CHAPTER 2

GETTING STARTED, BASICS AND EXAMPLES

In This Chapter...

Overview

This chapter is intended for the newcomer and includes brief descriptions of how to implement some common motion control solutions using CTRIO(2). The descriptions should give the newcomer a good understanding of what basic steps are required to implement the function. With this general understanding, specifics on each step can be sought out later in the manual.

Later in the chapter, two detailed examples walk the user through what is required to implement two of the most common functions, reading a quadrature encoder and generating a trapezoidal profile.

Basic Motion Functions, Summary of Examples

Get Position Using an Encoder

To read the position of an encoder, follow these basic steps in CTRIO Workbench:

1: Config I/O - Configure the appropriate inputs to recognize the encoder.

2: Optionally set up position scaling if desired.

3: Use I/O Map to map the CTRIO(2) to the controller's memory and print the I/O Map Report.

4: Download the configuration to the CTRIO(2) module and put the CTRIO(2) in Run.

5: Finally, go online with the controller to check that the encoder counts are appearing in the mapped address.

Get Rate Using an Encoder

To read the rate of an encoder, follow these basic steps in CTRIO Workbench:

1: Select Config I/O under Module Configuration to Configure the appropriate inputs to recognize the encoder.

2: Set scaling to rate - Scaling Wizard ruler button:

3: Choose the conversion parameters.

4: Use the Rate Scaling Calculator to verify the chosen settings.

5: Use I/O Map to map the CTRIO(2) to the controller's memory and print the I/O Map Report.

6: Download the configuration to the CTRIO(2) module and put the CTRIO(2) in Run.

7: Go online with the controller to check that encoder rate is appearing in the mapped address.

Measure Timing Between Pulse Edges

To measure the time between edges of a pulse in CTRIO Workbench:

1: Select Config I/O under Module Configuration to configure an input as Edge Timer, selecting the appropriate options (free-run is suggested for testing since it does not require interaction from the controller to function).

2: Optionally, set up scaling if desired.

3: Select I/O Map to map the CTRIO(2) to the controller memory and print the I/O Map Report.4: Download the configuration to the CTRIO(2) module and put the CTRIO(2) in Run.

5: Go online with the controller to check that pulse measurements are appearing in the mapped address.

Output Position Pulses

Several options are available for generating pulses for controlling drives, steppers, servos, etc.. Some options are only available on a CTRIO2 module, as noted. Those only available on CTRIO2 tend to be more versatile and are preferred. Shaded cells highlight the advantaged attributes.

Options	Position Source	Change target position on the fly	Accel / Decel ramps	Maximum pulse rate	Encoder Feedback Possible	Requires CTRIO2
Dynamic Positioning Plus	From controller	Yes	Specify separately	250kHz	Yes	Yes
Trapezoid Plus	From controller	Yes	Specify separately	250kHz	Yes	Yes
Dynamic Positioning	From controller	No	One setting for both	65kHz[1]	No	No
S-Curve	Hard-coded in pulse profile	No	Specify separately	65kHz[1]	No	No
Symmetrical S-Curve	Hard-coded in pulse profile	No	One setting for both	65kHz[1]	No	No
Trapezoid	Hard-coded in pulse profile	No	Specify separately	65kHz[1]	No	No
Free Form	Hard-coded in pulse profile	No	None	65kHz[1]	No	No

The following example uses Dynamic Positioning Plus on a CTRIO2. To use Dynamic Positioning Plus to send output pulses to an amplifier (without encoder feedback), follow these basic steps in CTRIO Workbench:

1: Select Config I/O under Module Configuration to configure the outputs to provide pulses appropriate for the amplifier.

2: Select Pulse Profiles at bottom of dialog box.

3: Optionally, give the profile a name.

4: Select Dynamic Positioning Plus and choose the Frequency Settings appropriate for the motor and system.

5: Note the File Number assigned.

6: Use I/O Map to map the CTRIO(2) to the controller's memory and print the I/O Map Report.

7: Download the configuration to the CTRIO(2) module and put the CTRIO(2) in Run.

[1] The older CTRIO outputs can only achieve 25kHz maximum. CTRIO2 outputs are limited to 65kHz when using these profiles.

Home an Output

There are several available options for finding home on an output. If using a CTRIO2, *Trapezoid with Limits* profile will nearly always be the best option. The following example uses Trapezoid with Limits profile on a CTRIO2.

The table below shows a comparison of the available methods of homing an output with a CTRIO(2) module. Shaded cells highlight the advantaged attributes.

To home an output with one or more limit switches, follow these basic steps in CTRIO Workbench.

Profile	Accel and Decel Ramps	Creep to second limit or position	Change velocity manually on the fly	Multiple limits or triggers on a single home search	Single input can act as multiple triggers	Profile defined in CTRIO Workbench	Requires CTRIO2
Trapezoid with Limits	Yes	Yes	No	Yes	Yes	Yes	Yes
Home Search*	No	No*	No	Yes	No	Yes	No
Run to Limit	No	No	Yes	No	No	No	No

*Home Search allows you to select 2nd limit or different speed.

1: Select Config I/O to configure the outputs to provide pulses appropriate for the chosen amplifier.

2: Configure one or more inputs as *Limit Out n* where *n* is the output configured above.

3: Select Pulse Profiles at bottom of dialog box.

4: Select Add to create a new profile, or select an existing named profile.

5: Optionally, give the profile a name (defaults to "File 1-" upon closing dialog box).

6: Select Trapezoid profile with Limits.

7: Configure the Decel and Stop Triggers.

8: Choose the Frequency Settings appropriate for the motor and system.

9: Note the File Number assigned.

10: Use I/O Map to map the CTRIO(2) to the controller's memory and print the I/O Map Report.

11: Download the configuration to the CTRIO(2) module and put the CTRIO(2) in Run.

NOTE: *The following steps are generic. Details are available in the manual.*

In the controller:

1: Command the CTRIO2 to load the Trapezoid with Limits profile (by file number) for the appropriate output.

2: Specify the direction (CW/CCW) to seek home.

3: Set the appropriate CTRIO2 Enable Output bit.

4: The controller can monitor status bits for visibility into the CTRIO's progress.

Output PWM Pulses1

To generate PWM outputs, follow these basic steps in CTRIO Workbench.

1: Config I/O - Configure an output for Step/*Direct*ion, the step output will have the PWM signal.

2: Use I/O Map to map the CTRIO(2) to the controller's memory and print the I/O Map Report.

3: Download the configuration to the CTRIO(2) module and put the CTRIO(2) in Run.

4: In the controller - (**Note:** These steps are generic. Details are available in the *Direct*LOGIC manual).

5: Set the command code for Velocity Mode.

6: Set Parameter 1 (Frequency).

7: Set Parameter 2 (Duty Cycle) from 1 to 99.

8: Set Parameter 3 (Number of Output Pulses) to FFFF FFFF for unlimited.

9: Set the appropriate CTRIO2 Enable Output bit to start pulses.

10: To stop pulses, reset the appropriate Enable Output bit.

11: To change the Duty Cycle, need source.

Programmable Limit Switch or Preset Table

To control one of the CTRIO(2) outputs with a PLS or Preset Table that monitors an encoder input, follow these basic steps in CTRIO Workbench.

1: Config I/O - Configure the appropriate inputs to recognize the encoder, noting the Channel and Function numbers assigned.

2: Optionally set up scaling if desired.

3: Configure an output as Discrete on Ch_/Fn_, using the Channel and Function numbers from the encoder input.

4: Discrete Tables…

5: Add Preset Table or Add PLS Table[1]

6: Optionally give the table a name.

7: Configure the table for the desired behavior.

8: Scales will be available if scaling was defined for the input.

9: Note the File Number assigned.

10: Use I/O Map to map the CTRIO(2) to the controller's memory and print the I/O Map Report.

11: Download the configuration to the CTRIO(2) module and put the CTRIO(2) in Run.

NOTE: *The following steps are generic. Details are available in the manual.*

In the controller:

1: Command the CTRIO2 to load the Table by its file number

2: Set the appropriate CTRIO(2) Enable Output bit

3: The controller can monitor status bits for visibility into the CTRIO's progress

Output Discrete On/Off from Ladder

The output points on a CTRIO(2) can be turned on and off from ladder, called Raw control. Keep in mind that they will not react as quickly as the outputs of a standard discrete output module since there is also the scan time of the CTRIO(2) that can add latency. To simply turn the output of a CTRIO(2) on or off from ladder, follow these basic steps in CTRIO Workbench:

1: Config I/O - Configure any outputs to be controlled as Raw.

2: Use I/O Map to map the CTRIO(2) to the controller's memory and print the I/O Map Report.

3: Download the configuration to the CTRIO(2) module and put the CTRIO(2) in Run.

NOTE: *These steps are generic. Details are available later in the manual.*

In the controller:

1: Set the appropriate CTRIO(2) Enable Output bit.

2: The controller can monitor status bits for visibility into the CTRIO's progress.

[1] PLS Tables are only available when using CTRIO2 hardware.

Detailed Example: Configure and Test a Quadrature Input

This example walks through the steps required to get the counts from a quadrature encoder connected to a CTRIO(2) to appear in the CPU memory of a *Direct*LOGIC PLC. The example uses DL06 hardware, but is applicable to DL05 and DL205 hardware as well.

The Basic Steps

1: Gather and connect the hardware (not covered here).

2: Launch CTRIO Workbench.

3: Use Config I/O to configure the appropriate inputs to recognize the encoder.

4: Use I/O Map to map the CTRIO(2) to the controller's memory and print the I/O Map Report.

5: Write the configuration to the CTRIO(2) module and put the CTRIO(2) in Run.

6: Use Monitor I/O to verify the encoder movement is being interpreted by the CTRIO(2) properly.

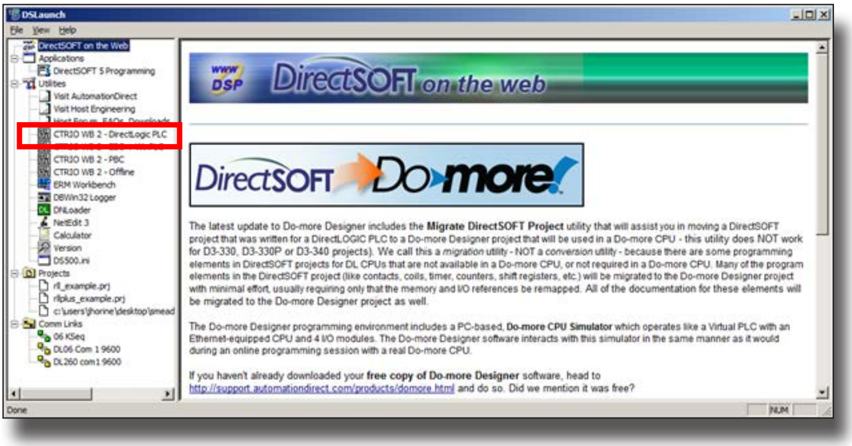
7: Launch *Direct*SOFT and use Data View to check that position data is appearing in the mapped addresses.

Equipment Needed

*Direct*LOGIC 06 base and H0-CTRIO2 installed in slot 4[1]. A quadrature encoder properly powered and connected to the H0-CTRIO2.

Launch CTRIO Workbench

When using CTRIO Workbench with a *Direct*LOGIC CPU and the latest version of *Direct*SOFT, the best way to launch it is from DSLaunch. DSLaunch is installed when installing *Direct*SOFT. Open DSLaunch.exe.



The latest update to Do-more Designer includes the **Migrate DirectSOFT Project** utility that will assist you in moving a DirectSOFT project that was written for a DirectLOGIC PLC to a Do-more Designer project that will be used in a Do-more CPU - this utility does NOT work for D3-330, D3-330P or D3-340 projects). We call this a *migration* utility - NOT a conversion utility - because there are some programming elements in DirectSOFT projects for DL CPUs that are not available in a Do-more CPU, or not required in a Do-more CPU. Many of the program elements in the DirectSOFT project (like contacts, coils, timer, counters, shift registers, etc.) will be migrated to the Do-more Designer project with minimal effort, usually requiring only that the memory and I/O references be remapped. All of the documentation for these elements will be migrated to the Do-more Designer project as well.

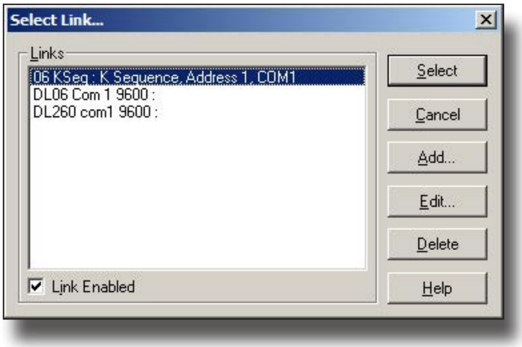
The Do-more Designer programming environment includes a PC-based, **Do-more CPU Simulator** which operates like a Virtual PLC with an Ethernet-equipped CPU and 4 I/O modules. The Do-more Designer software interacts with this simulator in the same manner as it would during an online programming session with a real Do-more CPU.

If you haven't already downloaded your **free copy of Do-more Designer** software, head to http://support.automationdirect.com/products/domore.html and do so. Did we mention it was free?

Click "CTRIO WB 2 – *Direct*LOGIC PLC" to open CTRIO Workbench. The Select

[1] If using *Direct*LOGIC 205 hardware, the CTRIO(2) cannot be installed directly next to the CPU slot.

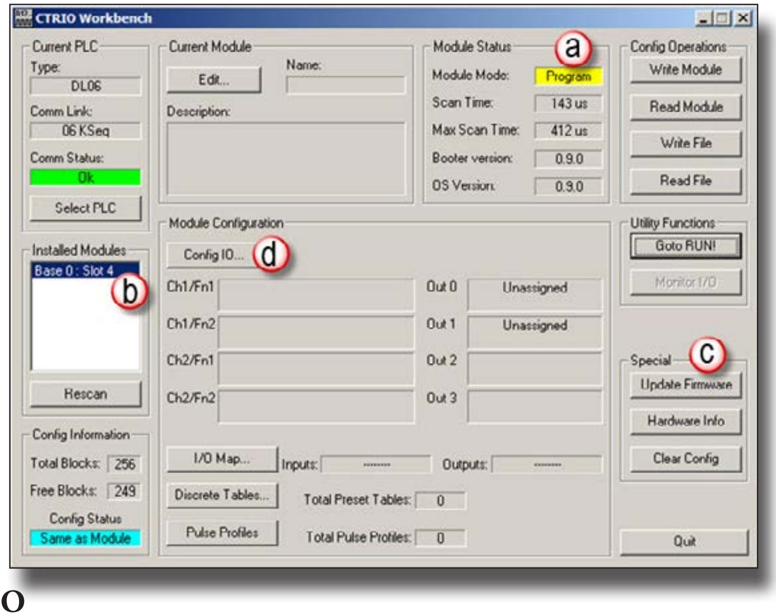
Link… dialog box appears. Select the appropriate PLC communication link to use. If it does not appear in the list, check power and the communications cable or, create a new link by clicking on 'Add'.


Select Link...

Links
06 KSeq : K Sequence, Address 1, COM1
DL06 Com 1 9600 :
DL260 com1 9600 :

☑ Link Enabled

Once connected, CTRIO Workbench will look like the one below. The H0-CTRIO2 shown is new and has no configuration. It is in Program mode (a). It is installed in slot 4 (b). Now is a good time to check the firmware version and make sure it's current (c).


CTRIO Workbench

Current PLC
Type: DL06
Comm Link: 06 KSeq
Comm Status: Ok

Current Module
Name:
Description:

Module Status
Module Mode: Program
Scan Time: 143 us
Max Scan Time: 412 us
Booter version: 0.9.0
OS Version: 0.9.0

Installed Modules
Base 0 : Slot 4

Module Configuration
Ch1/Fn1
Ch1/Fn2
Ch2/Fn1
Ch2/Fn2
Out 0 Unassigned
Out 1 Unassigned
Out 2
Out 3
Inputs: --------
Outputs: --------
Total Preset Tables: 0
Total Pulse Profiles: 0

Config Information
Total Blocks: 256
Free Blocks: 249
Config Status: Same as Module

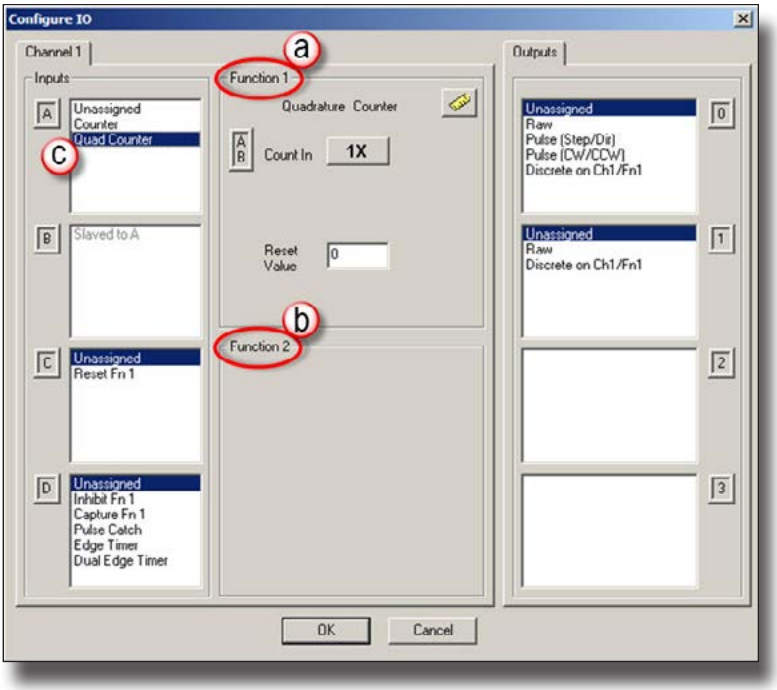
Config I/O

For this example, it is only necessary to configure the inputs of the H0-CTRIO2 to recognize a quadrature encoder. Click Config I/O (d) command box.

The Configure I/O panel is divided into two sections. One section is for Channel 1 Inputs (A-D) and the other section is for the Outputs (0-3). If working with an H2-CTRIO2, there would also be a tab for Channel 2.

Within Channel 1, note there are Function 1 (a) and Function 2 (b). Each Channel on a CTRIO(2) module may have up to two Functions assigned.

On Channel 1, select Quad Counter (c). This tells the CTRIO(2) to expect quadrature signals on Channel 1 Inputs A and automatically, B.



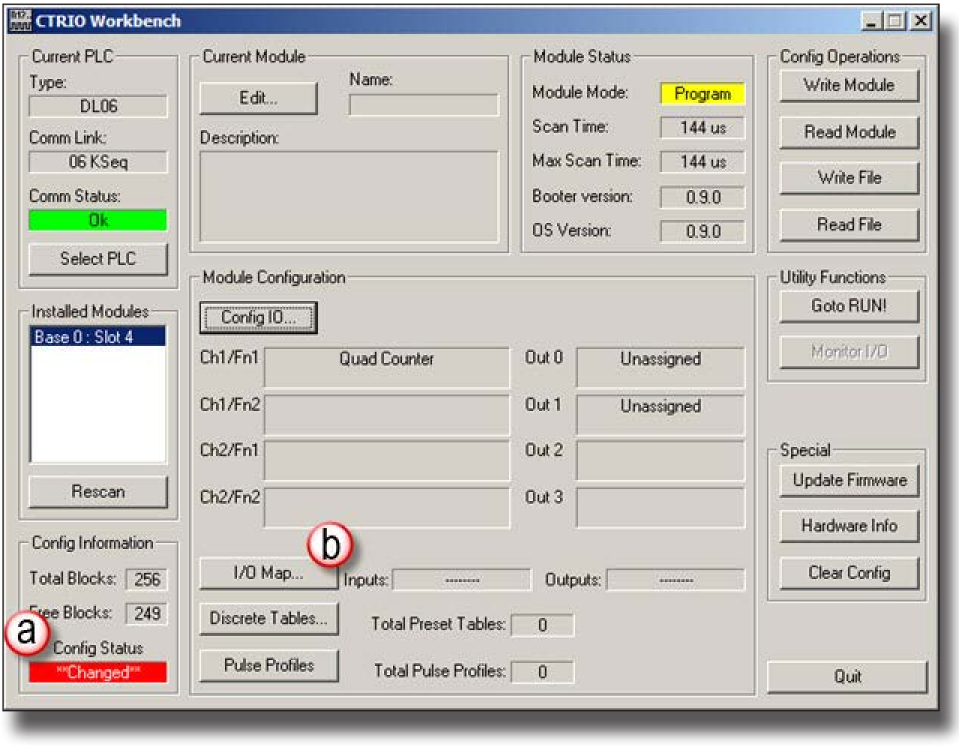
NOTE: Many other defaults change when a selection is made:

- The option for Input B to be a simple counter disappears and it is assigned as Slave to A and can no longer be directly changed.
- Some options for Input C disappear and a new option appears.
- Channel 1 Function 1 area displays Quadrature Counter and offers multiple options that apply to that counter input.
- New options for the Outputs appear that reference Ch1/Fn1, the quadrature input function created.

NOTE: *This automatic reconfiguring of available options is an important feature of CTRIO Workbench. The primary benefit is that it prevents the user from selecting options that will not work together. It is not possible to create an invalid configuration. However, keep this feature in mind when going about configuring a CTRIO(2). If a function cannot be found, it's likely that some dependency has not yet been enabled, or a feature that has been enabled is consuming an exclusive resource the desired feature also requires.*

Click OK to keep the changes and go back to the CTRIO Workbench home screen.

On the home screen, note the indication in the lower left (a), Config Status **Changed**. This indicates that the configuration in Workbench does not match what was read from the module. Before the new configuration is written to the module, there is one other thing that must be done. Click I/O Map (b). This brings up the I/O Map dialog box.



Type: DL06
Comm Link: 06 KSeq
Comm Status: Ok
Base 0 : Slot 4
Total Blocks: 256
Free Blocks: 249
Config Status **Changed**

Module Mode: Program
Scan Time: 144 us
Max Scan Time: 144 us
Booter version: 0.9.0
OS Version: 0.9.0

Ch1/Fn1 Quad Counter
Ch1/Fn2
Ch2/Fn1
Ch2/Fn2
Out 0 Unassigned
Out 1 Unassigned
Out 2
Out 3
Inputs: --------
Outputs: --------
Total Preset Tables: 0
Total Pulse Profiles: 0

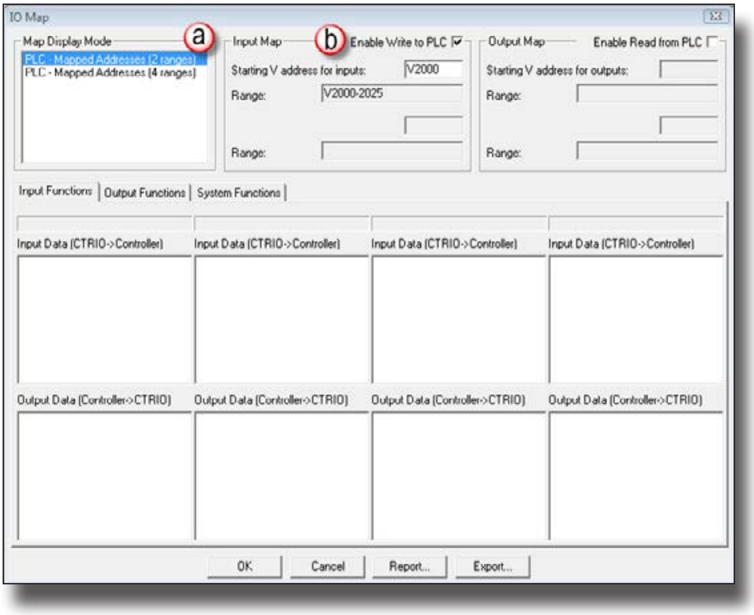
I/O Map

Depending on the configuration, the CTRIO(2) will have some number of variables to pass back and forth to the CPU. The data does not fit in the normal structure that most I/O modules use. Instead, the CTRIO(2) is able to write to and read from any CPU memory specified. In this panel, map the variables into CPU memory.

Be careful to map memory addresses that will not be used for any other purpose.

In Map Display Mode (a), select "PLC - Mapped Addresses (2 ranges)" mode.

In the Input Map box (b) point the CTRIO(2) where to write input data to the PLC memory. Click "Enable Write to PLC" and type in an address for "Starting V address for inputs". V2000 is used here as the starting address.



IO Map

Map Display Mode: PLC - Mapped Addresses (2 ranges) (selected); PLC - Mapped Addresses (4 ranges)

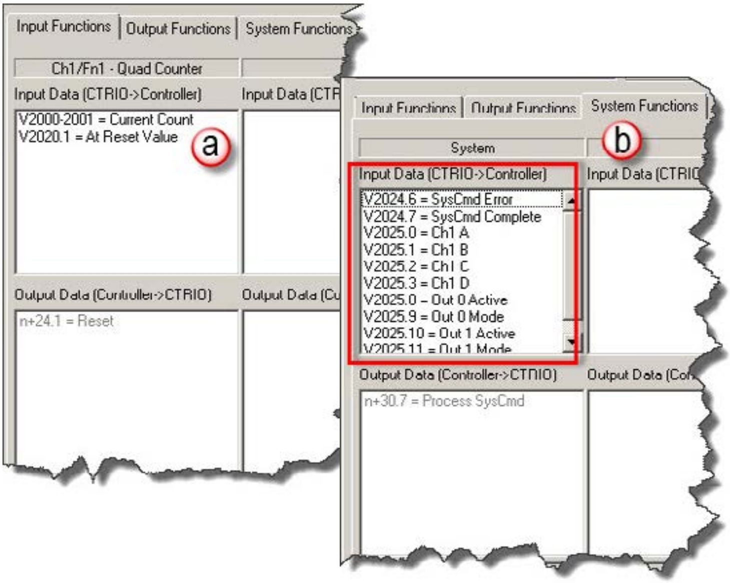
Input Map — Enable Write to PLC ☑
Starting V address for inputs: V2000
Range: V2000-2025
Range:

Output Map — Enable Read from PLC □
Starting V address for outputs:
Range:
Range:

a
b
Input Functions
Output Functions
System Functions
Input Data (CTRIO->Controller)
Input Data (CTRIO->Controller)
Input Data (CTRIO->Controller)
Input Data (CTRIO->Controller)
Output Data (Controller->CTRIO)
Output Data (Controller->CTRIO)
Output Data (Controller->CTRIO)
Output Data (Controller->CTRIO)
OK
Cancel
Report...
Export...

Note how entering the address changes the display under (a) Input Function tab, “Ch1/Fn1 – Quad Counter Input Data”. The two variables displayed are Current Count (32 bit integer that gets mapped to V2000-2001) and AT Reset Value (a bit that gets mapped to the second bit of V2020, or V2020.1). Clicking on the (b) “System Functions” tab reveals ten more status bit variables that were also mapped. Each of these variables provides status information to the CPU.

The “Output Map”(c) section tells the CTRIO(2) where to read variables out of the CPU and to be able to reset the counter from the ladder logic program.

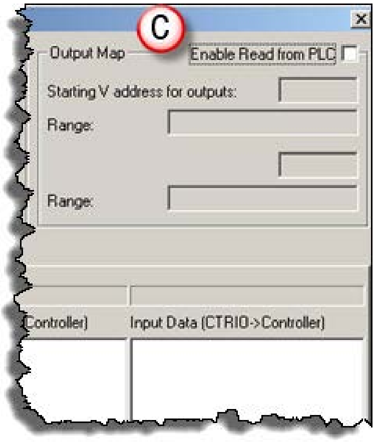


V2000-2001 = Current Count
V2020.1 = At Reset Value

n+24.1 = Reset

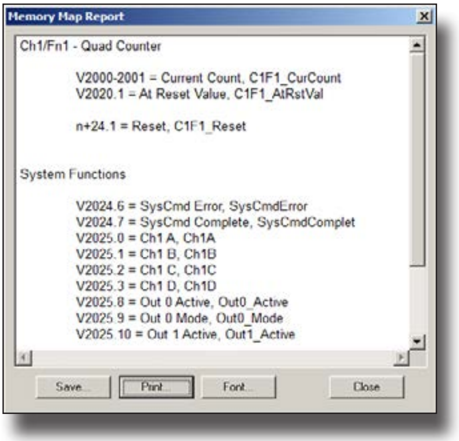
V2024.6 = SysCmd Error
V2024.7 = SysCmd Complete
V2025.0 = Ch1 A
V2025.1 = Ch1 B
V2025.2 = Ch1 C
V2025.3 = Ch1 D
V2025.0 – Out 0 Active
V2025.9 = Out 0 Mode
V2025.10 = Out 1 Active
V2025.11 = Out 1 Mode

n+30.7 = Process SysCmd


c
Output Map
Enable Read from PLC □
Starting V address for outputs:
Range:
Range:
Controller)
Input Data (CTRIO->Controller)

In this particular example, it is not necessary for the CTRIO(2) to read any addresses from the CPU. As configured, once the CTRIO(2) is in run mode, it will constantly report the counts to the CPU via the Current Count variable. As can be seen in the previous two images, if no addresses are specified in “Output Map”, there will be no way for the CTRIO(2) to get to the *Reset* or *Process SysCmd* bits, which are not needed here.

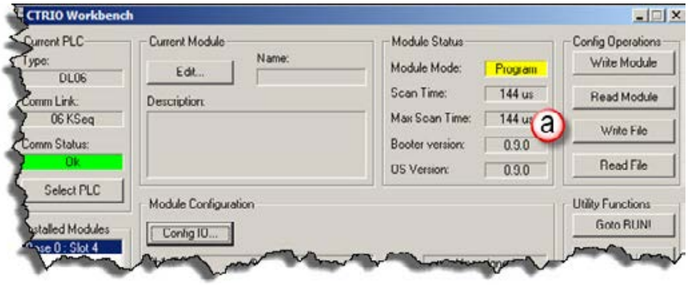
Once the I/O Map is set up, it is best to **print the "Memory Map Report"**. Click on the "Report…" button to access the dialog below. Click on "Print". Creating this hard copy provides a critical reference to have in hand while writing the control program. The report can also be saved to disk as a ".txt" file by clicking on "Save" and giving it a unique name. When finished, close this dialog then click OK on the I/O Map dialog to return to the CTRIO Workbench home screen.



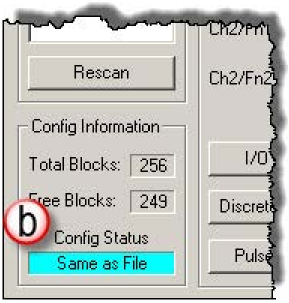
```
Ch1/Fn1 - Quad Counter

        V2000-2001 = Current Count, C1F1_CurCount
        V2020.1 = At Reset Value, C1F1_AtRstVal

        n+24.1 = Reset, C1F1_Reset


System Functions

        V2024.6 = SysCmd Error, SysCmdError
        V2024.7 = SysCmd Complete, SysCmdComplet
        V2025.0 = Ch1 A, Ch1A
        V2025.1 = Ch1 B, Ch1B
        V2025.2 = Ch1 C, Ch1C
        V2025.3 = Ch1 D, Ch1D
        V2025.8 = Out 0 Active, Out0_Active
        V2025.9 = Out 0 Mode, Out0_Mode
        V2025.10 = Out 1 Active, Out1_Active
```

Download the Configuration

Now that the configuration has been defined, press "Write File" (a) to save a copy of the configuration to disk. It will save as a ".cwb" file that should be kept together with the other project files and documents.



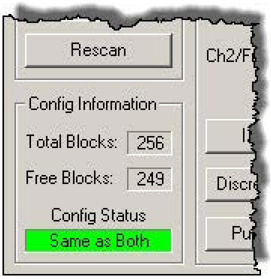
Type: DL06
Comm Link: 06 KSeq
Comm Status: Ok
Module Mode: Program
Scan Time: 144 us
Max Scan Time: 144 us
Booter version: 0.9.0
OS Version: 0.9.0



Total Blocks: 256
Free Blocks: 249
Config Status
Same as File

Note that the "Config Status" (b) changes to **Same as File**, indicating that the file on disk and the file in CTRIO workbench are identical.

Once saved, click “Write Module” to write the file to the CTRIO(2) module. The Config Status changes to **Same as Both**, indicating that the file on disk, the file in CTRIO workbench and the file in the CTRIO(2) module are identical.



Config Information

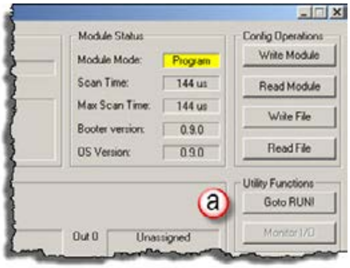
Total Blocks: 256

Free Blocks: 249

Config Status

Same as Both

At this point, the CTRIO(2) module is still in Program mode, doing nothing. Click “Goto RUN!” (a) to have the module begin execution. The module will begin interpreting input pulses as a quadrature input and writing the resulting counts to V2000-2001. Note (b) “Module Mode” changes to **Run**.



Module Status

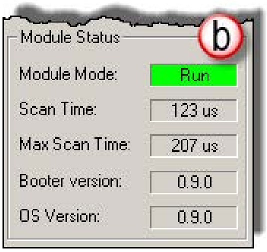
Module Mode: Program

Scan Time: 144 us

Max Scan Time: 144 us

Booter version: 0.9.0

OS Version: 0.9.0



Module Status

Module Mode: Run

Scan Time: 123 us

Max Scan Time: 207 us

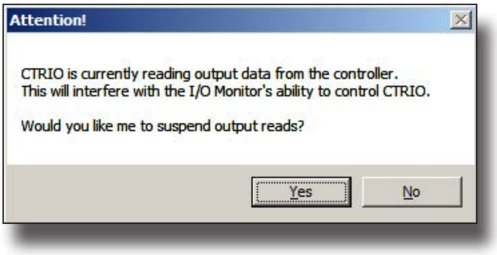
Booter version: 0.9.0

OS Version: 0.9.0

NOTE: *It is only necessary to manually switch the CTRIO(2) into Run or Program during testing. During normal operation, the CTRIO(2) follows changes in the mode of the CPU. If the CPU goes to Stop, the CTRIO(2) goes to Program. If the CPU goes to Run, the CTRIO(2) goes to Run.*

Monitor I/O

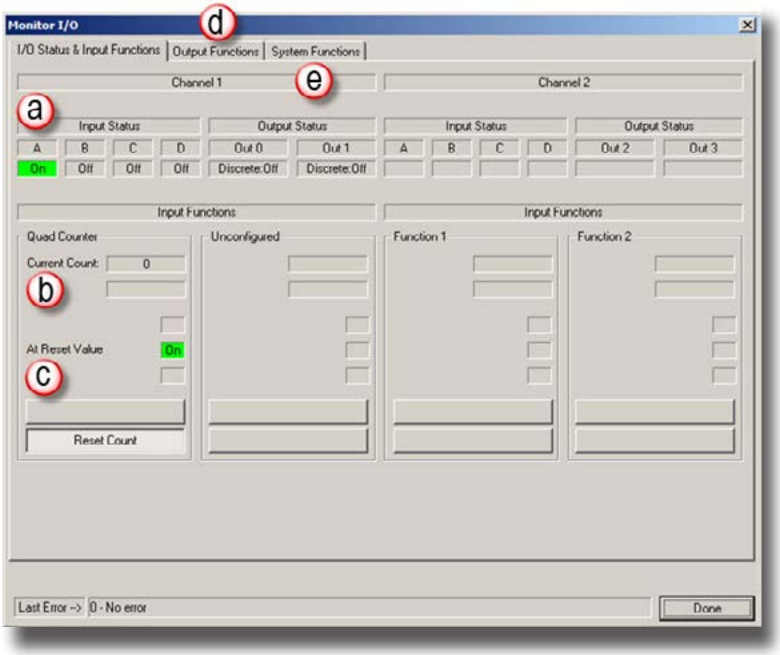
Before jumping over to using the CPU to work with the CTRIO(2) module, there is a very convenient way to verify everything is working as expected from within CTRIO Workbench. Click Monitor I/O. The following prompt may appear, asking whether the system should suspend output reads. Output reads are what the CTRIO(2) does to know what the CPU is telling it to do. Suspending them just means that the CTRIO(2) should temporarily stop taking commands from the CPU. The CTRIO(2) will be controlled from the Monitor I/O panel, so answer Yes.



CTRIO is currently reading output data from the controller.
This will interfere with the I/O Monitor's ability to control CTRIO.

Would you like me to suspend output reads?

The CTRIO(2) stops reading addresses from the CPU and Monitor I/O opens.



A	B	C	D	Out 0	Out 1	A	B	C	D	Out 2	Out 3
On	Off	Off	Off	Discrete:Off	Discrete:Off						

Current Count: 0

At Reset Value: On

Last Error --> 0 - No error

This dialog box displays a variety of important I/O information at a glance. The raw state of each discrete input point is indicated under **Input Status** (**a**). This is very handy for quickly checking the encoder wiring.

Give the encoder a twist to verify that Current Counts (**b**) follows the movement and At Reset Value (**c**) goes low. If no inputs ever appear On, check the encoder wiring carefully. If the text Quad Counter and Current Count do not appear, make sure the configuration was sent to the CTRIO(2) module.

The other tabs, not used in this example, have some very useful tools for monitoring and testing Output Functions (**d**) as well as System Functions (**e**).

Detailed Example: Configure and Test a Pulse Output with a Trapezoidal Profile

This example walks through the steps required to use a *Direct*LOGIC PLC to generate a trapezoidal profile on an output channel of a CTRIO(2). The example uses DL06 hardware, but is applicable to DL05, DL205 and DL405 hardware as well.

The Basic Steps

1: Gather and connect the hardware (not covered here).

2: Launch CTRIO Workbench.

3: Use Config I/O to configure the appropriate outputs to generate pulses appropriate for the amplifier in use.

4: Use I/O Map to map the CTRIO(2) to the controller's memory and print the I/O Map Report.

5: Use Pulse Profile Tables to create a Trapezoid Plus Pulse Profile and select its Frequency Settings.

6: Download the configuration to the CTRIO(2) module and put the CTRIO(2) in Run.

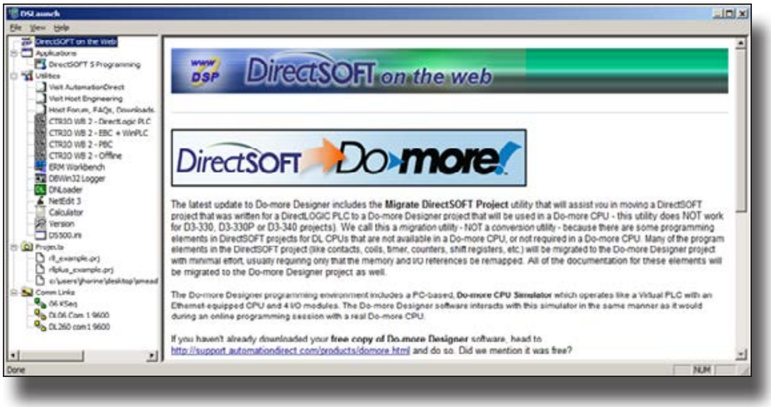
7: Use Monitor I/O to manually generate an output profile to make sure the configuration and hardware connections are correct.

8: Launch *Direct*SOFT and use Data View to manually generate an output profile.

Equipment Needed

*Direct*LOGIC 06 base and H0-CTRIO2 installed in slot 2[1]. A stepper amplifier and motor, properly powered and connected to the H0-CTRIO2. Alternately, the output activity can be monitored in Data View or in Monitor I/O in CTRIO Workbench.

Launch CTRIO Workbench



The latest update to Do-more Designer includes the **Migrate DirectSOFT Project** utility that will assist you in moving a DirectSOFT project that was written for a DirectLOGIC PLC to a Do-more Designer project that will be used in a Do-more CPU - this utility does NOT work for D3-330, D3-330P or D3-340 projects). We call this a *migration utility* - NOT a conversion utility - because there are some programming elements in DirectSOFT projects for DL CPUs that are not available in a Do-more CPU, or not required in a Do-more CPU. Many of the program elements in the DirectSOFT project (like contacts, coils, timer, counters, shift registers, etc.) will be migrated to the Do-more Designer project with minimal effort, usually requiring only that the memory and I/O references be remapped. All of the documentation for these elements will be migrated to the Do-more Designer project as well.

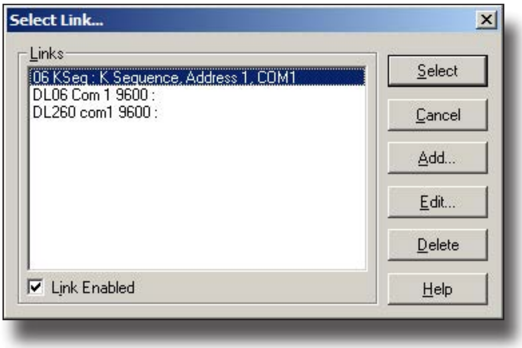
The Do-more Designer programming environment includes a PC-based, **Do-more CPU Simulator** which operates like a Virtual PLC with an Ethernet-equipped CPU and 4 I/O modules. The Do-more Designer software interacts with this simulator in the same manner as it would during an online programming session with a real Do-more CPU.

If you haven't already downloaded your **free copy of Do-more Designer** software, head to http://support.automationdirect.com/products/domore.html and do so. Did we mention it was free?

[1] If using *Direct*LOGIC 205 hardware, the CTRIO(2) cannot be installed directly next to the CPU slot.

The best way to launch CTRIO Workbench is from DSLaunch. DSLaunch is installed as part of a *Direct*SOFT install. Open DSLaunch.exe.

Click *CTRIO WB 2 – DirectLOGIC PLC* to open CTRIO Workbench. The *Select Link…* dialog appears asking which PLC communication link to use. Select the appropriate PLC. If it does not appear in the list, check power and the communications cable; or, create a new link.

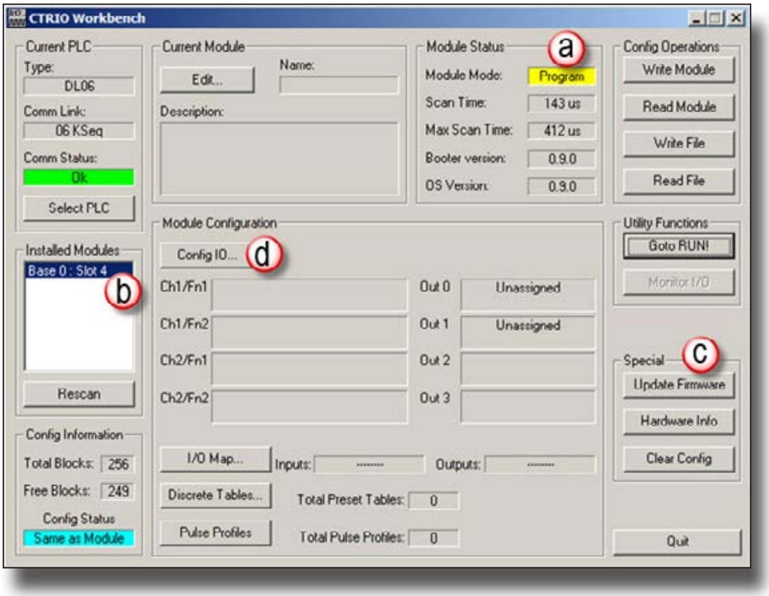

Select Link...

Links

06 KSeq : K Sequence, Address 1, COM1
DL06 Com 1 9600 :
DL260 com1 9600 :

☑ Link Enabled

Once connected, CTRIO Workbench will look like the one below. The H0-CTRIO2 shown is new and has no configuration. It is in Program mode (**a**). It is installed in slot 4 (**b**). Now is a good time to check the firmware version and make sure it's current (**c**).


CTRIO Workbench

Current PLC
Type: DL06
Comm Link: 06 KSeq
Comm Status: Ok

Installed Modules
Base 0 : Slot 4

Config Information
Total Blocks: 256
Free Blocks: 249
Config Status: Same as Module

Current Module
Name:
Description:

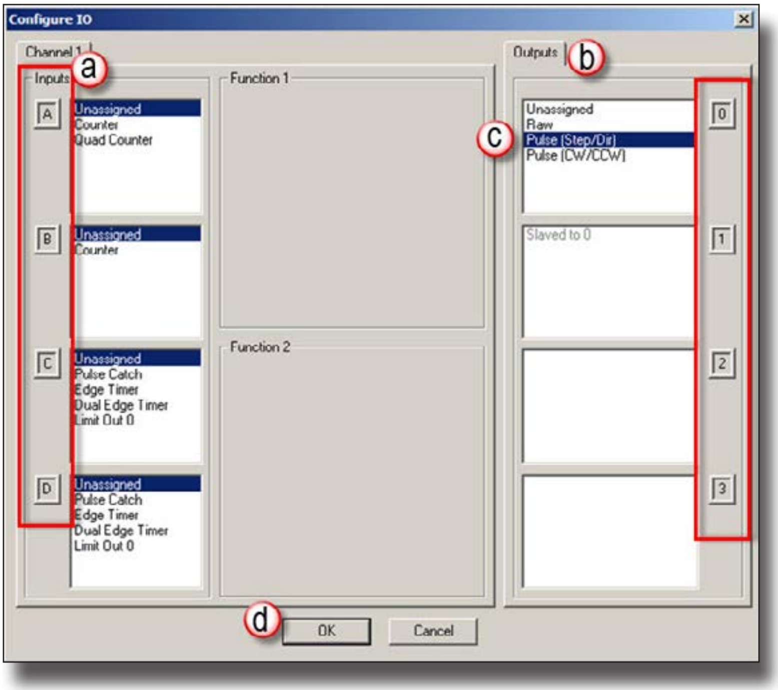
Module Status
Module Mode: Program
Scan Time: 143 us
Max Scan Time: 412 us
Booter version: 0.9.0
OS Version: 0.9.0

Module Configuration
Ch1/Fn1
Ch1/Fn2
Ch2/Fn1
Ch2/Fn2
Out 0: Unassigned
Out 1: Unassigned
Out 2
Out 3
Inputs: --------
Outputs: --------
Total Preset Tables: 0
Total Pulse Profiles: 0

Config I/O

For this example, it is only necessary to configure the outputs of the H0-CTRIO2 to generate pulses for the connected stepper amplifier. Click "Config I/O"(**d**) command box.

The Configure I/O panel is divided into two sections. One section (**a**) is for Channel 1 Inputs (A-D) and the other section (**b**) is for the Outputs (0-3). If working with an H2-CTRIO2, there would also be a tab for Channel 2. Within Channel 1, note there are Function 1 and Function 2. Each Channel on a CTRIO(2) module may have up to two Functions assigned.



On Output 1 (**c**), select the appropriate format for the step amplifier. The example uses Pulse (Step/Dir), (reference here to Output 0). This tells the CTRIO(2) that Output 0 will be used as the step signal. Automatically, Output 1 is assigned as the Direction signal.

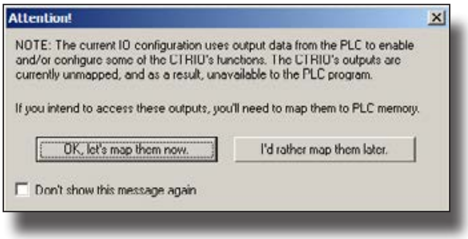
NOTE: Many other defaults change when a selection is made:

- Options for Output 1 to be Raw disappears and it is assigned as Slaved to 0 and can no longer be directly changed.
- An option is added to Channel 1 Inputs C and D for them to be Limit Out 0.

NOTE: *This automatic reconfiguring of available options is an important feature of CTRIO Workbench. The primary benefit is that it prevents the user from selecting options that will not work together. It is not possible to create an invalid configuration. However, keep this feature in mind when going about configuring a CTRIO(2). If a function cannot be found, it's likely that some dependency has not yet been enabled, or a feature that has been enabled is consuming an exclusive resource the desired feature also requires.*

Click "OK (**d**) to keep the changes and go back to the CTRIO Workbench home screen.

A dialog box appears warning that I/O has not yet been mapped to PLC memory. This warning only occurs if mapping the memory is required. Click "OK, let's map them now" to open the I/O Map dialog box (also accessible by clicking on "I/O Map" from the CTRIO Workbench home screen).



Attention!

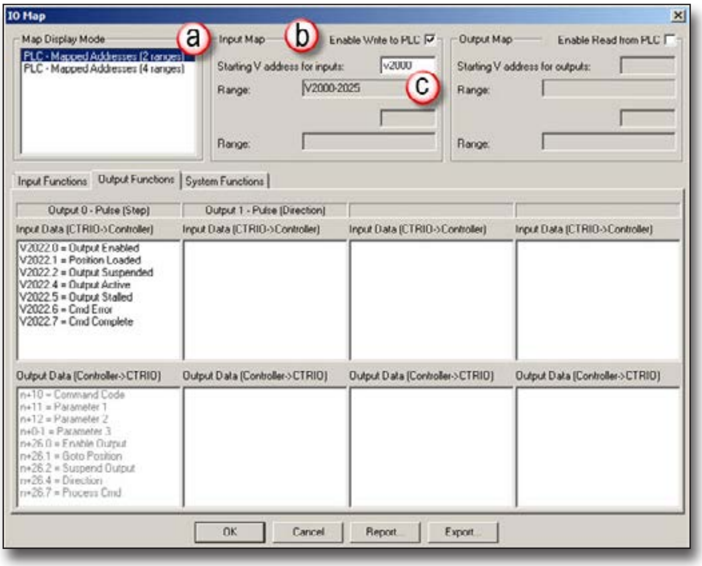
NOTE: The current IO configuration uses output data from the PLC to enable and/or configure some of the CTRIO's functions. The CTRIO's outputs are currently unmapped, and as a result, unavailable to the PLC program.

If you intend to access these outputs, you'll need to map them to PLC memory.

I/O Map

Depending on the configuration, the CTRIO(2) will have some number of variables to pass back and forth to the CPU. The data does not fit in the normal structure that most I/O modules use. Instead, the CTRIO(2) is able to write to and read from any CPU memory specified. In this panel, map the variables into CPU memory.

Be careful to map dedicated memory addresses that will only be used for this purpose.



IO Map

Map Display Mode: PLC - Mapped Addresses (2 ranges) / PLC - Mapped Addresses (4 ranges)

Input Map — Enable Write to PLC ☑

Starting V address for inputs: v2000

Range: V2000-2025

Output Map — Enable Read from PLC □

Output 0 - Pulse (Step)

Input Data (CTRIO->Controller)

V2022.0 = Output Enabled
V2022.1 = Position Loaded
V2022.2 = Output Suspended
V2022.4 = Output Active
V2022.5 = Output Stalled
V2022.6 = Cmd Error
V2022.7 = Cmd Complete

Output Data (Controller->CTRIO)

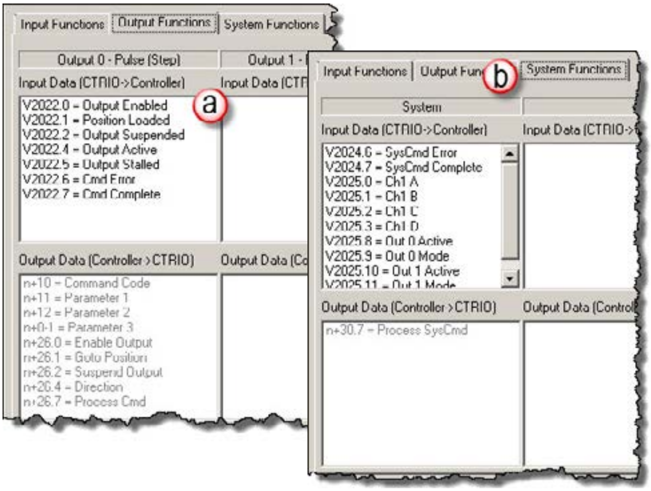
n+10 = Command Code
n+11 = Parameter 1
n+12 = Parameter 2
n+0-1 = Parameter 3
n+26.0 = Enable Output
n+26.1 = Goto Position
n+26.2 = Suspend Output
n+26.4 = Direction
n+26.7 = Process Cmd

a
b
c
Starting V address for outputs:
Range:
Range:
Range:
Input Functions
Output Functions
System Functions
Output 1 - Pulse (Direction)
Input Data (CTRIO->Controller)
Input Data (CTRIO->Controller)
Input Data (CTRIO->Controller)
Output Data (Controller->CTRIO)
Output Data (Controller->CTRIO)
Output Data (Controller->CTRIO)
OK
Cancel
Report...
Export...

In the Map Display Mode box (**a**), select "PLC - Mapped Addresses (2 ranges)" mode.

Input Map section tells the CTRIO(2) where to write input data into the PLC. Click "Enable Write to PLC" (**b**) check box and type in an address below (**c**) for "Starting V address for inputs". V2000 is used here.

Note how entering the address changes the display under “Output Functions”, “Output 0 – Pulse (Step)” Input Data (**a**). Each variable is a status bit the CTRIO(2) module will write to the CPU. Clicking on the “System Functions” (**b**) tab reveals ten more status bit variables that were also mapped to addresses in the V2000-2025 range.



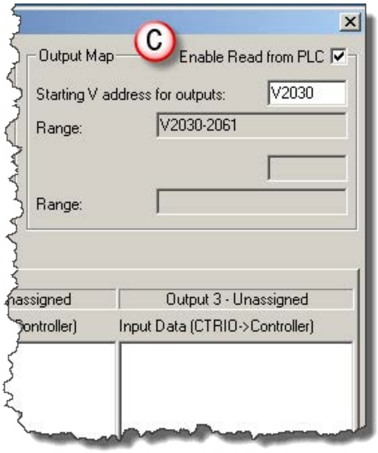
V2022.0 = Output Enabled
V2022.1 = Position Loaded
V2022.2 = Output Suspended
V2022.4 = Output Active
V2022.5 = Output Stalled
V2022.6 = Cmd Error
V2022.7 = Cmd Complete

n+10 = Command Code
n+11 = Parameter 1
n+12 = Parameter 2
n+0-1 = Parameter 3
n+26.0 = Enable Output
n+26.1 = Goto Position
n+26.2 = Suspend Output
n+26.4 = Direction
n+26.7 = Process Cmd

V2024.6 = SysCmd Error
V2024.7 = SysCmd Complete
V2025.0 = Ch1 A
V2025.1 = Ch1 B
V2025.2 = Ch1 C
V2025.3 = Ch1 D
V2025.8 = Out 0 Active
V2025.9 = Out 0 Mode
V2025.10 = Out 1 Active
V2025.11 = Out 1 Mode

n+30.7 = Process SysCmd

Output Map tells the CTRIO(2) where to read variables out of the CPU. Click “Enable Read from PLC” (**c**) and type in an address for Starting V address for outputs. The next available consecutive address, V2030, is used here. Note the Range field indicates V2030-2061 of the CPU will be used.



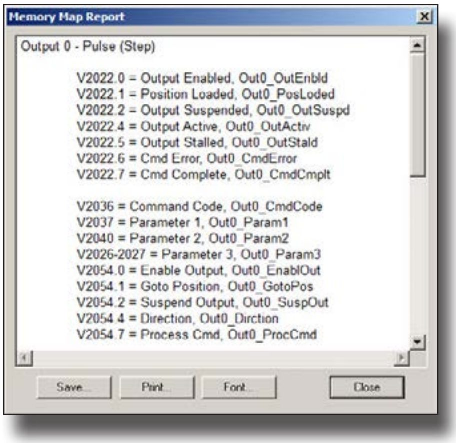
Enable Read from PLC ☑

Starting V address for outputs: V2030

Range: V2030-2061

Range:

Once the I/O Map is set up, it is best to **print the "Memory Map Report"**. Click on the "Report…" button to access the dialog below. Click on "Print". Creating this hard copy provides a critical reference to have in hand while writing the control program. The report can also be saved to disk as a ".txt" file by clicking on "Save" and giving it a unique name. When finished, close this dialog then click OK on the I/O Map dialog to return to the CTRIO Workbench home screen.



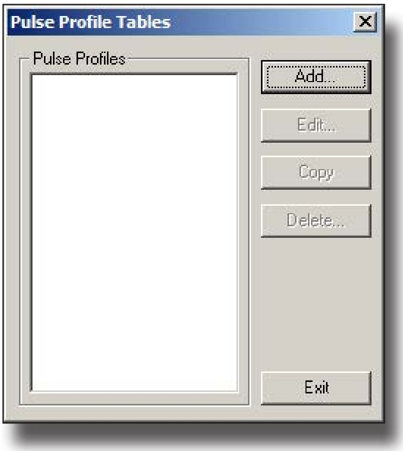
```
Output 0 - Pulse (Step)

        V2022.0 = Output Enabled, Out0_OutEnbld
        V2022.1 = Position Loaded, Out0_PosLoded
        V2022.2 = Output Suspended, Out0_OutSuspd
        V2022.4 = Output Active, Out0_OutActiv
        V2022.5 = Output Stalled, Out0_OutStald
        V2022.6 = Cmd Error, Out0_CmdError
        V2022.7 = Cmd Complete, Out0_CmdCmplt

        V2036 = Command Code, Out0_CmdCode
        V2037 = Parameter 1, Out0_Param1
        V2040 = Parameter 2, Out0_Param2
        V2026-2027 = Parameter 3, Out0_Param3
        V2054.0 = Enable Output, Out0_EnablOut
        V2054.1 = Goto Position, Out0_GotoPos
        V2054.2 = Suspend Output, Out0_SuspOut
        V2054.4 = Direction, Out0_Dirction
        V2054.7 = Process Cmd, Out0_ProcCmd
```

Pulse Profile Tables

For the CTRIO(2) module to generate a Trapezoidal Profile, the characteristics of the profile must be provided. As is the case with most controllers[1], the characteristics of the profile are defined as part of the CTRIO configuration. Multiple profiles may be added to the configuration to address different motion needs.

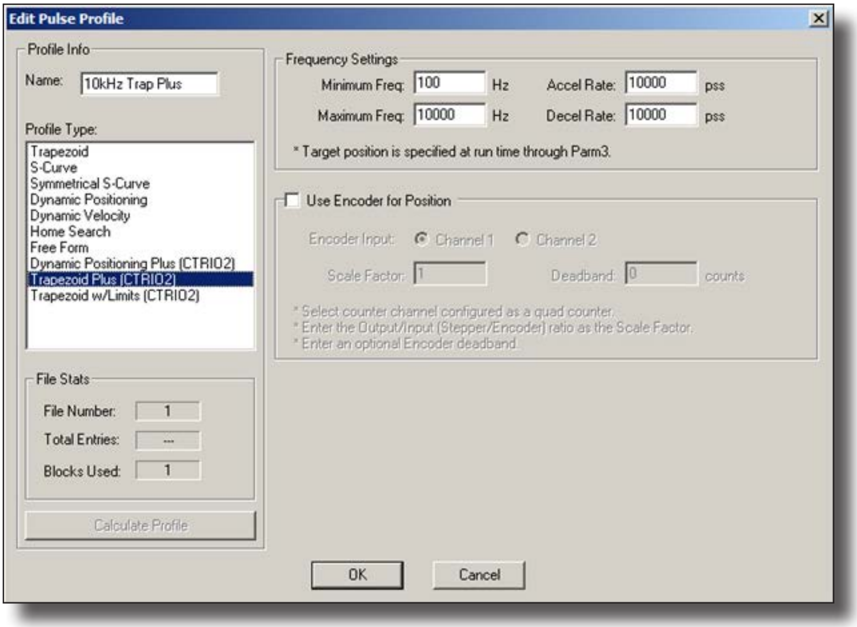
Using addresses covered in I/O Map, the controller will tell the CTRIO(2) to load one of its stored profiles. The CTRIO(2) will respond when complete. Then the CPU provides the other parameters of the move and tells the CTRIO(2) to start. During the move, the CTRIO(2) notifies the CPU of its status.

From the CTRIO Workbench home screen, press "Pulse Profiles" to access the Pulse Profile Tables panel. Click "Add…" to get to the Edit Pulse Profile panel.



[1] Using a CTRIO2 with a Do-more CPU allows use of a ladder instruction, CTAXCFG, where the characteristics of the profile are defined.

Edit Pulse Profiles offers a list of profile types on the left. Note the last three profiles are only valid for use with a CTRIO2. Select Trapezoid Plus (CTRIO2) and optionally give the profile a name; *10kHz Trap Plus* is used here. In the "Frequency Settings" section, enter appropriate frequency range for the application. The default values were left in place for the example. Press "OK" to return to **Pulse Profile Tables** box.



Edit Pulse Profile

Profile Info

Name: 10kHz Trap Plus

Profile Type:

- Trapezoid
- S-Curve
- Symmetrical S-Curve
- Dynamic Positioning
- Dynamic Velocity
- Home Search
- Free Form
- Dynamic Positioning Plus (CTRIO2)
- Trapezoid Plus (CTRIO2)
- Trapezoid w/Limits (CTRIO2)

File Stats

File Number: 1

Total Entries: ---

Blocks Used: 1

Frequency Settings

Minimum Freq: 100 Hz Accel Rate: 10000 pss

Maximum Freq: 10000 Hz Decel Rate: 10000 pss

* Target position is specified at run time through Parm3.

☐ Use Encoder for Position

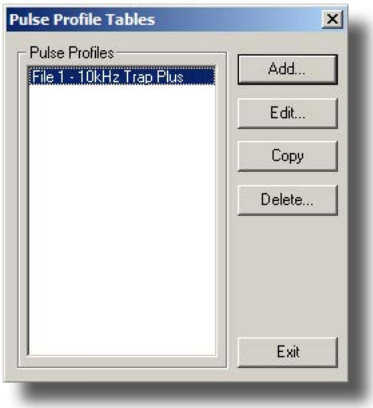
Encoder Input: ● Channel 1 ○ Channel 2

Scale Factor: 1 Deadband: 0 counts

* Select counter channel configured as a quad counter.
* Enter the Output/Input (Stepper/Encoder) ratio as the Scale Factor.
* Enter an optional Encoder deadband.

The **Pulse Profiles Tables** newly defined profile appears as *File 1 – 10kHz Trap Plus*. The assignment as File 1 is significant, as profiles will be specified in ladder by file number alone. Also, creating a PLS or Preset Table definition also results in creation of a File. Files are not just for defining Pulse Profiles.

Click "Exit" to return to the CTRIO Workbench home screen.



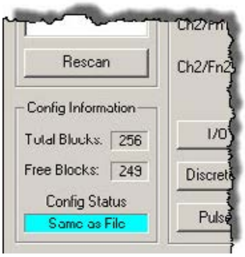
Pulse Profile Tables

Pulse Profiles

File 1 - 10kHz Trap Plus

Download the Configuration

Now that the configuration has been defined, click “Write File” to save a copy of the configuration to disk. It will save as a “.CWB” file that should be kept with the other files and documents for the project. Note that the Config Status changes to Same as File, indicating that the disk file and the CTRIO workbench file are identical.

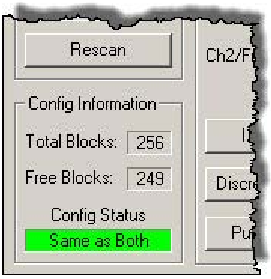


Config Information

Total Blocks: 256

Free Blocks: 249

Config Status: Same as File



Config Information

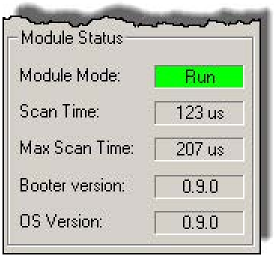
Total Blocks: 256

Free Blocks: 249

Config Status: Same as Both

Once saved, click “Write Module” to write the file to the CTRIO(2) module. Note that the Config Status changes to Same as Both, indicating that the file on disk, the file in CTRIO workbench and the file in the CTRIO(2) module are identical.

At this point, the CTRIO(2) module is still in Program mode. Click “Goto RUN!” to initiate program execution. The module will begin reading addresses from the CPU (refer to I/O map) to determine what action, if any, to take. Note that Module Mode changes to Run.



Module Status

Module Mode: Run

Scan Time: 123 us

Max Scan Time: 207 us

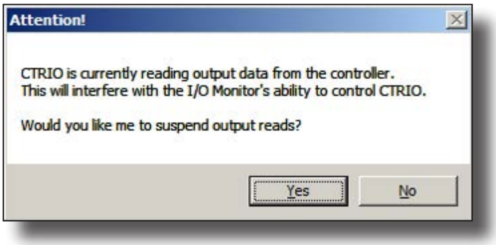
Booter version: 0.9.0

OS Version: 0.9.0

NOTE: *It is only necessary to manually switch the CTRIO(2) into Run or Program during testing. During normal operation, the CTRIO(2) follows changes in the mode of the CPU. If the CPU goes to Stop, the CTRIO(2) goes to Program. If the CPU goes to Run, the CTRIO(2) goes to Run.*

Monitor I/O

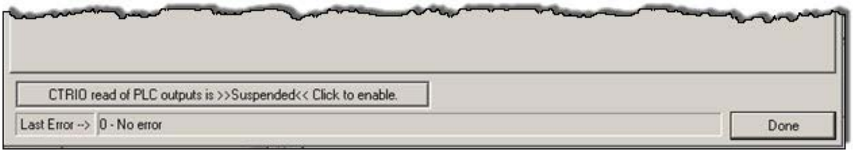
Before moving over to using the CPU to work with the CTRIO(2) module, there is a very convenient way to verify everything is working as expected from within CTRIO Workbench. Click “Monitor I/O” on Workbench Panel. The following prompt may appear, asking whether the system should suspend output reads. Output reads are what the CTRIO(2) does to know what the CPU is telling it to do. Suspending them just means that the CTRIO(2) should temporarily stop taking commands from the CPU. The CTRIO(2) is controlled from the Monitor I/O panel, answer Yes.



CTRIO is currently reading output data from the controller.
This will interfere with the I/O Monitor's ability to control CTRIO.

Would you like me to suspend output reads?

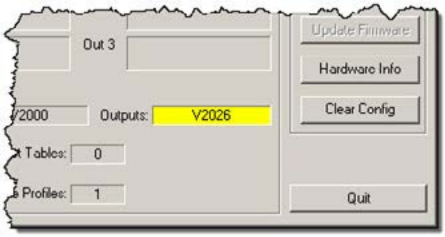
The CTRIO(2) stops reading addresses from the CPU and the Monitor I/O panel opens.



CTRIO read of PLC outputs is >>Suspended<< Click to enable.

Last Error --> 0 - No error

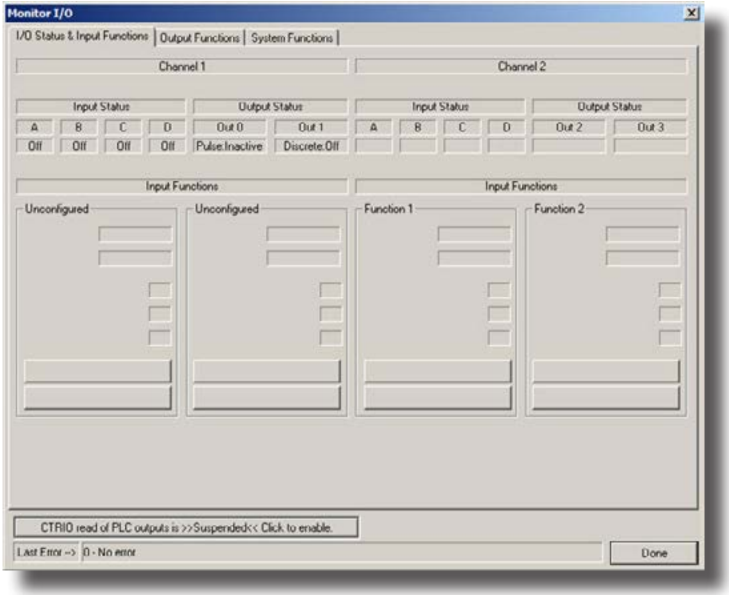
Notice the long button at the bottom of Monitor I/O panel, "CTRIO read of PLC outputs is >>Suspended<< Click to enable". When suspended, the Outputs address will be highlighted in yellow (This is true if you exit Monitor I/O without re-enabling reads).



Outputs: V2026

Tables: 0

Profiles: 1



A	B	C	D	Out 0	Out 1
Off	Off	Off	Off	Pulse:Inactive	Discrete:Off

CTRIO read of PLC outputs is >>Suspended<< Click to enable.

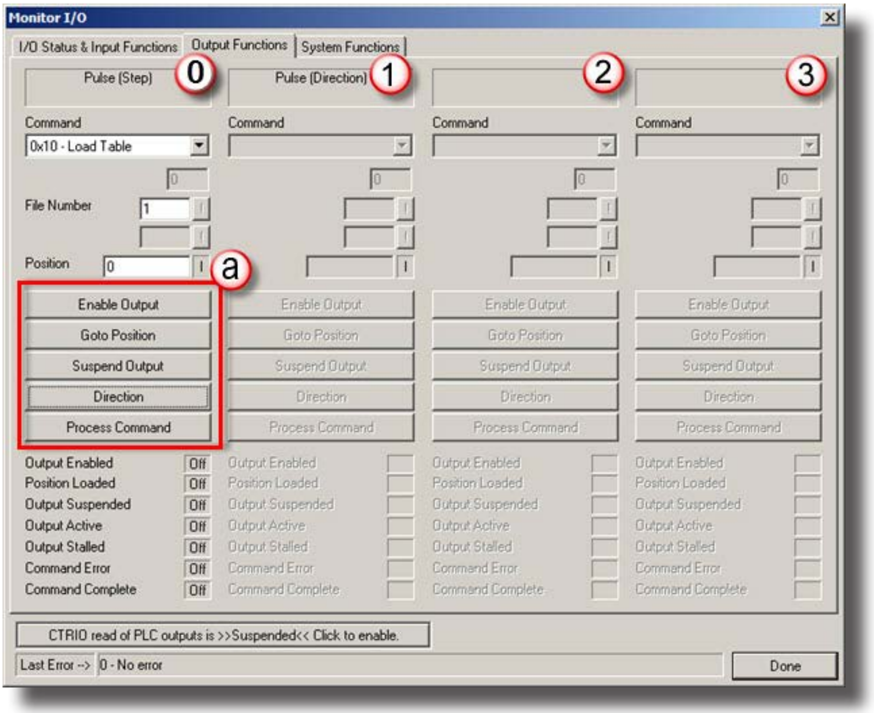
Last Error --> 0 - No error

Another indication that the output reads have been suspended can be seen on the Monitor I/O panel.

At first glance, not much is going on here. Most visible buttons are grayed out. This is because the default display tab is **I/O Status and Input Functions**. There are no Input Functions defined, for this example. The only relevant section of this display, for the purpose of this example, is **Output Status - Out 0 and Out 1**. The two fields show the current status of the **Pulse Output** as Inactive and the **Direction** (**Discrete Output**) as Off.

Click **Output Functions** tab.

The display for Output Functions is divided into four columns representing the 4 discrete output points (0, 1, 2, 3) of the hardware. The first two points (0,1)are assigned as Pulse (Step/ Dir) as indicated at the top of the first two columns (Outputs 0 & 1). The only available buttons appear under Output 0 (a) since Output 1 is slaved to Output 0. The fields and buttons under Output 0 will be used to execute a Trapezoid Plus move.



Command: 0x10 - Load Table

File Number: 1

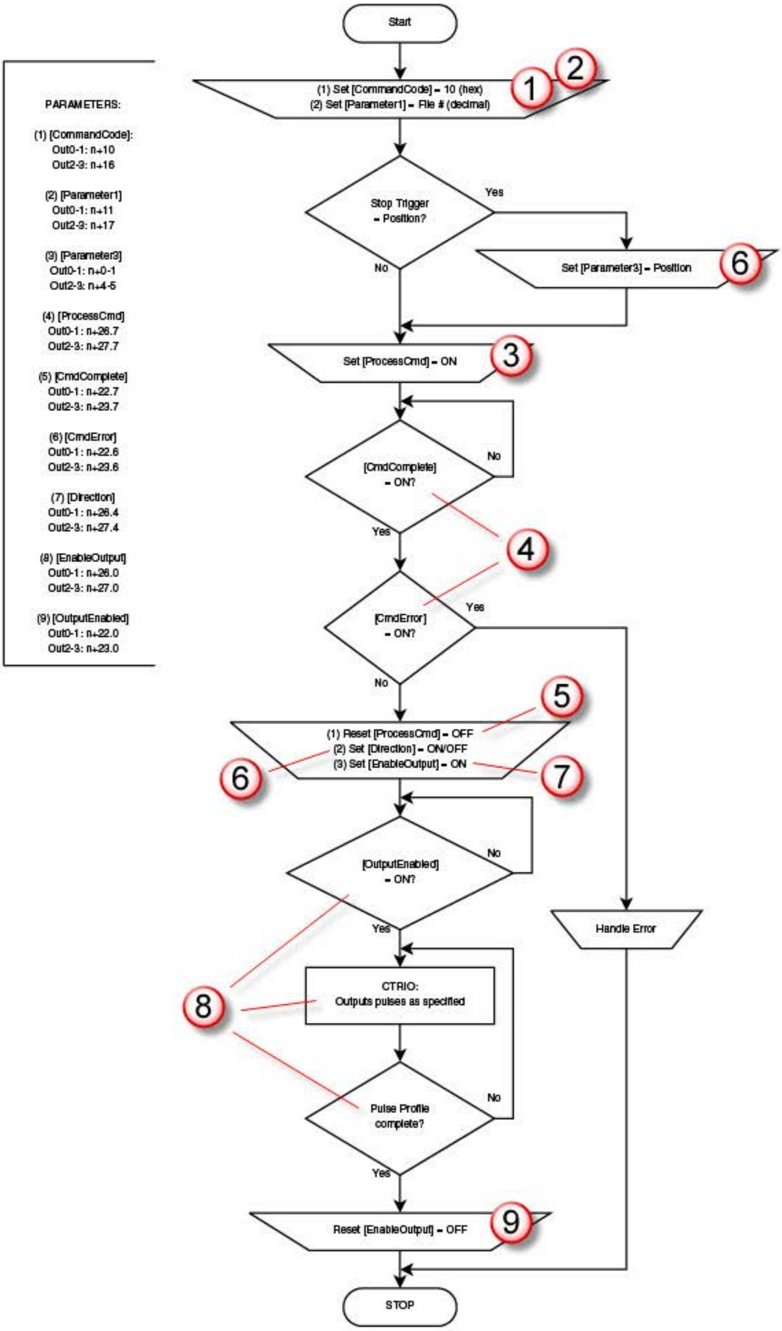
Position: 0

Output Enabled	Off
Position Loaded	Off
Output Suspended	Off
Output Active	Off
Output Stalled	Off
Command Error	Off
Command Complete	Off

CTRIO read of PLC outputs is >>Suspended<< Click to enable.

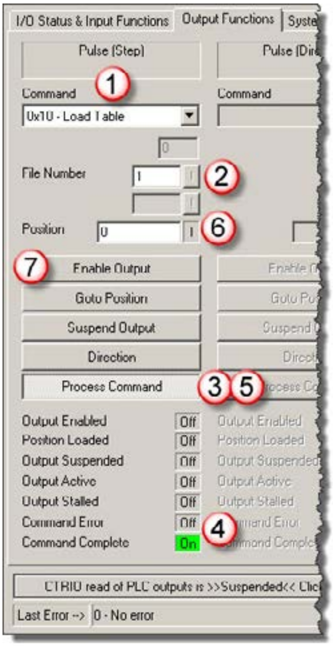
Last Error --> 0 - No error

Run Trapezoid w/Limits or Trapezoid PLUS (DL-PLC)



Flow Chart Example: Configure and Test a Pulse Output with a Trapezoidal Profile

The following steps are noted on the graphic below as well as the flow chart on the previous page.



Command: 0x10 - Load Table

File Number: 1

Position: 0

Output Enabled: Off
Position Loaded: Off
Output Suspended: Off
Output Active: Off
Output Stalled: Off
Command Error: Off
Command Complete: On

CTRIO read of PLC outputs is >>Suspended<< Clic

Last Error --> 0 - No error

1: For **Command**, select **0x10 – Load Table.** Command "0x10" is used for multiple functions: Load Preset Table, Load PLS or Load a Pulse Profile. We are loading a pulse profile in this example.

2: For **File Number**, select 1. File 1 is the Trapezoid Plus Pulse Profile configuration that was entered earlier. This is Parameter 1 in the flow chart and in the I/O mapping.

3: Press Process Command to tell the CTRIO(2) to act on the variables selected in steps 1 - 2.

4: The CTRIO(2) acts on the **Process Command** and reports its result as 'success' by setting the **Command Complete** status bit. This means the CTRIO(2) has successfully loaded File 1. If the CTRIO(2) had been unable to load the file for any reason, the **Command Error** status bit would have come on as well.

NOTE: *Loading a table is not instantaneous within the CTRIO(2). What the CTRIO(2) just did here took enough time that several CPU scans would lapse before Command Complete status bit turned On. Ladder code needs to be written to account for the asynchronous nature of interactions with the CTRIO(2). Stage is well suited for writing sequential asynchronous code.*

5: Press **Process Command** again to release (reset) it. The CTRIO(2) clears the **Command Complete** status bit.

6: For **Position**, type in the distance (total number of pulses) desired for the move. This is Parameter 3 in the flow chart and in the I/O mapping. The sign of this parameter sets the direction of the move. A positive value will keep Output 1 (Dir) off (With this profile, the **Direction** button (bit) is not used. Some other profiles use the **Direction** bit instead of the sign.).

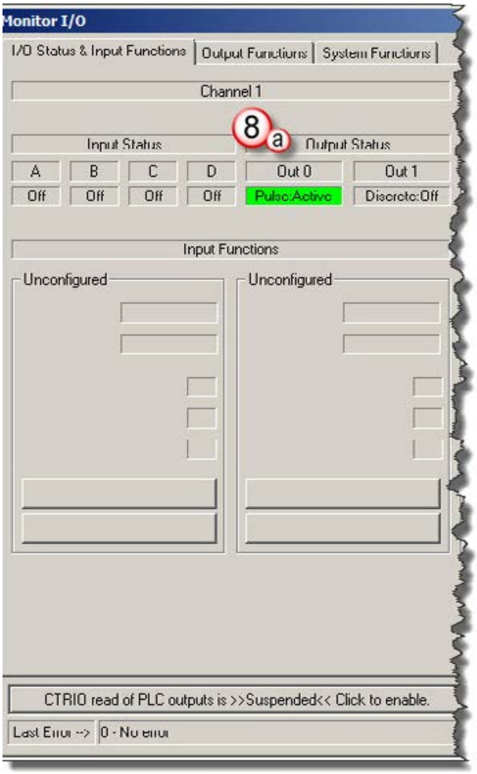
7: Press Enable Output to start the move.

8: The CTRIO(2):

a. Turns On the Output Active and Output Enabled status bits

b. Generates the appropriate pulse train on Output 0 and Output 1

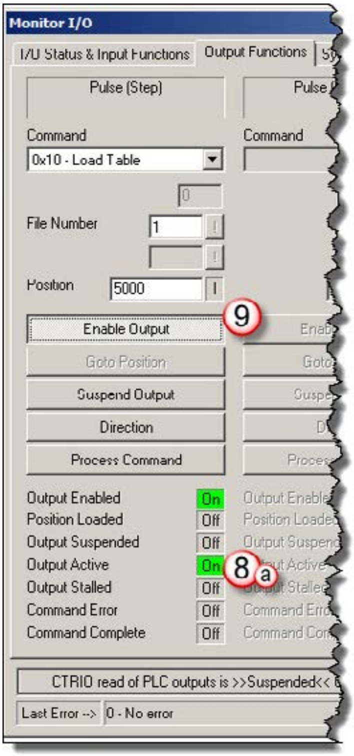
c. Turns Off the Output Active bit when the move completes



A	B	C	D	Out 0	Out 1
Off	Off	Off	Off	Pulse:Active	Discrete:Off

CTRIO read of PLC outputs is >>Suspended<< Click to enable.

Last Error --> 0 - No error



Command: 0x10 - Load Table

File Number: 1

Position: 5000

Output Enabled	On
Position Loaded	Off
Output Suspended	Off
Output Active	On
Output Stalled	Off
Command Error	Off
Command Complete	Off

CTRIO read of PLC outputs is >>Suspended<<

Last Error --> 0 - No error

9: Press the Enable Output button again to release (reset) it. The CTRIO(2) turns Off the Output Enabled status bit.

10: To initiate another move, repeat steps 6 – 9.

Ladder Example

A detailed ladder example is provided in *Chapter 9: Output Functions* later in this manual.