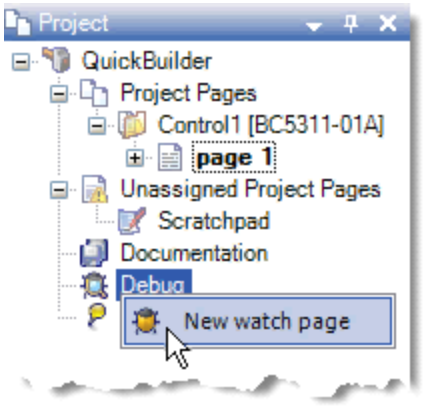


You will now see the following status window and can watch the program status.

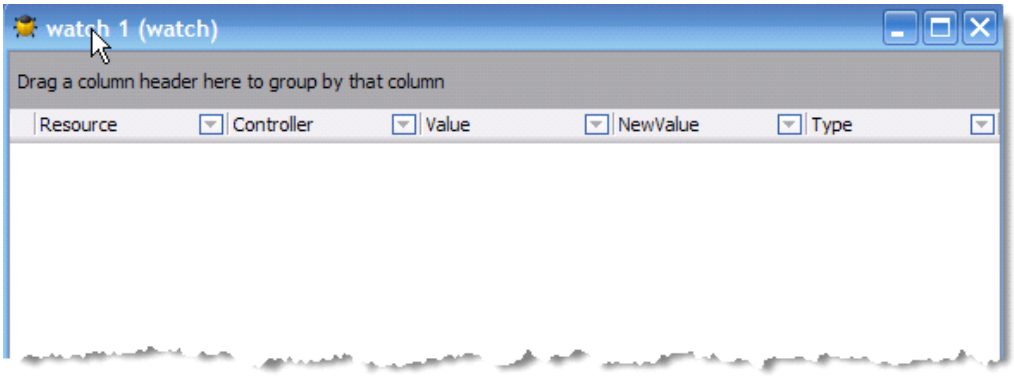


Another tool available to help verify/troubleshoot your program is the Watch Window.

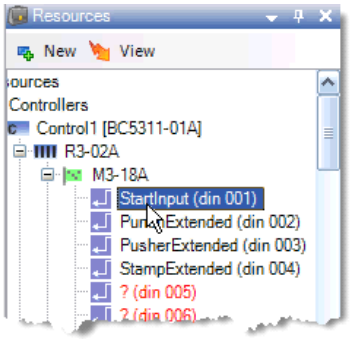
1. Right click on Debug in the Project Window.
2. Click New watch page. A Watch window like the one below will appear.



3. You can add resources to watch by simply dropping and dragging them into the watch window. Drag and drop **StartInput** from the Resources window into the Watch window.

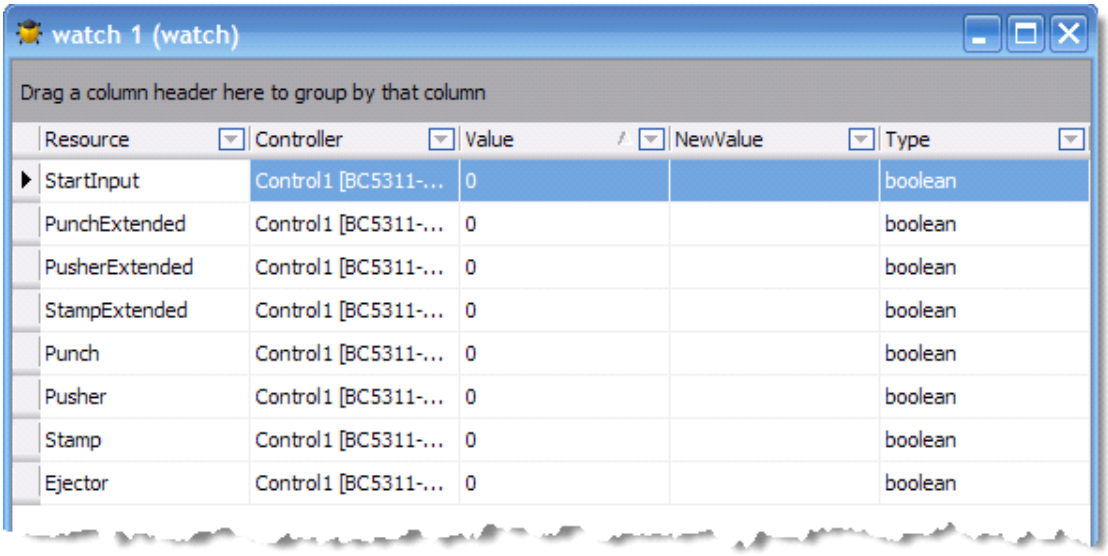


4. Drag and drop the other 3 inputs from the Resources Window into the Watch window.



5. Drag and drop **Punch** from the Resources window into the Watch window.

6. Drag and drop the other 3 outputs from the Resources window into the Watch window. Your watch window will now look like this:



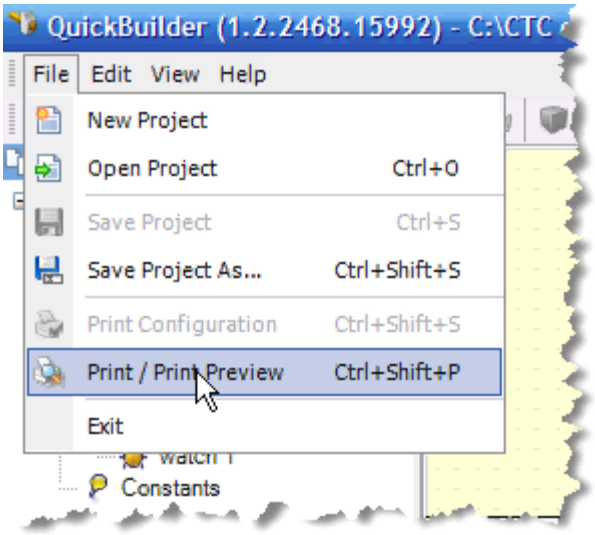
Resource	Controller	Value	NewValue	Type
StartInput	Control1 [BC5311-...	0		boolean
PunchExtended	Control1 [BC5311-...	0		boolean
PusherExtended	Control1 [BC5311-...	0		boolean
StampExtended	Control1 [BC5311-...	0		boolean
Punch	Control1 [BC5311-...	0		boolean
Pusher	Control1 [BC5311-...	0		boolean
Stamp	Control1 [BC5311-...	0		boolean
Ejector	Control1 [BC5311-...	0		boolean

Any changes in the status of the inputs or outputs will be reflected in the Value column. You can set an output by clicking on NewValue for that output and entering a 1 for on or 0 for off. This will only set the outputs and not force them. The program flow will be able to override any change you make in the Watch window.

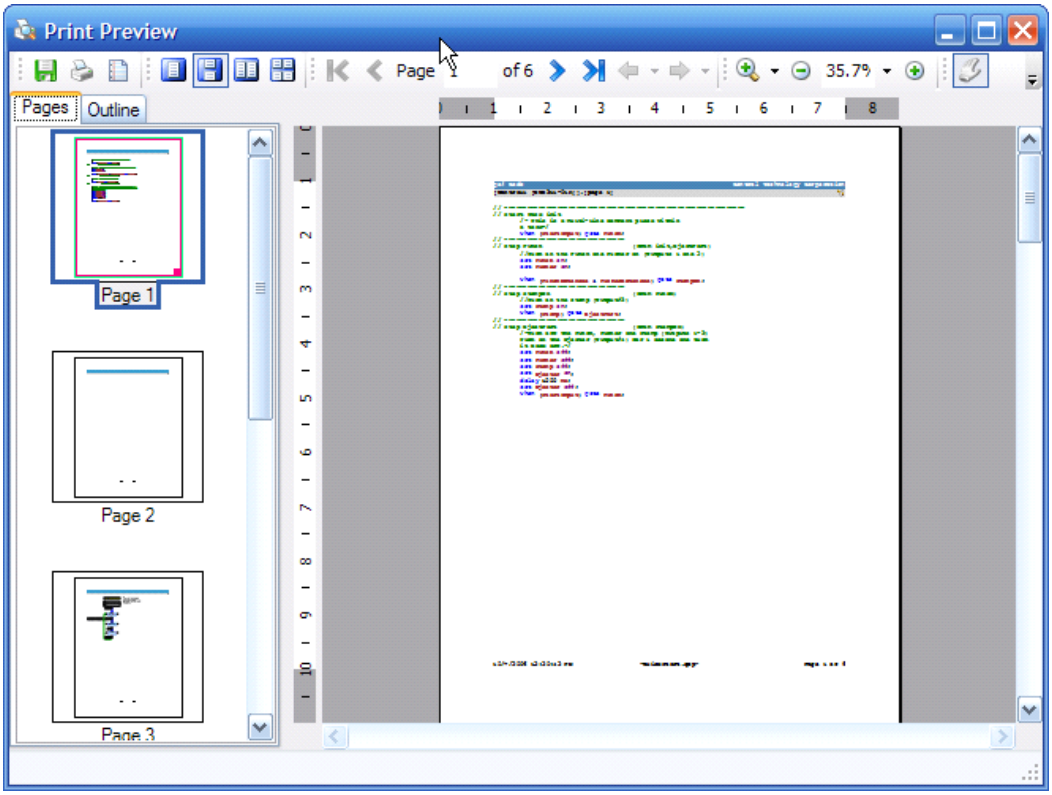
3.3.8 Maintenance

Now that we have verified our project is working properly we will create a report that contains all of the relevant documentation we will need for this project. This report may then be converted to a pdf.

1. Click on **File**.
2. Click on **Print/Print Preview**.



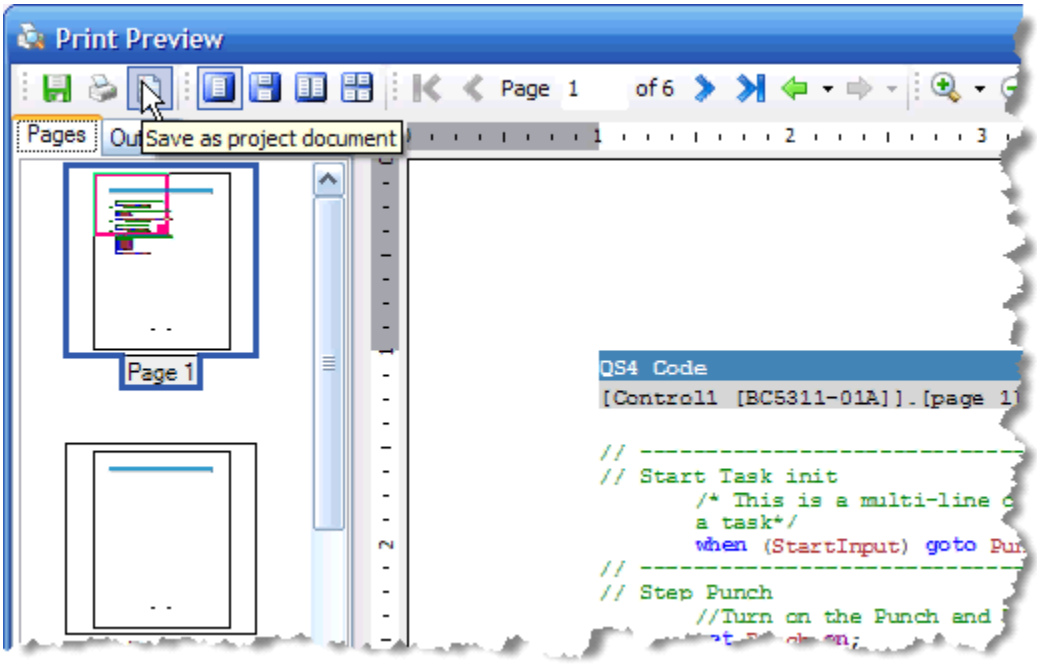
You will now see the following project information in a Print Preview format:



The information in your Print preview includes:

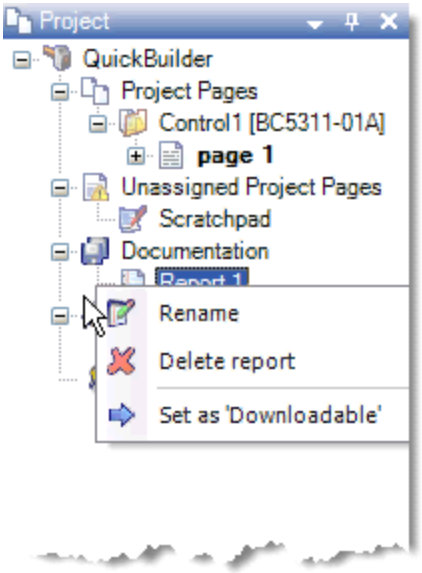
1. All of your QS4 Code, on all of the pages you have created (we did not discuss the use of pages and we only used one) See Appendix A for info on the use of pages.
2. Any QS4 Code you have placed in your ScratchPad (we did not use this feature either)
3. The SFC diagram from all of the pages you have created.
4. Any SFC diagrams in your ScratchPad.
5. A list of your contoller Resources showing where they are used in the code.

Click on the Save as project document icon shown below to save your document to the project.

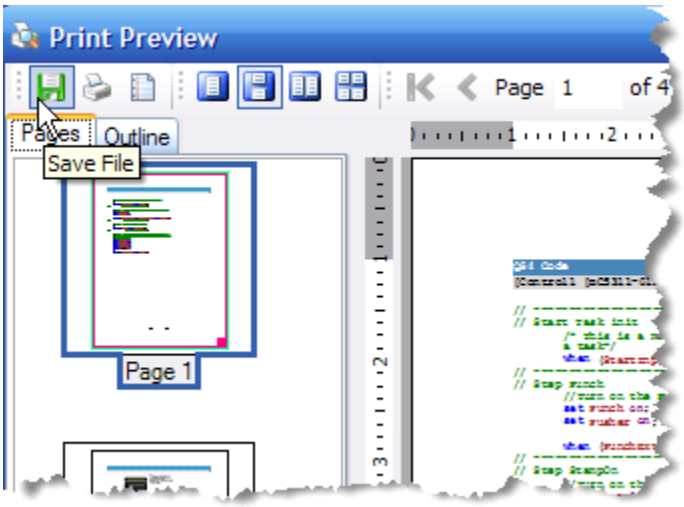


Notes:

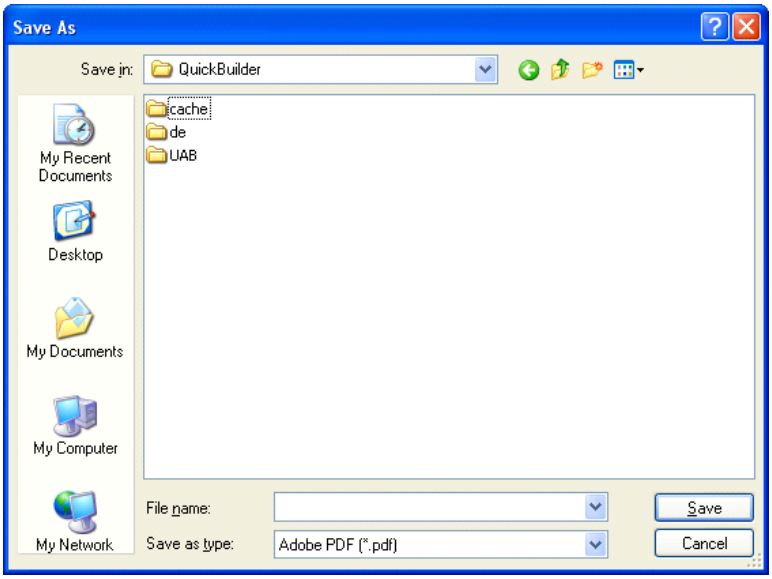
1. The project window will now include **Report1** as shown below. You can rename it by right clicking on **Report1**, selecting **Rename**, and typing a new name.
2. Notice there is also a **Set as 'Downloadable'** option. This will download this report to the controller on your next Publish command, so it can be retrieved at a later date. You can set it so it is 'not Downloadable' at any time after it is set as Downloadable, if you decide you don't want the document downloaded..



To save your document to a pdf, click on the Save File icon in the **Print Preview** pop-up:



You will see a pop-up screen that looks like this:

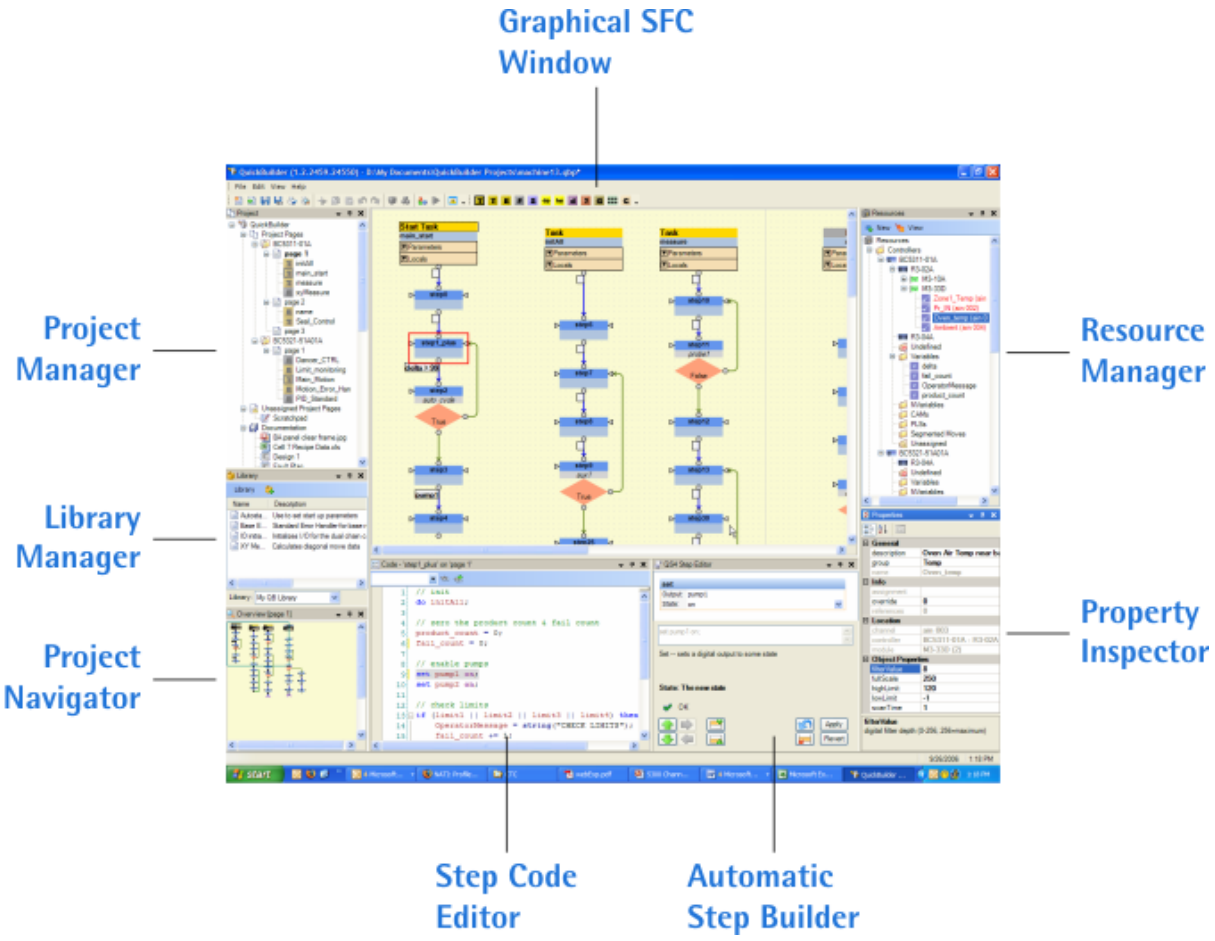


Enter a name for the pdf to save it.

4 Appendix A - Detailed Overview of the QuickBuilder Automation Suite Environment

This section takes you through a series of screen captures that will give you an overview of the major components of the QuickBuilder Automation Suite. With QuickBuilder you can handle all of your automation tasks from configuration, to programming, to application commissioning, to ongoing monitoring and maintenance in one unified environment.

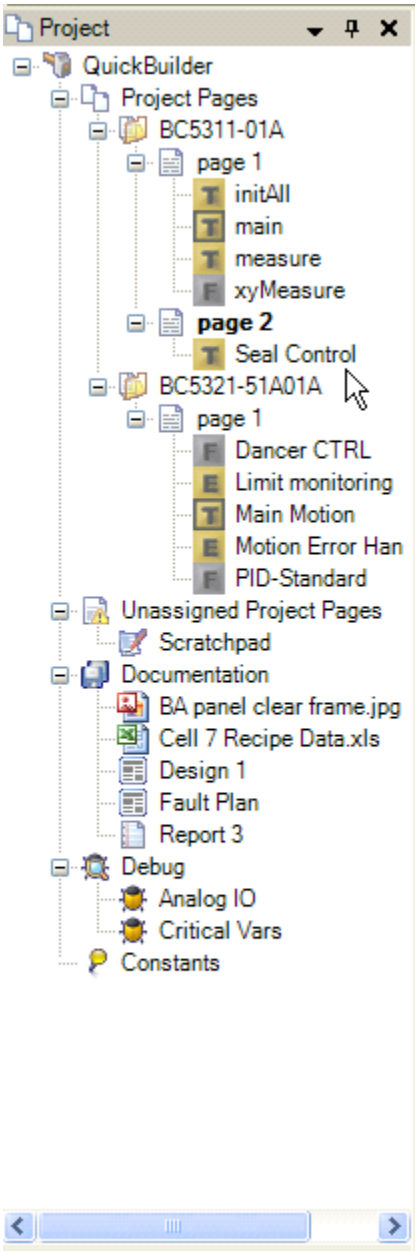
If you want to jump right into creating a program, skip to Appendix B.



4.1 Configuration

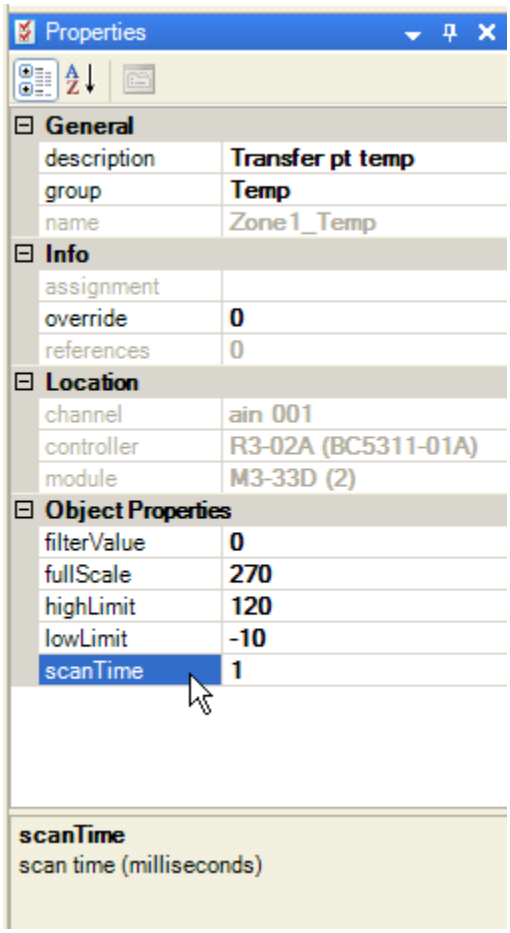
4.1.1 Project Manager

- Hierarchical view of major program elements
- Supports multi-controller projects
- Quickly drill down to specific tasks or functions
- Organizes project documentation
- Organizes debug views
- Gives you access to all pages created

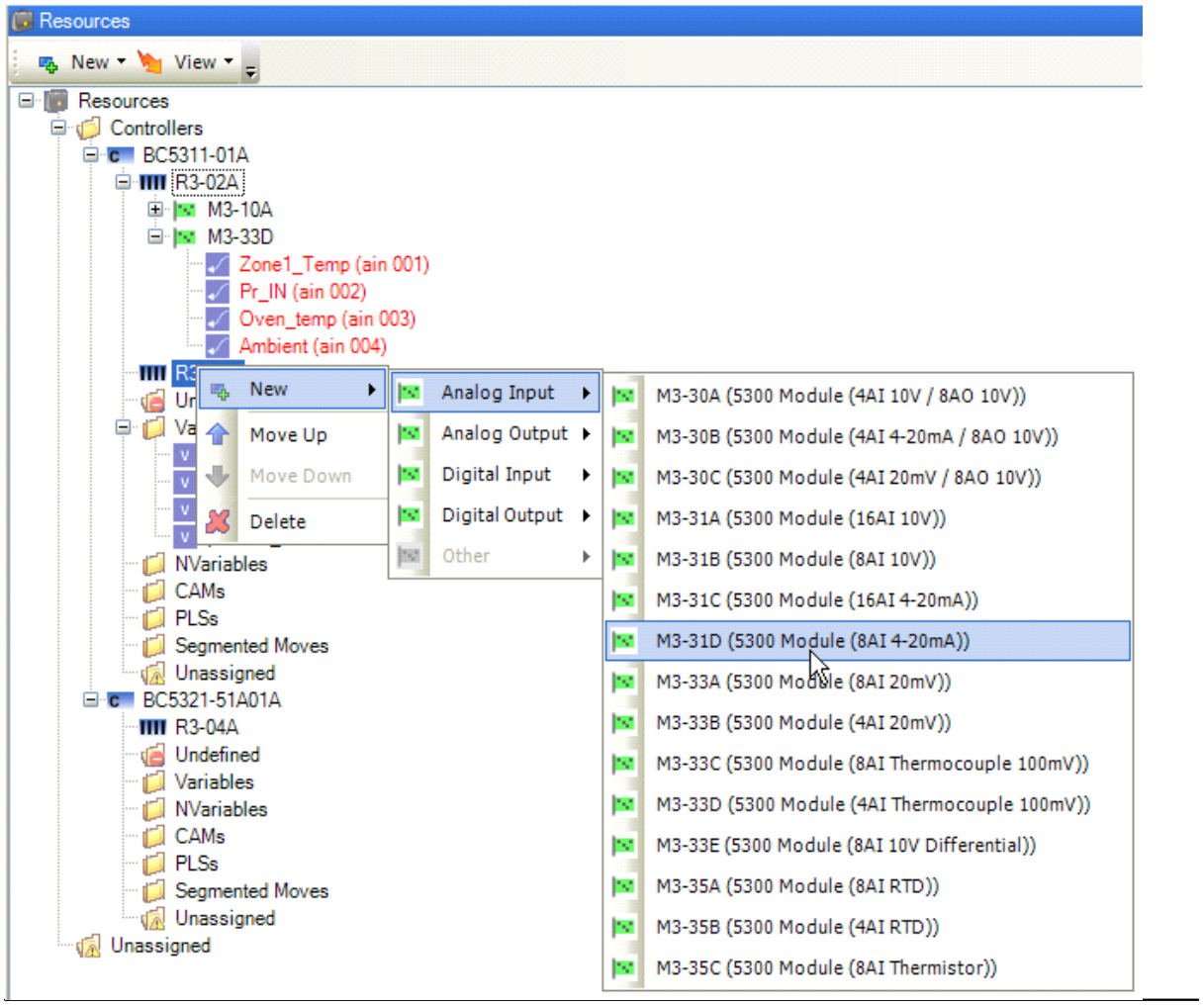


4.1.2 Property Inspector

- View and edit detailed properties of the selected item in the Resource manager
- Set initialization values

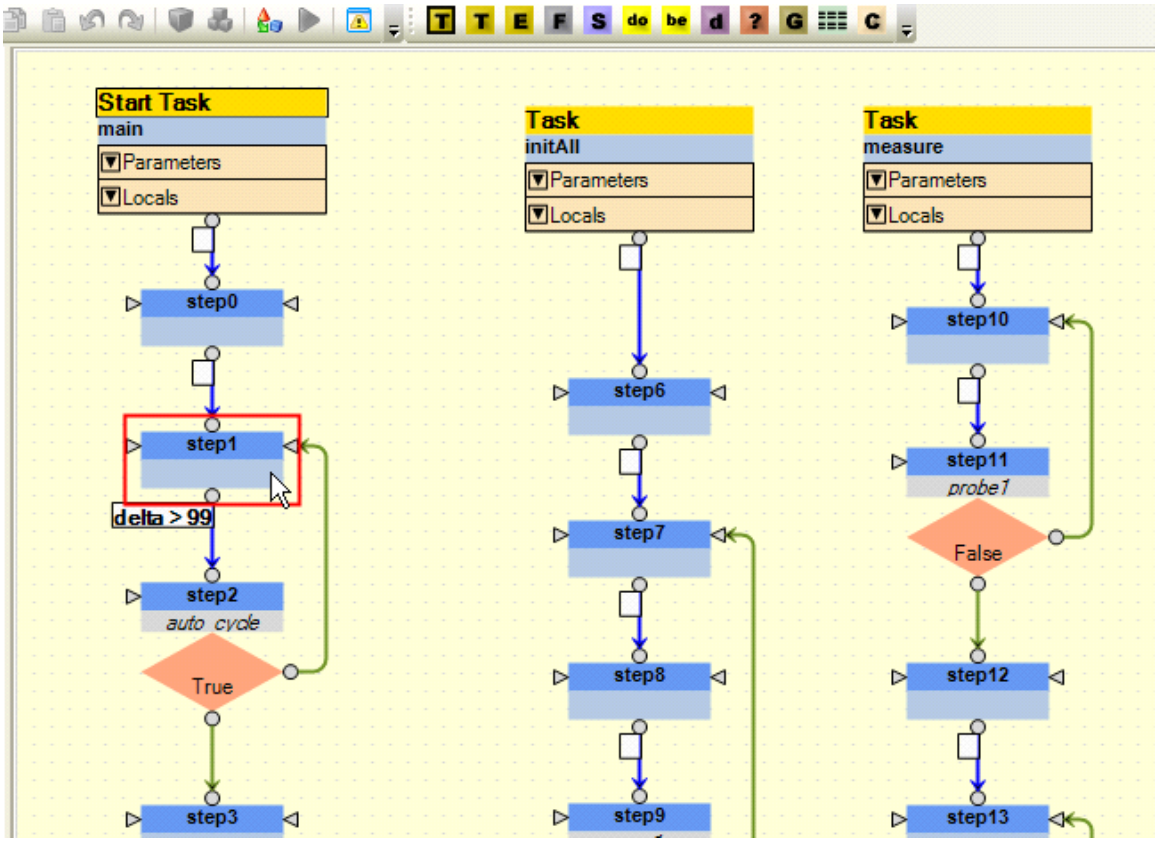


4.1.3 IO Configuration



4.2 Program

QuickBuilder combines the ease of use of CTC's proven state based Quickstep language with a graphical implementation of the modular IEC 61131-3 standard. The screen capture below shows the graphical representation of the sequential flow control of the program. QuickBuilder's graphical flowcharting divides the program into logical elements that are easy to manage and visualize. You are allowed to create multiple pages to organize your code into logical segments. There are no limits to the size of pages of the number of pages you decide to use.

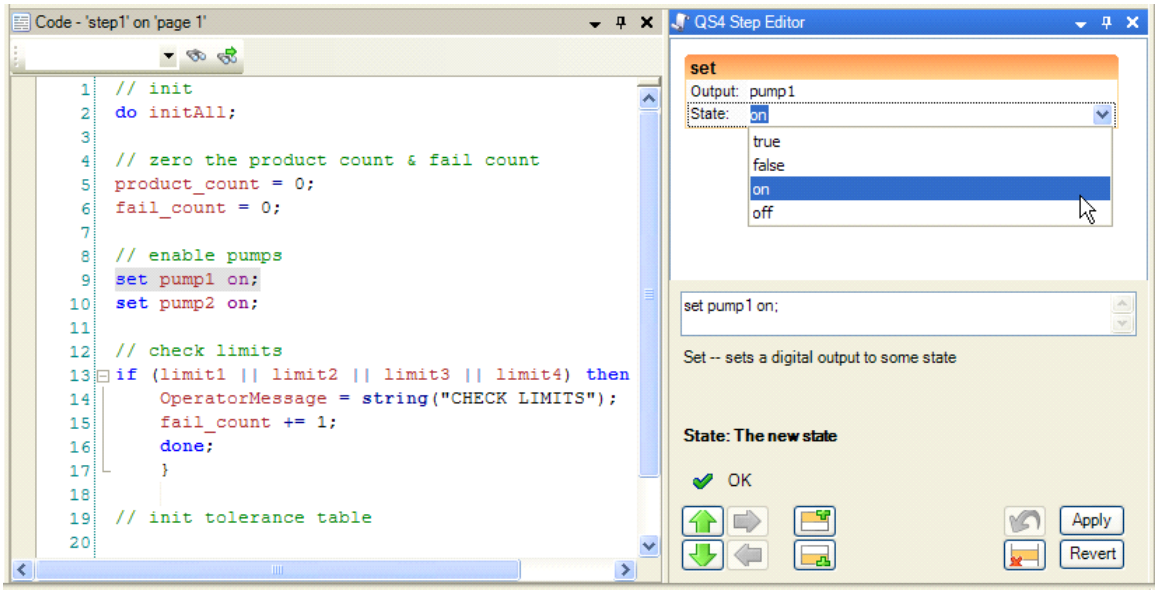


Features

- Task control
- Events
- Functions
- Subroutines
- Exception handling
- While
- Repeat … until
- For, by
- If, then, else
- Text/string commands
- Floating point support
- Advanced math & trig
- Multiple dimension arrays
- Timers
- Logic functions
- Resuable code segments
- Specialized motion commands

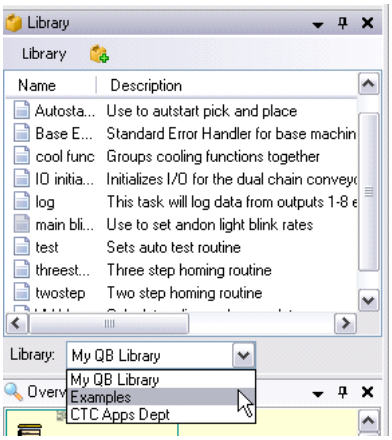
QuickStep 4 (QS4) is the primary programming language used in QuickBuilder applications. QS4 is the latest generation of CTC's famous state-based programming language. Now updated with floating point, the ability to handle advanced algorithms, strings, floats complex branching, etc. Patterned after 1131 structured text, but much easier to use!

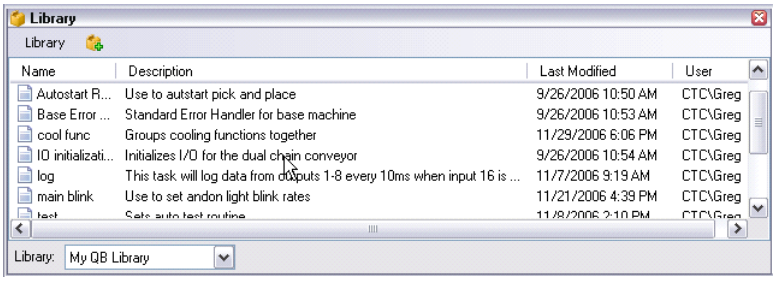
A powerful Step Editor allows beginners to quickly complete applications.



4.3 Program Libraries

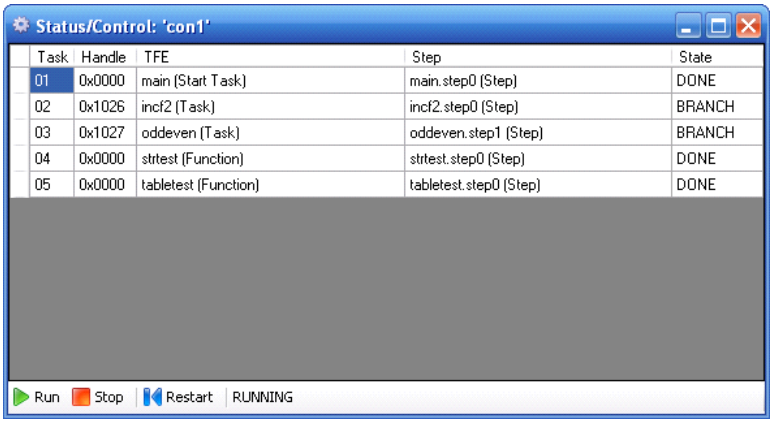
QuickBuilder's powerful Library Manager allows individuals and teams to easily share and re-use common code elements. Any portion of a project from a step, a series of steps, a task or an entire page can be saved as a library element. Library elements are stored in a folder that can be located on the local PC or shared server. Multiple libraries are supported.

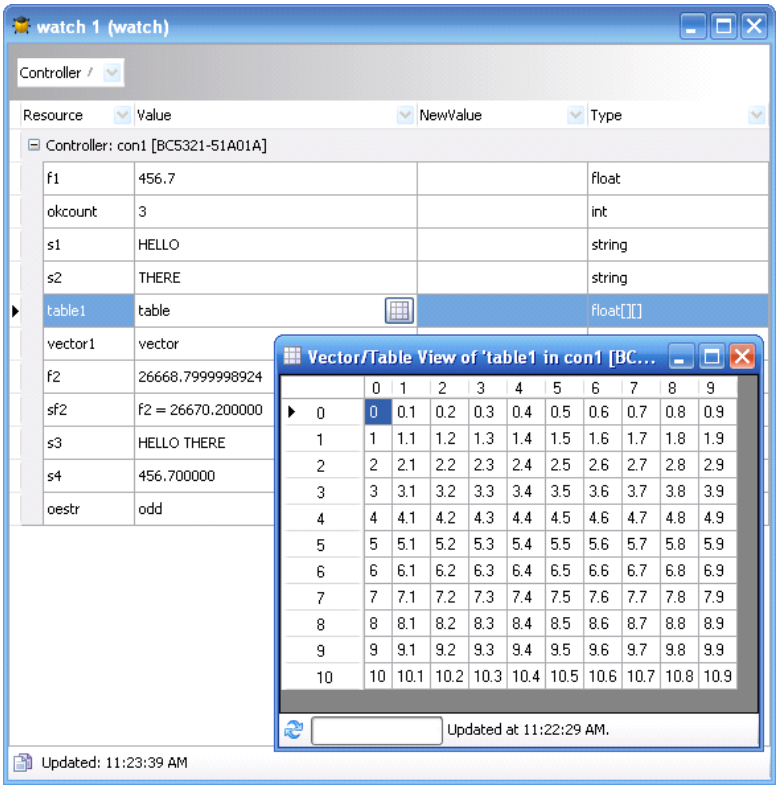




4.4 Test & Debug tools

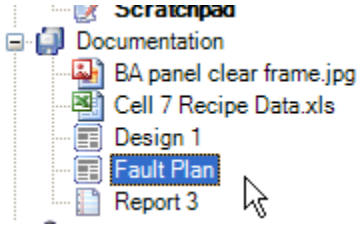
- Program flow control
- Multiple watch windows
- System state capture to file
- Array builder
- Forced I/O capability
- Data logging

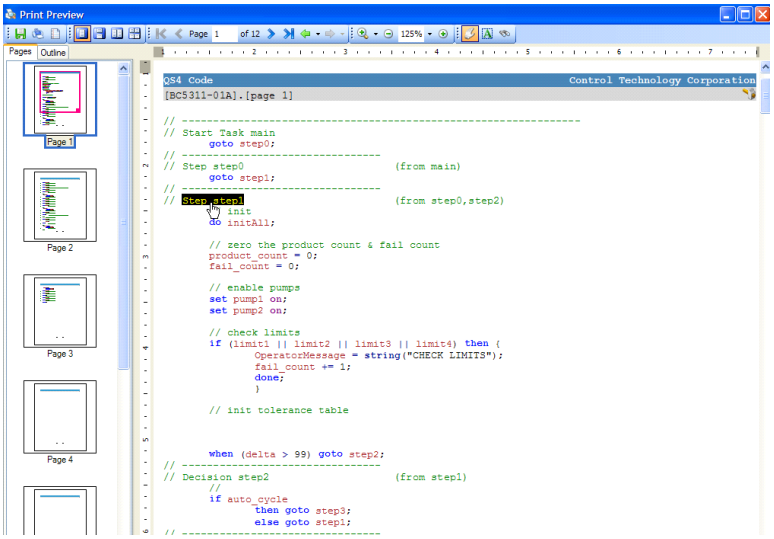




4.5 Maintenance

- Detailed auto report generation for system documentation, revision control, password protection
- Ability to upgrade system and module firmware via the network connection.
- Monitoring tools, remote diagnostics, logging, email exception reporting.





4.6 QuickBuilder benefits

Unified Development Environment

- Project planing
- Hardware configuration
- Network configuration
- Project design
- Project testing
- Project documentation
- Project maintenance
- Project security

Integrated Machine Control

- Motion control
- Logic control
- Web server integration
- Data logging
- Communications
- HMI integration
- Enterprise IT integration
- Exception handling

Improved Project Documentation

- Automatic PDF report generation
- Hyperlinked to actual project code
- Resource cross-reference
- Project information
- Graphical SFC documentation
- Step based code documentation
- Single and multi-line comments
- Integrated word processor
- Ability to embed third party documents, e.g., CAD drawings, Word files, spreadsheets, etc.
- Option to load complete project documentation set to the controller

Simplified User Interface

- Graphical .NET desktop
- Drag and drop flowchart builder
- Automatic step builder
- Color coded editing
- In-line error checking
- Extensive bubble help
- Auto-lookup and auto-completion
- Task specific windowing

5 Appendix B - QuickStep 2 Code

For those familiar with Quickstep 2, this is how you would code the project example in this document using Quickstep2.

```
[1] Start
   ;;; Step 1 turns all outputs off and then
   ;;; Waits for the start switch to be pressed.
  ---------------------------------------------------------------------------
   <TURN OFF ALL DIGITAL OUTPUTS>
  ---------------------------------------------------------------------------
   monitor Start_Switch_On goto Extend_the_Punch

[2] Extend_the_Punch
   ;;; The punch and pusher are turned on to punch out a new part.
  ---------------------------------------------------------------------------
   Punch_On
   Pusher_On
  ---------------------------------------------------------------------------
   monitor Punch_Extended & Pusher_Extended goto Stamp_Part

[4] Stamp_Part
   ;;; While the pusher holds the part in place the
   ;;; part is stamped.
  ---------------------------------------------------------------------------
   Stamp_On
  ---------------------------------------------------------------------------
   monitor Stamp_Extended goto Eject_Part

[6] Eject_Part
   ;;; Turn off the Punch, Stamp and Pusher and turn on the Ejector
   ;;; Delay 1000ms
  ---------------------------------------------------------------------------
   Punch_Off
   Stamp_Off
   Pusher_Off
   Ejector_On
  ---------------------------------------------------------------------------
```

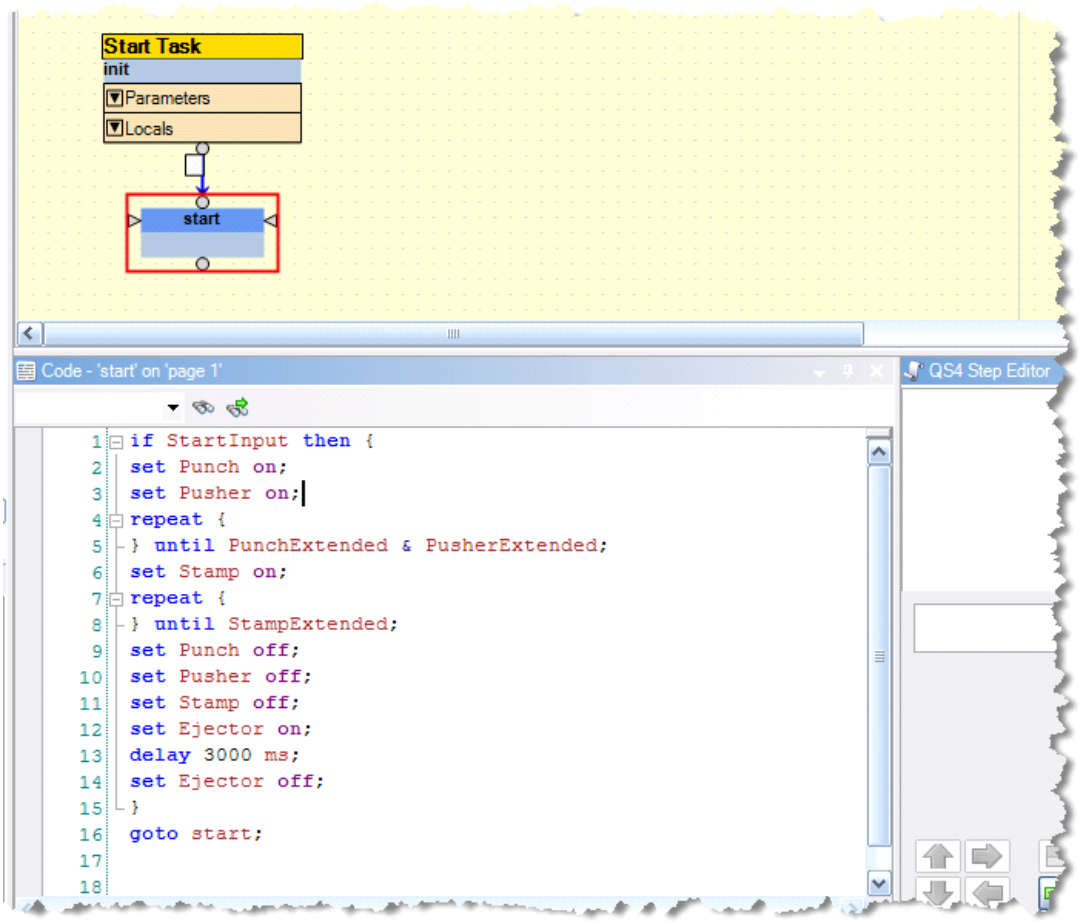
```
   Delay 1000 ms goto Eject_2

[7] Eject_2
   ;;; Turn off the Ejector and goto the start again
   ---------------------------------------------------------------------------
   Ejector_Off
   ---------------------------------------------------------------------------
   Goto Start
```

6 Appendix C - One Step Program

If you're familiar with Basic, C, Java or any other similar language you could write a program that does the exact same thing we previously created with our QuickStart program by simply creating a single step task as shown here.

This simple program lends itself to be programmed in one step due to its simplicity and small size. While you will not want to write a typical program in just one step, you can see that you can combine what used to be several "typical" QuickStep2.x steps into one QuickBuilder Step.



Index

- C -

Code window 33
Code/Comment Editor 18
Comment icon 28
comments, adding to QS4 program 28

- D -

debugging 42, 59
documentation, of code 48, 60

- E -

Errors and Warnings icon 42

- G -

Graphical SFC (Sequential Function Chart) Window 18, 56

- I -

installation of QuickBuilder:
- and Microsoft Vista 6
- software requirements 6

IP address setup 6

- M -

Model 5300 controller setup 6, 10
Model 5300 controller setup (within QuickBuilder): 19
- adding a controller 19
- I/O 25, 55
- I/O cards 20, 23
- rack 20

- O -

Overview View window 18

- P -

pdf generation 48, 60
program, sample: 15, 27
- adding a loop 37
- adding a step 30, 37
- compiling 42
- debugging 42
- documentation of, 48
- downloading to controller 42
- entering code 33, 37
- one-step program equivalent 64
- pdf generation 48
- publishing 42
- QuickStep 2 code equivalent 62
- saving 41
- translating 42

Project Manager 18, 53
Properties Window 18
Property Inspector 54
Publish and Run icon 42

- Q -

QS4 Code Editor 18
QuickBuilder benefits 61
QuickBuilder interface windows: 52
- Code/Comment Editor 18
- Graphical SFC (Sequential Function Chart) Window 18, 56
- Library Manager 58
- Overview View window 18
- Project Manager 18, 53
- Properties Window 18
- Property Inspector 54
- QS4 Code Editor 18
- Resource Window 18

QuickStep 2 code 62

- R -

RAMdisk, mounting 14
Resource Window 18

- S -

StartTask, creating 27
Statement icon 33
Status/Control window 42
Step Editor 33
Step icon 30

- T -

Task (Start) icon 27
Translate icon 42
troubleshooting 42, 59

- W -

Watch window 42, 59
WebMON 10