

Indexer 2 User Guide



P/N 16-01192
Revision 01
February 2019

TABLE OF CONTENTS

About This Manual.....	3
1: Introduction.....	7
1.1: Overview	8
2: Description and Operation.....	9
2.1: Operational Overview.....	10
2.2: Sequence Selection	11
2.3: Go Command.....	14
2.4: Priority Input Command.....	17
2.5: Indexer 2 Program Registers	17
2.6: Diagram of Typical Connections	19
3: Programming	20
3.1: Overview of Procedures	21
3.2: Basic Amplifier Setup	22
3.3: Indexer 2 Program Interface Tour	23
3.4: Menu Bar	24
3.5: Toolbar	27
3.6: Creating and Modifying Sequences	29
3.7: Adding Steps to a Sequence.....	31
3.8: Multi-axis Support.....	32
3.9: CVM Program Quick Copy Setup	33
3.10: Using Debug Mode.....	34
4: Functions.....	36
4.1: Wait Move Done	37
4.2: Wait for Delay Time.....	38
4.3: Wait for Event.....	39
4.4: Wait for Input Mask	41
4.5: Wait for Position	42
4.6: Wait for Input.....	43
4.7: Wait for Parameter	44
4.8: Set Current Limits.....	46
4.9: Set Tracking Windows.....	47
4.10: Set Gains.....	48
4.11: Velocity Move Position Mode	49
4.12: Move.....	50
4.13: Home.....	52
4.14: Current Move.....	55
4.15: Velocity Move Velocity Mode	56
4.16: Analog Velocity Mode	57
4.17: Analog Position Mode	58
4.18: Disable Drive	60
4.19: Camming Internal Master.....	61
4.20: Camming	62
4.21: Digital Position Mode.....	64
4.22: Velocity Loop Single Pole Output Filter.....	66
4.23: Set Output	67
4.24: Position Triggered Output	68
4.25: Math	70
4.26: If Register Set.....	71
4.27: Get Set Parameter	72
4.28: Save Restore Registers	74
4.29: Conditional Jump.....	75
4.30: Extended Math	76
4.31: Logic.....	77
4.32: CAN PDO Mapping	78

ABOUT THIS MANUAL

Overview and Scope

This manual describes the operation of the Copley Virtual Machine Indexer 2 Program developed by Copley Controls. The manual was written for the reader who has a basic knowledge of motion control theory and operation, Copley Controls drives, and Copley Controls CME 2 software.

Related Documentation

- [ASCII Interface Programmer's Guide](#)
- [Camming User Guide](#)
- [Parameter Dictionary](#)

Copley Controls software and related information can be found at:
<http://www.copleycontrols.com/Motion/Products/Software/index.html>

Product Warnings



DANGER: Hazardous voltages.

Exercise caution when installing and adjusting Copley Controls drives.

Failure to heed this warning can cause equipment damage, injury, or death.

DANGER



Risk of electric shock.

High-voltage circuits are connected to DC power on certain Copley Controls drives.

Failure to heed this warning can cause equipment damage, injury, or death.

DANGER

Using CME 2 can affect or suspend Indexer 2 Program operations.



When operating the drive under control of the Indexer 2 Program, use of CME 2 to change drive parameters can affect Indexer 2 Program operations in progress.

Using CME 2 to initiate motion can cause Indexer 2 Program operations to suspend. The operations may restart unexpectedly when the CME 2 move is stopped.

DANGER

Failure to heed this warning can cause equipment damage, injury, or death.



Moves in progress will run to completion after Indexer 2 Program is stopped.

Stopping the Indexer 2 Program does not stop any move in progress.

Failure to heed this warning can cause equipment damage, injury, or death.

DANGER

Drive faults and sequence errors are not equivalent.**WARNING**

A sequence error does not necessarily result in a drive fault condition. Each Indexer step function can result in a particular set of sequence errors, as described in [Functions](#).

A drive fault condition does not necessarily result in a sequence error. Thus it is possible that drive faults can occur without triggering any error handling response in the Indexer 2 Program.

Be aware of the differences between drive faults and sequence errors and program accordingly.

Revision History

Revision	Date	Comments
00	Sept 2014	Updated axis selection, added save/restore register
01	February 2019	Added CAN PDO Mapping section

CHAPTER

1: INTRODUCTION

1.1: Overview

The Indexer 2 Program is a powerful, easy to use indexer that runs on the Copley Virtual Machine (CVM), an embedded virtual machine available on most Copley Controls drives. Up to 32 sequences can be created with one or more steps which contain moves dwell times, I/O control, parameter changes, and conditional logic.

Typical applications involve a PLC or switch to activate the drive's digital inputs to select and execute sequences. The drive's digital outputs can be used to control machine processes or provide status feedback to the PLC.

The Indexer 2 Program also has 32 registers that can be controlled by issuing ASCII commands over the drive's serial port. These registers can be used to select and execute sequences and pass numeric parameters to the Indexer 2 Program.

Features of the Indexer 2 include:

- Simple, intuitive programming tools
- 32 programmable multi-step sequences
- Priority sequence, selected and executed with a single digital input
- 32 registers accessible via an RS-232 serial port and other interfaces
- Sequence selection and execution by digital input or register
- Program can start automatically on power up
- Programmable response to sequence errors

CHAPTER

2: DESCRIPTION AND OPERATION

2.1: Operational Overview

The Indexer 2 can be programmed to start using an input, a register, or an immediate Go on a power up or reset of the drive. Upon startup, the Indexer 2 Program sets the program registers to zero, then enters the main loop. The main loop continually polls the Priority input, then the Go command. If either is active, the selected sequence will be executed. During sequence execution, the Priority input, Go command, and sequence selection inputs are ignored. After successful completion of a sequence, the Indexer 2 Program returns to the main loop.

If an error occurs during a sequence, one of two responses can be programmed:

- The current sequence is aborted and the Indexer 2 Program reverts back to its main loop.
- The Indexer 2 Program starts immediate execution of another programmed sequence.

To Configure the Sequence Selection, Go Command, and Priority Input Command, select the *Setup* tab:

The screenshot displays the configuration interface for the Indexer 2, divided into three main sections:

- Sequence Selection:**
 - Use register to select sequence:
 - Use digital input:
 - Register number:
 - Number of BCD inputs:
 - Starting BCD input (LSB):
- Priority Input Command:**
 - Use Priority input:
 - Input number:
 - Trigger:
 - Edge
 - Level
 - Active:
 - Rising
 - Falling
- Go Command:**
 - Use register to initiate Go:
 - Use digital input:
 - Register number:
 - Input number:
 - Trigger:
 - Edge
 - Level
 - Active:
 - Rising
 - Falling
 - Use immediate single Go on startup or reset:

2.2: Sequence Selection

The Indexer 2 Program can store up to 32 sequences (0 to 31). Upon receiving a Go command, the program executes the sequence that has been selected by a register or by digital inputs.

Use a Register to Select a Sequence

When the Indexer 2 Program receives a Go command and **Use register to select sequence** is the sequence selection method, the program reads the first 5 bits (0-4) of the specified register to select a sequence.

To configure register selection, select **Use register to select sequence**, then enter the **Register number** (range R0-R31) that will contain the sequence number.

The screenshot shows a dialog box titled "Sequence Selection". It contains three radio buttons: "Use register to select sequence:" (selected), "Use digital input:", and "Use digital input:". Below the radio buttons are three input fields: "Register number:" with a text box containing "R0", "Number of BCD inputs:" with a dropdown menu showing "5", and "Starting BCD input (LSB):" with a dropdown menu showing "IN2".

Use Digital Inputs to Select a Sequence

Conflicting use of inputs is possible.



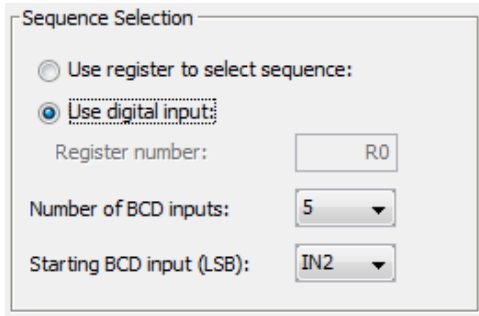
DANGER

It is possible to assign functions to the inputs through the CME 2 Input/Output configuration screen that would conflict with the use of the BCD sequence selection, Priority and Go inputs assigned by Indexer 2 Program. Both functions of the input would be in effect simultaneously and might cause unexpected results. Exercise care to avoid such conflicts when assigning input functions.

Failure to heed this warning can cause equipment damage, injury, or death.

When the Indexer 2 Program receives a Go command and **Use digital input** is the programmed sequence selection method, the drive's digital inputs are used to select the sequence to run. Up to five inputs can be used to represent a binary coded number with a decimal value between 0 and 31.

To configure Indexer 2 Program to use digital inputs for sequence selection, select **Use digital input**, then enter the appropriate values for the BCD Input fields:



Field	Description
Number of BCD inputs	Selects the number of inputs to be used for selecting sequences. The number of inputs chosen determines the number of sequences that can be selected using the inputs. For instance, 1 input can be used to select sequence 0 or sequence 1. Five inputs can be used to select any of the 32 sequences.
Starting BCD input (LSB)	Selects which input will be used for the least significant bit of the sequence address. The remaining address lines will be on consecutively higher inputs.

NOTE: The BCD inputs are only read when the Go command is received. The inputs can then be used for other purposes until the next Go command is expected. Exercise caution to avoid conflicting uses of an input.

The table below shows the number of input-selectable sequences per number of inputs used.

# of BCD Inputs	# of Sequences	Sequence # Range
5	32	0-31
4	16	0-15
3	8	0-7
2	4	0-3
1	2	0-1
0	1	0

For example, suppose the Indexer 2 Program has been configured to read three sequence selection inputs starting at IN2. This table shows the input states that would select each:

	Inputs		
Input Number	IN4	IN3	IN2
Decimal Equivalent	4	2	1
Sequence			

7	1	1	1
6	1	1	0
5	1	0	1
4	1	0	0
3	0	1	1
2	0	1	0
1	0	0	1
0	0	0	0

2.3: Go Command

A Go command is what triggers the Indexer 2 program to run the selected sequence. Programmable Go command trigger options include registers, digital inputs, and immediate single Go on startup or reset.

Use a Register to Initiate Go

When programmed to **Use Register to Initiate Go**, the Indexer 2 Program monitors bit 15 of the programmed Go register. If Bit 15 is set to 1, the program executes the selected sequence.

Select **Use register to initiate Go**, then enter the register number (range R0-R31). If **Use register to select sequence** was also selected, the sequence selection register number will be changed to match the Go command register number.

The screenshot shows a dialog box titled "Go Command". It has two radio buttons: "Use register to initiate Go:" (selected) and "Use digital input:". Below the "Use register to initiate Go:" option, there is a "Register number:" field with "R0" entered and an "Input number:" dropdown menu with "IN11" selected. Under the "Trigger:" section, there are two radio buttons: "Edge" (selected) and "Level". Under the "Active:" section, there are two radio buttons: "Rising" (selected) and "Falling". At the bottom, there is a checkbox labeled "Use immediate single Go on startup or reset:" which is currently unchecked.

To initiate a register-driven Go when the sequence will be selected by digital inputs, set Bit 15 of the Go register by writing to the Go register a hex value 0x8000, or the decimal equivalent, 32768 (all other bits will be ignored).

When Go command and sequence selection are both programmed to use a register, the same register is used for both purposes. In this case, use the same write operation to write the sequence to the register and set Bit 15 at the same time. For example, to execute Sequence 12, write to the register the hex value 0x800c or the decimal equivalent, 32780.

Note that the Indexer 2 Program clears Bit 15 before executing the sequence so that a register or digital input cannot trigger another Go until Bit 15 is reset. The state of Bit 15 does not affect the operation of Immediate Single Go.

Use a Digital Input to Initiate Go

When programmed to **Use Input to Initiate Go**, the Indexer 2 Program monitors the state of the programmed Go input. The Go can be programmed to happen whenever the input is at the specified level (low or high), or only on the rising or falling edge of an input transition.

Select **Use digital input**, then enter the input number and configure the input.



Field	Description
Input number	Selects which input will be used to execute the sequence.
Trigger	<ul style="list-style-type: none"> Edge: Input must transition for the sequence to start. Level: Input must be at the correct level for the sequence to start. Note that a level-triggered Go command will cause the sequence to repeat until the level becomes inactive.
Active	<p>With Edge Trigger:</p> <ul style="list-style-type: none"> Rising: Sequence will start on a low to high transition of the Go input. Falling: Sequence will start on a high to low transition of the Go input. <p>With Level Trigger:</p> <ul style="list-style-type: none"> Hi: Sequence will start when the Go input is high. Lo: Sequence will start when the Go input is low.

Use Immediate Single Go on Startup or Reset

This option programs the Indexer to trigger a Go command when the drive is powered up or reset or the Run button is pushed.

Important Note

When using Immediate Single GO, the Indexer 2 program will start executing as soon as the drive completes it's reset or power up. The system may not be ready for motion at this point. **It is recommended to put the Wait Delay step before any motion step with a delay time that allows the system to initialize and be ready for motion. This is especially true, but not limited to, systems where algorithmic phase initialization is being used.**

If the Indexer is programmed to **Use register to select sequence**, then the sequence executed will be number 0, because all registers are cleared when the program starts.

If sequence selection is programmed to **Use digital input**, then the states of the programmed selection inputs determine which sequence is executed.

After immediate single Go, further Go commands are executed as programmed.

Go Command

Use register to initiate Go:
 Use digital input:

Register number:

Input number:

Trigger:
 Edge Level

Active:
 Rising Falling

Use immediate single Go on startup or reset:

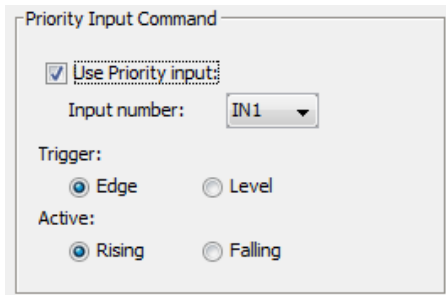
2.4: Priority Input Command

In addition to the 32 normal sequences, the Indexer 2 Program supports a Priority sequence. Unlike other sequences, the Priority sequence:

- is selected and initiated with a single digital input
- receives process priority over other sequence selection methods

Note that a Priority sequence does not stop or override any sequence in progress.

Click **Use Priority input**.



Field	Description
Input number	Selects which input will be used to execute the Priority sequence.
Trigger	<ul style="list-style-type: none"> • Edge: Input must transition for the Priority sequence to start. • Level: Input must be at the correct level for the Priority sequence to start. Note that a level-triggered Priority Input command will cause the Priority sequence to repeat until the level becomes inactive.
Active	<p>With Edge Trigger:</p> <ul style="list-style-type: none"> • Rising: Sequence will start on a low to high transition of the Priority input. • Falling: Sequence will start on a high to low transition of the Priority input. <p>With Level Trigger:</p> <ul style="list-style-type: none"> • Hi: Sequence will start when the Priority input is high. • Lo: Sequence will start when the Priority input is low.

2.5: Indexer 2 Program Registers

The Indexer 2 Program has 32 registers that can be used to select sequences, initiate Go commands, and set parameters in individual steps. Each register is 32 bits long.

NOTE: The Indexer 2 program sets all registers to zero at startup.

In many Indexer 2 Program function parameter fields, references to the program's 32 registers can be used in place of hard-coded numbers. For instance, move parameters can be entered as numbers or as register references. A register reference takes the form Rn , where n is a register number (0-31).

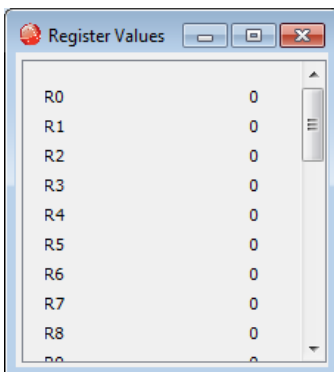
Distance:	<input type="text" value="58000"/>	counts	Distance:	<input type="text" value="R0"/>	counts
Velocity:	<input type="text" value="350"/>	rpm	Velocity:	<input type="text" value="R1"/>	0.1 counts/s
Accel:	<input type="text" value="500"/>	rps ²	Accel:	<input type="text" value="R2"/>	10 counts/s ²
Decel:	<input type="text" value="500"/>	rps ²	Decel:	<input type="text" value="R3"/>	10 counts/s ²

Some parameters have different units when using registers. Be sure to express all values in the units called for in the field label. When the step is executed, the Indexer 2 program performs the register value substitutions. Invalid data in a register will result in a sequence error.

Control applications (HMI, PLC, or PC-based programs) can use any of the supported protocols to read and write the Indexer 2 Program registers. Supported protocols include the Copley ASCII Interface, CANopen, and EtherCAT.

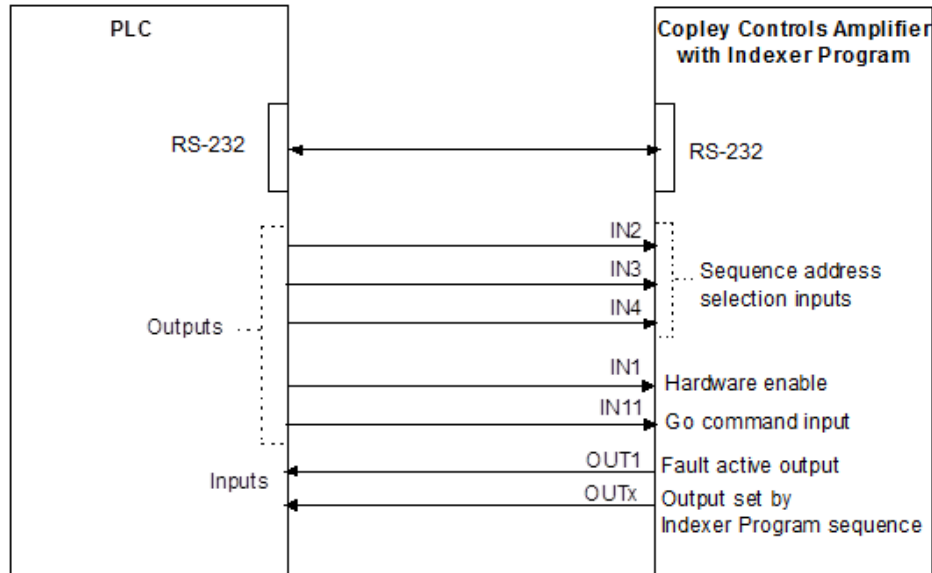
For experimentation with setting/reading register values, the ASCII Command Line can be used by selecting **Tools**→**ASCII Command Line** from the CME 2 menu bar. See [ASCII Interface Programmer's Guide](#).

Register values can also be viewed while the Indexer 2 Program is running by selecting **View**→**Register Values** from the menu bar of the Indexer 2 Program screen.



2.6: Diagram of Typical Connections

In a typical Indexer 2 Program application, the connections between a Copley Controls drive and a PLC resemble those shown below.



Here, inputs IN2-IN4 are used for sequence selection. Input IN11 is programmed to initiate the Indexer 2 Program **Go** command. The drive's hardware enable input, IN1, is also controlled by the PLC. OUT1 is configured to go active when a drive fault occurs, and any of the drive's digital outputs can be set by a step in an Indexer 2 Program sequence. The serial connection can be used to set register values.

CHAPTER

3: PROGRAMMING

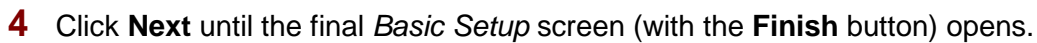
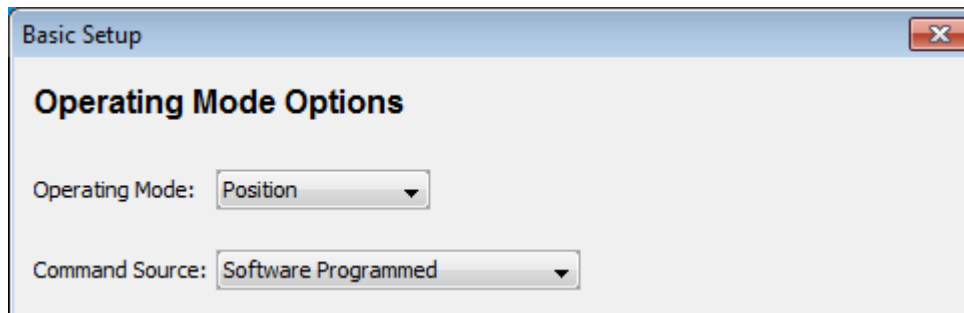
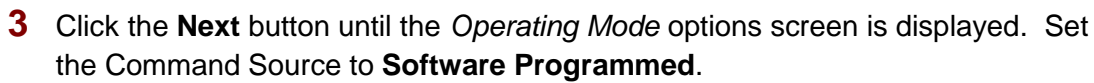
3.1: Overview of Procedures

The following is a typical procedure for creating an Indexer 2 Program.

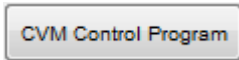
- 1 Set up, tune and test the drive.** The system should be able to safely perform the types of moves that the Indexer 2 program will command.
- 2 Save drive settings to flash.** The Indexer 2 Program uses the drive's flash memory for default values.
- 3 Optionally set up the drive to run in software programmed mode.** See [Basic Amplifier Setup](#).
- 4 Configure sequence selection, priority, and Go command.**
- 5 Program the sequences.**
- 6 Save the program to the drive's flash memory.**
- 7 Save the program to disk.**
- 8 Make sure the drive is hardware enabled and free of active faults.**
- 9 Run the program.**
- 10 Select the appropriate sequence.**
- 11 Activate the Go command.**
- 12 Debug the Indexer 2 Program as required.**
- 13 Stop the Indexer 2 Program.**
- 14 Configure the Indexer 2 Program for auto start if required.**
- 15 Save the final version of the program to the drive's flash memory.**
- 16 Save the final version of the program to disk.**

3.2: Basic Amplifier Setup

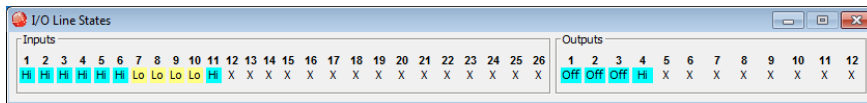
Any operating mode/command source combination can be used with an Indexer 2 program. However, in applications where the drive is to be controlled only by the Indexer 2 program as a standalone device, it is recommended to set Software Programmed as the Command Source.



3.3: Indexer 2 Program Interface Tour



To open the Indexer 2 screen, click **CVM Control Program**. The IO Line States screen will also be displayed so that the inputs can be monitored while running and debugging programs.



Menu Bar →

Tool Bar →

Tabs →

Sequence List →

Memory Usage

Program Status

Sequence Number selected by BCD inputs

Tab	Use
Program	Create and modify sequences
Setup	Set up sequence selection, and Go command
Debug	Allows the program to be debugged by single-stepping and monitoring registers, inputs, etc.

3.4: Menu Bar

File Menu	Description
Save Control Program	Saves the current CVM program in the PC's memory to a disk file with a .ccp filename extension.
Open Control Program	Opens a saved CVM program. The program contained in this file will replace the current program in the PC's memory.*
New Control Program	Starts a new CVM control program. Note that using this command deletes the current CVM program from the PC's memory.*
Delete Control Program in Amplifier Flash	Removes the CVM control program that is currently stored in the drive's flash memory.
*NOTE: When a new CVM control program is started or an existing one is opened from a file (.ccp), Enable Control Program on Startup is automatically selected.	









View Menu	Description
Assembly Code	Displays the assembly language code of the CVM program in the PC's memory.
CVM Directory	Opens the <i>CVM Directory</i> screen, which displays all files stored in the CVM flash memory.
Program Text	Displays a text version of the Indexer 2 program.
Register Values	Displays the current values of the Indexer 2 Program's 32 registers (R0-R31).
Bean Versions	Displays the ID number and version of all the functions installed in the CME 2 CVM library (typically C:\Program Files\Copley Motion\CME 2\CvmLib).

Run Menu	Description
Run Control Program	Starts Indexer 2 Program execution. CAUTION: Depending on setup configuration and input line state, motion could start immediately. NOTE: The Run Control Program menu selection is disabled if the Indexer 2 Program has been changed but not saved to drive flash memory.
Stop Control Program	Stops Indexer 2 Program execution. CAUTION: Any programmed moves in progress will continue until finished.
Enable Control Program on Startup*	Configures the Indexer 2 Program to auto start when the drive is powered up or reset. This choice is the default setting.*
Disable Control	Disables auto start of the Indexer 2 Program.*


Program on Startup*	
*NOTE: When a new CVM control program is started or an existing one is opened from a file (.ccp), Enable Control Program on Startup is automatically selected.	

Tools Menu	Description
Clear All CVM Flash	Deletes all CVM control programs, Cam tables and gain scheduling tables from the drive CVM flash memory. Available only when Indexer 2 Program is not running. CAUTION: This command will delete all data stored in the drive CVM flash memory. It should only be used when it is believed that this memory has been corrupted.

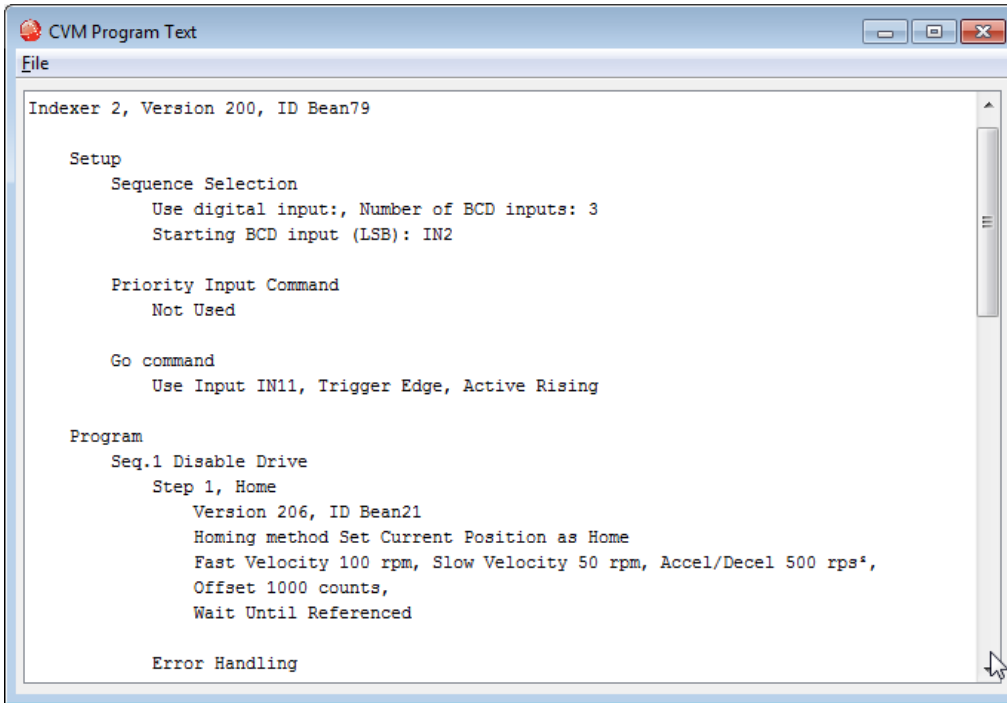
3.5: Toolbar

Icon	Tool	Description
	Save Control Program to disk	Saves the current CVM program in PC memory to a disk file with a <i>.ccp</i> filename extension.
	Restore Control Program from disk	Opens a saved CVM program. The program contained in this file will replace the current program in the PC memory.*
	Save Control Program to drive flash memory	Saves the current CVM program in PC memory to drive flash memory. This will overwrite any program currently in the drive
	Restore Control Program from drive flash memory	Replaces the current CVM program in PC memory with program stored in drive flash memory.
	Run Control Program	Starts program execution. CAUTION: Depending on setup configuration and input line state, motion could start immediately. NOTE: The Run Control Program menu selection is disabled if the program has been changed but not saved to drive flash memory.
	Stop Control Program	Stops program execution. CAUTION: Any programmed moves in progress will continue until finished.
	Single-Step Control Program	Starts the CVM program in single-step mode and opens the <i>Debug</i> tab for debugging purposes. See Error! Not a valid result for table.
	Program Text	Displays a window which contains a text version of the Indexer 2 program.
*NOTE: When a control program is restored from disk, Enable Control Program on Startup is automatically selected.		

Program Text

To view a text version of the Indexer 2 program, click  in the Tool Bar. The CVM Program Text screen will be displayed which show the setup and program in text form. The values entered for each step in the program will be shown.

To save the text to a file, select Save from the File menu and select a filename and location.



```
CVM Program Text
File
Indexer 2, Version 200, ID Bean79

Setup
  Sequence Selection
    Use digital input:, Number of BCD inputs: 3
    Starting BCD input (LSB): IN2

  Priority Input Command
    Not Used

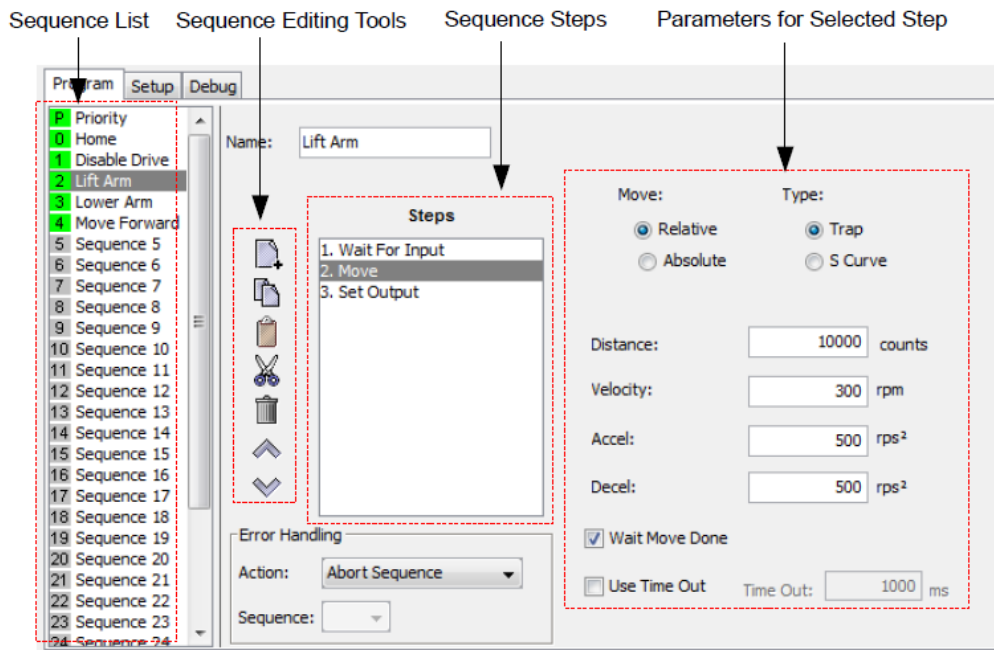
  Go command
    Use Input IN11, Trigger Edge, Active Rising

Program
  Seq.1 Disable Drive
    Step 1, Home
      Version 206, ID Bean21
      Homing method Set Current Position as Home
      Fast Velocity 100 rpm, Slow Velocity 50 rpm, Accel/Decel 500 rps²,
      Offset 1000 counts,
      Wait Until Referenced

  Error Handling
```

3.6: Creating and Modifying Sequences

The *Program* tab contains tools for creating sequences.










Sequence List

A green sequence number indicates that the sequence contains steps. A grey sequence number indicates that the sequence is empty.

Sequence Name

An optional name which has a maximum of 16 characters.

Sequence Editing Tools

Icon	Tool/Field	Description
	Name	Optional sequence name which has a maximum of 16 characters.
	Add Step	Opens the <i>Indexer Functions</i> screen. When a Function is selected from this screen and added, the function appears as a step in the sequence.
	Copy Step	Place a copy of the selected step on the Indexer 2 Program's clipboard.
	Paste Step	Paste the contents of the Indexer 2 Program's clipboard in the selected sequence. The step is added after the selected step.
	Cut Step	Remove the selected step from the sequence and place it on the Indexer 2 Program's clipboard.
	Delete Step	Remove the selected step from the sequence. After the step is deleted it cannot be retrieved.
	Step Up	Moves the selected step to the previous position in the sequence.
	Step Down	Moves the selected step to the next position in the sequence.

Error Handling


Error Handling

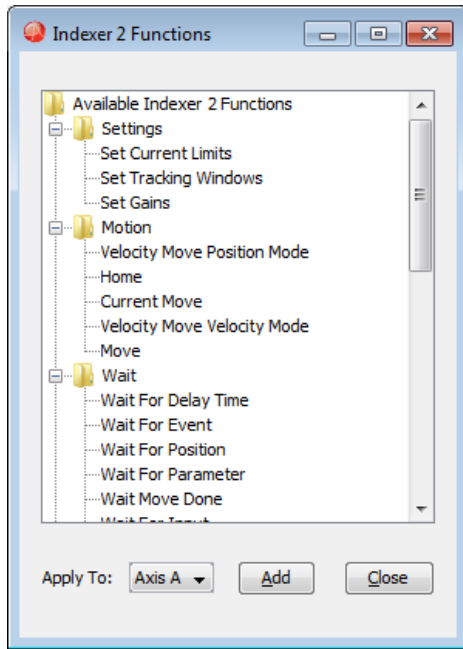
Action: Abort Sequence ▼

Sequence: ▼

Action	Abort Sequence: When a sequence error occurs, the Indexer 2 Program will immediately abort the sequence and wait for a Go command. If a Go command is received, the program will run the currently selected sequence
Sequence	Jump to Sequence: On error, the Indexer 2 Program will immediately execute the sequence selected in the Sequence field (no Go command required). NOTE: Sequence errors do not necessarily result in drive faults and drive faults do not necessarily result in sequence errors.

3.7: Adding Steps to a Sequence

- 1 Select the sequence in the sequence list.
- 2 Click **Add New Step**  on the *Program* tab to open the *Indexer Functions* screen.

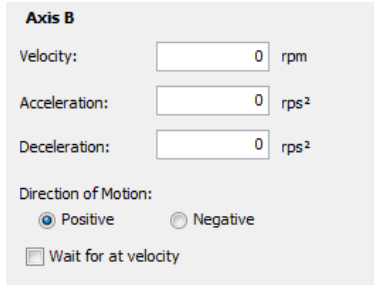


For function descriptions see [Functions](#).

- 3 Select the function in the list. To select multiple functions, hold the Control key while selecting functions. The order that the functions were selected will be the order that they will appear in the list of steps.
- 4 Select the axis that the function is to be applied to. Axis selection is only available on multi-axis drives.
- 5 Click **Add**.

3.8: Multi-axis Support

Starting with CME 2 V7.1, axis selection is done when the step is added as shown in [Adding Steps to a Sequence](#). The selected axis is then displayed at the top of the parameter setup of each step as shown below:



Axis B

Velocity: rpm

Acceleration: rps²

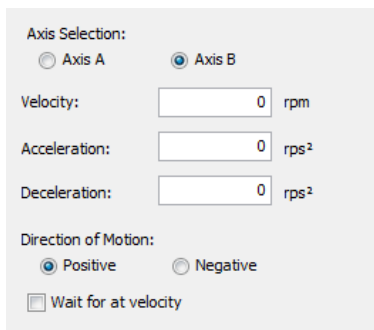
Deceleration: rps²

Direction of Motion:

Positive Negative

Wait for at velocity

In versions of CME 2 prior to V7.1, axis selection was done in the parameter setup of each step as shown below:



Axis Selection:

Axis A Axis B

Velocity: rpm

Acceleration: rps²

Deceleration: rps²

Direction of Motion:

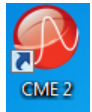
Positive Negative

Wait for at velocity

3.9: CVM Program Quick Copy Setup

Use this procedure to load a CVM Program into a drive.

Make sure the drive has either a direct serial bus connection to the PC running CME 2, or a CAN connection to the gateway drive that has the serial bus connection.

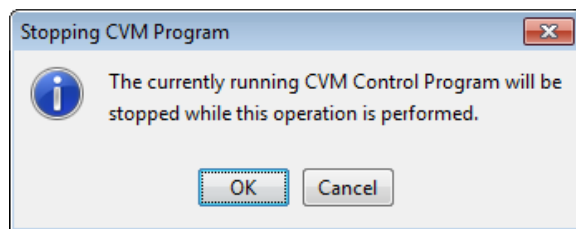


Start CME 2 by clicking the CME 2 shortcut icon on the Windows desktop.

In the CME 2 *Copley Neighborhood*, select the drive that you wish to set up.

From the **File** menu choose **Restore CVM Control Program**.

If a CVM control program was running, a prompt appears:



If the prompt appears, click **OK**.

When the *Restore Control Program from Disk* screen opens, navigate to the folder containing the appropriate *.ccp* file. Highlight the file name and then click **Open** to load the file data into drive flash.

This procedure also results in the setting of the Indexer 2 Program option [Enable Control Program on Startup](#). This configures the program to auto start when the drive is powered up or reset.



3.10: Using Debug Mode

Overview

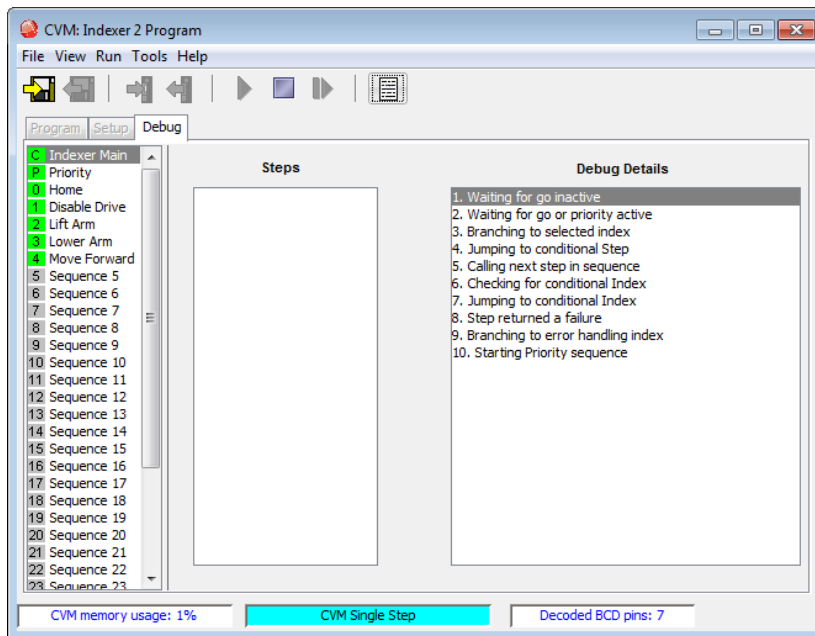
Debug mode is used to troubleshoot Indexer 2 Programs.


Start Debug Mode

To single-step a sequence from the beginning:




- 1 Click **Save to Flash**  and then Single Step Control Program .

The CVM program begins executing at the Indexer 2 Program Main loop level. *Debug Details* contains the list of instructions associated with the current sequence step or Indexer 2 Main program.



- 2 Click **Single Step Control Program**  as needed to proceed to the next step. If the Indexer 2 Program is at a Wait step, the condition for the wait must be satisfied before the program will proceed.


Start Debug Mode While Program is Running

- 1 Click **Stop Program**  and then **Single Step Control Program** .
- 2 Click **Single Step Control Program**  as needed to proceed to the next step. If the program is at a Wait step, the condition for the wait must be satisfied before the program will proceed.

Exit Debug Mode

Click **Stop Control Program** .

OR

Click **Run Control Program**  to start running the Indexer 2 Program from the current location.

CHAPTER

4: FUNCTIONS

4.1: Wait Move Done

Wait Move Done pauses execution of the sequence until the move in progress is completed.

Parameter	Description
Use time out	If selected and the move is not completed within the <i>Time out</i> period, the Indexer 2 Program performs the action selected from the choices below (Use Error Handling or Use Jump to Sequence).
Time out	Time period used by the time out option. Register Units: ms.
Use Error Handling on time out	If Time Out occurs, generate a sequence error.
Use Jump to Sequence on time out	If Time Out occurs, jump to the specified sequence.
Detect Motion Abort	If selected, a motion abort will result in a sequence error.

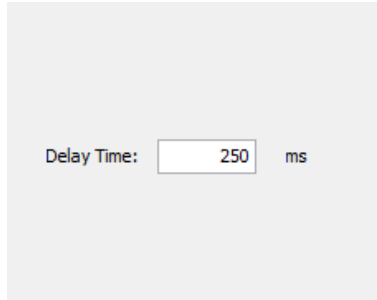
Errors

A sequence error will occur if during the wait time:

- The drive becomes hardware disabled.
- The drive faults.
- A software travel limit is reached or a hardware limit switch is activated.
- A register used for a function parameter contains a value that is not valid for the parameter.

4.2: Wait for Delay Time

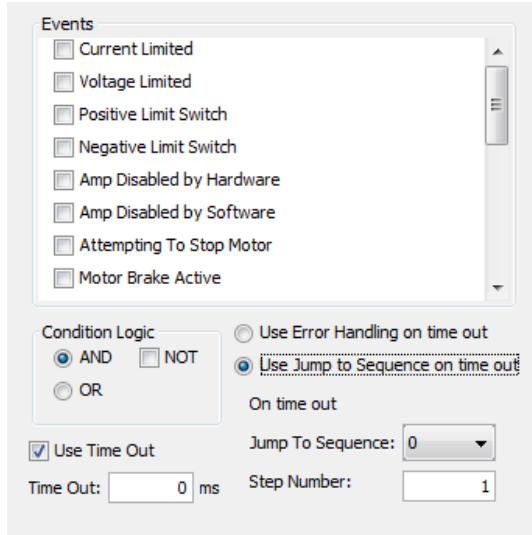
Wait for Delay Time pauses execution of the sequence for the specified amount of time.



Parameter	Description
Delay Time	The amount of time that the sequence will pause. Register units: ms.

4.3: Wait for Event

Wait for Event pauses execution of the sequence until the specified event conditions are met.



Control	Description
AND	Wait until all selected events are true.
OR	Wait until any of the selected events happen.
NOT	Wait until the selected event conditions are not present.
Use time out	If selected and the wait condition is not met within the <i>Time out</i> period, the Indexer 2 Program performs the action selected from the choices below (Use Error Handling or Use Jump to Sequence).
Time out	Time period used by the time out option. Register Units: ms.
Use Error Handling on time out	If Time Out occurs, generate a sequence error.
Use Jump to Sequence on time out	If Time Out occurs, jump to the specified sequence.

Wait for Event: Events	
Event	Description
Current Limited	The current output is being limited by the I ² T algorithm or a latched current fault has occurred.
Voltage Limited	Current loop is commanding the full bus voltage in an attempt to control current. Often occurs when motor is running as fast as the available bus voltage allows.
Positive Limit Switch	Axis has contacted positive limit switch.
Negative Limit Switch	Axis has contacted negative limit switch.
Positive Software Limit	Actual position has exceeded the positive software limit setting.
Negative Software Limit	Actual position has exceeded the negative software limit setting.
Following Error	Following error has occurred.
Following Warning	Following error has reached the programmed warning limit.
Velocity Limited	The velocity command (from analog input, PWM input, or position loop) has exceeded the programmed velocity limit.
Acceleration Limited	In velocity mode, motor has reached a programmed acceleration or deceleration limit.
Velocity Outside of Tracking Window	Difference between target and actual velocity has exceeded the window.
Position Outside of Tracking Window	The following error has exceeded the programmed value.
Amp Disabled by Hardware	Amplifier enable input(s) is not active.
Amp Disabled by Software	Amplifier is disabled by a software command.
Attempting to Stop Motor	The drive, while in velocity or position mode, has been disabled. In velocity mode, drive is using the programmed Fast Stop Ramp. In position mode, the drive is using the programmed Abort Deceleration. The output remains active until the drive is re-enabled.
Motor Brake Active	Motor brake activated.
PWM Outputs Disabled	The drive's PWM outputs are disabled.
Home Switch Active	Axis has contacted the home limit switch.
In Motion	Motor is moving, or has not settled after a move. Settled when it comes within the position tracking window and stays there for the tracking time at the end of a move. Once settled, it remains settled until a new move is started.
Phase not Initialized	Set when the drive has not yet initialized its phase when using phase initialization (wake and wiggle).

Errors

A sequence error will occur if this function is executed when a register used for a function parameter contains a value that is not valid for the parameter.

4.4: Wait for Input Mask

This function configures an Indexer 2 Program step to wait for the states of the drive’s digital inputs to match the specified mask before continuing to the next step.

Parameter	Description
Hi Lo X (IN 1 – IN 16)	Defines the mask: Hi: Mask condition is met if the input is high. Lo: Mask condition is met if the input is low. X: Don't care.
Use Time Out	If selected and the mask condition is not met within the <i>Time out</i> period, the Indexer 2 Program will jump to the specified step in the specified sequence.
Time out	Time period used by the time out option. Register Units: ms
On Time Out Jump to Sequence	If Time Out occurs, jump to the specified step in the specified sequence.

Errors

A sequence error will occur if a register used for a function parameter contains a value that is not valid for the parameter.

4.5: Wait for Position

Wait for Position pauses execution of the sequence until the axis position meets the programmed criteria.

Parameter	Description
Position	Wait until the axis position is, as specified: <ul style="list-style-type: none"> • Greater than or equal to the specified position or • Less than or equal to the specified position. Register Units: counts.
Actual Position	Use the actual position for the comparison position.
Limited Position	Use the limited position for the comparison position. Typically used for stepper motors operated in open-loop stepper mode.
Use Time Out	If selected and the position condition is not met within the <i>Time out</i> period, the Indexer 2 Program will jump to the specified sequence.
Time Out	Time period used by the time out option. Register Units: ms
On Time Out Jump to Sequence	If Time Out occurs, jump to the specified sequence.

Errors

A sequence error will occur if during the wait time:

- The drive becomes hardware disabled.
- The drive faults.
- A software travel limit is reached or a hardware limit switch is activated.
- A register used for a function parameter contains a value that is not valid for the parameter.

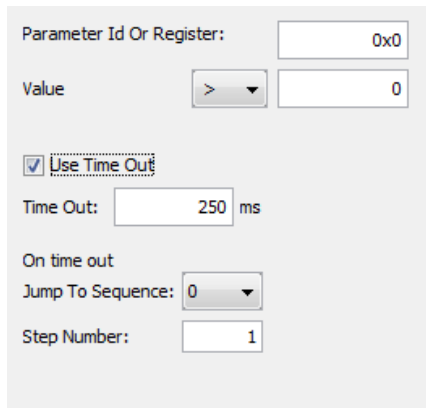
4.6: Wait for Input

Wait for Input pauses execution of the sequence until the specified input condition is met.

Parameter	Description
Input	Selects which input will be monitored.
Trigger	<ul style="list-style-type: none"> Edge: Will wait for an edge on the selected input. Level: Will wait for a level condition on the selected input. If the selected level is present on the input at the start of the wait, the sequence will continue to the next step.
Active State	With Edge Trigger: <ul style="list-style-type: none"> Rising: Sequence will continue on a low to high transition of the selected input. Falling: Sequence will continue on a high to low transition of the selected input. With Level Trigger: <ul style="list-style-type: none"> Hi: Sequence will continue when the selected input is high. Lo: Sequence will continue when the selected input is low.
Use time out	If selected and the input condition is not met within the <i>Time out</i> period, the Indexer 2 Program performs the action selected from the choices below (Use Error Handling or Use Jump to Sequence).
Time out	Time period used by the time out option. Register Units: ms.
Use Error Handling on time out	If Time Out occurs, generate a sequence error.
Use Jump to Sequence on time out	If Time Out occurs, jump to the specified step in the specified sequence.

4.7: Wait for Parameter

Wait for Parameter pauses execution of the sequence until the specified condition is met.



Parameter Id Or Register:

Value

Use Time Out

Time Out: ms

On time out

Jump To Sequence:

Step Number:

Parameter	Description
Parameter ID Or Register	The ID of the drive parameter or register which will be monitored.
Parameter Value	The value of the monitored parameter or register content is compared to the value in the field, using the chosen comparison operator.
Use time out	If selected and the parameter condition is not met within the <i>Time out</i> period, the Indexer 2 Program jumps to the specified step in the specified sequence.
Time out	Time period used by the time out option. Register Units: ms.
On time out Jump to Sequence	If Time Out occurs, jump to the specified step in the specified sequence.

Notes

For a list of parameter IDs, see the *Copley Parameter Dictionary*. Use the ASCII ID value.

Parameter IDs can be entered as decimal or hexadecimal numbers, using the 0x notation (for instance, 0x00c1). When the Indexer 2 Program reloads, all values will show in decimal form.

Due to the finite scan time of this function, the “equal to operator” should not be used with rapidly changing parameters such as actual velocity or actual current. Instead, chose “equal to or greater than” or “equal to or less than”.

Errors

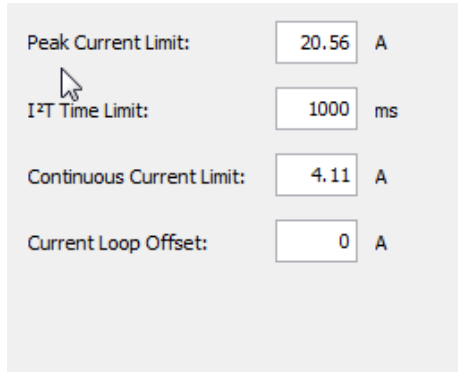
A sequence error will occur if:

- The specified parameter ID does not exist.
- The specified parameter ID is for a flash only parameter.

- The specified parameter returns more than two words.

4.8: Set Current Limits

The Set Current Limits function changes the current limits.



Peak Current Limit: A

I²T Time Limit: ms

Continuous Current Limit: A

Current Loop Offset: A

Parameter	Description
Peak Current Limit	Maximum current that can be generated by the drive for a short duration of time. This value cannot exceed the peak current rating of the drive. Register Units: 0.01 Amps
I ² T Time Limit	Maximum amount of time that the peak current can be applied to the motor before it must be reduced to the continuous limit or generate a fault. Register Units: ms
Continuous Current Limit	Maximum current that can be constantly generated by the drive. Register Units: 0.01 Amps
Current Loop Offset	Sets a current offset. Register Units: 0.01 Amps

Errors

A sequence error will occur if this function is executed when a register used for a function parameter contains a value that is not valid for the parameter.

4.9: Set Tracking Windows

This function modifies the velocity and position tracking windows.

Position

Tracking Window: counts

Tracking Time: ms

Following Error Fault: counts

Following Error Warning: counts

Velocity

Tracking Window: rpm

Tracking Time: ms

Position Tracking	
Parameter	Description
Tracking Window	Width of the tracking window. Register units: counts.
Tracking Time	Position must remain in the tracking window for this amount of time to be considered tracking. Register Units: ms
Following Error Fault	The level at which the following error produces a fault. Register units: counts.
Following Error Warning	The level at which the following error produces a warning. Register units: counts.
Velocity Tracking	
Parameter	Description
Tracking Window	Width of the tracking window. Register units: 0.1 counts/s.
Tracking Time	Velocity must remain in the tracking window for this amount of time to be considered tracking. Register Units: ms

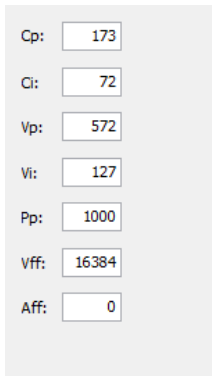
Errors

A sequence error will occur if this function is executed when a register used for a function parameter contains a value that is not valid for the parameter.

4.10: Set Gains

This function sets new gain values for the current, velocity and position loops during a sequence. This can be used, for instance, when the load on the axis has been changed.

The new values will stay in effect for all sequences until changed again by another Set Gains function or by a drive reset.



A screenshot of a control interface for setting gains. It features seven rows, each with a parameter label on the left and a numerical input field on the right. The parameters and their values are: Cp: 173, Ci: 72, Vp: 572, Vi: 127, Pp: 1000, Vff: 16384, and Aff: 0.

Parameter	Description
Cp	Current loop proportional gain.
Ci	Current loop integral gain.
Vp	Velocity loop proportional gain.
Vi	Velocity loop integral gain.
Pp	Position loop proportional gain.
Vff	Velocity feed forward gain.
Aff	Acceleration feed forward gain.

4.11: Velocity Move Position Mode

This function is used to change the operating mode of the drive to programmed position mode and configures a constant velocity trajectory.

Velocity: rpm

Acceleration: rps²

Deceleration: rps²

Direction of Motion:

Positive Negative

Wait for at velocity

Parameter	Description
Velocity	Commanded velocity. Register units: 0.1 counts/s. (Positive values only)
Acceleration	Acceleration rate. Register units: 10 counts/s ² .
Deceleration	Deceleration rate. Register units: 10 counts/s ² .
Direction of Motion	Positive or negative.
Wait for at velocity	If selected, sequence execution will wait at this step until the commanded velocity has reached the new value.

Errors

A sequence error will occur if this function is executed when:

- The drive is hardware disabled.
- The drive is faulted.
- Motor phasing is not initialized.
- A register used for a function parameter contains a value that is not valid for the parameter.

4.12: Move

The Move function executes a profile move using the specified parameters.

Parameter	Description
Move	Type of move: <ul style="list-style-type: none"> Relative: The axis will move the specified distance from the current position. Absolute: The axis will move to the specified absolute position.
Type	Type of profile: <ul style="list-style-type: none"> Trap: The move will use a trapezoidal profile. S Curve: The move will use an S curve profile.
Distance	Distance of a relative move. Register units: counts.
Position	Destination position of an absolute move. Register units: counts.
Velocity	Velocity during the constant velocity portion of the move. Register units: 0.1 counts/s
Accel	Acceleration rate of trapezoidal profile or maximum acceleration/deceleration rate of S curve profile. Register units: 10 counts/s ²
Decel (Trap move only)	Deceleration rate of trapezoidal profile. Register units: 10 counts/s ²
Jerk (S-Curve only)	Rate of change of acceleration and deceleration of the S curve profile. Register units: 100 counts/s ³
Wait Move Done	If selected, sequence execution will wait at this step for the commanded move to finish. If an error occurs during the move, the sequence will exit as programmed.

Notes

If **Wait Move Done** is not selected and a second move is commanded, the second move executes immediately with the following results:

- If the second move is a relative, trapezoidal move, the axis moves the relative distance from where the axis was when the second move was executed.
- If the second move is an absolute, trapezoidal move, the axis moves to the position specified by the second move.
- If an S curve profile move is executed while another move is still in progress, an error occurs.

Errors

A sequence error will occur if:

- The drive is hardware disabled when the Move begins, or becomes hardware disabled during the Move.
- The drive has a fault when the move begins or a fault occurs during the move.
- A software travel limit is reached or a hardware limit switch is activated during the move.

4.13: Home

The Home function executes a home command using the method and parameters specified.

Homing Method:

Direction of Motion: Positive Negative

Fast Velocity: rpm

Slow Velocity: rpm

Accel/Decel: rps²

Offset: counts

Current Limit: A

Delay Time: ms

Wait Until Referenced

Parameter	Description
Homing Method	Selects which homing method will be used. See Error! Reference source not found.
Direction of Motion	Sets the initial direction of motion. Positive or Negative.
Fast Velocity	The velocity used to find a limit or home switch. Also used when moving to an offset position, or a resolver or Servo Tube index position. Register units: 0.1 counts/s.
Slow Velocity	The velocity used to find a switch edge, index pulse, or hard stop. Register units: 0.1 counts/s.
Accel/Decel	The acceleration and deceleration rate used during homing. Register units: 10 counts/s ² .
Offset	The axis will execute a move of this distance after the reference is found. The actual position will be reset to 0 at this new position and will now be considered home. Register units: counts.
Current Limit	The current used during the home to hard stop method. Register units: 0.01 A.
Delay Time	The amount time the current limit is applied during the home to hard stop. Register units: ms.
Wait until referenced	If selected, sequence execution will wait at this step for the home move to finish. If a homing error occurs during homing, the Indexer 2 Program will use the Error Handling action selected from the choices below: <ul style="list-style-type: none"> • Abort Sequence.

	• Jump to the specified sequence.
--	-----------------------------------

Notes

Do not execute a move function until the homing function is complete.

Errors

A sequence error will occur if:

- The drive is, or becomes, hardware disable during the Homing function.
- The drive has a fault at the beginning of the Home function or a fault occurs during the function.

Other sequence errors may occur that are specific to the homing method selected.
See [Error! Reference source not found.](#)

4.14: Current Move

This function is used to change the operating mode of the drive to programmed current mode and configure the mode.

Current: A

Current Ramp: mA/s

Wait for at commanded current

Parameter	Description
Current	Commanded current. Register units: 0.01 A.
Current Ramp	Rate at which the current will change to its commanded value. Register Units: mA/s
Wait for at commanded current	If selected, sequence execution will wait at this step until the commanded current has ramped to the new value.

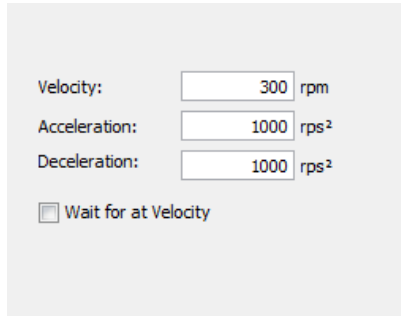
Errors

A sequence error will occur if this function is executed when:

- The drive is hardware disabled.
- The drive is faulted.
- Motor phasing is not initialized.
- A register used for a function parameter contains a value that is not valid for the parameter.

4.15: Velocity Move Velocity Mode

This function is used to change the operating mode of the drive to programmed velocity mode and configures a constant velocity trajectory.



Velocity: rpm
Acceleration: rps²
Deceleration: rps²
 Wait for at Velocity

Parameter	Description
Velocity	Commanded velocity. Register units: 0.1 counts/s. (Positive values only)
Acceleration	Acceleration rate. Register units: 10 counts/s ² .
Deceleration	Deceleration rate. Register units: 10 counts/s ² .
Wait for at velocity	If selected, sequence execution will wait at this step until the commanded velocity has reached the new value.

Errors

A sequence error will occur if this function is executed when:

- The drive is hardware disabled.
- The drive is faulted.
- Motor phasing is not initialized.
- A register used for a function parameter contains a value that is not valid for the parameter.

4.16: Analog Velocity Mode

This function is used to change the operating mode of the drive to analog velocity mode and configure the mode.

Parameter	Description
Scaling	Sets the input-voltage to-velocity command ratio. Register units: Counts / second = 10 V.
Dead Band	Sets dead band. The drive treats anything within the dead band ranges as zero, and subtracts the dead band value from all other values. Register units: mV
Offset	Offset applied to the input voltage. Register units: mV.
Invert Command	Inverts polarity of drive output with respect to input signal.

Errors

A sequence error will occur if this function is executed when:

- The drive is hardware disabled.
- The drive is faulted.
- Motor phasing is not initialized.
- A register used for a function parameter contains a value that is not valid for the parameter.

4.17: Analog Position Mode

This function is used to change the operating mode of the drive to Analog Position mode and configure the mode.

Configuration

Scaling: counts = 10V

Dead Band: mV

Invert Command

Trajectory Limits

Velocity: rpm

Accel: rps²

Decel: rps²

Parameter	Description
Scaling	Controls the input-to-command ratio. Position change produced by +10 Vdc of input. Register units: Counts = 10 V.
Dead Band	Sets dead band. The drive treats anything within the dead band ranges as zero, and subtracts the dead band value from all other values. Register units: mV
Invert Command	Inverts polarity of drive output with respect to input signal.
Velocity	Maximum velocity that will be commanded during a move. Register units: 0.1 counts/s.
Accel	Maximum acceleration that will be used during a move. Register units: 10 counts/s ² .
Decel	Maximum deceleration that will be used during a move. Register units: 10 counts/s ² .

Notes

The analog position command operates as a relative motion command. When the drive is enabled the voltage on the analog input is read. Then, any change in the command voltage will move the axis a relative distance, proportional as programmed to the change in voltage, from its position when enabled.

To use the analog position command as an absolute position command, the drive should be homed every time it is enabled. A simple way to accomplish this is to program the priority sequence to use the same input as the enable input. The priority sequence would be programmed to home the axis. In this way, the axis will be automatically homed every time the drive is enabled.

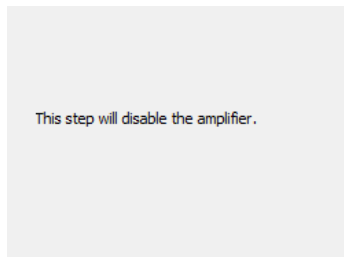
Errors

A sequence error will occur if this function is executed when:

- The drive is hardware disabled.
- The drive is faulted.
- Motor phasing is not initialized.
- A register used for a function parameter contains a value that is not valid for the parameter.

4.18: Disable Drive

This function is used to software disable the drive.



Disable Drive Notes

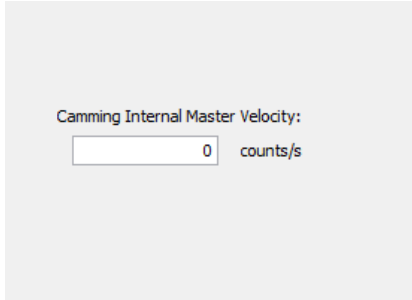
The drive will automatically re-enable with the execution of any function, such as Home or Move, which requires motion.

Errors

This function does not generate an error.

4.19: Camming Internal Master

This function is used to change the velocity of the Camming Internal Master.



Parameter	Description
Velocity	Velocity of the Camming Internal Master. Register units: 0.1 counts/s.

Errors

A sequence error will occur if this function is executed when a register used for a function parameter contains a value that is not valid for the parameter.

4.20: Camming

This function is used to change the operating mode of the drive to Camming and to configure the Camming mode.

The screenshot shows a configuration window for the Camming function. It is divided into four main sections:

- Master Input:** Two radio buttons are present. 'External' is selected, and 'Internal' is unselected. Next to 'Internal' is a text input field containing '0' followed by the unit 'counts/s'.
- Trigger Type:** Three radio buttons are present. 'Use Input: IN2' is selected. Below it are two more radio buttons: 'Edge' (selected) and 'Level' (unselected).
- Offset:** Two text input fields are present. The first is labeled 'Forward:' and contains '0' followed by 'counts'. The second is labeled 'Reverse:' and also contains '0' followed by 'counts'.
- Active Cam Table:** A dropdown menu is present, currently showing 'CAM 0'.

Parameter	Description
Master Input	External: Source of camming master pulse is external. Internal: Source of camming master pulse is internal generator. Velocity of internal master pulse is set using the counts/s field. It can also be reset using the Camming Internal Master function.
Trigger Type	Controls how execution of a set of Camming moves is triggered: None (Continuous): The active Cam Table is executed continuously. Use Master (Secondary) Encoder Index: The active Cam Table is executed when the drive receives an index pulse from the camming master encoder. Index pulses received during execution are ignored. Use Input, Edge: The active Cam Table begins executing on the rising edge of the designated input pin. Use Input, Level: The active Cam Table repeats continuously as long as the designated input is active.
Offset	Forward: A delay (in counts) applied before the active Cam Table is executed when the Master is moving forward. Reverse: A delay (in counts) applied before the active Cam Table is executed when the Master is moving in reverse.
Active Cam Table	The table containing the move profile that the drive will execute when the camming trigger is activated.

Notes

- The velocity of internal master pulse can be changed using the Camming Internal Master function.
- The drive will stay in the camming mode of operation until another Indexer 2 Program function changes it or the drive is reset or power cycled.

Errors

A sequence error will occur if this function is executed when:

- The drive is hardware disabled.
- The drive is faulted.
- Motor phasing is not initialized.
- A register used for a function parameter contains a value that is not valid for the parameter.

4.21: Digital Position Mode

Change the operating mode of the drive to Digital Position mode and configure the mode.

Control Input: Pulse and Direction Pulse Up / Pulse Down Quadrature

Increment Position On: Rising Edge Falling Edge

Stepping Resolution
 Input Pulses = Output counts
 Invert Command

Trajectory Limits
 Velocity: rpm
 Accel.: rps²
 Decel.: rps²

Parameter	Description
Control Input	<p>Pulse and Direction: One input takes a series of pulses as motion step commands, and another input takes a high or low signal as a direction command.</p> <p>Pulse Up / Pulse Down: One input takes each pulse as a positive step command, and another takes each pulse as a negative step command.</p> <p>Quadrature: A/B quadrature commands from a master encoder (via two inputs) provide velocity and direction commands.</p>
Increment position on	<p>Rising Edge: Increment position on the rising edge of the input pulse.</p> <p>Falling Edge: Increment position on the falling edge of the input pulse.</p>
Stepping Resolution	<p>Input Pulses: Number of Input Pulses/Counts required to produce given number of output counts.</p> <p>Output Counts: Number of Output Counts per given number of input pulses. Range: 1 to 32,767. Register Units: Pulses/Counts.</p>
Invert Command	When selected, inverts commanded direction.
Velocity	Velocity during the constant velocity portion of the move. Register units: 0.1 counts/s.
Accel	Acceleration rate of trapezoidal profile. Register units: 10 counts/s ² .
Decel	Deceleration rate of trapezoidal profile. Register units: 10 counts/s ² .

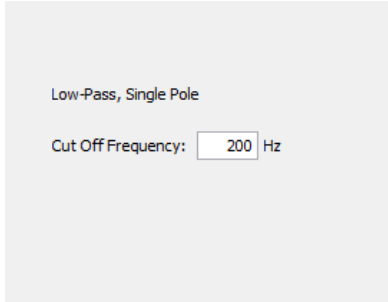
Errors

A sequence error will occur if this function is executed when:

- The drive is hardware disabled.
- The drive is faulted.
- Motor phasing is not initialized.
- A register used for a function parameter contains a value that is not valid for the parameter.

4.22: Velocity Loop Single Pole Output Filter

This function modifies the Velocity Loop Output Filter.



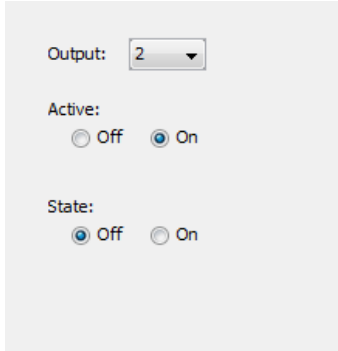
Parameter	Description
Cut Off Frequency	The frequency for which the output of the circuit is -3 dB of the nominal.

Errors

This function does not generate sequence errors.

4.23: Set Output

Set Output sets an output to the specified state.



Parameter	Description
Output	Selects which output will be set.
State	<ul style="list-style-type: none"> • Hi: Sets the selected output high or off. • Lo: Sets the selected output low or on.

Notes

When this function is executed, it changes the configuration of the selected output to "Program Control, Active Low." The "Program Control, Active Low" configuration will stay in effect until the drive is reset. Consider this effect when programming digital outputs.

4.24: Position Triggered Output

This function programs the drive to activate a digital output when the specified position criteria are met.

Output: 1

Configuration

- In Position Window
- Trigger At Position
- Trigger Positive Motion
- Trigger Negative Motion

Output Active High Output Active Low

Upper Position: 80000 counts

Lower Position: 20000 counts

Use Actual Position

Use Limited Position

Basic Setting	Description
Output	The digital output that is being programmed.
Output Active High	Output goes high when activated.
Output Active Low	Output goes low when activated.
Use Actual Position	Use the actual position for the target position.
Use Limited Position	Use the limited position for the target position. Typically used for stepper motors operated in open-loop stepper mode.

Configuration	Description and Parameters
In Position Window	Activates the output while the axis is in the window between the programmed Upper Position and Lower Position .
Trigger at Position	Activates the output for the programmed Time when the axis travels through the programmed Position .
Trigger Positive Motion	Activates the output for the programmed Time when the axis travels in the positive direction through the programmed Position .
Trigger Negative Motion	Activates the output for the programmed Time when the axis travels in the negative direction through the programmed Position .

Notes

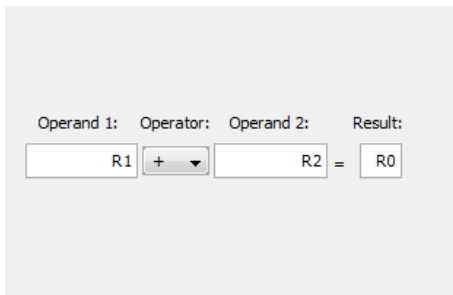
When this function is executed, it changes the configuration of the selected output to the chosen Position Triggered Output configuration. This configuration of the output will stay in effect until the drive is reset or the output configuration is changed by another function.

Errors

A sequence error will occur if this function is executed when a register used for a function parameter contains a value that is not valid for the parameter.

4.25: Math

This function is used to perform basic integer math and assign the result to an Indexer 2 Program



Parameter	Description
Operand 1	An operand. Value can be an integer constant or a reference to an Indexer 2 Program register. *
Operator	Addition, Subtraction, Multiplication, or Division.
Operand 2	An operand. Value can be an integer constant or a reference to an Indexer 2 Program register.*
Result	The register to which the math results will be written. For division, the result value is the integer portion of the quotient.
Remainder	The register to which the remainder portion of the quotient will be written.
For multiplication operations, the Math function imposes a value range of -32768 through 32767. To multiply numbers outside of this range, use the function Extended Math .	

Notes

The Math function does not support carry on addition or borrow on subtraction. Thus, if a Math operation results in a number that exceeds the capacity of the register (between $2^{31}-1$ and -2^{31}) the results will be incorrect (but no error will be generated).

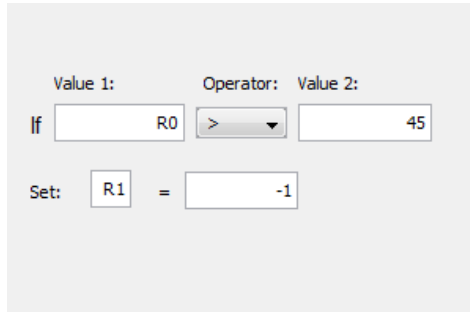
Errors

A sequence error will occur if:

- During multiplication, one of the operands is a register with a value less than -32768 or greater than 32767.
- During division, Operand 2 is a register with a value equal to 0.

4.26: If Register Set

This function is used to conditionally assign a value to an Indexer 2 Program register.



Parameter	Description
Value 1	One of the two values being compared. Value can be an integer constant or a reference to an Indexer 2 Program register.
Operator	<ul style="list-style-type: none"> • Greater than • Less than • Equal to • Not equal to • Greater than or equal • Less than or equal.
Value 2	One of the two values being compared. Value can be an integer constant or a reference to an Indexer 2 Program register.
Set	On the left side of the equation, enter the register to be set. On the right side of the equation, enter an integer constant or a reference to an Indexer 2 Program register.

Notes

When the Indexer 2 Program has been set up to use a register to select a sequence, the If Register Set function can determine which sequence will execute after the current sequence has finished.

Errors

This function does not generate an error.

4.27: Get Set Parameter

This function is used to get or set the value of one the drive's internal parameters.

Parameter ID #: 0x24 Register #: R0
 Get: Save To: R0
Parameter ID #: 0x0 Value: R0
Memory Option:
 Ram Flash

Setting	Description
Get	Copies the current value of the specified drive parameter into the register.
Parameter ID #	The ID of the drive parameter that the function will read. For a list of drive parameter IDs, see the <i>Copley Parameter Dictionary</i> . Use the ASCII ID value given in that book.
Register #	The value of the drive parameter will be copied into this register.
Set	Copies a value into the specified drive parameter.
Parameter ID #	The ID of the drive parameter that the function will write.
Value	The value in this field or the value in the specified register will be copied to the drive parameter.
Memory Option	Selects whether to perform the get or set on the RAM or Flash version of the parameter.

Notes

Parameter IDs can be entered as decimal or hexadecimal numbers. When entering hexadecimal numbers, use the 0x notation (for instance, 0x00c1). The Indexer 2 Program will convert and display all values in hexadecimal form.

Errors

A sequence error will occur if:

- The specified parameter ID does not exist.
- The specified parameter returns more than two words.
- A set is attempted on a read only parameter.
- A set is attempted and the value is illegal for the parameter being set.

4.28: Save Restore Registers

This function allows either a single register or all registers to be saved to or restored from flash.



The maximum number of times flash can be written to is 10,000. Exceeding this number of writes may cause the drive's flash to fail. This function is not intended for frequent writes to flash.

Restore flash value to register
 Save register value to flash

All Registers

Register Number:

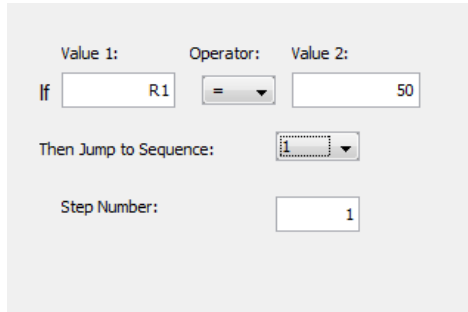
Setting	Description
Restore flash value to register	Copies the previously stored register value from flash to the specified register.
Save register value to flash	Saves the specified register to flash
All Registers	When selected, all 32 registers will be saved/restored
Register Number	The register to be saved or restored. Valid entries are R0 through R31.

Errors

This function does not generate sequence errors.

4.29: Conditional Jump

This function is used to conditionally jump to a specified step within a specified sequence. If the conditional statement is not true, the Indexer 2 Program continues to the next step in the original sequence.



The screenshot shows a configuration window for a Conditional Jump function. It contains the following elements:

- Value 1:** A text input field containing "R1".
- Operator:** A dropdown menu with "=" selected.
- Value 2:** A text input field containing "50".
- Then Jump to Sequence:** A dropdown menu with "1" selected.
- Step Number:** A text input field containing "1".

Setting	Description
Value 1	One of the two values being compared. Value can be an integer constant or a reference to an Indexer 2 Program register.
Operator	<ul style="list-style-type: none"> • Greater than • Less than • Equal to • Not equal to • Greater than or equal • Less than or equal.
Value 2	One of the two values being compared. Value can be an integer constant or a reference to an Indexer 2 Program register.
Then Jump to Sequence	Sequence to jump to when condition is met. Note that this can be the same sequence that initiates the Conditional Jump.
Step Number	Step within sequence to jump to when condition is met.

Notes

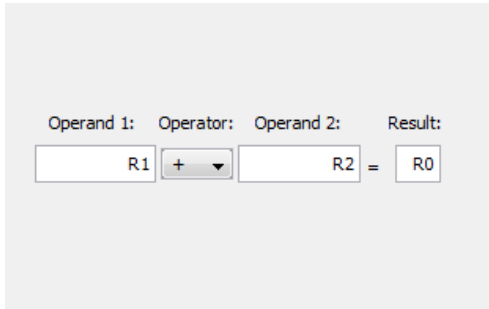
If a jump is made to a sequence that has not been programmed or to a step in a sequence that does not exist, an error is not generated. Instead, the Indexer 2 Program will return to an idle state waiting for the next Go command.

Errors

This function does not generate sequence errors.

4.30: Extended Math

This function is used to perform basic integer math and assign the result to an Indexer 2 Program register. Extended Math is identical to [Math](#) except that Extended Math does not impose a size limit on multiplication operands.



Operand 1: R1 Operator: + Operand 2: R2 Result: R0

Parameter	Description
Operand 1	An operand. Value can be an integer constant or a reference to an Indexer 2 Program register.
Operator	Addition, Subtraction, Multiplication, or Division
Operand 2	An operand. Value can be an integer constant or a reference to an Indexer 2 Program register.
Result	The register to which the math results will be written. For division, the result value is the integer portion of the quotient.
Remainder	The register to which the remainder portion of the quotient will be written.

Notes

The Math function does not support carry on addition or borrow on subtraction. Thus, if a Math operation results in a number that exceeds the capacity of the register (between $2^{31}-1$ and -2^{31}) the results will be incorrect (but no error will be generated).

Errors

A sequence error will occur if:

- During division, Operand 2 is a register with a value equal to 0.

4.31: Logic

This function is used to perform logical operations on drive parameter values or Indexer 2 register values.

Get Parameter or Register:

Logic Operator:

Value or Register:

Result:

Parameter	Description
Get Parameter or Register	Operand. Specify a drive parameter ID number or Indexer 2 register number.
Logic Operator	Operation: AND, OR, XOR
Value or Register	Operand. Specify a numerical value or an Indexer 2 register number.
Result	Specify the number of the Indexer 2 register where you want to put the results of the logic operation.

Notes

Parameter IDs can be entered as decimal or hexadecimal numbers. When entering hexadecimal numbers, use the 0x notation (for instance, 0x00c1). The Indexer 2 Program will convert and display all values in hexadecimal form.

Errors

A sequence error will occur if:

- The specified parameter ID does not exist.
- The specified parameter ID is for a flash only parameter.
- The specified parameter returns more than two words.

4.32: CAN PDO Mapping

The CAN PDO Mapping Tool is used to map PDOs.

The image shows two side-by-side screenshots of the CAN PDO Mapping Tool interface for Axis A. Both screenshots have 'PDO Number' set to 1 and 'CAN Message ID' set to 0x00000000, with '29 Bits' selected for the message size.

Left Screenshot (TxPDO Configuration):

- Radio buttons: TxPDO From Drive, RxPDO To Drive
- Section: TxPDO Type
 - RTR, PDO Type Value: 252
 - Dropdown: SYNC + RTR, Period: 1
- Table: Objects To Map

Index	Sub-Index	Size in Bits	
1:	0x0000	0x00	16
2:	0x0000	0x00	16
3:	0x0000	0x00	16
4:	0x0000	0x00	16

Right Screenshot (RxPDO Configuration):

- Radio buttons: TxPDO From Drive, RxPDO To Drive
- Section: RxPDO Type
 - Sync, Async
 - PDO Type Value: 0
- Table: Objects To Map

Index	Sub-Index	Size in Bits	
1:	0x0000	0x00	16
2:	0x0000	0x00	16
3:	0x0000	0x00	16
4:	0x0000	0x00	16

Parameter	Description
PDO Number	Number of the PDO to be mapped. Range 1-8
CAN Message ID	This object holds the CAN object ID used by the PDO. Available in 29 or 11-bit size
RTR	Remote Transfer Request
PDO Type	RxPDOs are configurable as Sync or Async. TxPDOs are configurable as SYNC+RTR or EVENT+RTR if RTR is enabled. If RTR not enabled, options are SYNC in which case Period is configurable, EVENT, or SYNC+EVENT
Period	Transmission of PDO every n'th period
Index	4-digit object Index. Input in Hexadecimal Format (0x1234)
Sub-Index	2-digit object Sub-Index. Input in Hexadecimal Format (0x01)
Size in Bits	Sizes range from 8, 16, and 32 bits.

Notes

For a list of CANopen Object IDs, see the *CANopen Programmer's Guide*. CAN Message IDs with an 11-bit size will receive priority from the drive over IDs with a 29-bit size. The total number of mappable bits in any one PDO is 64. Mapping more than 64 bits will not return an error, but the bits that exceed 64 will not be mapped.

Errors

A parameter error will occur if:

- The value of **CAN Message ID** is out of range. Valid range is 0x0 to 0x3FFFFFFF for 29-bits and 0x0 to 0x7FF for 11-bits.
- The value of **Period** is out of range. Valid range is 0x1 to 0xF0, or 1 to 240.

Indexer 2 User Guide

P/N 16-01192
Revision 01
February 2019

© 2019
Copley Controls
20 Dan Road
Canton, MA 02021 USA
All rights reserved