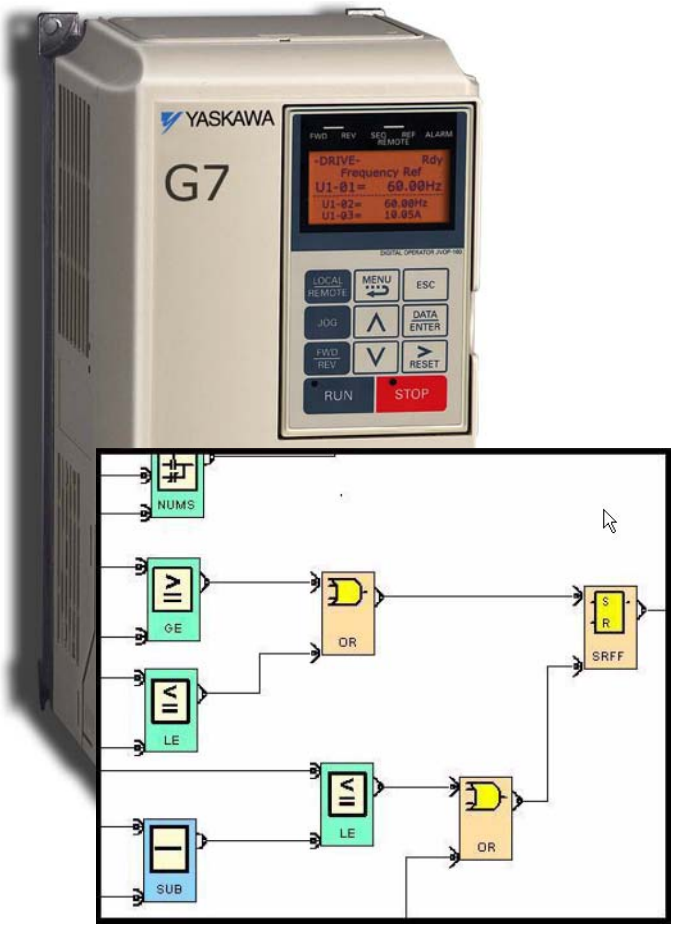
DriveWorksEZ™ Technical Manual



Model: CIMR-G7U **Document Number: TM.DWEZ.01**

This page intentionally left blank.

Thank you for choosing this YASKAWA ELECTRIC AMERICA software product. Proper installation and use of this software will ensure proper product performance, and will prevent damage to connected drive products.

Please read this manual thoroughly before using this software product and operate the software and any connected drive products with care.

Please refer to the drive product technical manual for proper handling and care of any connected drive.

◆ Notice

1. This manual describes the functions of the product and relations with other products. You should assume that anything not described in this manual is not possible.
2. Although care has been given in documenting the product, please contact your YASKAWA ELECTRIC AMERICA representative if you have any suggestions on improving this manual.
3. The drive product contains potentially dangerous parts under the cover. Do not attempt to open the cover under any circumstances. Doing so may result in injury or death and may damage the product. Never attempt to repair or disassemble the product.
4. We recommend that you add the following precautions to any instruction manuals you prepare for the system into which the product is being installed or incorporated.
   - Precautions on the dangers of high-voltage equipment.
   - Precautions on touching the terminals of the drive product even after power has been turned OFF. (These terminals are live even with the power turned OFF.)
5. Specifications and functions may be changed without notice in order to improve product performance.

◆ Items to Check Before Unpacking

Check the following items before removing the product from the package:

- Has the correct product been delivered (i.e., the correct model number and specifications)?
- Has the product been damaged in shipping?

◆ Notice

YASKAWA ELECTRIC AMERICA products are created and manufactured for use according to proper procedures by a qualified operator and only for the purposes described in this manual.

The following conventions are used to indicate and classify precautions in this manual. Always heed the information provided with them. Failure to heed precautions can result in injury to people or damage to property.

⚠ **DANGER** indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.

⚠ **WARNING** indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury.

⚠ **Caution** indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury, or property damage.

◆ YASKAWA Product References

All YASKAWA and YASKAWA ELECTRIC AMERICA products are capitalized in this manual.

The abbreviation “PC” means Personal Computer, and is not used as an abbreviation for anything else.

Table of Contents

1

DriveWorksEZ™ Introduction

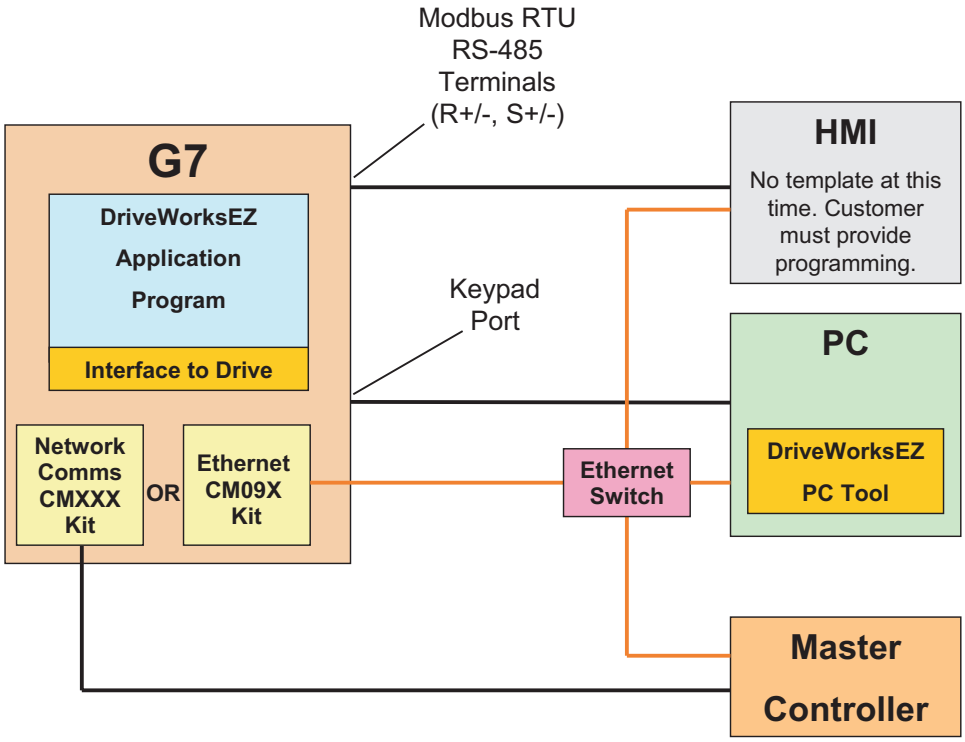
DriveWorksEZ™ Overview

DriveWorksEZ™ is a system that allows the user to create a custom application software program that resides in the Yaskawa G7 drive. The program is created by connecting functions that are already programmed in the drive together to create an application program. These connection are stored in the special parameters that are part of the DriveWorksEZ™ software.

The system is made up of two parts.

- The DriveWorksEZ™ software that is part of the G7 drive firmware. This firmware contains all of the actual program function blocks that are connected together to create an application program.
- The DriveWorksEZ™ PC Tool that is used to create, save, monitor, and edit the application program in the G7 drive equipped with the DriveWorksEZ™ firmware.

The figure below shows an overview of the DriveWorksEZ™ system.



◆ Overview of the DriveWorksEZ™ G7 Drive Software

- Utilizes all of the drive's I/O, including two channels of PG (Pulse Generator) feedback.
- Access to the G7 drive speed and torque commands.
- Access to some G7 drive internal functions.
- Monitor values are available.
- Digital and analog functions are available to create complex control function.
- Data can be passed between drive communications and the DriveWorksEZ™ program.
- Various access levels to DriveWorksEZ™ parameters are protected by passwords.
- Independent initialization of DriveWorksEZ™ separate from standard drive initialization.

◆ Overview of the DriveWorksEZ™ PC Tool Software

- The PC tool utilizes drag and drop graphical function blocks to create the program.
- Input and output values of the function blocks can be monitored on-line.
- Templates for pre-engineered programs can be created.
- A DriveWorksEZ™ program, plus data, is called a project.
- Projects can be created off-line and on-line.
- A program is divided into pages (maximum of 8).
- A subroutine can be created that can be used anywhere in the current project or it can be imported into a different project.

DriveWorksEZ™ System Requirements

For proper operation of DriveWorksEZ™, the G7 drive and the PC running the PC Tool both have minimum requirements.

◆ Minimum PC Configuration

- **Processor:** Pentium III, 300MHZ or greater
- **Hard Disk:** 100Mb of free space
- **Memory:** 128Mb or greater
- **Operating system:** Windows 2000, Windows XP, Windows XP Professional
- **Video:** Minimum 800 x 600 resolution

◆ Minimum Drive Configuration

- **Control board:** ETC618058-S3040 or newer
- **Drive software installed:**

  Control Board CPU Flash version VSG134041

  Control Board CPU Flash version VSG133041

The revision of the Control Board Flash can be confirmed by viewing the U1-14 monitor (FLASH ID) or viewing the U1-28 monitor (CPU ID).

■DriveWorksEZ™ (DWEZ) is compatible with these Drive Option Cards

- AI-14B
- DI-16H
- DI-08
- DO-02
- PG-X2
- PG-W2
- CM090
- CM092
- AO-08
- AO-12

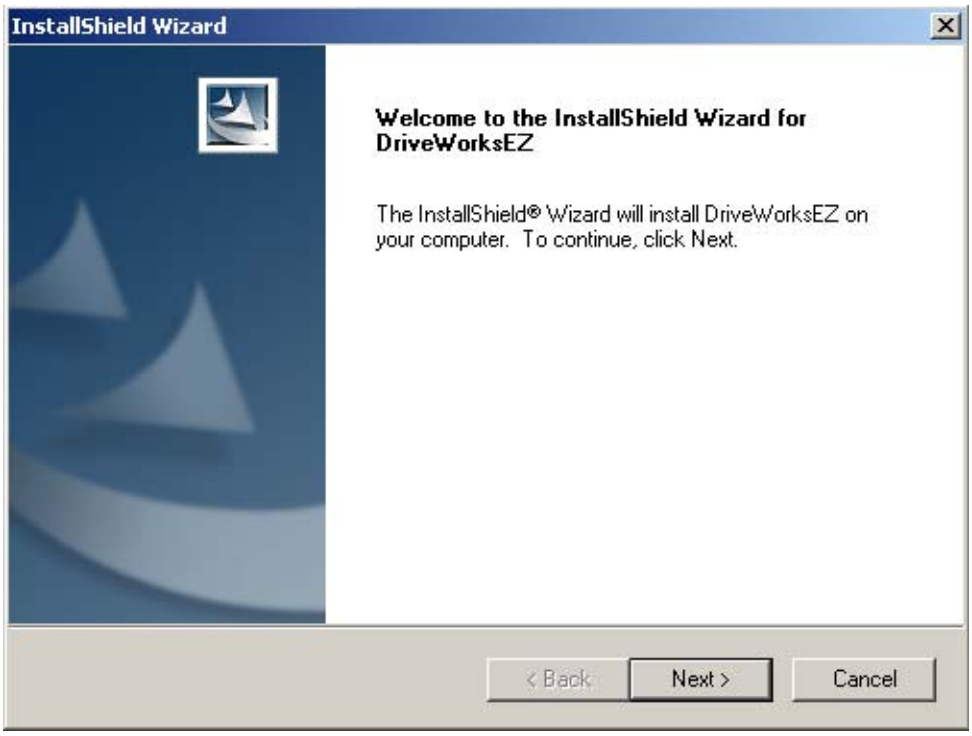
Installing the DriveWorksEZ™ PC Software

INFO

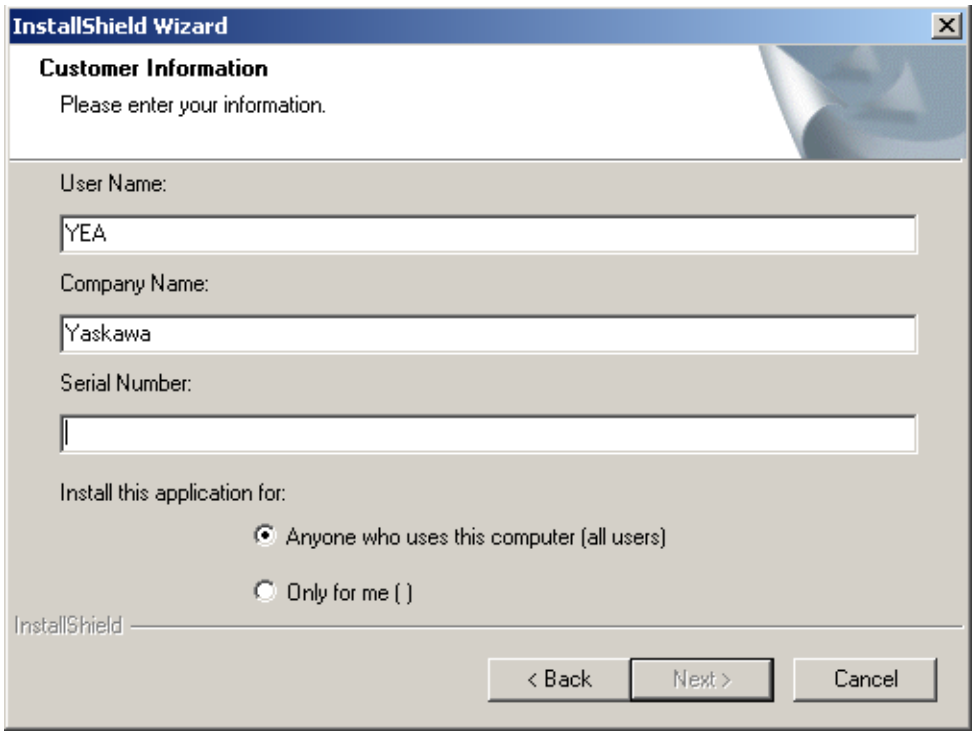
When installing a newer version of DriveWorksEZ™, it is recommended that you first uninstall the previous version and reboot your PC to make sure the installation executes correctly.

To install DriveWorksEZ™ from the CD.DWEZ.01 CD-ROM, please follow these steps below.

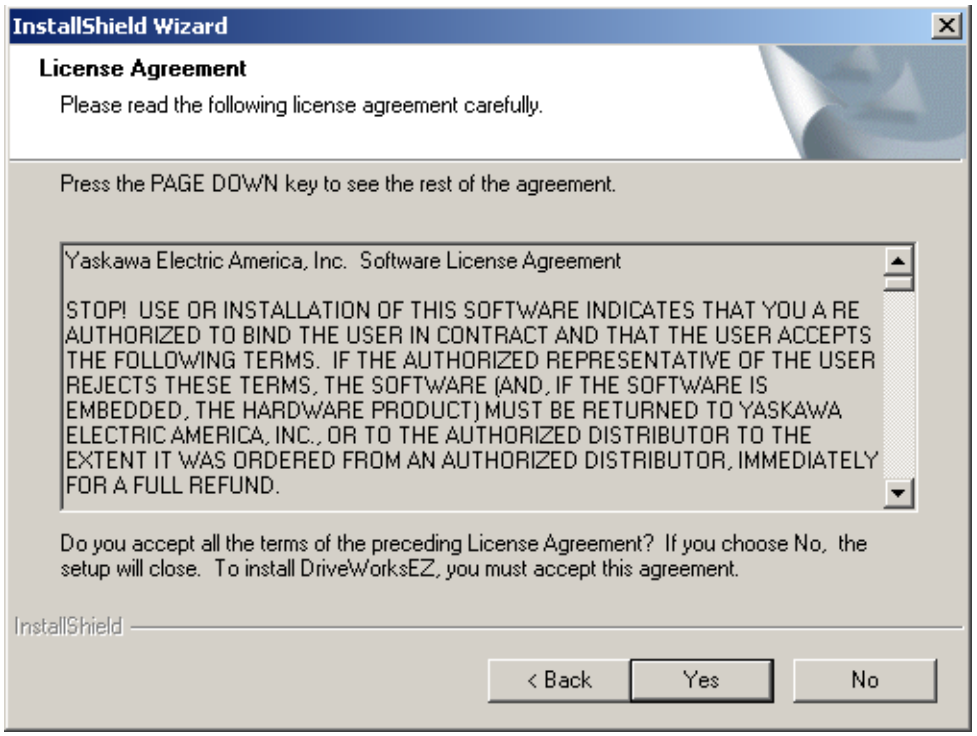
1. Close all open programs currently running on the PC to prevent interference with the installation process.
2. Insert the CD-ROM into the CD-ROM drive.
3. If *Auto-play* is enabled, the CD-ROM will automatically start when the CD-ROM is inserted. The CD ROM main navigation menu will appear in a new window on the PC display. Click on the *Software menu* button and select the DriveWorksEZ™ PC tool installation icon.
4. If *Auto-play* is not enabled, a manual method may be used to start installation. If the CD-ROM drive is not D, replace D with the appropriate CD-ROM drive letter.
   a. Insert CD.DWEZ.01 into the PC CD-ROM drive.
   b. Open WINDOWS Explorer and click on the drive letter where the CD is located.
   c. Find the installation file.
   d. Double click on the setup.exe icon to start the installation. The installation program will start and a welcome screen will appear.
5. An initial screen will appear. Click *Next*.



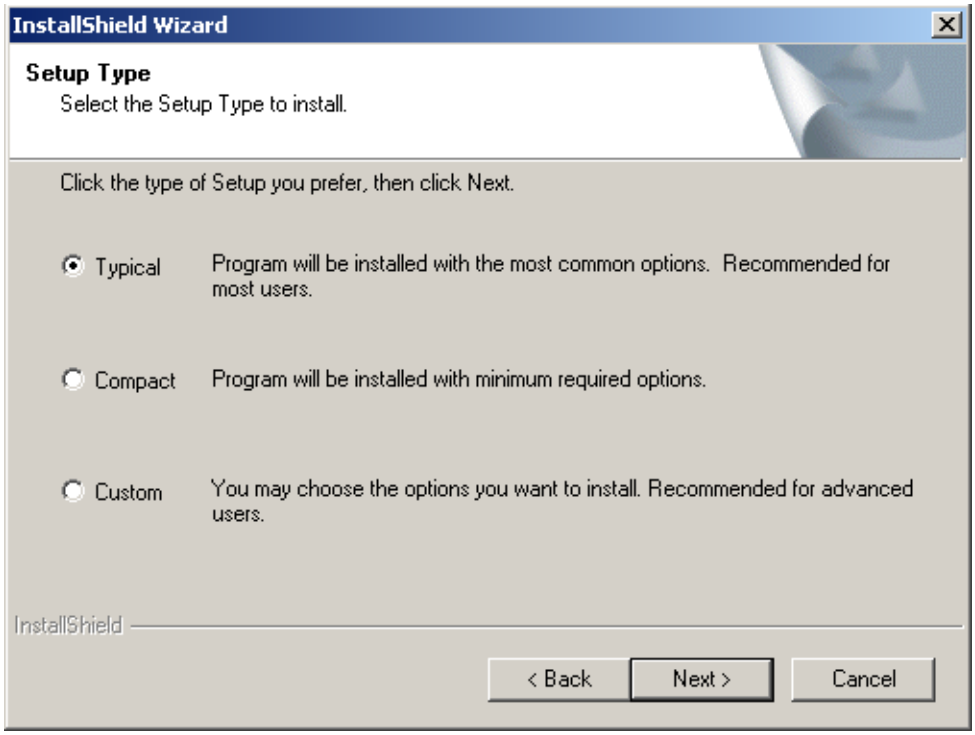
6. A User Name and Serial Number screen will appear. Fill in the appropriate information. Use the serial number obtained from the training class. Click *Next*.



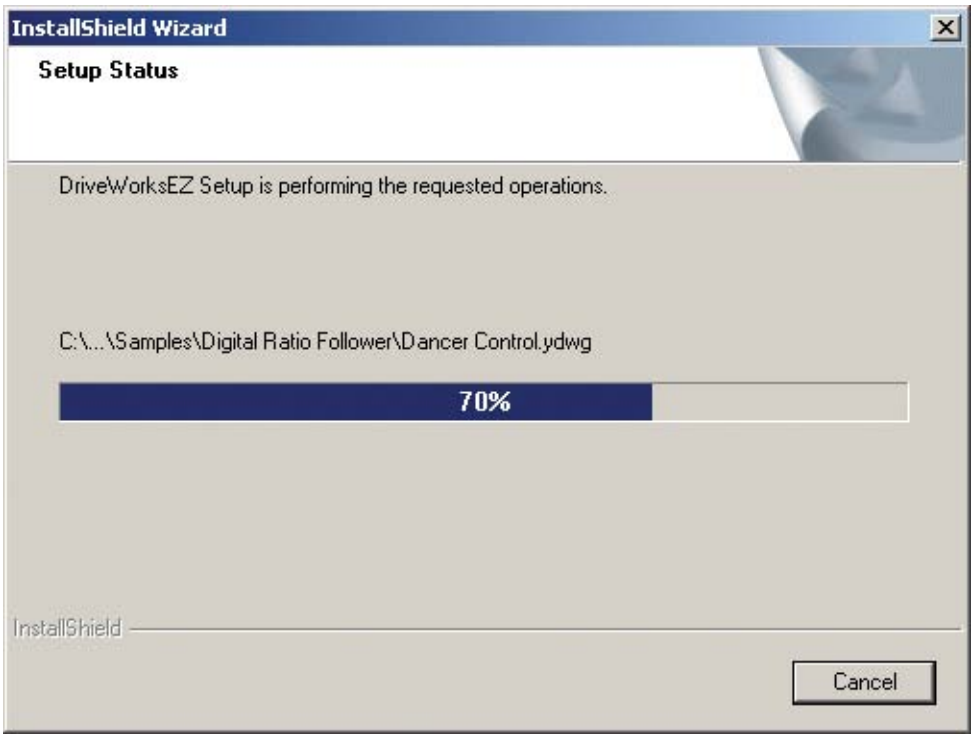
7. A License Agreement screen will appear. After reading the agreement, click *Yes*.



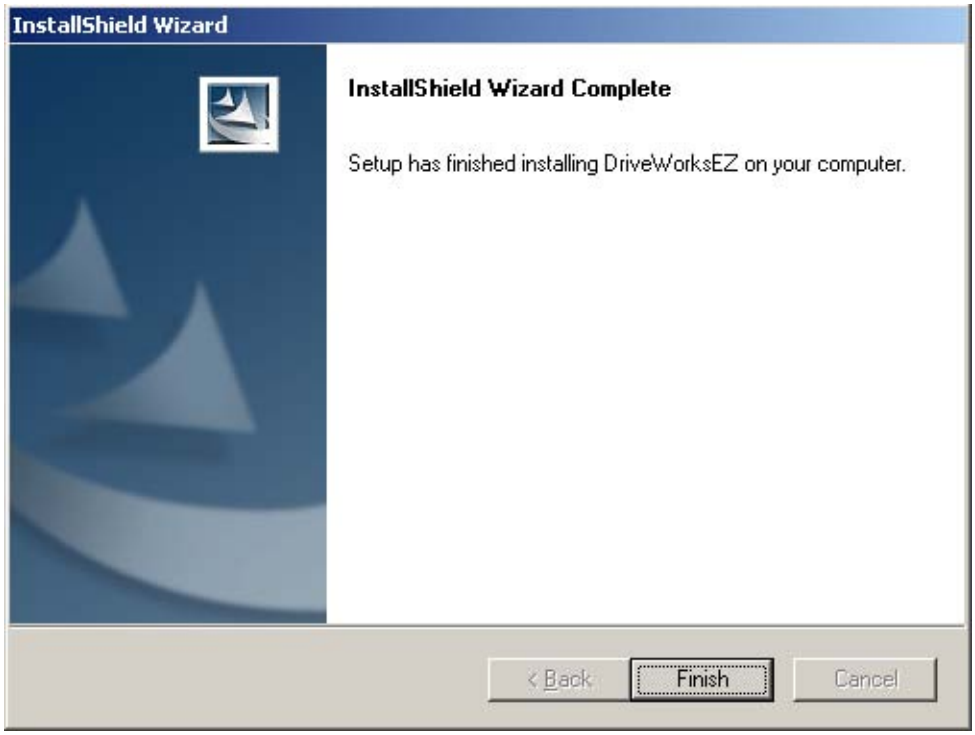
8. A Setup Type screen will appear. Select *Typical*, then click *Next*.



DriveWorksEZ™ will now be installed on the PC. A status bar indicates the progress of the installation. The default installation path is: C:\Program Files\Yaskawa\DriveWorksEZ.



9. Click on *Finish* to complete the installation process.



INFO

During installation, errors can occur, especially in laptop computers, that will prevent DriveWorksEZ™ from installing correctly. Two of the errors are **license.dat** file error and **–119 data move** error.

These errors are caused when the installation program cannot install some of the active-X components. The typical causes are:

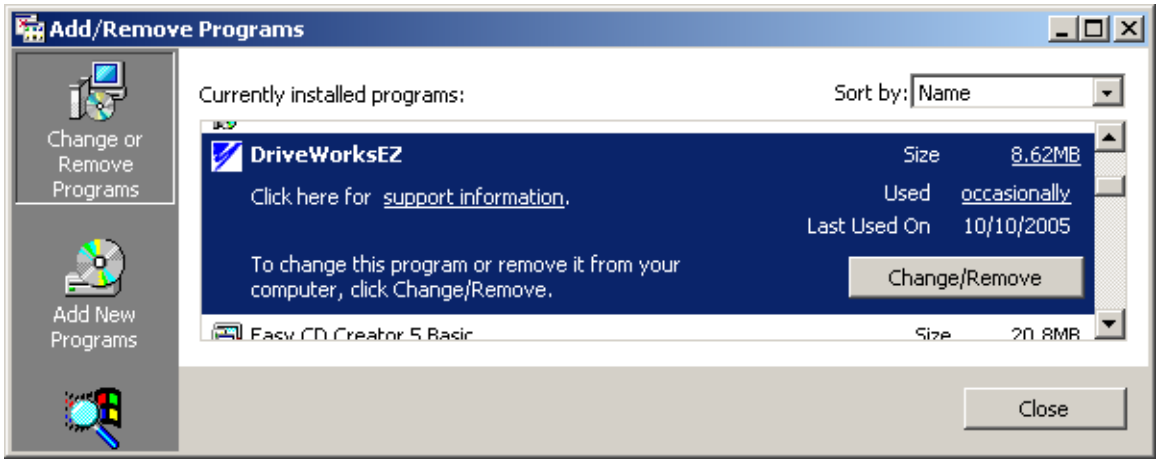
1. Other programs running that are interfering with the installation process. Check the task manager to be sure that no other programs are running. Check that other devices' support software have closed down their communication tasks and background programs especially if you have any other drive or PLC support products loaded on your PC.
2. Another program is using another version of one of the active-X programs used by DriveWorksEZ™, such as spread.ocx or strip.ocx.

Uninstalling the DriveWorksEZ™ PC Software

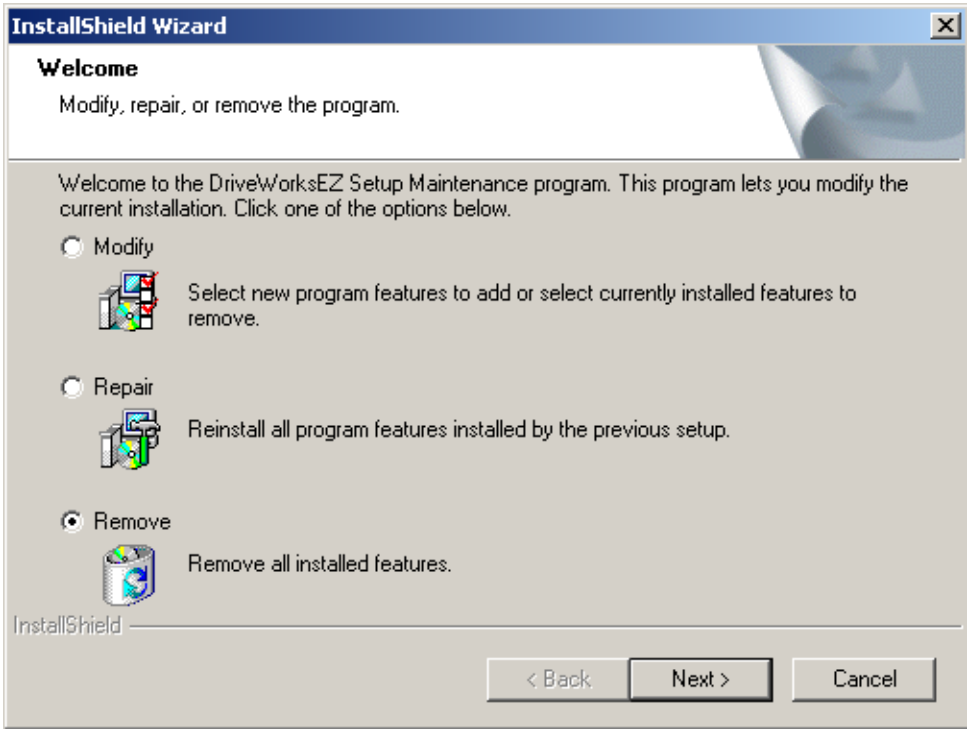
It is recommended that you uninstall a previous version of DriveWorksEZ™ before installing the newer version. To uninstall DriveWorksEZ™, follow the steps below.

1. From the Windows Start menu, select Control Panel. Double click on *Add or Remove Programs*.

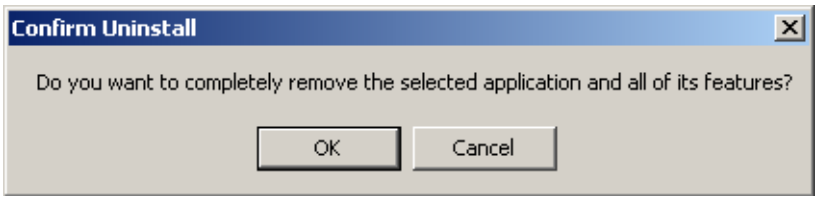
2. Select *DriveWorksEZ* from the list of programs.



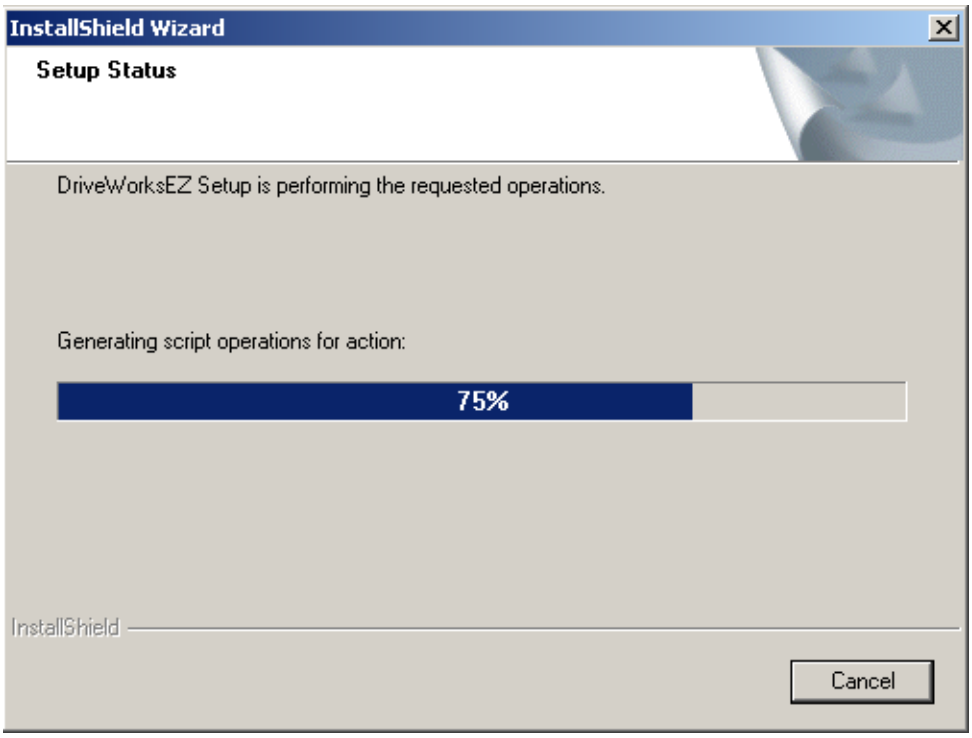
3. Click the *Change/Remove* button. The Welcome screen will appear. Select *Remove* to uninstall DriveWorksEZ™ from the PC. Click *Next* to continue.



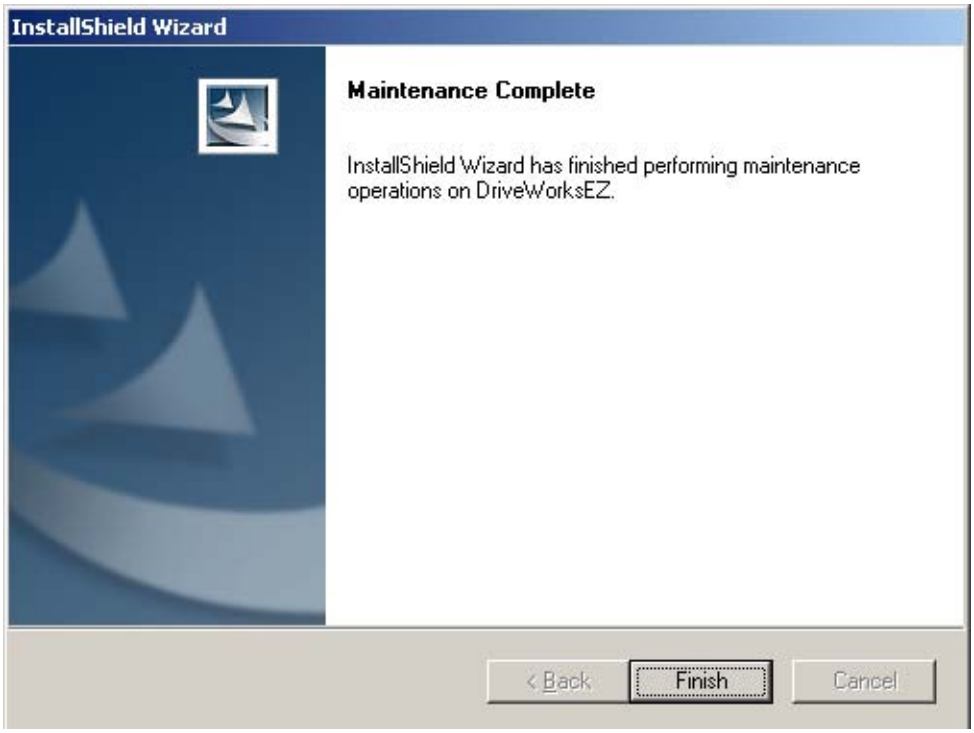
4. A warning message will appear asking you to confirm your selection. Click *OK* to continue.



5. The uninstall program will run and remove the DriveWorksEZ™ program. A status bar indicates the progress of the removal.



6. Click *Finish* to complete the uninstall process.



This page intentionally left blank.

2

Getting Started

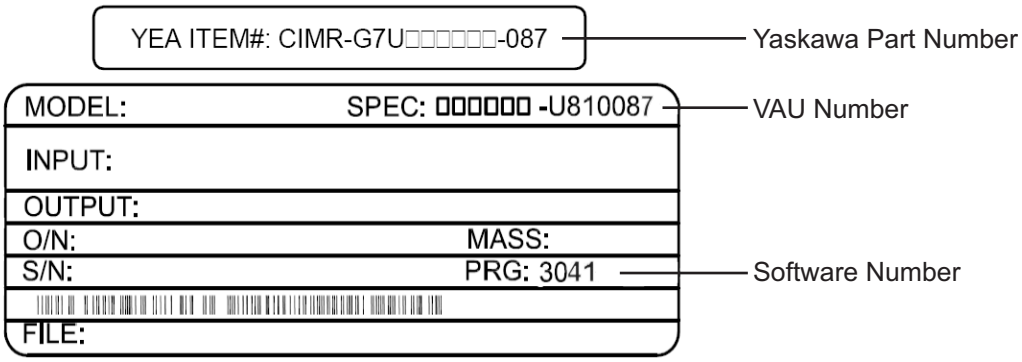
Drive Labelling for Custom Software

The drive will be modified in the following ways when DriveWorksEZ™ software is installed.

◆ Control Board

- ETC618058-S2015 is changed to ETC618058-S3040.
- The box label would be ETC618058-087 if the control board is ordered separately.

◆ Nameplate



- Additional sticker above nameplate: YEA ITEM#: CIMR-G7U*- 087
- SPEC with VAU number: 40P71B (example) changed to SPEC: 40P71B-U810087
- PRG: Number changed to PRG: 3041

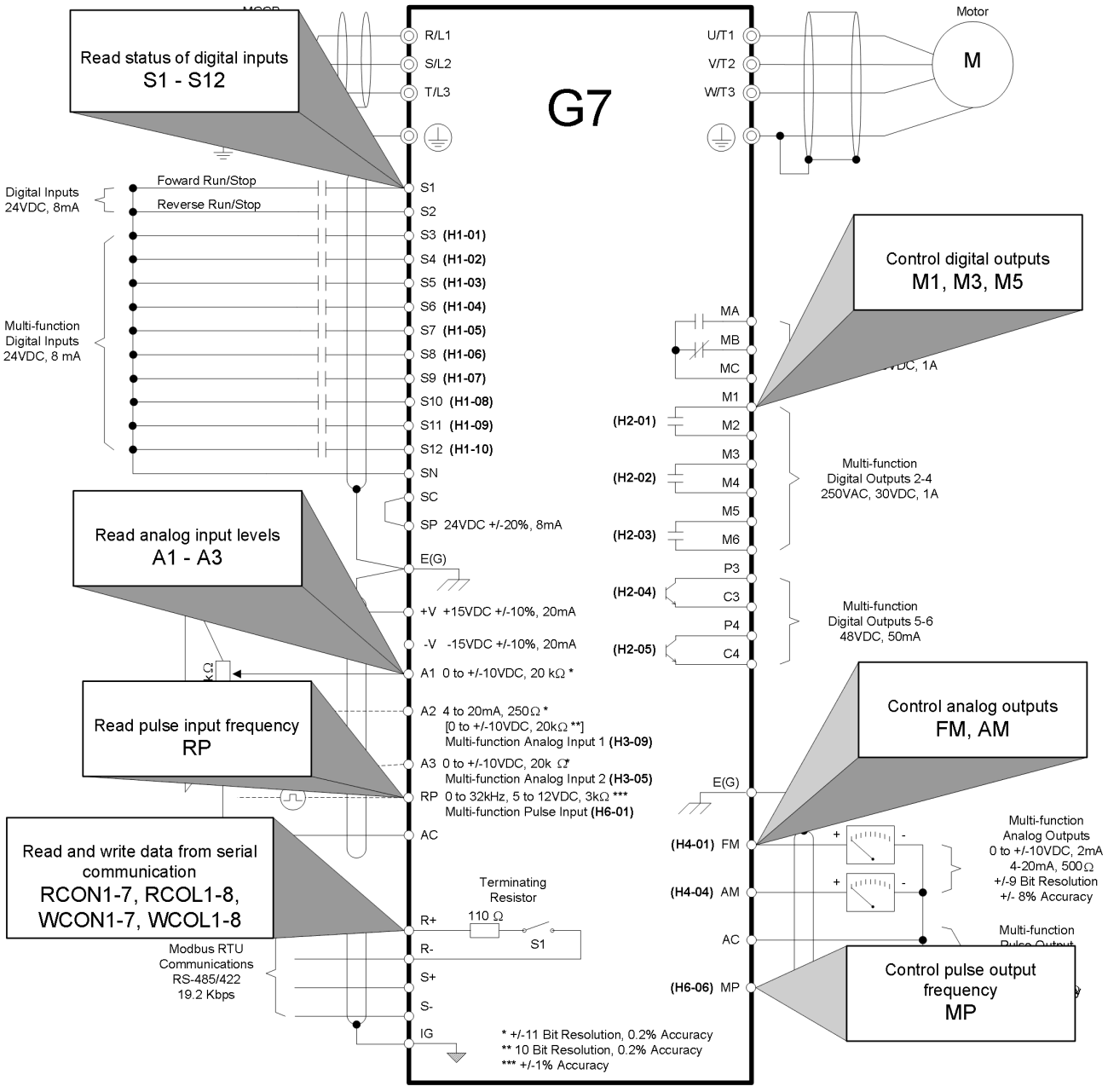
◆ Front Cover

- SPEC: 40P71B (example) changed to SPEC: 40P71B-U810087

◆ Packaging Box

- Yaskawa Item # CIMR-G7U*- 087
- SPEC: 40P71B (example) changed to SPEC: 40P71B-U810087

DriveWorksEZ™ Access Points for Physical I/O

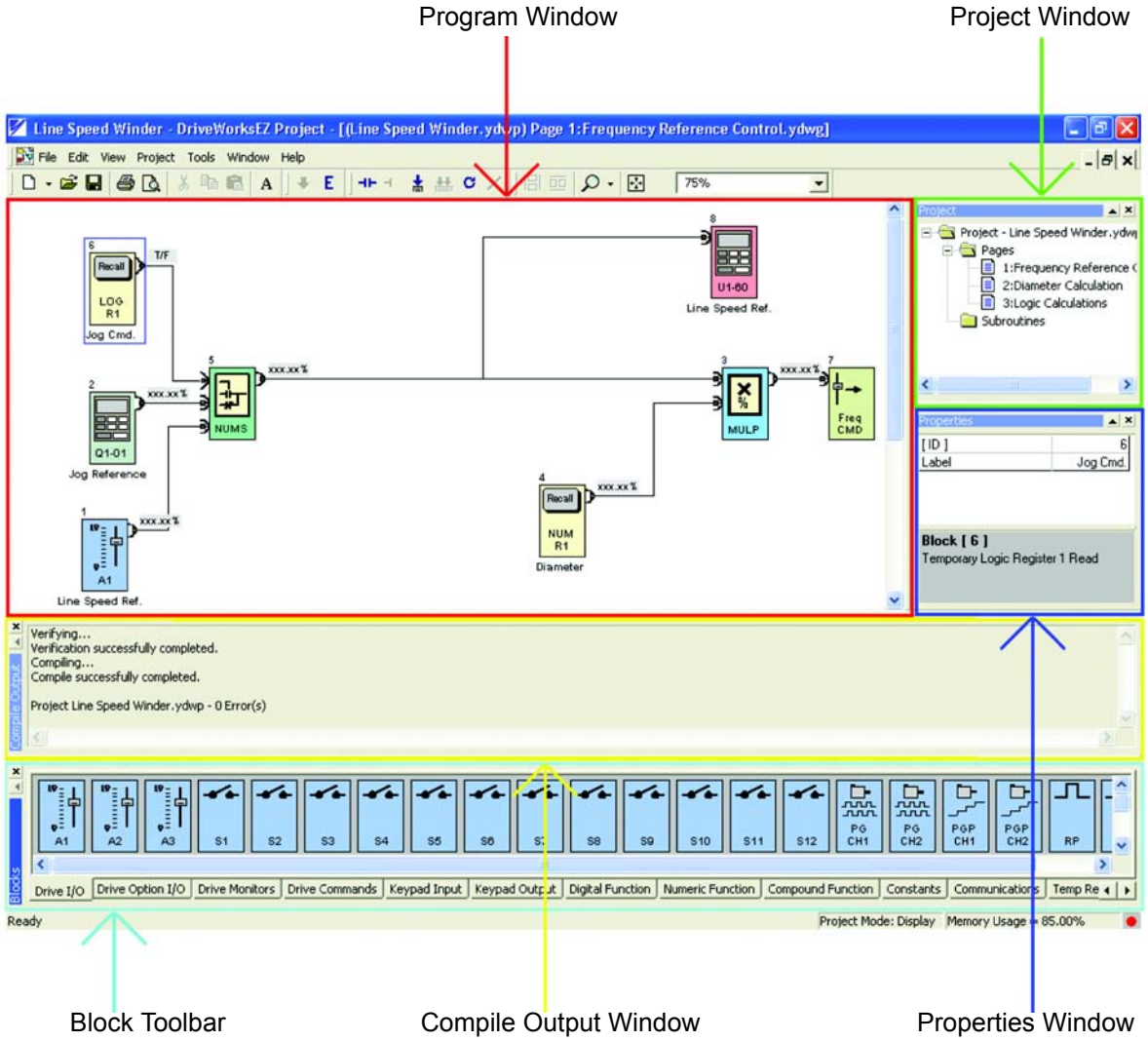


DriveWorksEZ™ Feature Overview

◆ DriveWorksEZ™ Programming Environment Features

- All windows can be resized.
- All windows except the *Program Window* can be moved around or hidden.
- The programming canvas is a virtually unlimited area that will be used to create the programs by dragging, dropping, and connecting blocks.
- New pages are added by using the *Project* pull down menu.
- Up to eight pages can be used in a single DriveWorksEZ™ project.

◆ DriveWorksEZ™ Navigation Window Overview



Program Window – Displays the graphical program layout. Blocks can be dragged from the *Block Toolbar* into the *Program Window*. Those blocks can then be connected together forming a program.

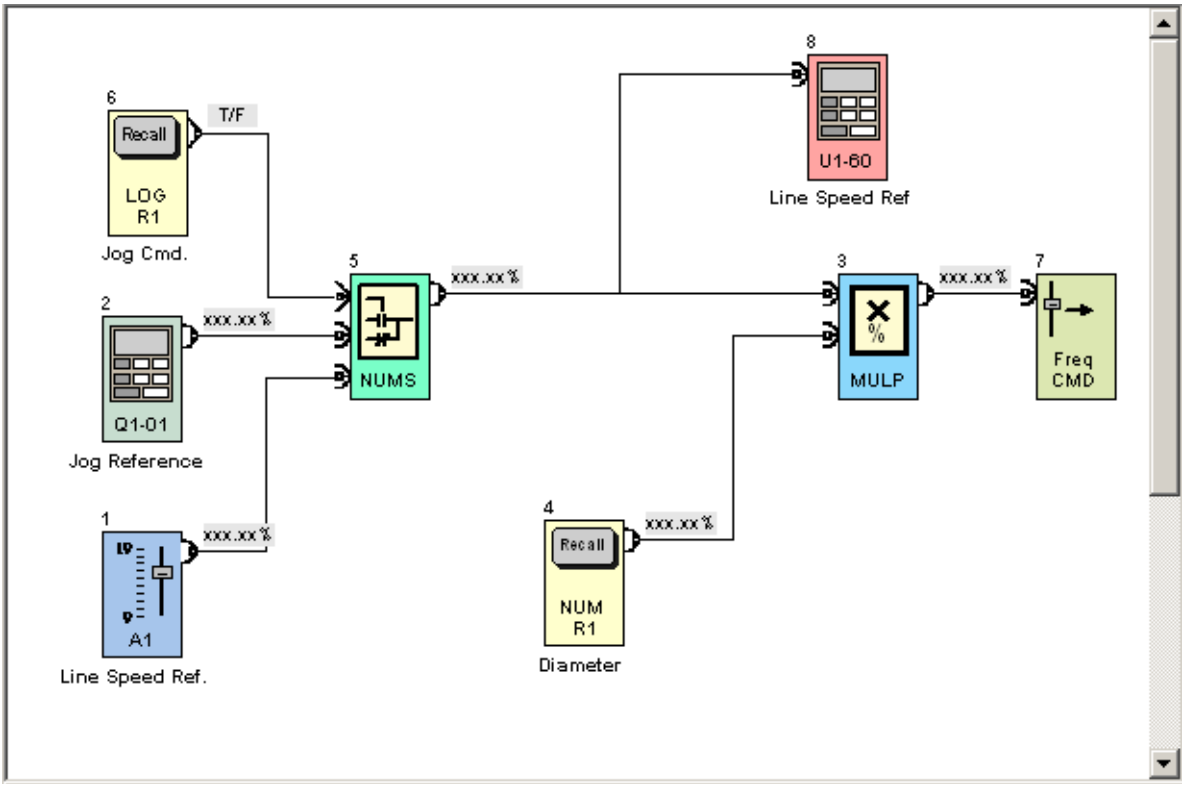
Project Window – Displays and allows editing of program information in a file manager-type style.

Properties Window – Used to view and edit block property and parameter information.

Block Toolbar – Used to drag and drop blocks on the *Program Window*.

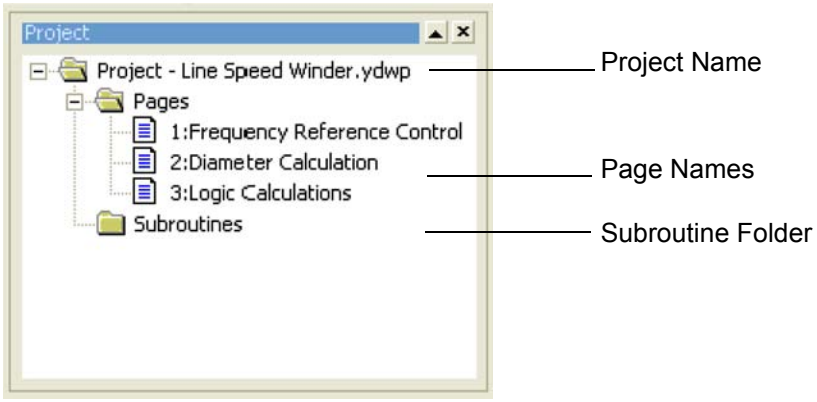
Compile Output Window – Used to display compiler status and results.

■Program Window



- The programming canvas is a virtually unlimited area that is used to create the programs by dragging, dropping, and connecting blocks.
- New pages are added by using the *Project* pull-down menu.
- Up to eight pages can be used in a single DriveWorksEZ™ project *Project Window*.

■Project Window



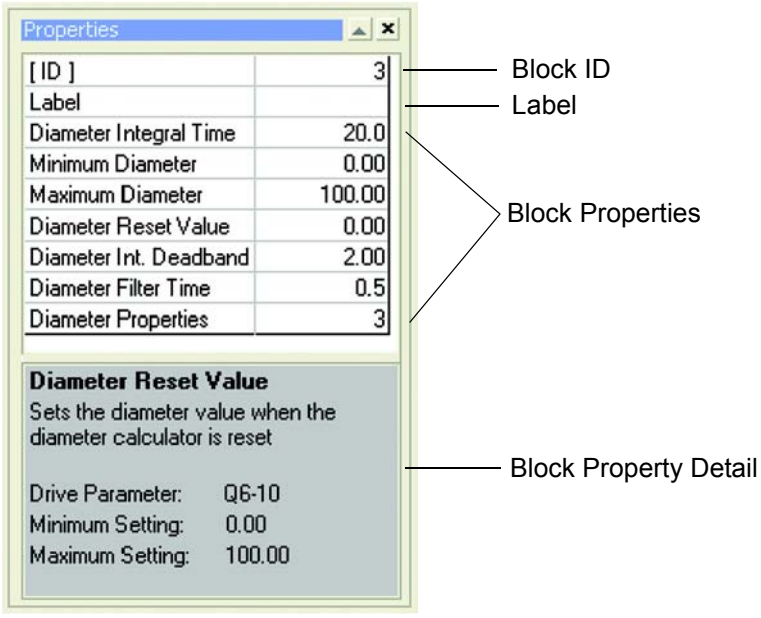
- Used as a quick navigation tool for the current project.
- Right-clicking the page and selecting *Print* can print individual pages.

Project Name – The name of the project. This field cannot be changed.

Page Names – The names of the pages. Page names can be changed by single-clicking on the name.

Subroutine Folder – Contains the subroutines associated with the project.

■Properties Window



- The central location for programming blocks, setting scope, and associated drive parameters.

Block ID – Every block has a unique ID number that is assigned as the block is dragged from the toolbar to the canvas. This ID number is displayed in brackets to the left of the block type and as a property in the block properties window. This is automatically assigned by DriveWorksEZ™.

Label – All blocks will have a Label property. The label will appear below the block on the canvas. The font of the label will change with respect to the block size, so that the label is always approximately the same width of the block. A label may be entered to describe what the block is or what it is doing.

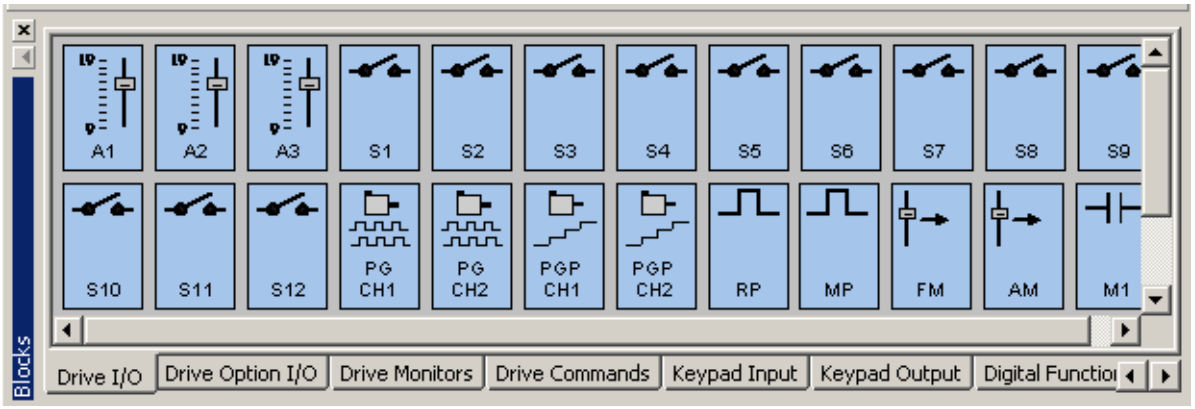
Block Properties – Contains a list with all of the properties (parameters) that affect the block. Clicking the name of the property will bring up the Block Property Details for that particular property.

Block Property Detail – When a property is clicked, the details of that property will appear here. Typically this will include a short description of the property, associated drive parameter number, minimum setting, and maximum setting. More detailed information for each block/property can be found in the help file and elsewhere in this manual.

INFO

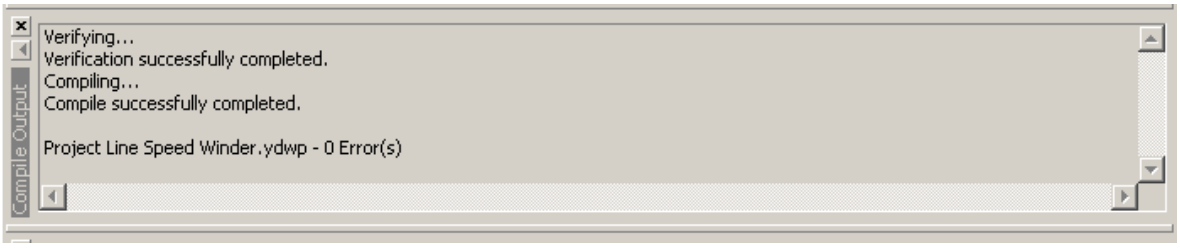
If the drive is connected and on-line, properties may be changed and then written to the drive by right-clicking the value and selecting *Write to Inverter*.

■Block Toolbar



- This tool bar contains all programming blocks represented by icons. The user will drag and drop icons on the canvas to create programs.
- A small description of the block will appear when the mouse pointer is over a block for more than one second.
- Blocks are organized into tabs of like types (Drive I/O, Drive Option I/O, Drive Monitors, etc.).
- The Block Toolbar can be viewed/hidden by the *View* > *Toolbars* menu or by right-clicking the main menu bar.

■Compile Output Window



- This form reports errors that result after compiling the project. The errors are listed line-by-line, in sentence format.
- This window automatically displays after starting the compile or can be opened by using the *View* > *Toolbars* menu.
- Each time the project is recompiled, the contents of the text box are replaced with new compile information. If the compile is successful "0 Error(s)" will be displayed. If there are errors in either the program or any of the subroutines, those errors will be displayed.

■Status Bar

The status bar is used to inform the user of operations that are taking place. This status bar is located on the very bottom of the screen, and can be added/removed by using the *View* menu.

Text Message

The text message on the far left side of the status bar will inform the user about the status of the application:

- Ready
- Compiling
- Connecting To Drive
- Switch To Edit

Monitoring Indicator

When actively monitoring a program with the PC tool, a small *heartbeat* indicator will appear in the status bar just to the left of the Project Mode indicator.

Project Mode Indicator

The Project Mode indicator provides further details as to the status of the application.

- Edit
- Display
- Monitor

Memory Usage Indicator

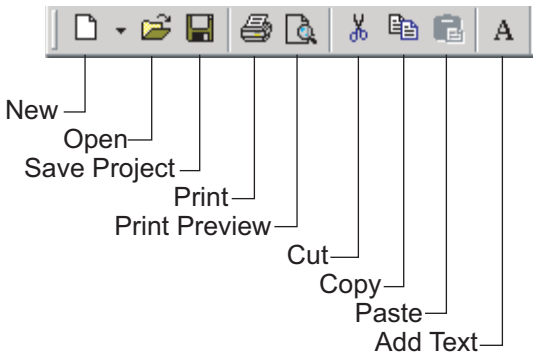
This indicates the number of connections used in the project. The numeric readout indicates the amount of connections used in percent. The value is calculated from the total number of connections.

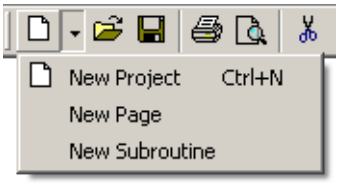
Simulated LED Display

An “LED” display in the lower right corner indicates whether the PC Tool is connected with a drive.

- Red = Off-line (not connected with a drive)
- Green = On-line (connected to a drive and communication with the drive is OK)

■Toolbar – Standard





New – Start a new project, a new page, or a new subroutine.

Open – Open up an existing project.

Save Project – Save project to hard disk.

Print – Print the block diagram or the report.

Print Preview – Bring up the block diagram on the screen as it will appear on the paper after it is printed.

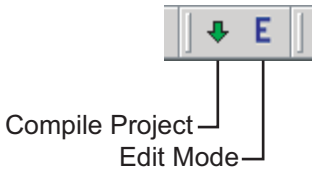
Cut – Remove the selected blocks and connections from the canvas and place them on the clipboard.

Copy – Copy the selected blocks and connections from the canvas to the clipboard.

Paste – Bring up the special paste cursor. Position the cursor in the desired spot on the canvas with the mouse, and left-click once to copy the blocks and connections from the clipboard back onto the canvas.

Add Text – Text can be added to the project page then moved anywhere on that page. Font style and size are fixed and cannot be changed.

■Toolbar – Compile



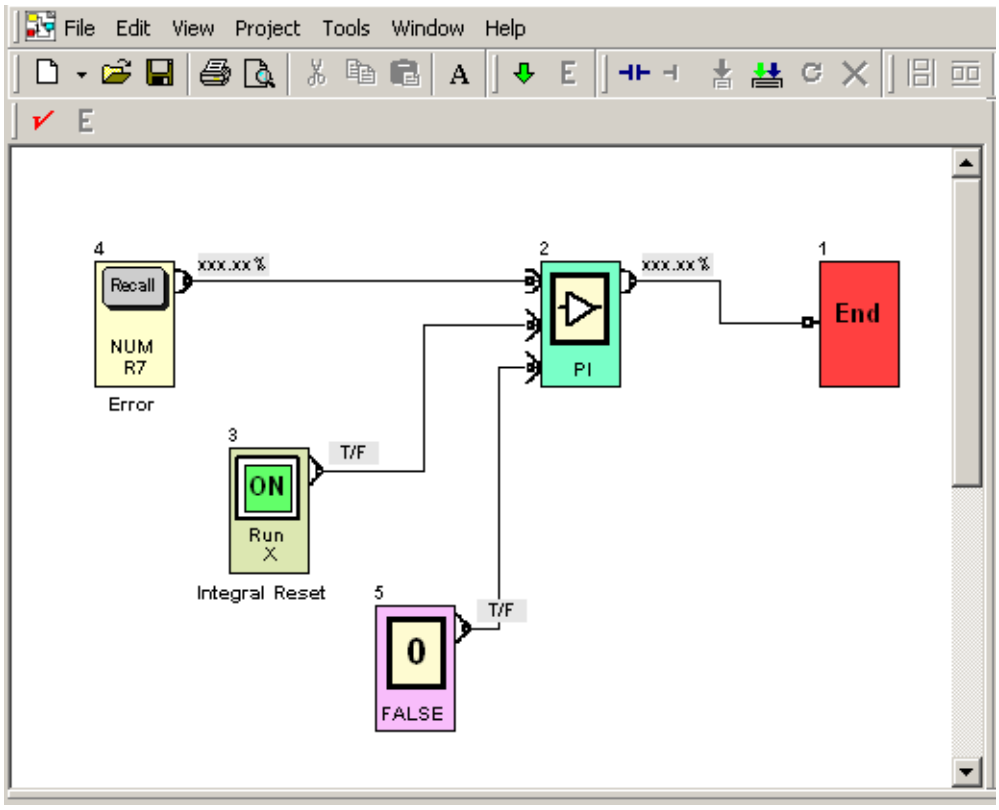
Compile Project Mode – Takes the pages of the open project and checks the connections for errors and then translates the project into a connection list.

Edit Project Mode – Allows the project to be further modified after a compilation.

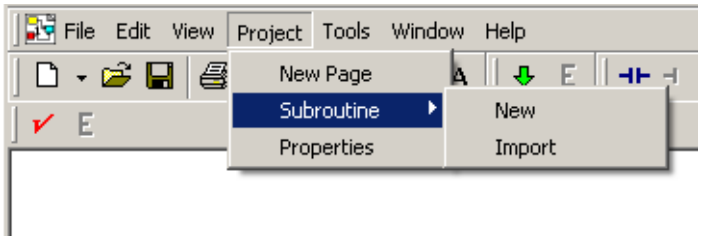
■Subroutine Function Blocks

A subroutine can be useful when it is necessary to repeatedly use the result of a calculation or logic sequence, particularly when that same result can be used in other DriveWorksEZ™ projects.

For example, a user may wish to create a subroutine to automatically reset the PI block's integral every time the drive loses its *Run* command. The user can create a subroutine that can, in effect, be pasted into future DriveWorksEZ™ projects.



Subroutines can be created and used within DriveWorksEZ™ projects as function blocks. Subroutines may also be copied from one DriveWorksEZ™ project to another.



A new subroutine can be created by selecting the *Project* pull-down menu and selecting *Subroutine* and then *New*. The name given to the subroutine will also be the name used for the subroutine file saved in the project folder.

Subroutine files use the extension *.ydws.

Once the subroutine is named, a subroutine page with that name will be displayed in the *Project Navigator* window. Clicking in the navigator on the subroutine page will display a subroutine page containing only the End block. The End block is the effective output that will be sent out of the subroutine block whenever it is used in the DriveWorksEZ™ Project.

All of the function blocks that can be used in the DriveWorksEZ™ project pages can be used in the subroutine. Create the subroutine using the function blocks and the next time the *Save File* function is implemented or the DriveWorksEZ™ project is compiled, the subroutine will be saved.

IMPORTANT

When a "Once Per Project" type of function block, such as the Diameter block, is used in a subroutine, that block cannot be used in the DriveWorksEZ™ project if the subroutine block is used in the project.

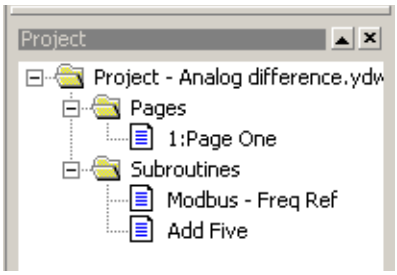
To use an already created subroutine within a project, import the subroutine from any other project that already contains the subroutine by selecting *Project* > *Subroutine* > *Import* and then searching for the *.ydws file.

IMPORTANT

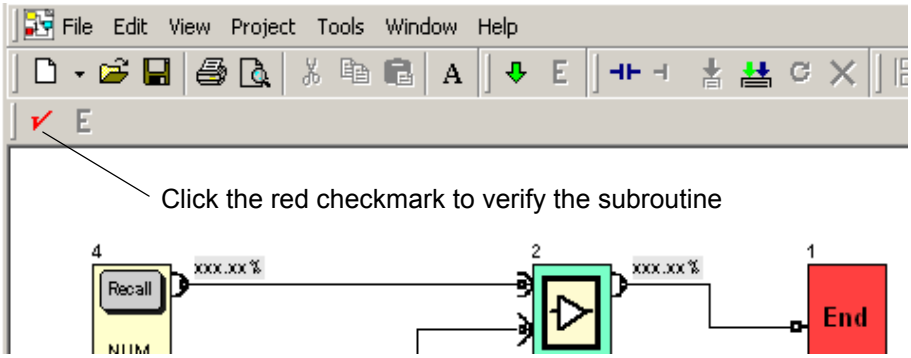
Whether a subroutine is created or imported, the number of connections used in the subroutine(s) will count towards the limit of 40 connections maximum per DriveWorksEZ™ project, but only if the subroutine block is used in a project page.

Once the subroutine has been created or imported, the DriveWorksEZ™ PC Tool must verify it before the subroutine block can be used in the project. When the DriveWorksEZ™ project that contains the subroutine is compiled, the subroutine is not checked for errors but must be individually verified.

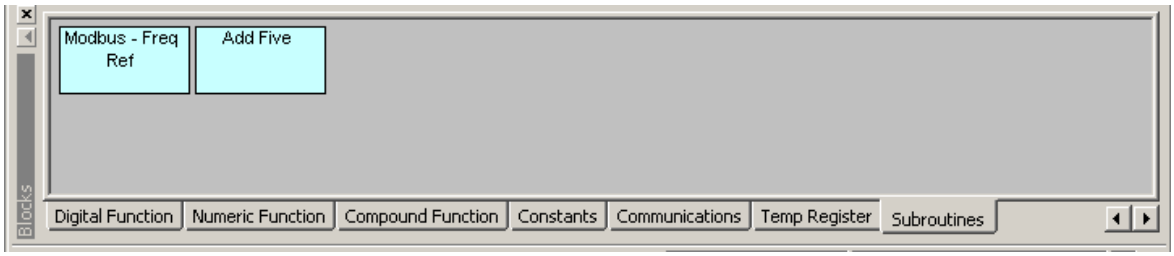
To verify a new or imported subroutine, open the subroutine page in the *Project* window of the PC Tool.



Once the subroutine page is open, the subroutine can be verified. Use the *RED checkmark* found between the *Toolbar* and the actual subroutine page to initiate the verification.



All subroutines created in a project or imported to the project are available for use in the project pages by dragging the subroutine blocks from the subroutine toolbar onto the page.

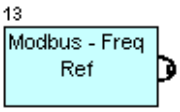


If the subroutine has not been verified within the current project as described above, attempting to drag the subroutine block onto the project page will result in the following warning.

Subroutine blocks will appear on the project page as aqua colored blocks with only a single numeric or logic output depending on which type of information the subroutine block will pass to the blocks connected to it.

Example:

A subroutine with a numeric output connection type.



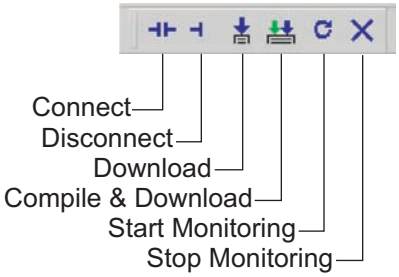
IMPORTANT

Only five subroutines may be used in a single project. More may be created or imported but only five may be used.

IMPORTANT

Subroutines may not be nested within other subroutines.

■Toolbar – Drive



Connect – Establishes communication with drive per DriveWorksEZ™ Properties settings provided the drive is set for DriveWorksEZ™ access.

Disconnect – Disconnects from drive and closes down communications.

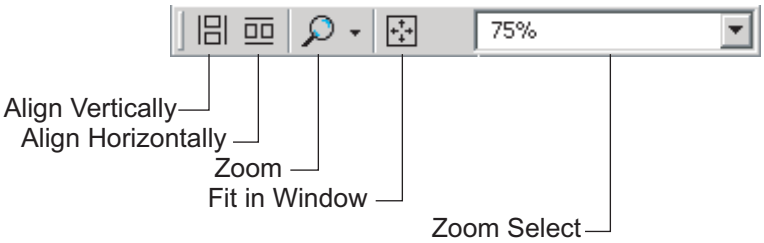
Download – Downloads all associated parameters and connection list.

Compile and Download – Translates the project into a connection list then downloads the list and all associated parameters.

Start Monitoring – Reads the value or status of each internal connection and displays them inside of the block diagram. This feature is very useful for troubleshooting and debugging.

Stop Monitoring – Stops all monitoring of the block diagram.

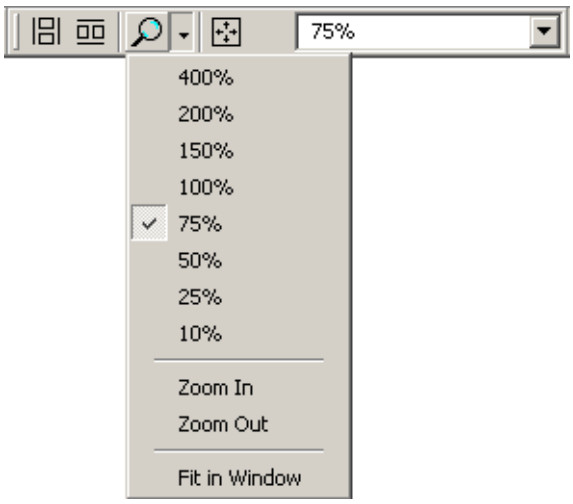
■Toolbar – Canvas



Align Vertically – This tool vertically aligns the highlighted blocks. It will only be enabled if more than one block is highlighted.

Align Horizontally – This tool horizontally aligns the highlighted blocks. It will only be enabled if more than one block is highlighted.

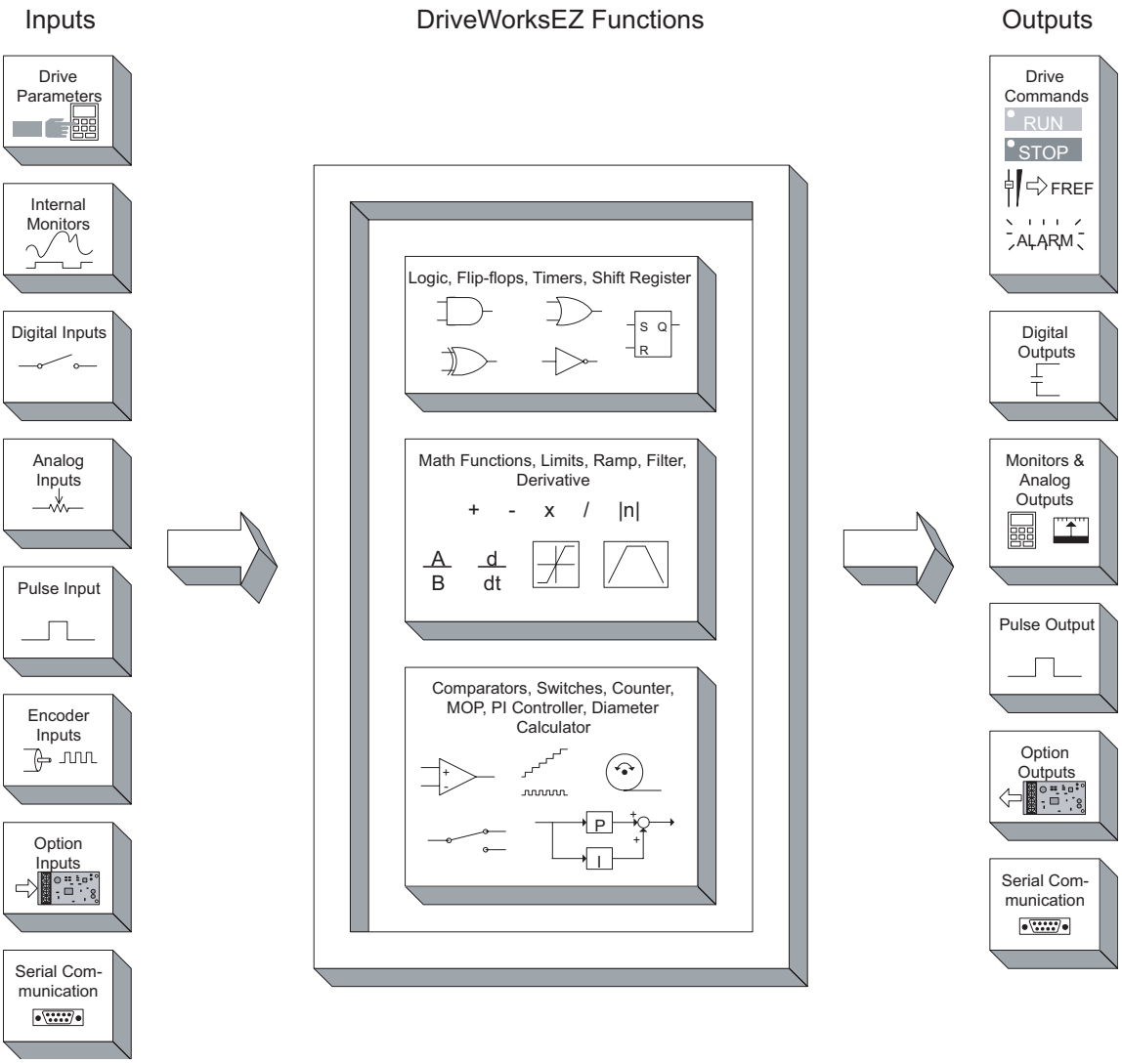
Zoom – Drop-down menu, which allows selection of multiple zoom levels, plus an additional selection of fit in window.



Fit In Window – Adjusts to the zoom level so all of the blocks will fit in the window.

Zoom Select – Drop-down menu, which allows an additional way of selecting the zoom level.

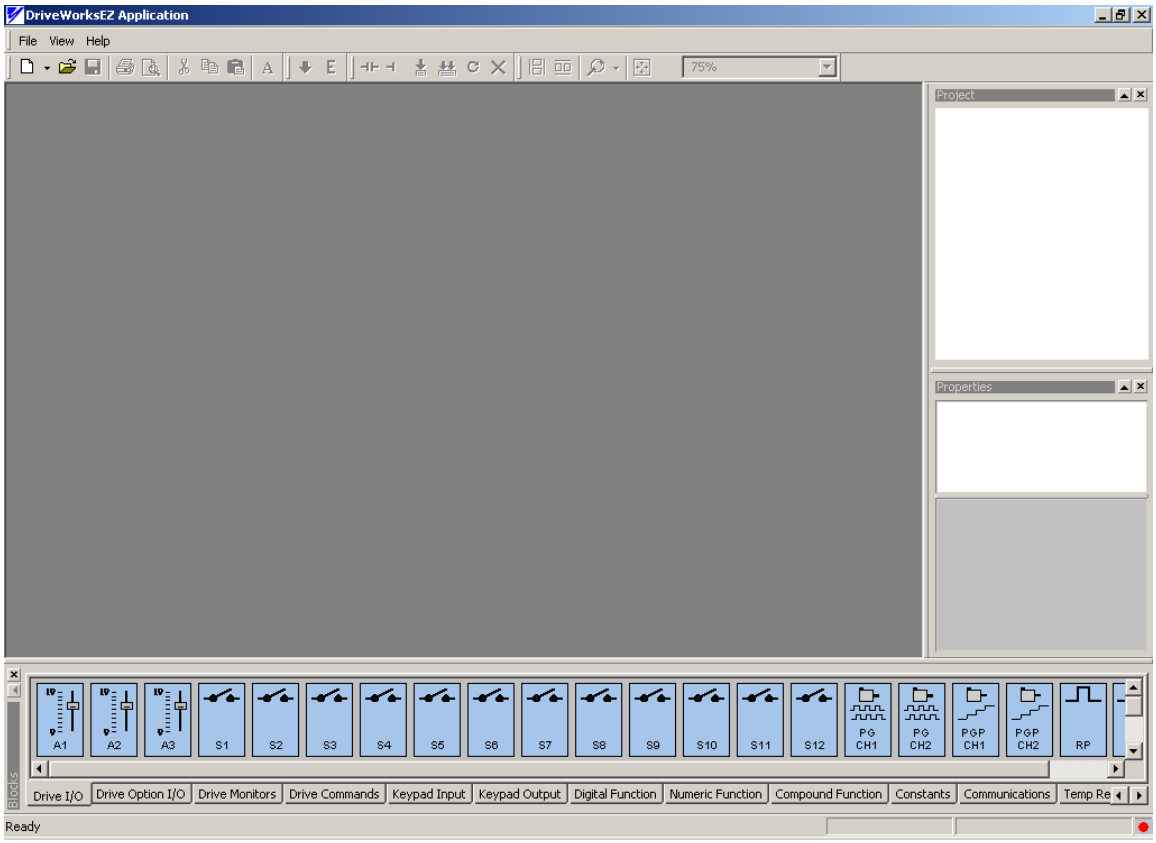
Overview of DriveWorksEZ™ Block Diagrams



Creating a New Project in DriveWorksEZ

◆ Step 1: Create a New Project

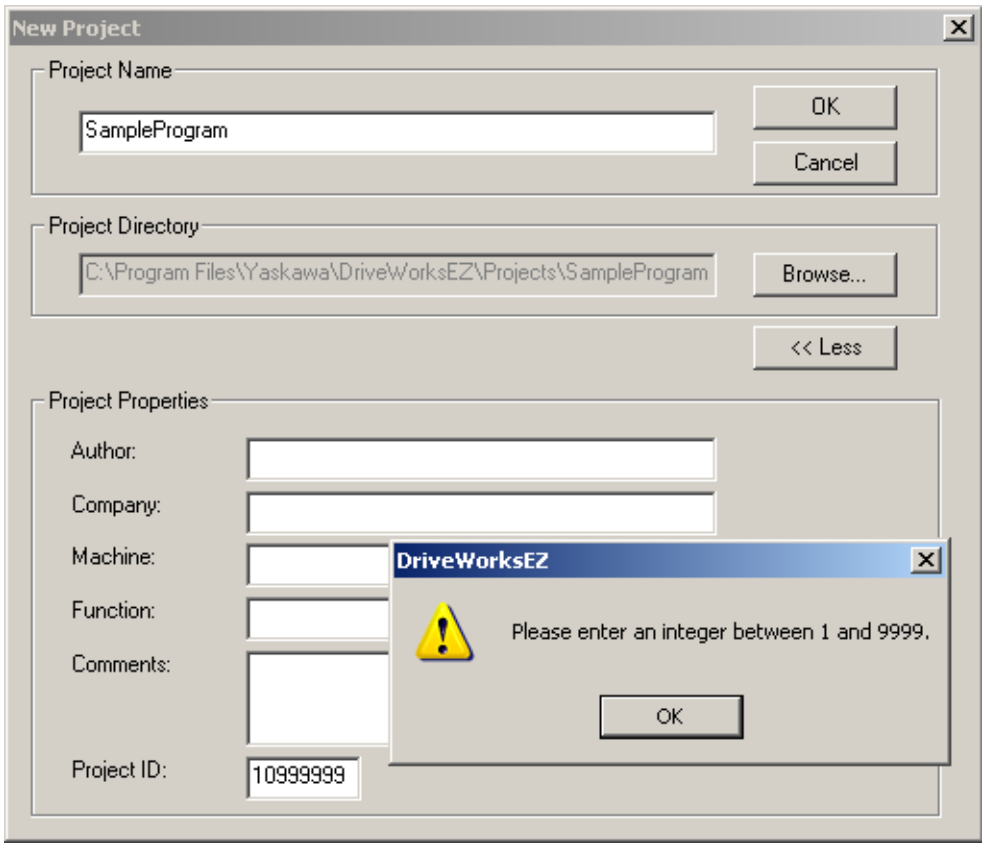
1. Launch the DriveWorksEZ™ application by accessing *Programs* > *Yaskawa* > *DriveWorksEZ* or click on the DriveWorksEZ desktop icon.
2. The application should launch.



3. To create a new project, click on *File* > *New Project*. A new project window will open up. Type in "SampleProgram" in the entry window as shown below.

INFO

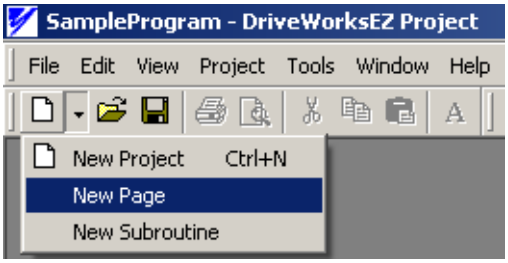
The *Project ID* field will only accept integer values between 1–9999.

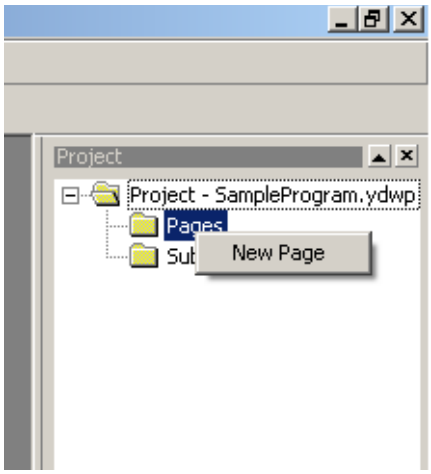


4. Click on *OK* to accept the entry.

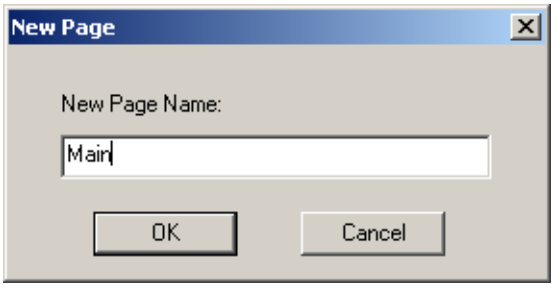
◆ Step 2: Create a New Page

1. Click on the *New* toolbar icon and select *New Page*, or right-click on the page folder in the Project Navigator window.

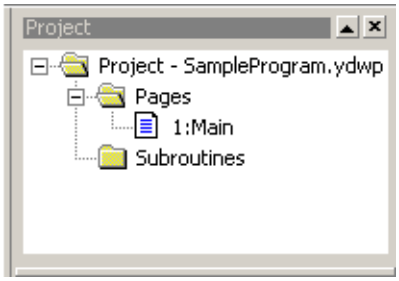




2. Type “Main” in the New Page Name window that appears. Click on *OK*.



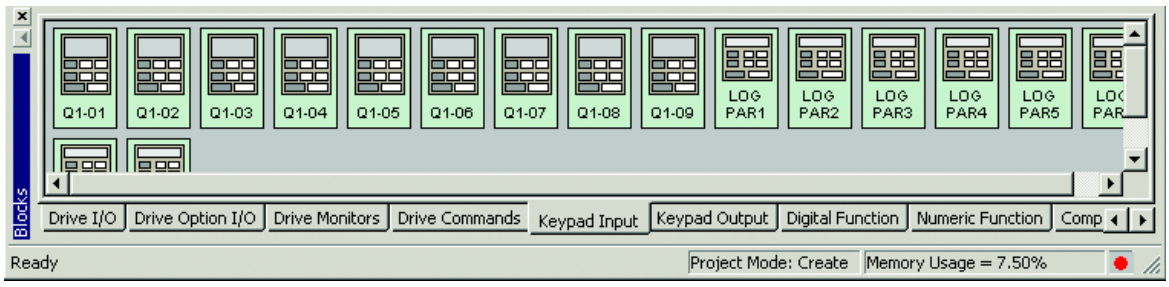
A page called Main is now in the *Project Navigator*. Notice that the dark gray blank area is now a window with a white background.



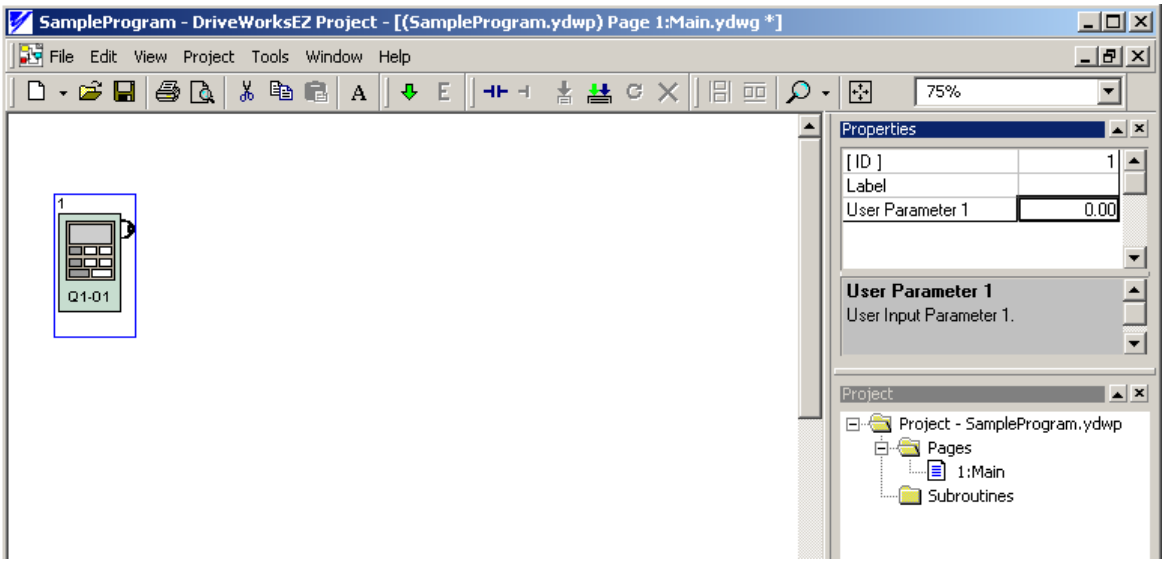
◆ Step 3: Create the Program

The sample program to be created takes the sum of two keypad inputs, divides it by a third keypad input, and displays the result on the Keypad monitor U1-60.

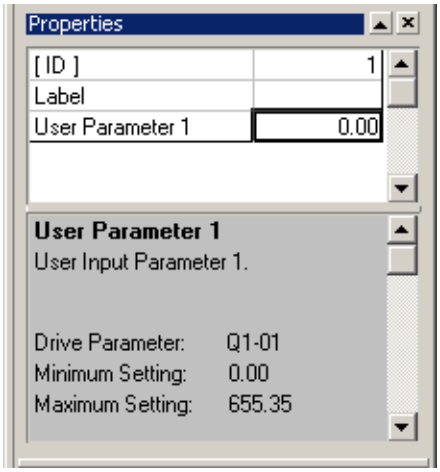
1. On the *Block Toolbar*, click the *Keypad Input* tab.



2. Click on the Q1-01 block and drag it onto the page (white area) while holding down the left mouse button. The block will have a gray box around it indicating it has been selected. Drag it to the top left-hand corner of the Page window.

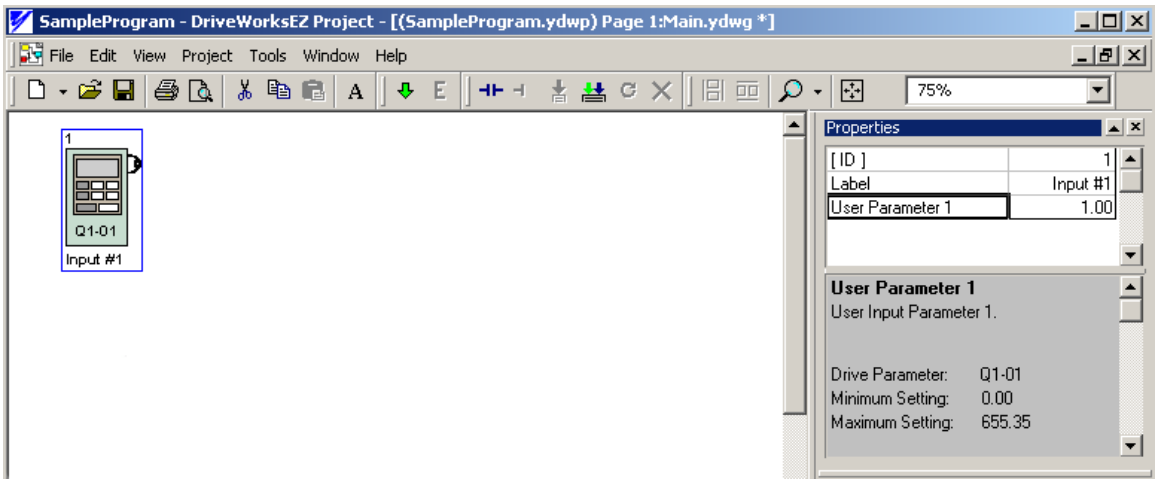


3. The properties of this block can now be edited. In the *Properties* window is displayed a spreadsheet of information. One cell is called *Label* and is used to label the selected block. The second cell is the actual parameter Q1-01 value. Click on the 0.00 in the *User Parameter* cell to see the properties of the parameter in the lower portion of the *Properties* window. This holds true for any block with parameters.

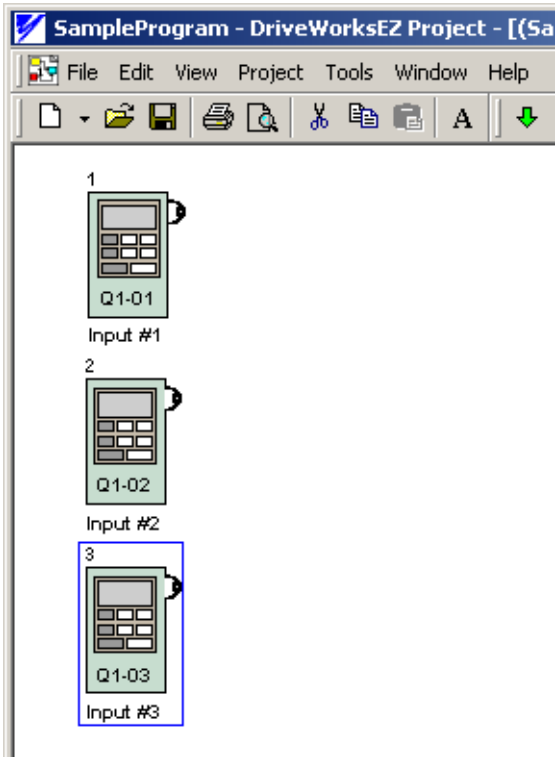


4. Enter 1 in Q1-01 by clicking on the cell that has 0.00 and type in the number *1* in the cell and then press *Enter*.

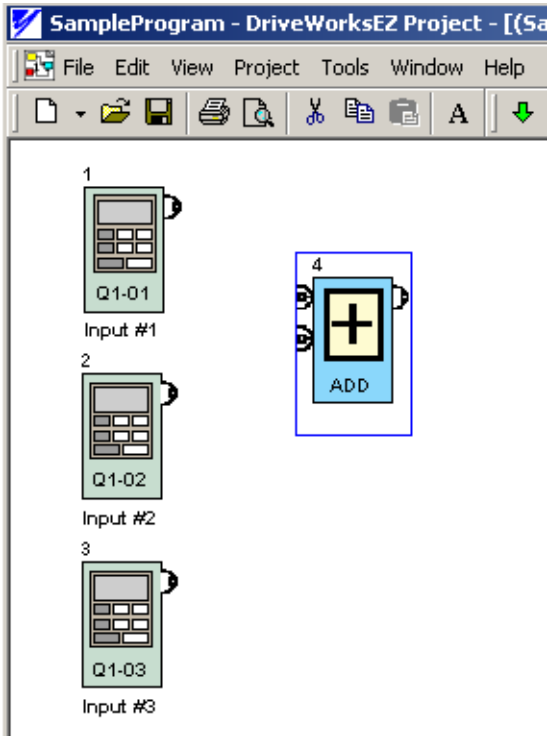
5. Click on the cell to the right of the label and enter “Input #1.” Notice that when you click on the *Label* cell, the function of that cell is shown in the block property detail area below it



6. Drag Q1-02 onto the page below Q1-01. Label it “Input #2” and set the parameter value to 3.00.
7. Drag Q1-03 onto the page below Q1-02. Label it “Input #3” and set the parameter value to 2.00.

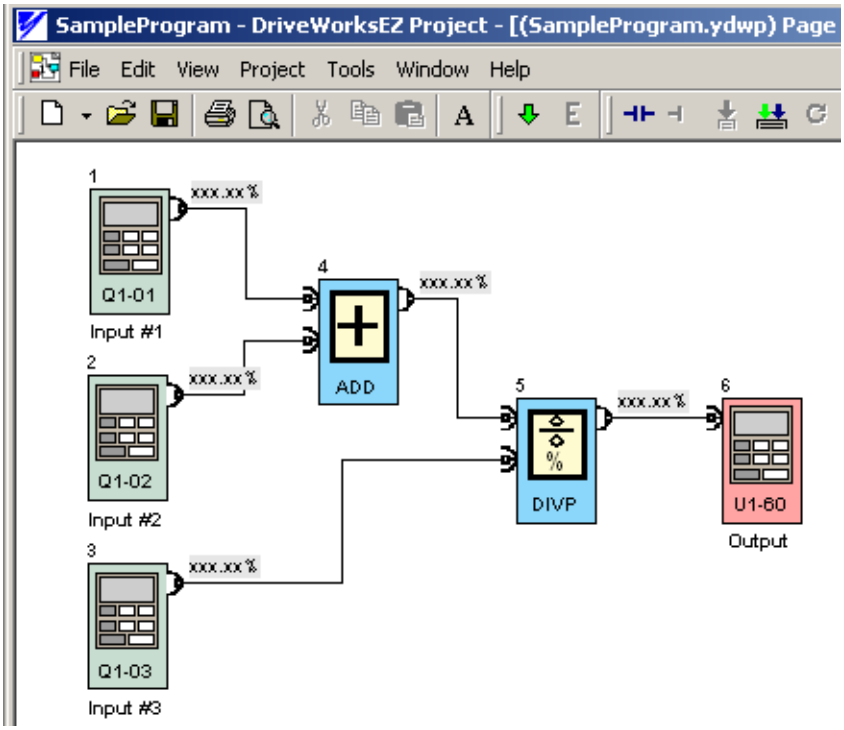


8. Add the math functions to the program. Select the *Numeric Function* tab on the *Block Toolbar* and drag the ADD block onto the page to the right of the Q1-01 and Q1-02 blocks.



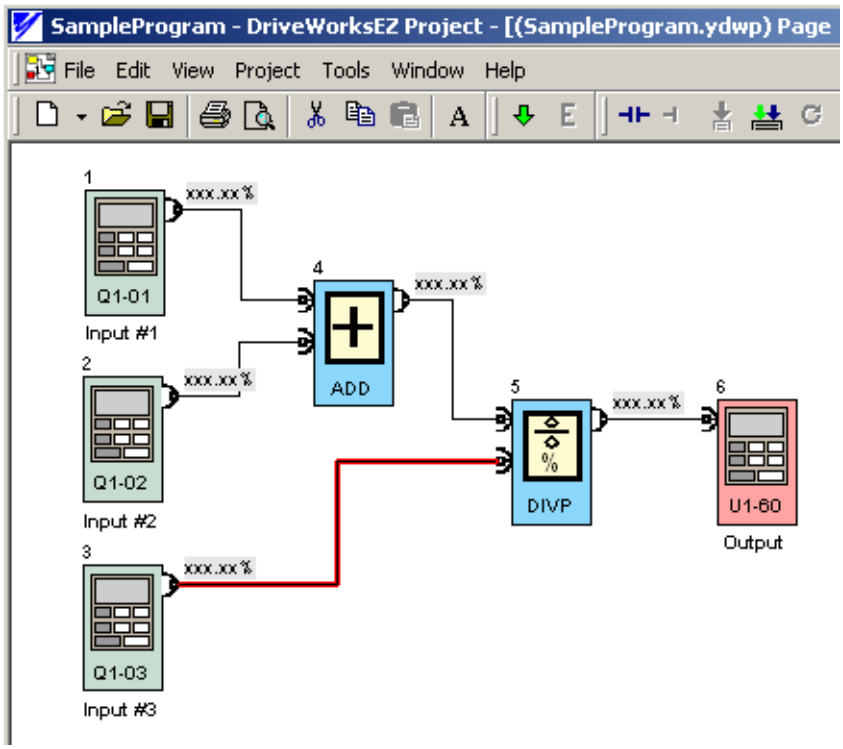
9. To make the first connection, position the cursor over the numeric output node of the Q1-01 block. Click and hold the left mouse button and move the cursor towards the numeric input upper node of the ADD block. A short line will be noticed on the output node indicating that the PC Tool is in the line draw mode. The PC Tool will automatically finish the connection, so the left mouse button can be released. A place holder for the monitor value will display the current value of that connection.
10. Connect the Q1-02 block to the lower port of the ADD block.
11. Select the DIVP block and position it to the right of the ADD block.
12. Connect the output of the ADD block to the upper input node of the DIVP block and connect the Q1-03 block to the lower node.

13. From the *Keypad Output* tab on the *Block Toolbar*, select and drag the *U1-60 block* onto the page. Label it "Output." Connect it to the output of the DIVP block. The program is now complete.



INFO

Mistaken connections can easily be deleted. Position your cursor over the connection line you wish to delete. Click on the line and it will turn red. Use the *Delete* key to remove the line.

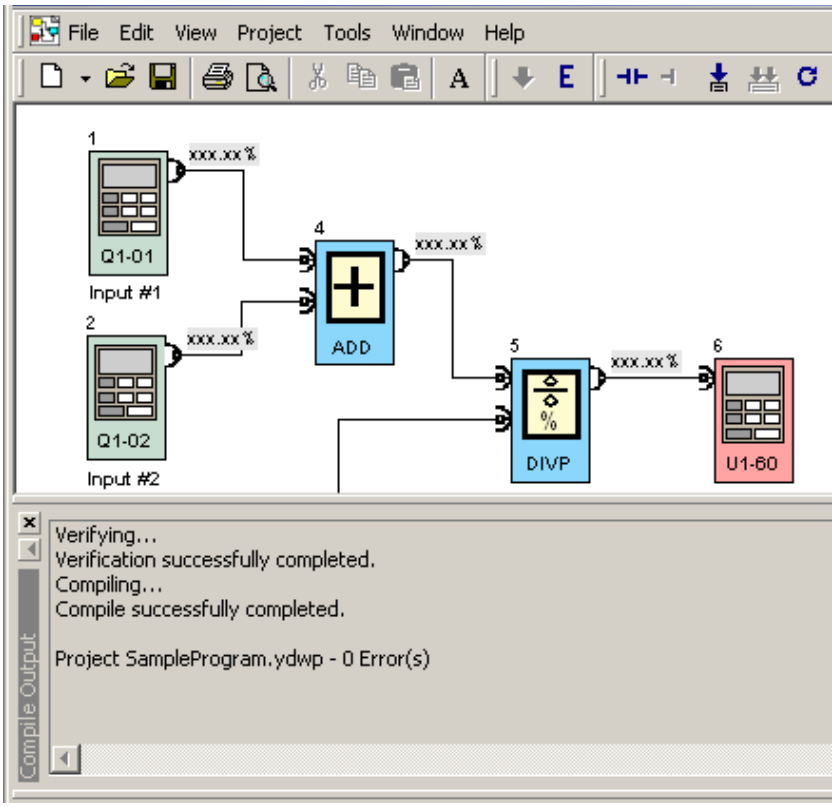


◆ Step 4: Save the Project

To save the project, use the *File* > *Save Project* function. Always save before compiling the project as a good practice.

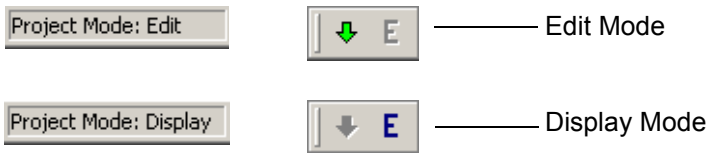
◆ Step 5: Compile the Project

To compile the project, use the *Tools* > *Compile* function. When starting the compile, the compiler window will appear at the bottom of the display. It will show the status of the compile. The compiler checks that all of the proper connections have been made and that all of the programming rules are followed.



IMPORTANT

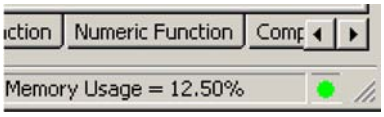
After a project is compiled, the editing feature of the PC tool is disabled. This is illustrated by a greyed out compile icon on the toolbar and the status bar indicating "Project Mode: Display."



◆ Step 6: Go Online and Download the Project

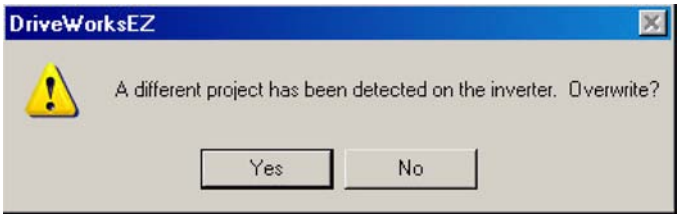
1. The drive must be in the DWEZ mode before connecting and downloading the project. Go to parameter A1-01 and set it to "423 DWEZ opr access."
2. Connect the PC Tool to the drive cable in the proper manner. See ***Connecting the Drive to the PC***.
3. Enable the connect function to go online with the drive. The status bar on the left side will show that the PC tool connection function is in process.

When the connection process is complete, the red dot on the right side of the status bar will turn green.



4. To download the project, select the download function. The status bar will show the progress of the download. The download goes through several phases and it may take some time to complete.

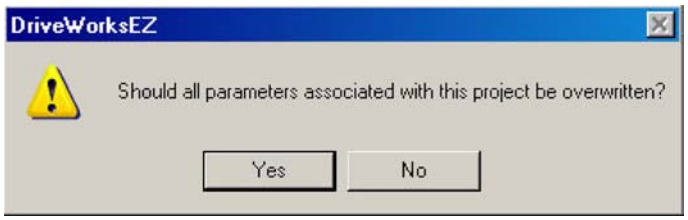
   If another project has been downloaded to the drive, a message will appear asking if it is OK to overwrite. In most cases you can click *Yes* to continue.



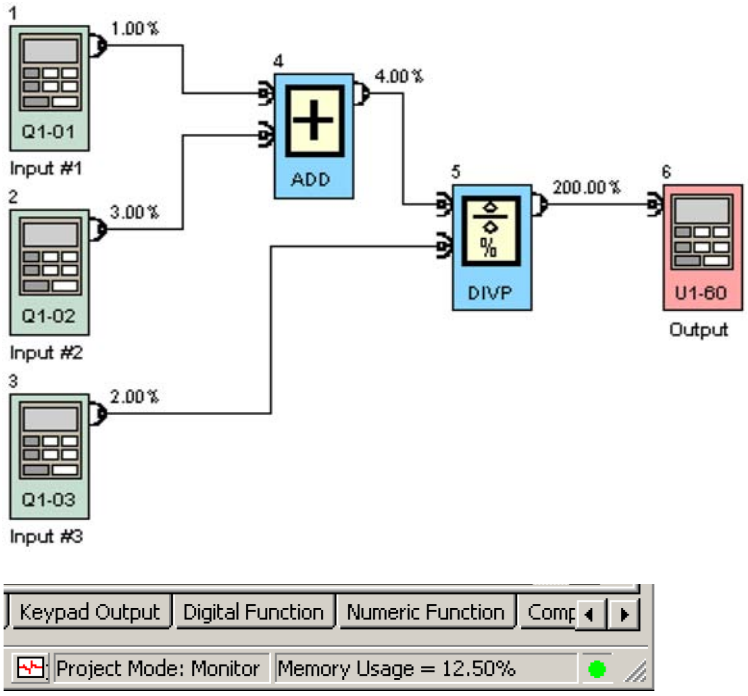
When the download is finished, the status bar on the left hand side will show Ready.

◆ Step 7: Monitor the Project.

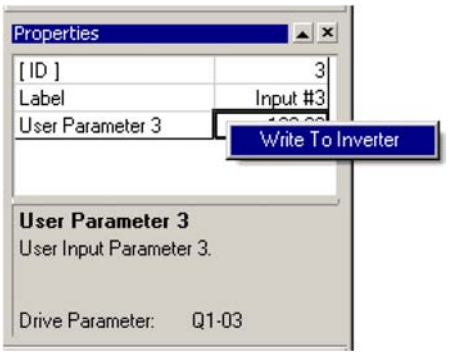
1. To monitor a project, there should be matching versions of the program on the PC tool and the drive. If this is not so, the PC tool will detect this when starting the monitor and ask you to compile and download a fresh version of the project.
2. Start the monitor by selecting *Tools* > *Start Monitoring*. The PC tool will prompt to overwrite parameters in the drive. Click *Yes* to accept, and the monitoring process will start.



When the monitor is running, the placeholder monitor values will change to the actual drive values and the monitor icon will appear on the right side of the status bar



3. To change a value and see the results, select the Q1-03 Input #3 block on the page. Go to the *Properties* window and click User Parameter #3 data field, which now reads 2.00. Change the value to 100.00% by double clicking on the cell and typing in "100" and press *Enter*. To write the new value to the drive, right-click on the cell and left-click on *Write to Inverter*.



4. The value of the U1-60 block should now be 4.00%.
5. Check the program with the drive keypad by stopping the Monitor and going off line before changing the PC tool cable to the drive keypad.

Connecting the Drive to the PC

To connect the G7 drive to the PC, follow the procedure outlined below:

1. Remove the operator keypad.
2. Connect cable P/N UWR00468-2 to the connector for the drive keypad (1CN).
3. Connect the 9-pin D-sub-miniature to the PC serial port.

IMPORTANT

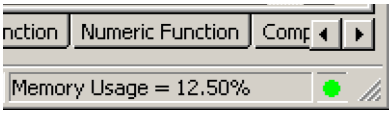
Before starting the online process, be sure to check the following:

1. The PC is connected to the drive correctly.
2. There are no other programs running on the PC that could interfere with the PC serial port.
3. The communications settings for DriveWorksEZ™ match the drive settings.
4. Open either a new project or an existing project.

To start the online process, perform the following operations:

1. Open a valid project already saved or create a new project.
2. Go to the Main Menu, click on *Tools* > *Connect* or push the button on the toolbar.

The PC tool will then connect to the drive. The red dot on the status bar will turn green indicating that the PC tool is on line with the G7 drive.



IMPORTANT

If a communication error window appears during the process of going on line, check the following:

1. The communication cable between the drive and the PC is the correct type and is installed correctly.
2. The drive is powered up.
3. The communication parameters in DriveWorksEZ™ are set correctly.

INFO

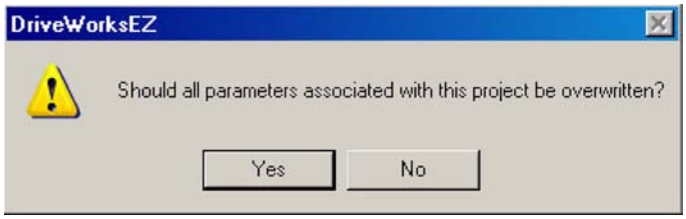
Ethernet communication may also be used to connect the PC to the G7 drive with either the CM090 (Modbus TCP/IP) or CM092 (EtherNet/IP) option kit.

DriveWorksEZ™ and Drive Initialization

The procedures below detail the recommended way of correcting a situation where a DriveWorksEZ™ program is loaded into the drive but the standard drive parameters have been set back to their default settings by a subsequent drive initialization.

If it is necessary to initialize a drive loaded with a DriveWorksEZ™ program, the following is suggested prior to operation of the drive:

- Use the drive digital operator to reset the Parameter Access Level parameter to DriveWorksEZ™ access (A1-01=423).
- Reconnect to the drive with a computer that has the DriveWorksEZ™ PC Tool and the DriveWorksEZ™ project files. Download the project to the drive or attempt to enter the Monitor mode. In either case, the PC Tool will recognize the fact the parameters do not match and will query if the user wishes to overwrite the changed parameters.



Another solution to prevent incorrect operation of the DriveWorksEZ™ programming due to a drive initialization is to save the settings of the standard parameters with a User Default save (o2-03=1). See the *Precautions* section after the first successful operation of the DriveWorksEZ™ program.

Once a User Default save has been performed, the drive can be set back to the proper DriveWorksEZ™ setting by doing a User Initialization (A1-01=1110).

If a drive loaded with a DWEZ program is initialized via parameter A1-03, the DriveWorksEZ™ program might not operate correctly.

Initializing the drive will set all standard drive parameters (A1-01 through o3-03) back to their default settings while not affecting the DriveWorksEZ™ parameters (the Q parameters).

This situation occurs because:

- The Parameter Access Level parameter (A1-01) will be changed from DWEZ Access (A1-01=423) back to Advanced Access (A1-01=2), which will disable the DWEZ program.
- The DWEZ sets any drive I/O used by the program to NOT USED in order to avoid conflicts between the use of the I/O in the DWEZ program and the default operation of the I/O.

  Example:
  DWEZ uses the A2 analog input to input a Torque Reference;
  however the default setting of the A2 input is Frequency Bias.

Precautions and Guidelines

The following rules must be observed when connecting the DriveWorksEZ™ function blocks.

◆ Rules for Setting User Parameter Default Values Using Drive Parameter o2-03

The standard drive provides the option of configuring any and all of the programming parameters and then saving the parameters as *User Initialization Values*. After configuring the drive, set parameter o2-03= *1: Set Defaults*, to save the parameters to a User Initialization memory location. Once this has been done, the *Initialize Parameters* parameter (A1-03) will offer the choice of *1110:User Initialize*. Choosing A1-03= *1110: User Initialize*, will reset all modified parameters back to the previous state.

Refer to the *Standard Drive Instruction Manual* for more specifics on setting o2-03.

■Precautions When Setting o2-03 in a Drive with DWEZ Firmware

The memory area reserved for *user parameter default values* is only 50 words large (100 bytes).

A very large number of modified drive parameters may exceed the reserved memory for storing *user parameter default values*.

If the user tries to store default parameter data larger than 50 words (100 bytes) by setting o2-03 = 1 with a large number of Modified Constants, an *Out of Memory* or *MAX Param Change* error is displayed. This is true for standard drive software as well as DWEZ software.

Out of Memory or *MAX Param Change* errors may occur more easily when using DWEZ with a drive because each connection in a DWEZ program requires two parameter changes. Therefore, for a modest program of 25 connections, the entire user area would already be exhausted, and that doesn't even include standard drive parameter changes or other Q parameter changes.

If *Out of Memory* or *MAX Param Change* errors occur, then none of the user parameters are stored as defaults in the drive. In this case, the user default values should be stored in the DWEZ project file for later retrieval.

◆ I/O Parameters Are Set to *Unused* by DWEZ When the DWEZ Program Is Using an I/O Point

Example:

When an A2 or A3 block is used in the program, the DWEZ PC tool sets that input to not used (1F) when the project is downloaded to the drive after compiling the project.

■Precautions when Setting Drive Parameters via the Drives Keypad or Drive Wizard Support Tool

The operation of a DWEZ program can be affected by the setting or changing of drive parameters. The DWEZ tool uses both standard and specially designed drive parameters in the DWEZ function blocks. It is possible to modify the function of a DWEZ program by not only changing parameter values in DWEZ but also by using the drives keypad or the DriveWizard Support Tool.

The following table lists standard drive parameters that exist within function blocks in DWEZ. Use the function block properties window in DWEZ to determine which standard drive parameters are included in your DWEZ program.

Standard Drive Parameters that exist within Function Blocks in DWEZ

DWEZ Block Names	Description	Drive Parameter
A1	Terminal A1 Signal	H3-01
A1	Terminal A1 Gain	H3-02
A1	Terminal A1 Bias	H3-03
A1	Filter Average Time	H3-12
A2	Terminal A2 Signal	H3-08
A2	Terminal A2 Selection	H3-09
A2	Terminal A2 Gain	H3-10
A2	Terminal A2 Bias	H3-11
A2	Filter Average Time	H3-12
A3	Terminal A3 Signal	H3-04
A3	Terminal A3 Selection	H3-05
A3	Terminal A3 Gain	H3-06
A3	Terminal A3 Bias	H3-07
A3	Filter Average Time	H3-12
AM	Terminal AM Sel	H4-04
AM	Terminal AM Gain	H4-05
AM	Terminal AM Bias	H4-06
AM	AO Level Select 2	H4-08
DI16H	DI Input	F3-01
FM	Terminal FM Sel	H4-01
FM	Terminal FM Gain	H4-02
FM	Terminal FM Bias	H4-03
FM	AO Level Select 1	H4-07

DWEZ Block Names	Description	Drive Parameter
M1	Terminal M1-M2 Sel	H2-01
M3	Terminal M3-M4 Sel	H2-02
M5	Terminal M5-M6 Sel	H2-03
MP	Pulse Moni Sel	H6-06
MP	Pulse Output Scaling	H6-07
P3	Terminal P3 Sel	H2-04
P4	Terminal P4 Sel	H2-05
RP	Pulse Input Scaling	H6-02
RP	Pulse Input Gain	H6-03
RP	Pulse Input Bias	H6-04
RP	Pulse Input Filter Time	H6-05
S3	Terminal S3 Selection	H1-01
S4	Terminal S4 Selection	H1-02
S5	Terminal S5 Selection	H1-03
S6	Terminal S6 Selection	H1-04
S7	Terminal S7 Selection	H1-05
S8	Terminal S8 Selection	H1-06
S9	Terminal S9 Selection	H1-07
S10	Terminal S10 Selection	H1-08
S11	Terminal S11 Selection	H1-09
S12	Terminal S12 Selection	H1-10
SFS	Accel Time 1	C1-01
SFS	Decel Time 1	C1-02

◆ Rules Enforced While Program is Being Drawn

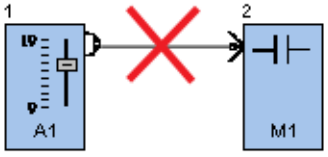
The DriveWorksEZ™ PC tool prevents the following rules from being violated as the application program is created on the screen.

■The number of connections cannot exceed 40.

The memory usage indicator on the lower right hand status bar indicates the percentage of maximum connections used.

■Numeric outputs can only be connected to numeric inputs.

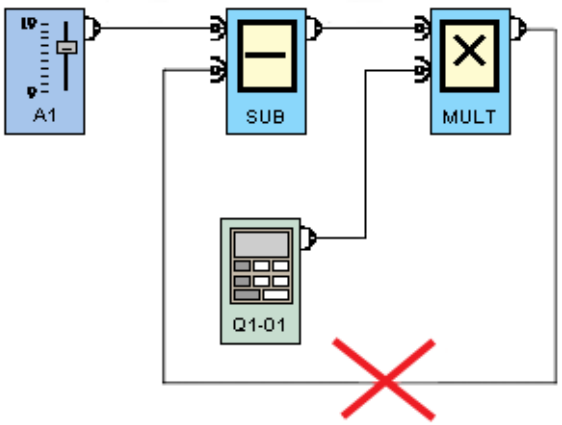
Logic outputs can only be connected to logic inputs.



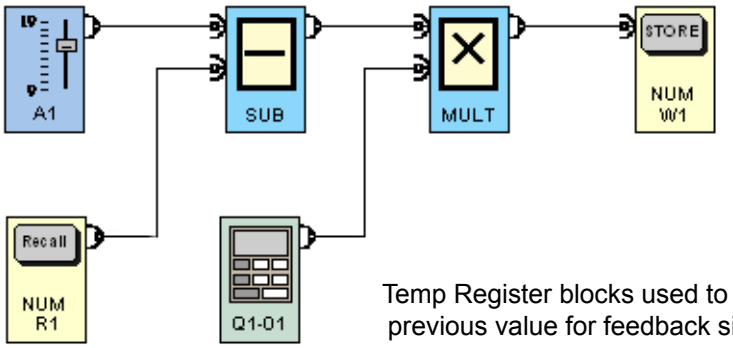
NOT PERMITTED:
Block with numeric output connected to block with logic input

■The output of a block cannot be directly connected to an input of a prior block in the sequence.

This type of connection is typically used in control block diagrams that require feedback. If it is necessary to make such a connection, temp register blocks should be used as shown below.



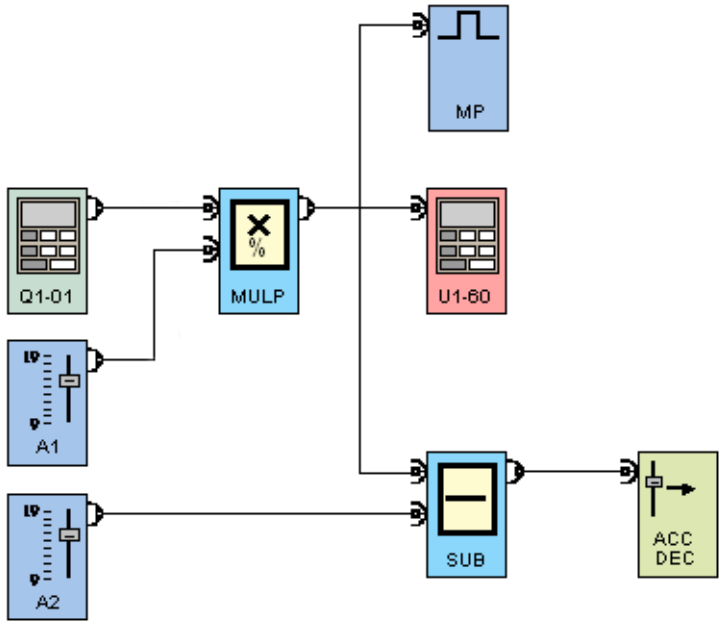
NOT PERMITTED:
Output connected to previous block input



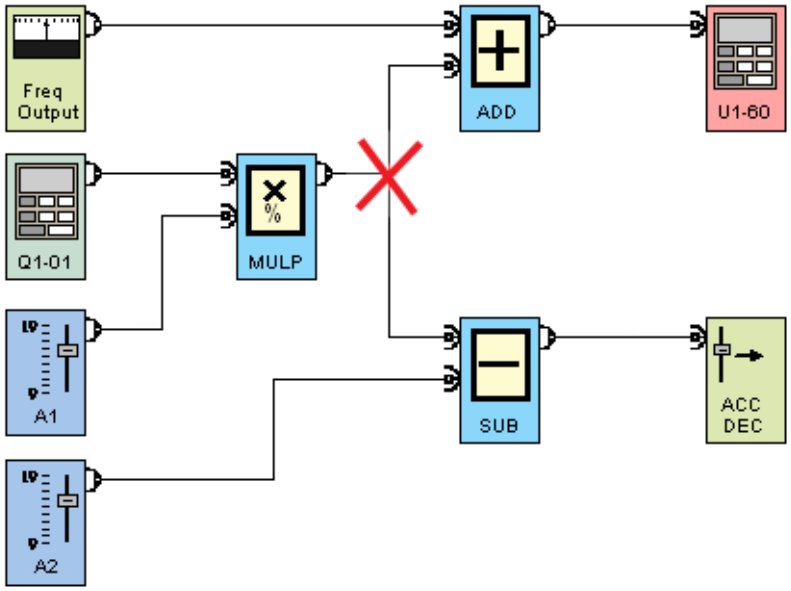
Temp Register blocks used to store previous value for feedback signal

■The output of any block can be connected to multiple output blocks, but can only be connected to one operation block.

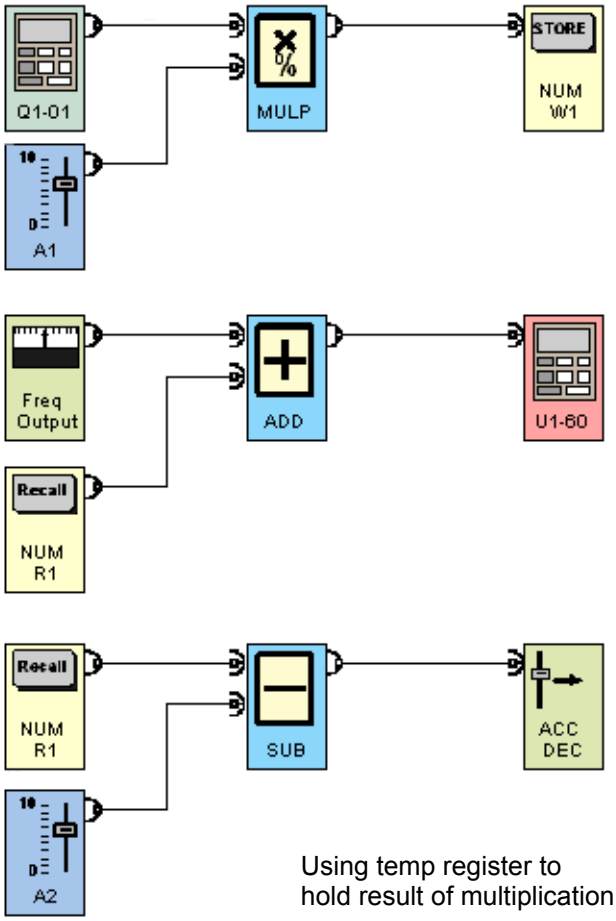
Temp register blocks should be used if it is necessary to connect one output to multiple operation blocks.



Output connected to multiple output blocks, but only one operation block



NOT PERMITTED: One output connected to multiple operation blocks



Using temp register to hold result of multiplication

◆ Rules Enforced at Compile Time

The DriveWorksEZ™ PC tool checks to see if the following rules are being violated when the application program is compiled.

■All inputs and outputs for each block in the program must be connected.

If this is not satisfied, a *Missing Connection* error will occur during compilation.

■The following blocks can only be used once per program:

- CNTR
- DIA
- DFILT
- DERV
- INTVL TMR
- MOP
- NUML1
- NUML2
- NUML3
- PI
- RAMP

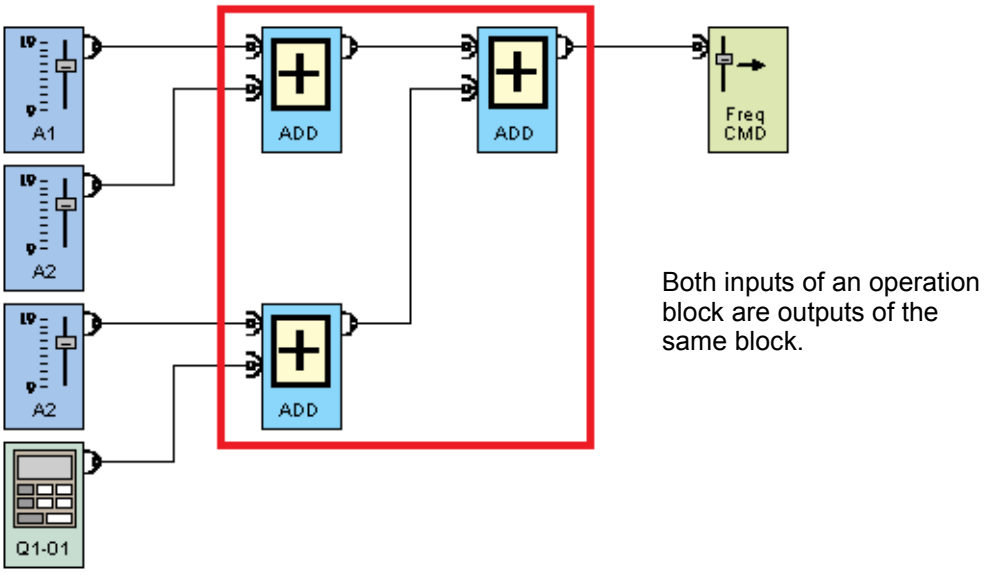
- Shift Cntrl
- SMOP
- SRFF 1
- SRFF 2
- SRFF 3
- TMR 1
- TMR 2
- TMR 3
- TMR 4
- 1 Shot
- LE (when Q5-09 > 0.00%)*
- GE (when Q5-11 > 0.00%)*

*No compiler error occurs, but the DriveWorksEZ™ program will not operate correctly.

If any of these blocks are used more than once, a *Block XXX can be used only once in a project* error will occur during compilation.

■ An operation block cannot be connected to both inputs of itself.

The DriveWorksEZ™ PC tool will automatically correct this issue at compile time by adding some temporary registers to the program behind the scenes. This is transparent to the user, but should be noted as the memory usage of the program will increase slightly as compiling occurs. If for some reason this memory increase causes the maximum number of connections to be exceeded, the compiler will generate an error stating that the project compile required the use of data write/read blocks. This causes the maximum number of links to be exceeded.



◆ Programming Limits

- 40-block limitation (2.5% x 40 blocks = 100%)
- Up to eight pages
- Uploading a program from a drive with DWEZ is only possible by the program originator

◆ Program Execution

The program function blocks are executed in the order of the connection sequence. The program is executed every 5ms (10ms when A1-02 = open loop vector 2).

This page intentionally left blank.

3

Function Blocks

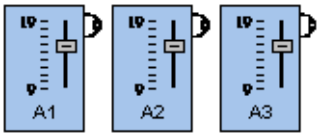
This chapter provides descriptions of the function blocks available in DriveWorksEZ™.

Drive I/O Tab Function Blocks

This table provides a description for each Drive I/O function block that appears in DriveWorksEZ™ and its location within the manual.

Block Name	Description	Page
A1	Analog Input	3
A2	Analog Input	3
A3	Analog Input	3
S1	Digital Input	5
S2	Digital Input	5
S3	Digital Input	5
S4	Digital Input	5
S5	Digital Input	5
S6	Digital Input	5
S7	Digital Input	5
S8	Digital Input	5
S9	Digital Input	5
S10	Digital Input	5
S11	Digital Input	5
S12	Digital Input	5
PG CH1	Encoder Input (Speed)	6
PG CH2	Encoder Input (Speed)	6
PGP CH1	Encoder Input (# of pulses)	7
PGP CH2	Encoder Input (# of pulses)	7
RP	Pulse Input	8
MP	Pulse Output	9
FM	Analog Output	10
AM	Analog Output	10
M1	Digital Output	11
M3	Digital Output	11
M5	Digital Output	11
P3	Digital Output	11
P4	Digital Output	11

◆ Analog Input



Block Names: A1, A2, A3

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	Numeric	Proportional to analog signal on drive terminal

Block Properties – A1

Name	Default	Min.	Max.	Parameter	Description
Terminal A1 Signal	0	0	1	H3-01	Selects the signal level type for Terminal A1 0: 0–10VDC 1: +/-10VDC
Terminal A1 Gain	100.0%	0.0	1000.0	H3-02	Sets the gain level for Terminal A1 Input
Terminal A1 Bias	0.0%	100.0	100.0	H3-03	Sets the bias level for the Terminal A1 Input
Filter Average Time	0.03sec	0.00	2.00	H3-12	Sets the input filter time for all three analog input terminals, A1, A2 & A3

Block Properties – A2

Name	Default	Min.	Max.	Parameter	Description
Terminal A2 Signal	0	0	1	H3-08	Selects the signal level type for Terminal A2 0: 0–10VDC 1: +/-10VDC 2: 4–20mA (Set switch S1-2 to the I position on the input terminal strip)
Terminal A2 Gain	100.0%	0.0	1000.0	H3-10	Sets the gain level for Terminal A2 Input
Terminal A2 Bias	0.0%	-100.0	100.0	H3-11	Sets the bias level for the Terminal A2 Input
Filter Average Time	0.03sec	0.00	2.00	H3-12	Sets the input filter time for all three analog input terminals, A1, A2 & A3

Block Properties – A3

Name	Default	Min.	Max.	Parameter	Description
Terminal A3 Signal	0	0	1	H3-04	Selects the signal level type for Terminal A3 0: 0–10VDC 1: +/-10VDC
Terminal A3 Gain	100.0%	0.0	1000.0	H3-06	Sets the gain level for Terminal A3 Input
Terminal A3 Bias	0.0%	-100.0	100.0	H3-07	Sets the bias level for the Terminal A3 Input
Filter Average Time	0.03sec	0.00	2.00	H3-12	Sets the input filter time for all three analog input terminals, A1, A2 & A3

Description

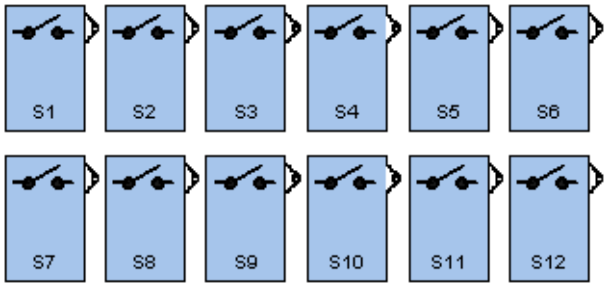
A1 has one numeric output corresponding to the level of analog input terminal A1 (normally used for the primary frequency reference). Can be unipolar (0.00% to 100.00%) or bipolar (-100.00% to 100.00%), depending on the setting of parameter H3-01.

A2 has one numeric output corresponding to the level of multi-function analog input terminal A2. Parameter H3-09 must be set to 1F (Not Used) for this block to function properly. The data output by A2 can be unipolar (0.00% to 100.00%) or bipolar (-100.00% to 100.00%), depending on the setting of parameter H3-08.

A3 has one numeric output corresponding to the level of multi-function analog input terminal A3. Parameter H3-05 must be set to 1F (Not Used) for this block to function properly. The data output by A3 can be unipolar (0.00% to 100.00%) or bipolar (-100.00% to 100.00%), depending on the setting of parameter H3-04.

When an A2 or A3 block is used in the program, the PC tool will set that input to Not Used (1F) when the project is downloaded to the drive after compiling the project.

◆ Digital Input



Block Names: S1, S2, S3, S4, S5, S6, S7, S8, S9, S10, S11, S12

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	LOGIC	Indicates the status of a digital input terminal

Block Properties – S1 to S12

Only [ID] and Label properties are present.

Description

The S1 to S12 blocks each have a logic output that indicates the status of terminals S1 to S12. Terminals S1 to S12 still function normally, even when their corresponding function blocks are programmed into the connection sequence. The H1-XX parameter associated with one of the terminals S3 to S12 will automatically be set to *F* (Not Used) when the corresponding block is used in the connection sequence. DriveWorksEZ™ recognizes the input terminal regardless of its H1-XX setting.

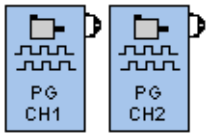
Example

Terminal S4 is used in the connection sequence. When the program is compiled and downloaded to the Drive, parameter H1-02 will be automatically set to *F* (Not Used). Using the digital operator, that parameter could be changed back to the factory setting of *14* (Fault Reset). When Terminal S4 is closed it will work properly in the connection diagram and also act as a fault reset command.

When the corresponding input is closed the logic output will be TRUE. When the input is open, the output will be FALSE.

When S1 or S2 are used in the program without using the drive Commands Run Fwd or Run Rev, respectively, they still retain their Forward/Reverse or run/stop functionality.

◆ Pulse Generator Input – Speed (Frequency)



Block Names: PG CH1, PG CH2

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	Numeric	Proportional to pulse generator speed (frequency)

Block Properties – PG CH1

Name	Default	Min.	Max.	Parameter	Description
Max Pulses CH1	32,020Hz	0	65,535	Q2-01	Input frequency on PG card that will result in 100.00% output

Block Properties – PG CH2

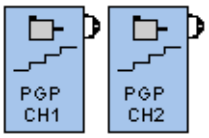
Name	Default	Min.	Max.	Parameter	Description
Max Pulses CH2	32,020Hz	0	65,535	Q2-02	Input frequency on PG card that will result in 100.00% output

Description

The PG CH1 and PG CH2 blocks each have one numeric output corresponding to the pulse frequency of channels 1 and 2 of the PG-W2 option card. PG CH1 also functions with single channel PG option cards (PG-X2 and PG-B2). Parameters Q2-01/Q2-02 set the pulse frequency that will produce data of 10000 (100.00%) on the numeric output. With Q2-01/Q2-02 at default setting, the output of the PG CH or PG CH2 block will be 10000 (100.00%) when running a motor with a 1024 PPR encoder in the forward direction at 1800RPM.

If the output frequency of the encoder exceeds 65535Hz for a particular application, the output of the PG block could be submitted to a scaling block (Scale 1 or Scale 2) to obtain proper scaling.

◆ Pulse Generator Input–Pulse Count



Block Names: PGP CH1, PGP CH2

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	Numeric	Corresponding to the total number of encoder lines accumulated

Block Properties – PGP CH1 & PGP CH2

Only [ID] and Label properties are present.

Description

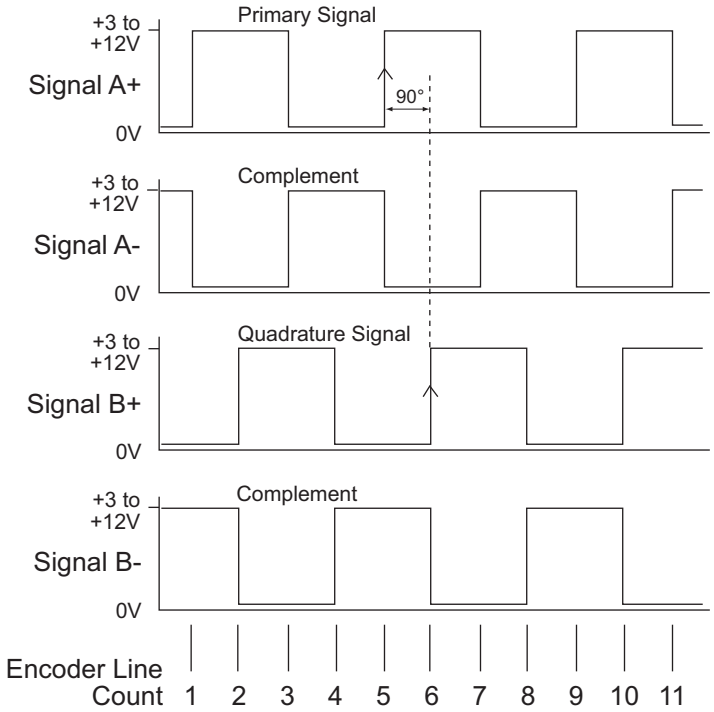
The PGP CH1 and PGP CH2 blocks each have one numeric output corresponding to the total number of encoder lines accumulated in channels 1 and 2 of the PG-W2 option card. PGP CH1 also functions with single channel PG cards (PG-X2 & PG-B2). The numeric output will be from 0 to 655.35%. The value will roll over to 0 on the next incremental pulse count after 655.35%.

INFO

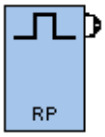
An *encoder line* is ¼ of an encoder pulse.

Example

1 revolution of a 1024 PPR encoder would result in an output of 4096 (40.96%). See the following diagram.



◆ Pulse Input



Block Name: RP

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	Numeric	Proportional to signal frequency on terminal RP

Block Properties – RP

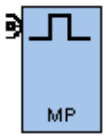
Name	Default	Min.	Max.	Parameter	Description
Pulse Input Scaling	1440	1000	32,000	H6-02	Input frequency on terminal RP that will result in 100.00% output
Pulse Input Gain	100.0%	0.0	1000.0	H6-03	Sets the gain level for terminal RP input
Pulse Input Bias	0.0%	-100.0	100.0	H6-04	Sets the bias level for terminal RP input

Description

RP has one numeric output corresponding to the frequency of the pulse train input RP. When the pulse train input frequency equals the value programmed in Pulse Train Input Scaling (H6-02) the numeric output will be 10000 (100.00%).

When an RP or MP block is used in the program, the PC tool will set that input to Not Used when the project is downloaded to the drive after compiling the project.

◆ Pulse Output



Block Name: MP

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Controls the output frequency on terminal MP. Scaling is controlled by parameter H6-07

Block Properties – MP

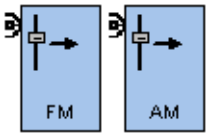
Name	Default	Min.	Max.	Parameter	Description
Pulse Output Scaling	1440	0	32,000	H6-07	Sets the output frequency of terminal MP when the input is 10000 (100.00%)

Description

The numeric input of MP is used to control the pulse train output. The data is scaled based on Pulse Train Output Scaling (H6-07). Data of 10000 (100.00%) will produce the pulse frequency set in H6-07 at the pulse output terminal MP. When MP is programmed into the connection sequence, control of terminal MP is assumed by DriveWorksEZ™ and the setting of H6-06 is Not Used.

When an RP or MP block is used in the program, the PC tool will set that input to Not Used when the project is downloaded to the drive after compiling the project.

◆ Analog Output



Block Names: FM, AM

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Controls the voltage on the analog monitor terminal

Block Properties – FM

Name	Default	Min.	Max.	Parameter	Description
Terminal FM Gain	100.0%	0.0	1000.0	H4-02	Sets the gain level for Terminal FM Output
Terminal FM Bias	0.0%	-110.0	110.0	H4-03	Sets the bias level for the Terminal FM Output
AO Level Select 1	0	0	2	H4-07	Selects the signal level type for Terminal FM 0: 0–10VDC 1: +/-10VDC 2: 4–20mA (Set jumper CN15, CH1, to *I* on the input terminal strip)

Block Properties – AM

Name	Default	Min.	Max.	Parameter	Description
Terminal AM Gain	50.0%	0.0	1000.0	H4-05	Sets the gain level for Terminal AM Output
Terminal AM Bias	0.0%	-110.0	110.0	H4-06	Sets the bias level for the Terminal AM Output
AO Level Select 1	0	0	2	H4-08	Selects the signal level type for Terminal FM 0: 0–10VDC 1: +/-10VDC 2: 4–20mA (Set jumper CN15, CH2, to *I* on the input terminal strip)

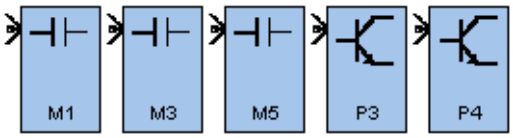
Description

FM: The numeric input of the FM block is used to control analog output terminal FM. Data of 10000 (100.00%) will produce 10V at terminal FM. When H4-07 (AO Level Select 1) is set to 1, data of -10000 (-100.00%) will produce -10V at the FM block. When the FM block is programmed into the connection sequence, control of terminal FM is assumed by DriveWorksEZ™ and the setting of H4-01 is ignored.

AM: The numeric input of the AM block is used to control analog output terminal AM. Data of 10000 (100.00%) will produce 10V at terminal AM. When H4-08 (AO Level Select 2) is set to 1, data of -10000 (-100.00%) will produce -10V at the AM block. When the AM block is programmed into the connection sequence, control of terminal AM is assumed by DriveWorksEZ™ and the setting of H4-04 is ignored.

When an AM or FM block is used in the program, the PC tool will respectively set H4-04 or H4-01=31 (Not Used) when the project is downloaded to the drive after compiling.

◆ Digital Output



Block Names: M1, M3, M5, P3, & P4

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	LOGIC	Controls the multi-function output terminal

Block Properties – M1, M3, M5, P3, & P4

Only [ID] and Label properties are present.

Description

M1, M3, M5, P3, and P4 each have a logic input that controls the state of terminals M1, M3, M5, P3, and P4, respectively. When any of these blocks are programmed into the connection sequence, DriveWorksEZ™ assumes control of the corresponding multi-function output terminal and the setting of H2-0X will be set to Not Used.

When the logic input of any of the above function blocks is TRUE, the corresponding output will close. When the input is FALSE, the output will open.

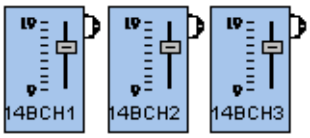
When an M1, M3, M5, P3, or P4 block is used in the program, the PC tool will set H2-XX = 1F (Not Used) when the project is downloaded to the drive after compiling.

Drive Option I/O Tab Function Blocks

This table provides a description for each Drive Option I/O function block that appears in DriveWorksEZ™ and its location within the manual.

Block Name	Description	Page
14BCH1	Analog Input	13
14BCH2	Analog Input	13
14BCH3	Analog Input	13
DI16H	Digital Input	15
DI-08 In 1	Digital Input	16
DI-08 In 2	Digital Input	16
DI-08 In 3	Digital Input	16
DI-08 In 4	Digital Input	16
DI-08 In 5	Digital Input	16
DI-08 In 6	Digital Input	16
DI-08 In 7	Digital Input	16
DI-08 In 8	Digital Input	16
DO-02 Out 1	Digital Output	17
DO-02 Out 2	Digital Output	17

◆ AI-14B Option – Analog Input



Block Names: 14BCH1, 14BCH2, 14BCH3

Block Type: Input

Use: Multiple times per program.

I/O Table

Connections	Signal Type	Description
Output	Numeric	Proportional to analog signal on drive terminal

Block Properties – 14BCH1

Name	Default	Min.	Max.	Parameter	Description
AI-14B Ch1 Gain	100.0%	0.0	1000.0	Q2-04	Sets the gain level for Terminal TC1 Input
AI-14B Ch1 Gain	0.0%	-100.0	100.0	Q2-05	Sets the bias level for the Terminal TC1 Input
AI-14B Signal Level Select	0	0	2	Q2-10	Selects the signal level type for all option board terminals 0: 0–10VDC 1: +/-10VDC 2: 4–20mA (Move all jumpers on option board to the *I* position)

Block Properties – 14BCH2

Name	Default	Min.	Max.	Parameter	Description
AI-14B Ch2 Gain	100.0%	0.0	1000.0	Q2-06	Sets the gain level for Terminal TC2 Input
AI-14B Ch2 Gain	0.0%	-100.0	100.0	Q2-07	Sets the bias level for the Terminal TC2 Input
AI-14B Signal Level Select	0	0	2	Q2-10	Selects the signal level type for all option board terminals 0: 0–10VDC 1: +/-10VDC 2: 4–20mA (Move all jumpers on option board to the *I* position)

Block Properties – 14BCH3

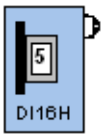
Name	Default	Min.	Max.	Parameter	Description
AI-14B Ch3 Gain	100.0%	0.0	1000.0	Q2-08	Sets the gain level for Terminal TC3 Input
AI-14B Ch3 Gain	0.0%	-100.0	100.0	Q2-09	Sets the bias level for the Terminal TC3 Input
AI-14B Signal Level Select	0	0	2	Q2-10	Selects the signal level type for all option board terminals 0: 0–10VDC 1: +/-10VDC 2: 4–20mA (Move all jumpers on option board to the *I* position)

Description

Blocks 14BCH1, 14BCH2, and 14BCH3 output the value for each of the three channels of the AI-14B analog input option. When DriveWorksEZ™ detects one of the AI-14B blocks in use, it will disable the functionality in the basic drive software and the values from the input card go to the DriveWorksEZ™ software only. The three analog inputs A1, A2, and A3 remain unchanged. Therefore, the user can achieve a total of six channels of analog input.

Parameter F2-01 MUST be set “0: 3-channel individual” for the AI-14B option board to work correctly.

◆ DI-16H2 Option – BCD/Binary Digital Input



Block Name: DI16H

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	Numeric	Convert 16 discrete digital inputs into a numeric value

Block Properties – DI16H

Only [ID] and Label properties are present.

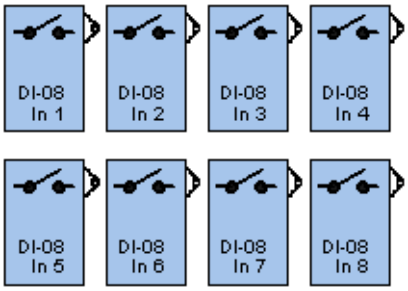
Description

This block allows the user the ability to add a BCD (Binary Coded Decimal) digital input to the program. When this block is added to the program, parameter F3-01 will automatically be set to 6 and O1-03 value will be ignored. The range of input is -199.98% to +199.98%.

Terminal Block	Terminal #	Description	Example 1 12.34%	Example 2 195.62%
TC1	1	*Hundredths* BCD Digit (000.0X%) [1]	0	1
	2		1	0
	3		0	0
	4	*Tenths* BCD Digit (000.X0%)	1	0
	5		1	1
	6		0	1
	7		0	0
	8	*Ones* BCD Digit (00X.00%)	0	1
	9		1	0
	10		0	1
TC2	1	*Tens* BCD Digit (0X0.00%)	0	0
	2		1	1
	3		0	0
	4		0	0
	5		0	1
	6	*Hundreds* BCD Digit (X00.00%)	0	1
	7	SIGN (0 = Positive, 1 = Negative)	0	1
	8	SET (Read Data)	1	1
	9	Input Signal Common	-	-

1. The Least significant bit is not available. Terminals 1, 2, and 3 represent the most significant bits of the *Hundredths* BCD digit. Finest resolution using this option will be 0.02%.

◆ DI-08 Option – Digital Input



Block Names: DI-08 In 1, DI-08 In 2, DI-08 In 3, DI-08 In 4, DI-08 In 5, DI-08 In 6, DI-08 In 7, DI-08 In 8

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	LOGIC	Indicates the status of the DI-08 option board digital input terminal

Block Properties – DI-08 In 1 thru DI-08 In 8

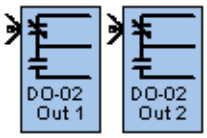
Only [ID] and Label properties are present.

Description

There are eight blocks that correspond to each of the eight digital inputs of the DI-08 input option card. This will allow the user eight additional digital inputs. Each block will correspond to one of the digital inputs. When any of the blocks are used in the program, parameter numbers F3-01 and O1-03 are Not Used for this option.

When the corresponding input is closed, the logic output will be TRUE. When the corresponding input is open, the logic output will be FALSE.

◆ DO-02C Option – Digital Output



Block Names: DO-02 Out 1, DO-02 Out 2

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	LOGIC	Controls the multi-function output terminal

Block Properties – DO-02 Out 1, DO-02 Out 2

Only [ID] and Label properties are present.

Description

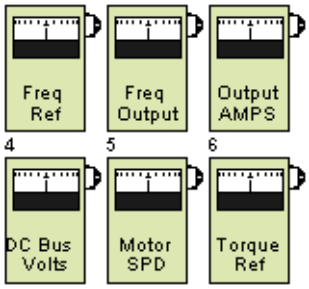
Setting the logic input of DO-02 OUT1 or DO-02OUT2 to TRUE triggers the first or second output of the DO-02C form C relay output. When either of the two blocks is selected, the F5-01 or F5-02 parameters are ignored and the logic for the output comes from the DriveWorksEZ™ program.

Drive Monitors Tab Function Blocks

This table provides a description for each Drive Monitor function block that appears in DriveWorksEZ™ and its location within the manual.

Block Name	Description	Page
Freq Ref	Input	19
Freq Output	Input	19
Output AMPS	Input	19
DC Bus Volts	Input	19
Motor SPD	Input	19
Torque Ref	Input	19
Drive Mon 1	Input	20
Drive Mon 2	Input	20
Run X	Input	22
Run Rev	Input	22
Run Fwd	Input	22
@ Zero Speed	Input	22
@ Set Speed	Input	22
Drive Ready	Input	22
Drive Alarm	Input	22
Drive Fault	Input	22
Drive Regen	Input	22
Drive TQLIM	Input	22
Drive ZS end	Input	22

◆ Numeric Monitors



Block Names: Freq Ref, Freq Output, Output Amps, DC Bus Volts, Motor SPD, Torque Ref

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	Numeric	Proportional to internal drive signal

Block Properties – Freq Ref, Freq Output, Output Amps, DC Bus Volts, Motor SPD, Torque Ref

Only [ID] and Label properties are present.

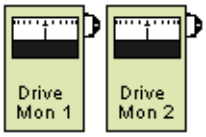
Description

These blocks allow various drive monitors to be used inside of a DriveWorksEZ™ program. Each block has one numeric output representing its associated monitor.

Block Name	U1 Monitor	Description	Range	Scaling 100.00% =	Notes
Freq Ref	U1-01	Frequency Reference	-100.00%–100.00%	Maximum Frequency	Max. Frequency is parameter E1-04 (or E3-02 for Motor 2)
Freq Output	U1-02	Output Frequency	-100.00%–100.00%	Maximum Frequency	Max. Frequency is parameter E1-04 (or E3-02 for Motor 2)
Output Amps	U1-03	Output Current	0.00%–100.00%	Drive Rated Output Current	As shown on the drive-data nameplate
DC Bus Volts	U1-07	DC Bus Voltage	0.00%–100.00%	400V or 800V	230V Drive > 400V 460V Drive > 800V
Motor SPD	U1-05 / U1-02[1]	Motor Speed	-100.00%–100.00%	Maximum Frequency	Max. Frequency is parameter E1-04 (or E3-02 for Motor 2)
Torque Ref[2]	U1-09	Torque Reference	-300.00%–300.00%	Motor Rated Torque	As determined by the motor parameters

1. The Motor SPD block will report motor speed in all modes except V/f, where it will report output frequency.
2. Available only in a *Vector* control method (A1-02 = 2, 3, or 4). If the control method is V/f (A1-02 = 0 or 1), this block will always report 0.00%.

◆ Numeric Monitors – Selectable



Block Names: Drive Mon 1, Drive Mon 2

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	Numeric	Proportional to internal drive signal

Block Properties – Drive Mon 1

Name	Default	Min.	Max.	Parameter	Description
Monitor Sel 1	0	0	17	Q2-11	Selects which U1 monitor value is output (see table below)

Block Properties – Drive Mon 2

Name	Default	Min.	Max.	Parameter	Description
Monitor Sel 2	0	0	17	Q2-12	Selects which U1 monitor value is output (see table below)

Description

These two blocks allow the user access to additional U1-XX drive numeric monitors that are available to output to an analog output. All of the monitors are scaled as 100% being full scale and are scaled similar to the analog output. Each block will have a Q2 parameter that will contain the selection to be the output of the respective monitor block.

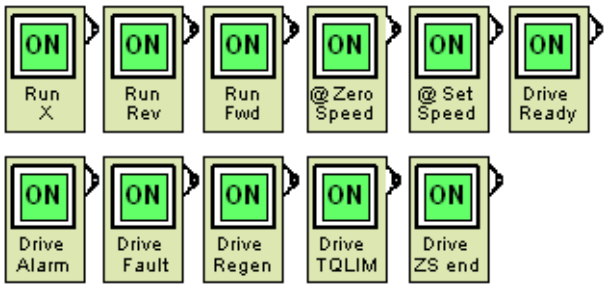
Q2-11 / Q2-12 Value	U1 Monitor	Description	Range	Scaling 100.00% =	Notes
0	N/A	None	N/A	N/A	
1	U1-08	Output Power	+/- 100.00%	Inv Capacity	1
2	U1-18	Motor Secondary Amps (Iq)	+/- 100.00%	Mtr rated sec. amps	2
3	U1-19	Motor Excitation Amps (Id)	+/- 100.00%	Mtr rated sec. amps	2
4	U1-21	ASR Input	+/- 100.00%	Max Freq	3, 4
5	U1-22	ASR Output	+/- 100.00%	Mtr rated sec. amps	4, 5
6	U1-24	PID Feedback	+/- 100.00%	Max Freq	3
7	U1-26	Voltage Reference (Vq output)	+/- 100.00%	200/400 V	
8	U1-27	Voltage Reference (Vd output)	+/- 100.00%	200/400 V	
9	U1-32	ACR (q) output	+/- 100.00%	100% ACR Output	
10	U1-33	ACR (d) output	+/- 100.00%	100% ACR Output	
11	U1-36	PID Input	+/- 100.00%	Max Freq	3
12	U1-37	PID Output	+/- 100.00%	Max Freq	3
13	U1-38	PID Reference	0–100.00%	Max Freq	3
14	U1-42	Estimated Motor Flux	0–100.00%	Rated motor flux	

Q2-11 / Q2-12 Value	U1 Monitor	Description	Range	Scaling 100.00% =	Notes
15	U1-43	Motor Flux Comp	+/- 100.00%	Mtr rated sec. amps	2
16	U1-44	ASR output w/o filter	+/- 100.00%	Mtr rated sec. amps	2
17	U1-45	Feed Forward control	+/- 100.00%	Mtr rated sec. amps	2

Notes:

1. Drive capacity is available in the *Specifications* section of the G7 drive technical manual, (TM.G7.01).
2. As determined by the motor parameters.
3. Maximum frequency is parameter E1-04 (or E3-02 for Motor 2).
4. Available only when the control method is: V/f w/PG, Flux Vector, Open Loop Vector 2 (A1-02 = 1, 3, or 4).
5. When control method is V/f w/PG (A1-03 = 1), ASR output is a percentage of maximum frequency.

◆ Logic Monitors



Block Names: Run X, Run Rev, Run Fwd, @ Zero Speed, @ Set Speed, Drive Ready, Drive Alarm, Drive Fault, Drive Regen, Drive TQLIM, Drive ZS end

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	LOGIC	Internal drive status signal

Block Properties – Run X, Run Rev, @ Zero Speed, @ Set Speed, Drive Ready, Drive Alarm, Drive Fault, Drive Regen, Drive TQLIM, Drive ZS end

Only [ID] and Label properties are present.

Description

These blocks allow various internal drive status bits to be used inside of a DriveWorksEZ™ program. Each block has one logic output.

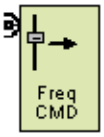
Block Name	Output = TRUE
Run X	Drive is in run mode (voltage output or run command is given)
Run Rev	Drive is running in reverse/Motor rotation is reverse
Run Fwd	Drive is in the forward mode (independent of speed or reference polarity)
@ Zero Speed	Drive is at a zero speed state (motor speed < 0.5Hz)
@ Set Speed	Drive output speed (freq) = frequency reference (parameter L4-02 sets hysteresis level)
Drive Ready	Drive is in the ready mode (not alarmed or faulted)
Drive Alarm	Drive is in an alarm state and not in the program mode
Drive Fault	Drive is in a fault state and not in the program mode
Drive Regen	Drive is regenerating (same as selection 1D for H2 parameters)
Drive TQLIM	Drive is in torque limit (same as selection 30 for H2 parameters)
Drive ZS end	Drive is in Zero Servo mode and function is completed (same as selection 33 for H2 parameters)

Drive Commands Tab Function Blocks

This table provides a description for each Drive Command function block that appears in DriveWorksEZ™ and its location within the manual.

Block Name	Description	Page
Freq CMD	Output	24
Torq CMD	Output	25
ACC DEC	Output	26
T Comp CMD	Output	27
ASR PGAIN	Output	28
ASR Int time	Output	29
PID PGAIN	Output	30
PID Int time	Output	31
PID Dv time	Output	32
PID Offset	Output	33
Fwd CMD	Output	34
Rev CMD	Output	35
BB CMD	Output	36
DCINJ	Output	36
FSTOP	Output	36
ASR I Rst	Output	36
SPD TORQ	Output	36
Zero Servo	Output	36
PID Disbl	Output	37
PID I Hold	Output	37
PID I Rst	Output	37
ALARM	Output	38
FAULT	Output	38

◆ Frequency Set Command



Block Name: Freq CMD

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Frequency Reference (100% = Max. Freq.)

Block Properties – Freq CMD

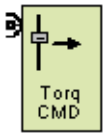
Only [ID] and Label properties are present.

Description

The numeric input of the Freq CMD block is used to set the drive's frequency reference. Data of 100.00% would result in a frequency reference equal to the Maximum Frequency parameter E1-04. Positive or negative values are accepted. When this block is programmed into the connection sequence, control of the frequency reference is assumed by DriveWorksEZ™ and the setting of parameter b1-01 is ignored.

When this block is selected, all of the standard frequency references to the soft starter (SFS) are disconnected and only the value input to the Freq CMD block is used. There is no indication to the user via the keypad that this function is operational.

◆ Torque Command/Torque Limit



Block Name: Torq CMD

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Torque Reference (100% = Motor Rated Torque)

Block Properties – Torq CMD

Name	Default	Min.	Max.	Parameter	Description
Maximum Torque Reference	300.00%	0.00	300.00	Q2-03	Limits the maximum torque reference or torque command supplied by this block.

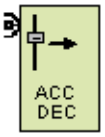
Description

The numeric input of the Torq CMD block is used to set the drive torque reference or torque limit. Data of 100.00% will result in a torque limit or torque reference of the motor rated torque. Positive or negative values are accepted. The torque is limited based on the setting of Max Torque Ref (Q2-03). The table below shows a summary of the functionality of the Torq CMD block.

Speed / Torque Selections	Open Loop Vector (A1-02 = 2)	CLV/ Open Loop Vector 2 (A1-02 = 3 or 4)
Speed Mode d5-01 = 0 and H1-XX not = 71 and SP/TORQ block is Not Used.	Torque Limit	Torque Limit
Torque Mode d5-01 = 1 and H1-XX not = 71 and SPD/TORQ block is Not Used.	Torque Limit	Torque Reference
Switch Between Speed and Torque d5-01 = 0 and H1-XX = 71 or SPD/TORQ block is used.	Torque Limit	Off (FALSE) = Torque Limit On (TRUE) = Torque Reference
Torque Mode d5-01 = 1 and H1-XX = 71 or SPD/TORQ block is used.	Torque Limit	Torque Reference (Logic input is ignored)

When this block is used and the drive is in torque mode, the standard analog torque reference (H3-0X = 13) is not functional. When this block is used and the drive is in speed mode, other analog torque references (H3-0X = 13) are active as torque limits. There is no indication to the user via the keypad that this function is operational.

◆ Accel/Decel Command



Block Name: ACC DEC

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Modifies selected accel and decel times

Block Properties – ACC DEC

Only [ID] and Label properties are present.

Description

The numeric input of the ACC DEC block is used to adjust the soft-starter (accel/decel) times. Data is input as a percentage of the selected accel/decel values. Negative values will be treated as 0.

$$\text{Accel time} = \frac{\text{Numeric input data} \times \text{C1-01}^*}{10000}$$

* Selected accel time (C1-01, C1-03, C1-05, or C1-07)

$$\text{Decel time} = \frac{\text{Numeric input data} \times \text{C1-02}^{**}}{10000}$$

** Selected decel time (C1-02, C1-04, C1-06, or C1-08)

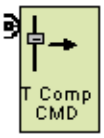
Example

If C1-01 is set to 10.0 seconds and the numeric input to the ACC DEC block is 40.00%, then the actual acceleration time used by the soft-starter would be 4.0 seconds.

The selected C1-XX/C2-XX, C3-XX/C4-XX, C5-XX/C6-XX, or C7-XX/C8-XX acceleration/deceleration times will be modified in the soft-starter (SFS) as described above.

There is no indication to the user via the keypad that this function is operational.

◆ Torque Compensation Command



Block Name: T Comp CMD

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Internally adds to the Iq Reference

Block Properties – T Comp CMD

Only [ID] and Label properties are present.

Description

The numeric input of the T Comp CMD block is used to set the torque compensation value (not in V/F mode). This value is added to the torque producing current reference (Iq). A value of 100.00% will add 100.00% of rated Iq (torque producing current). Note that if an analog input is programmed for Torque Compensation and the T Comp CMD block is used in a program, the analog input will be ignored.

When this block is used, the setting of 14 (Torque Compensation) for Analog input terminal A2 or A3 becomes the same as 1F, Not Used.

◆ Automatic Speed Regulator (ASR) P-Gain Set Command

Block Name: ASR PGAIN

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Changes the gain of the speed regulator

Block Properties – ASR PGAIN

Name	Default	Min.	Max.	Parameter	Description
ASR P Ramp Time	0.50sec	0.00	10.00	Q2-13	Sets the time it will take for the ASR P Gain to change from the minimum (0.00%) value to the maximum value (100.00%)

Description

The numeric input of the ASR PGAIN block is used to control the gain of the Automatic Speed Regulator (ASR). The ASR is only active when the control method is V/f w/PG, Flux Vector, or Open Loop Vector 2 (A1-02 = 1, 3, or 4).

Normally the gain value the ASR uses originates from parameter C5-01. However, when this block is contained in the program, the C5-01 parameter is ignored and the ASR uses the data from this block instead.

Data of 10000 (100.00%) will result in an ASR gain of 300.00. Data of 0 (0.00%) will result in an ASR gain of 1.00. Values greater than 10000 (100.00%) are clamped at 10000 and less than 0 (0.00%) are clamped at 1.00.

Example

If the numeric input to the ASR PGAIN block is 5.00%, the ASR gain will be 15.00.

To prevent sudden changes in torque, the actual input to the ASR PGAIN block is internally ramped, with the ramp time being controlled by parameter Q2-13.

There is no indication to the user that this function is active.

◆ Automatic Speed Regulator (ASR) I-Time Set Command

Block Name: ASR Int time

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Changes the integral time of the speed regulator

Block Properties – ASR Int Time

Name	Default	Min.	Max.	Parameter	Description
ASR I Ramp Time	0.50sec	0.00	10.00	Q2-14	Sets the time it will take for the ASR I Time to change from the minimum (0.00%) value to the maximum value (100.00%)

Description

The numeric input of the ASR Int time block is used to control the integral time of the Automatic Speed Regulator (ASR). The ASR is only active when the control method is V/f w/PG, Flux Vector, or Open Loop Vector 2 (A1-02 = 1, 3, or 4).

Normally the integral time the ASR uses originates from parameter C5-02. However, when this block is contained in the program the C5-02 parameter is ignored and the ASR uses the data from this block instead.

Data of 10000 (100.00%) will result in an ASR integral time of 10.00 seconds. Data of 0 (0.00%) will result in an ASR integral time of 0.00 seconds. Values greater than 10000 (100.00%) are clamped at 10000 and less than 0 (0.00%) are clamped at 0.

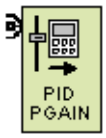
Example

If the numeric input to the ASR Int time block is 5.00%, the ASR integral time will be 0.50 seconds.

To prevent sudden changes in torque, the actual input to the ASR Int time block is internally ramped, with the ramp time being controlled by parameter Q2-14.

There is no indication to the user that this function is active.

◆ PID Gain Set Command



Block Name: PID PGAIN

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Changes the gain of the drive's internal PID process controller

Block Properties – PID PGAIN

Only [ID] and Label properties are present.

Description

The numeric input of the PID PGAIN block is used to control the proportional gain of the drive's internal PID process controller. The drive's internal PID process controller is only active when parameter b5-01 is set to a value greater than zero.

Normally the drive's internal PID proportional gain originates from parameter b5-02. However, when this block is contained in the program, the b5-02 parameter is ignored and the drive's internal PID process controller uses the data from this block instead.

Data of 10000 (100.00%) will result in a PID proportional gain of 25.00. Data of 0 (0.00%) will result in an a PID proportional gain of 0.00. Values greater than 10000 (100.00%) are clamped at 10000 and less than 0 (0.00%) are clamped at 0.

Example

If the numeric input to the PID PGAIN block is 15.00%, the PID proportional gain will be 3.75.

There is no indication to the user that this function is active.

◆ PID Integral Time Set Command

Block Name: PID Int time

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Changes the integral time of the drive's internal PID process controller

Block Properties – PID Int Time

Only [ID] and Label properties are present.

Description

The numeric input of the PID Int time block is used to control the integral time of the drive's internal PID process controller. The drive's internal PID process controller is only active when parameter b5-01 is set to a value greater than zero.

Normally the drive's internal PID integral time originates from parameter b5-03. However, when this block is contained in the program, the b5-03 parameter is ignored and the drive's internal PID process controller uses the data from this block instead.

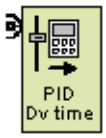
Data of 10000 (100.00%) will result in a PID integral time of 360.0 seconds. Data of 0 (0.00%) will result in a PID integral time of 0.00 seconds. Values greater than 10000 (100.00%) are clamped at 10000 and less than 0 (0.00%) are clamped at 0.

Example

If the numeric input to the PID Int time block is 2.00%, the PID integral time will be 7.2 seconds.

There is no indication to the user that this function is active.

◆ PID Derivative Time Set Command



Block Name: PID Dv time

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Changes the derivative time of the drive's internal PID process controller

Block Properties – PID Dv Time

Only [ID] and Label properties are present.

Description

The numeric input of the PID Dv time block is used to control the derivative time of the drive's internal PID process controller. The drive's internal PID process controller is only active when parameter b5-01 is set to a value greater than zero.

Normally the drive's internal PID derivative time originates from parameter b5-05. However, when this block is contained in the program, the b5-05 parameter is ignored and the drive's internal PID process controller uses the data from this block instead.

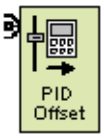
Data of 10000 (100.00%) will result in a PID derivative time of 10.00 seconds. Data of 0 (0.00%) will result in an a PID derivative time of 0.00 seconds. Values greater than 10000 (100.00%) are clamped at 10000 and less than 0 (0.00%) are clamped at 0.

Example

If the numeric input to the PID Dv time block is 40.00%, the PID derivative time will be 4.0 seconds.

There is no indication to the user that this function is active.

◆ PID Offset Set Command



Block Name: PID Offset

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Adds additional output to the drive's internal PID process controller

Block Properties – PID Offset

Only [ID] and Label properties are present

Description

The numeric input of the PID Offset block is used to control the amount of output added to the drive's internal PID process controller. The drive's internal PID process controller is only active when parameter b5-01 is set to a value greater than zero.

Normally the drive's internal PID offset originates from parameter b5-07. However, when this block is contained in the program, the b5-07 parameter is ignored and the drive's internal PID process controller uses the data from this block instead.

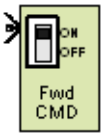
Data of 10000 (100.00%) will result in a PID offset value of 100.0% (max. frequency). Data of 10000 (-100.00%) will result in an a PID offset value of -100.0% (-max. frequency). Values greater than 10000 (100.00%) are clamped at 10000 and less than -10000 (-100.00%) are clamped at -10000.

Example

If the numeric input to the PID Offset block is 5.00%, parameter E1-04 will be 60.0Hz, the amount of offset added is 3Hz.

There is no indication to the user that this function is active.

◆ Run FWD Command



Block Name: Fwd CMD

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	LOGIC	True = Drive runs in the forward direction

Block Properties – Fwd CMD

Only [ID] and Label properties are present.

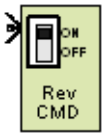
Description

The logic input to the Fwd CMD block is used to control the forward run command. When the input data is TRUE, the forward run command is given. When this block is programmed into the connection sequence, control of the run commands are assumed by DriveWorks EZ™ and the setting of parameter b1-02 is ignored.

The drive digital input S1 does not control the run forward command of drive as the Fwd CMD block assumes that control. If either Fwd CMD, Rev CMD, or both are used in a program, S1 and S2 do not control the run/stop of the drive. This is true for 2-wire or 3-wire initialization.

If both the Fwd CMD and Rev CMD are TRUE, “EF” will appear on the digital operator and the drive will ramp to a stop, then baseblock, regardless of control method.

◆ Run REV Command



Block Name: Rev CMD

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	LOGIC	True = Drive runs in the reverse direction

Block Properties – Rev CMD

Only [ID] and Label properties are present.

Description

The logic input to the Rev CMD block is used to control the reverse run command. When the input data is TRUE, the reverse run command is given. When this block is programmed into the connection sequence, control of the run commands are assumed by DriveWorksEZ™ and the setting of parameter b1-02 is ignored.

The drive digital input S2 does not control the run reverse command of drive as the Rev CMD block assumes that control. If either Fwd CMD, Rev CMD, or both are used in a program, S1 and S2 do not control the run/stop of the drive. This is true for 2-wire or 3-wire initialization.

If both the Fwd CMD and Rev CMD are TRUE, an EF will appear on the digital operator and the drive will ramp to a stop, then baseblock, regardless of control method.

◆ Miscellaneous Logic Commands

Block Name: BB CMD, DCINJ, FSTOP, ASR I Rst, SPD TORQ, Zero Servo

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	LOGIC	Functions are described in the table below

Block Properties – BB CMD, DCINJ, FSTOP, ASR I Rst, SPD TORQ, Zero Servo

Only [ID] and Label properties are present.

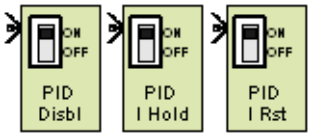
Description

There are several logic commands that duplicate some of the existing multi-function inputs. When these blocks are used, the logic commands are logically *OR*ed with the existing multi-function input. Further information on these functions can be found in the G7 drive technical manual (TM.G7.01) under *Multi-function Contact Input Functions.*

Block Name	Description	Equivalent Multi-Function Input (H1-0X =)
BB CMD	TRUE: Drive is base-blocked (output transistors are off) FALSE: No Effect	8
DC INJ	TRUE: DC Injection FALSE: No Effect	60
FSTOP	TRUE: Drive ramps to stop on the C1-09 decel ramp. FALSE: No Effect	15
ASR I Rst[1]	TRUE: ASR Integrator is reset to zero. FALSE: No Effect	E
SPD TORQ[1]	TRUE: Drive enters torque mode. FALSE: Drive enters speed mode.[2]	71
Zero Servo[3]	TRUE: Drive enters the zero servo mode FALSE: No Effect	72

1. Effective only when control method is V/f w/PG, Flux Vector, or Open Loop Vector 2 (A1-02 = 1, 3 or 4).
2. Effective only when drive is set to speed mode (d1-01 = 0).
3. Effective only when control method is Flux Vector (A1-02 = 3).

◆ PID Logic Command



Block Name: PID Disbl, PID I hold, PID I Rst

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	LOGIC	Functions are described in the table below

Block Properties – PID Disbl, PID I hold, PID I Rst

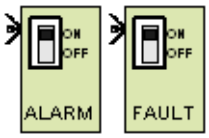
Only [ID] and Label properties are present.

Description

There are several logic commands that duplicate some of the existing multi-function inputs. These logic commands control the drive's internal PID process controller. When these blocks are used, the inputs are logically *OR*ed with the existing multi-function input. Further information on these functions can be found in the G7 drive technical manual (TM.G7.01) under *Multi-function Contact Input Functions*.

Block Name	Description	Equivalent Multi-Function Input (H1-0X =)
PID Disbl	TRUE: PID is disabled (equivalent of setting b5-01 = 0). FALSE: No Effect	19
PID I Hold	TRUE: PID Integrator holds at its present value. FALSE: No Effect	31
PID I Rst	TRUE: PID Integrator resets to zero. FALSE: No Effect	30

◆ Alarm and Fault Command



Block Name: ALARM, FAULT

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	LOGIC	Functions are described in the table below

Block Properties – ALARM, FAULT

Only [ID] and Label properties are present.

Description

Setting the logic input to the ALARM block to TRUE will cause a DWALM (DriveWorksEZ™ Alarm) to be displayed on the digital operator. The drive will continue to run. When the input is changed to FALSE, the alarm will clear.

Setting the logic input to the FAULT block to TRUE will cause a DWFAL (DriveWorksEZ™ Fault) to be displayed on the digital operator. The fault contact will close and the drive will coast to a stop. The fault must be reset via the digital operator, external terminals, or serial communication.

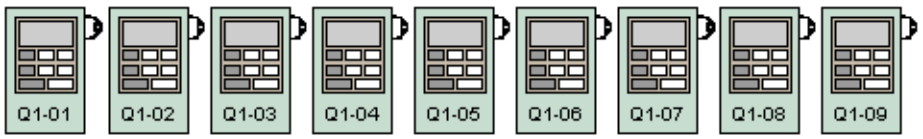
The run command on the G7 drive must be removed prior to allowing a reset of the DWFAL fault.

Keypad Input Tab Function Blocks

This table provides a description for each Keypad Input function block that appears in DriveWorksEZ™ and its location within the manual.

Block Name	Description	Page
Q1-01	Input	40
Q1-02	Input	40
Q1-03	Input	40
Q1-04	Input	40
Q1-05	Input	40
Q1-06	Input	40
Q1-07	Input	40
Q1-08	Input	40
Q1-09	Input	40
LOG PAR1	Input	41
LOG PAR2	Input	41
LOG PAR3	Input	41
LOG PAR4	Input	41
LOG PAR5	Input	41
LOG PAR6	Input	41
LOG PAR7	Input	41
LOG PAR8	Input	41

◆ Keypad Input – Keypad Input Function Blocks



Block Names: Q1-01, Q1-02, Q1-03, Q1-04, Q1-05, Q1-06, Q1-07, Q1-08, Q1-09

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	Numeric	Proportional to parameter setting

Block Properties – Q1-01 thru Q1-09

Only [ID] and Label properties are present.

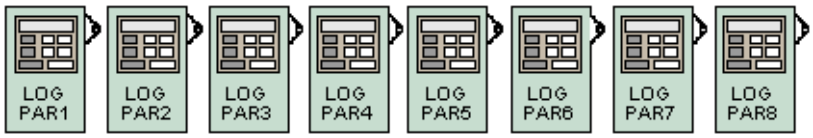
Description

These function blocks link a set of user keypad parameters of the G7 drive (Q1-XX) into the DriveWorksEZ™ Program.

Block	Output Type	Description	Unit	Range	Default	During Run
Q1-01	Numeric	Customer Param. 1	0.01%	0.00–655.35%	0.00%	Yes
Q1-02	Numeric	Customer Param. 2	0.01%	0.00–655.35%	0.00%	Yes
Q1-03	Numeric	Customer Param. 3	0.01%	0.00–655.35%	0.00%	Yes
Q1-04	Numeric	Customer Param. 4	0.01%	0.00–655.35%	0.00%	Yes
Q1-05	Numeric	Customer Param. 5	0.01%	0.00–655.35%	0.00%	Yes
Q1-06	Numeric	Customer Param. 6	0.01%	0.00–655.35%	0.00%	Yes
Q1-07	Numeric (bi-polar)	Customer Param. 7	0.01%	+/-999.9%[1]	0.00%	Yes
Q1-08	Numeric (bi-polar)	Customer Param. 7	0.01%	+/-999.9%[1]	0.00%	Yes
Q1-09	Numeric (bi-polar)	Customer Param. 7	0.01%	+/-999.9%[1]	0.00%	Yes

1. The hundredths decimal place is not shown for Q1-07 through Q1-09 due to display limitations. When these blocks/parameters are used in a program, a setting of 0.1% will show up in the block diagram as 0.10% or an integer value of 10.

◆ Keypad Input – LOGIC Inputs



Block Names: LOG PAR1, LOG PAR2, LOG PAR3, LOG PAR4, LOG PAR5, LOG PAR6, LOG PAR7, LOG PAR8

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	LOGIC	See description in table below

Block Properties – LOG PAR1, LOG PAR2, LOG PAR3, LOG PAR4, LOG PAR5, LOG PAR6, LOG PAR7, LOG PAR8

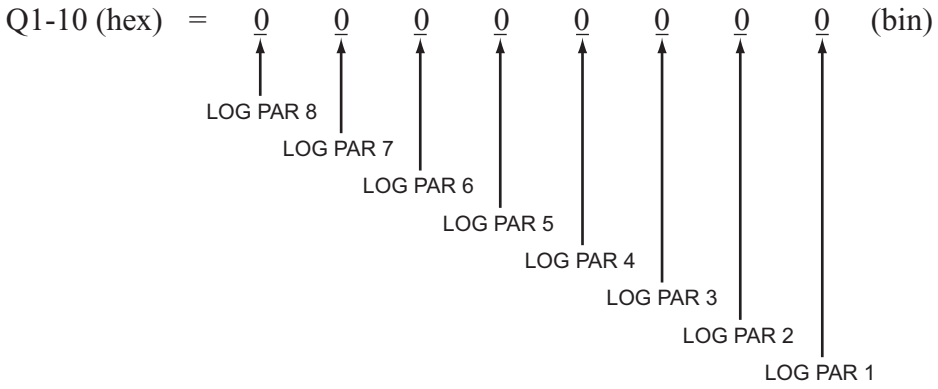
Only [ID] and Label properties are present.

Description

These function blocks link a keypad parameter (Q1-10) to eight discrete logic outputs in the G7 drive into the DriveWorksEZ™ Program.

Parameter Q1-10 Bit Definition Table

Block	Output Type	Description	Bit Position (Binary)	Bit Position (Hex)	Default	During Run
LOG PAR 1	LOGIC	Bit 1	xxxxxxx0	1h	0	Yes
LOG PAR 2	LOGIC	Bit 2	xxxxxx0x	2h	0	Yes
LOG PAR 3	LOGIC	Bit 3	xxxxx0xx	4h	0	Yes
LOG PAR 4	LOGIC	Bit 4	xxxx0xxx	8h	0	Yes
LOG PAR 5	LOGIC	Bit 5	xxx0xxxx	10h	0	Yes
LOG PAR 6	LOGIC	Bit 6	xx0xxxxx	20h	0	Yes
LOG PAR 7	LOGIC	Bit 7	x0xxxxxx	40h	0	Yes
LOG PAR 8	LOGIC	Bit 8	0xxxxxxx	80h	0	Yes



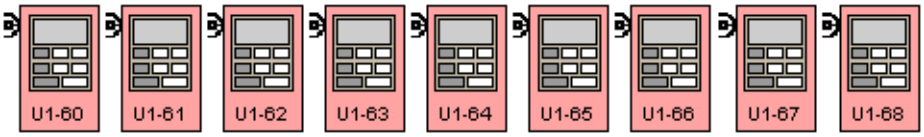
0 = FALSE, 1 = TRUE

Keypad Output Tab Function Blocks

This table provides a description for each Keypad Output function block that appears in DriveWorksEZ™ and its location within the manual.

Block Name	Description	Page
U1-60	Output	43
U1-61	Output	43
U1-62	Output	43
U1-63	Output	43
U1-64	Output	43
U1-65	Output	43
U1-66	Output	43
U1-67	Output	43
U1-68	Output	43
LOG MON1	Output	44
LOG MON2	Output	44
LOG MON3	Output	44
LOG MON4	Output	44
LOG MON5	Output	44
LOG MON6	Output	44
LOG MON7	Output	44
LOG MON8	Output	44

◆ Keypad Output – Monitors



Block Names: U1-60, U1-61, U1-62, U1-63, U1-64, U1-65, U1-66, U1-67, U1-68

Block Type: Output

Use: One time per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Keypad will display value present on block input

Block Properties – U1-60 thru U1-68

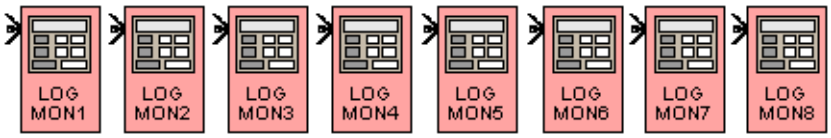
Only [ID] and Label properties are present.

Description

These function blocks link a set of user keypad monitors of the G7 drive (U1-60-U1-68) into the DriveWorksEZ™ Program.

Block	Input Type	Description	Unit	Range
U1-60	Numeric	Customer Monitor 1	0.01%	0–655.35%
U1-61	Numeric	Customer Monitor 1	0.01%	0–655.35%
U1-62	Numeric	Customer Monitor 2	0.01%	0–655.35%
U1-63	Numeric	Customer Monitor 3	0.01%	0–655.35%
U1-64	Numeric	Customer Monitor 4	0.01%	0–655.35%
U1-65	Numeric	Customer Monitor 5	0.01%	0–655.35%
U1-66	Numeric (bi-polar)	Customer Monitor 6	0.01%	+/- 327.67%
U1-67	Numeric (bi-polar)	Customer Monitor 7	0.01%	+/- 327.67%
U1-68	Numeric (bi-polar)	Customer Monitor 8	0.01%	+/- 327.67%

◆ Keypad Output – LOGIC Monitor



Block Names: LOG MON1, LOG MON2, LOG MON3, LOG MON4, LOG MON5, LOG MON6, LOG MON7, LOG MON8

Block Type: Output

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Input	LOGIC	See description in table below

Block Properties – U1-69

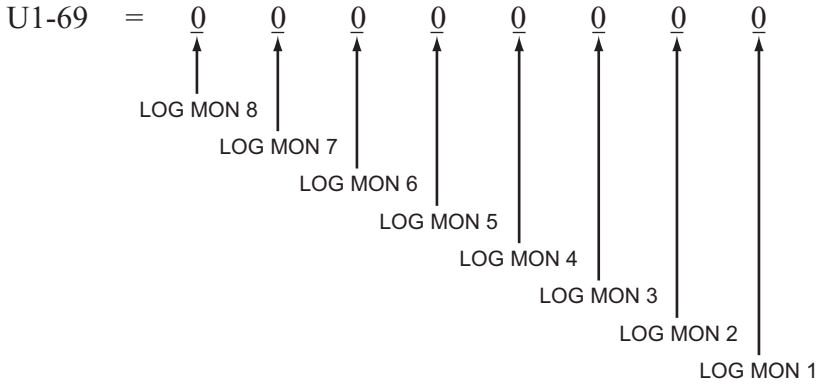
Only [ID] and Label properties are present.

Description

These function blocks link a set of logic outputs to a keypad monitor (U1-69) of the G7 drive into the DriveWorksEZ™ program.

Monitor U1-69 Bit Definition Table

Block	Output Type	Description	Bit Position
LOG MON 1	LOGIC	Bit 1	xxxxxxx0
LOG MON 2	LOGIC	Bit 2	xxxxxx0x
LOG MON 3	LOGIC	Bit 3	xxxxx0xx
LOG MON 4	LOGIC	Bit 4	xxxx0xxx
LOG MON 5	LOGIC	Bit 5	xxx0xxxx
LOG MON 6	LOGIC	Bit 6	xx0xxxxx
LOG MON 7	LOGIC	Bit 7	x0xxxxxx
LOG MON 8	LOGIC	Bit 8	0xxxxxxx



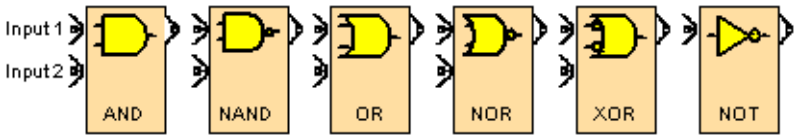
0 = FALSE, 1 = TRUE

Digital Function Tab Function Blocks

This table provides a description for each Digital Function function block that appears in DriveWorksEZ™ and its location within the manual.

Block Name	Description	Page
AND	Operation	46
NAND	Operation	46
OR	Operation	46
NOR	Operation	46
XOR	Operation	46
NOT	Operation	46
SRFF 1	Operation	47
SRFF 2	Operation	47
SRFF 3	Operation	47
TMR 1	Operation	48
TMR 2	Operation	48
TMR 3	Operation	48
1 Shot	Operation	49
INTVL TMR	Operation	50
0.1sec Pulses	Input	51
1.0sec Pulses	Input	51
Shift Bit 1	Input	52
Shift Bit 2	Input	52
Shift Bit 3	Input	52
Shift Bit 4	Input	52
Shift Cntrl	Output	53

◆ Digital Logic – Basic Gates



Block Names: AND, NAND, OR, NOR, XOR, NOT

Block Type: Operation

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Input(s)	LOGIC	See truth tables below
Output	LOGIC	See truth tables below

Block Properties – AND, NAND, OR, NOR, XOR, NOT

Only [ID] and Label properties are present.

Description

These blocks are used to provide basic logic functions in DriveWorksEZ™ projects. The following truth tables show how the logic blocks operate.

Truth Table – AND

Input 1	Input 2	Output
FALSE	FALSE	FALSE
TRUE	FALSE	FALSE
FALSE	TRUE	FALSE
TRUE	TRUE	TRUE

Truth Table – NAND

Input 1	Input 2	Output
FALSE	FALSE	TRUE
TRUE	FALSE	TRUE
FALSE	TRUE	TRUE
TRUE	TRUE	FALSE

Truth Table – OR

Input 1	Input 2	Output
FALSE	FALSE	FALSE
TRUE	FALSE	TRUE
FALSE	TRUE	TRUE
TRUE	TRUE	TRUE

Truth Table – NOR

Input 1	Input 2	Output
FALSE	FALSE	TRUE
TRUE	FALSE	FALSE
FALSE	TRUE	FALSE
TRUE	TRUE	FALSE

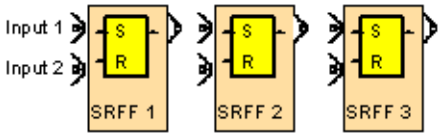
Truth Table – XOR

Input 1	Input 2	Output
FALSE	FALSE	FALSE
TRUE	FALSE	TRUE
FALSE	TRUE	TRUE
TRUE	TRUE	FALSE

Truth Table – NOT

Input 1	Output
FALSE	TRUE
TRUE	FALSE

◆ Digital Logic – Set/Reset Flip Flops



Block Names: SRFF 1, SRFF 2, SRFF 3

Block Type: Operation

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input 1	LOGIC	Set input to SR Flip-Flop
Input 2	LOGIC	Reset input to SR Flip-Flop
Output	LOGIC	See truth table below

Block Properties – SRFF 1, SRFF 2, SRFF 3

Only [ID] and Label properties are present.

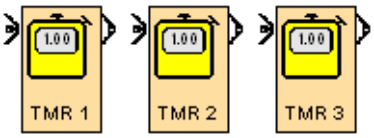
Description

SRFF1, SRFF2, and SRFF3 function like a set/reset flip-flop. Each has two logic inputs (set and reset) and one logic output. The truth table is as follows:

Truth Table – SRFF 1, SRFF 2, SRFF 3

Input 1	Input 2	Output
FALSE	FALSE	No Change
TRUE	FALSE	TRUE
FALSE	TRUE	FALSE
TRUE	TRUE	No Change

◆ Digital Logic – On-Delay Timers



Block Names: TMR 1, TMR 2, TMR 3

Block Type: Operation

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	LOGIC	Timer Input
Output	LOGIC	Timer Output

Block Properties – TMR 1

Name	Default	Min.	Max.	Parameter	Description
Timer 1 Delay	0.0sec	0.0	600.0	Q3-01	On-delay time for the TMR 1 block

Block Properties – TMR 2

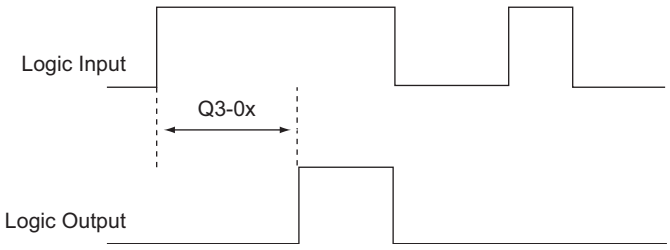
Name	Default	Min.	Max.	Parameter	Description
Timer 2 Delay	0.0sec	0.0	600.0	Q3-02	On-delay time for the TMR 2 block

Block Properties – TMR 3

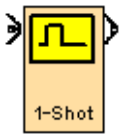
Name	Default	Min.	Max.	Parameter	Description
Timer 3 Delay	0.0sec	0.0	600.0	Q3-03	On-delay time for the TMR 3 block

Description

TIMR1, TMR2, and TMR3 all have one logic input and one logic output. The timers function as on-delay timers, with the delay set by Timer Delay parameters (Q3-01 through Q3-03). The logic input must be TRUE for at least the Q3-0X time in order for the logic output to change to TRUE.



◆ Digital Logic – 1-Shot Timer



Block Name: 1-Shot

Block Type: Operation

Use: Once per program

I/O Table

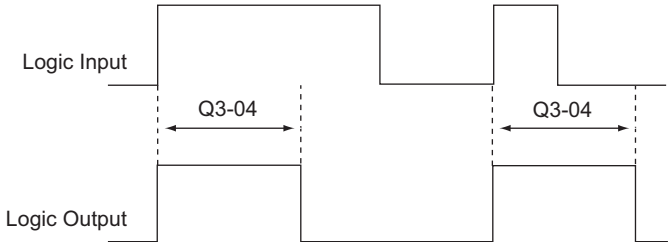
Connections	Signal Type	Description
Input	LOGIC	Timer Input
Output	LOGIC	Timer Output

Block Properties – 1 Shot

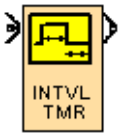
Name	Default	Min.	Max.	Parameter	Description
1-Shot On-Time	0.0sec	0.0	600.0	Q3-04	On-time for the 1-Shot block

Description

1-Shot has one logic input and one logic output. This is a timer with a variable pulse width. This timer will generate a pulse triggered on the leading edge of the timer input transition from FALSE to TRUE for the length of time set in Q3-04 regardless of the input state. The output will not go high again until the input transitions FALSE to TRUE after the output is FALSE.



◆ Digital Logic – Interval Timer



Block Name: INTVL TMR

Block Type: Operation

Use: Once per program

I/O Table

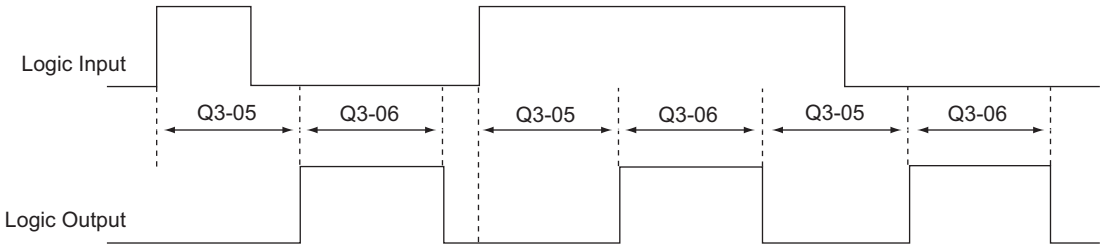
Connections	Signal Type	Description
Input	LOGIC	Timer Input
Output	LOGIC	Timer Output

Block Properties – INTVL TMR

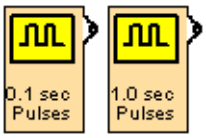
Name	Default	Min.	Max.	Parameter	Description
Interval Off-Time	0.0sec	0.0	600.0	Q3-05	Interval timer *Off* time
Interval On-Time	0.0sec	0.0	600.0	Q3-06	Interval timer *On* time

Description

INTVL TMR is an interval timer with variable on and off time settings. This timer will continue to time as long as the input is TRUE. The off-time will occur first, and then the on-time will occur. If the input is still TRUE when the on time is over, the cycle will repeat. If the input goes FALSE during the cycle, the cycle will always complete to the end of the on-time. The cycle will always repeat until the input is FALSE at the end of the on-time.



◆ Digital Logic – Pulse Generators



Block Names: 0.1sec Pulses, 1.0sec Pulses

Block Type: Input

Use: Multiple times per program

I/O Table

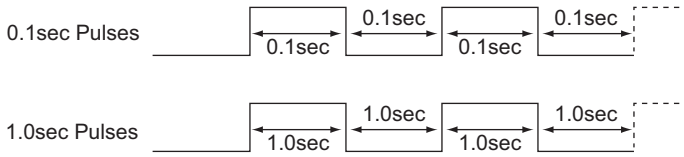
Connections	Signal Type	Description
Output	LOGIC	True/False output with 50% duty cycle

Block Properties – 0.1sec Pulses, 1.0sec Pulses

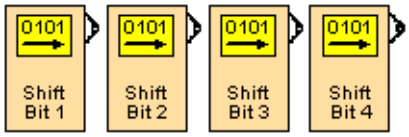
Only [ID] and Label properties are present.

Description

These blocks provide a continuous pulse train output that switches between TRUE and FALSE by the timing intervals as described below. This timing results in a pulse generator-type output.



◆ Digital Logic – 4-bit Shift Register Outputs



Block Name: Shift Bit 1, Shift Bit 2, Shift Bit 3, Shift Bit 4

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	LOGIC	Data output of shift register

Block Properties – Shift Bit 1, Shift Bit 2, Shift Bit 3, Shift Bit 4

Only [ID] and Label properties are present.

Description

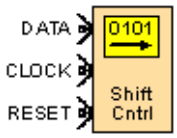
This block is used in conjunction with the Shift-Ctrl block. These four blocks represent the logic outputs of the 4-bit shift register.

Truth Table – 4-Bit Shift Register

Reset	Clock	Data	Shift bit 1	Shift bit 2	Shift bit 3	Shift bit 4
TRUE	X	X	FALSE	FALSE	FALSE	FALSE
FALSE	⎍	TRUE	TRUE	Shift Bit 1	Shift Bit 2	Shift Bit 3
FALSE	⎍	FALSE	FALSE	Shift Bit 1	Shift Bit 2	Shift Bit 3
FALSE	FALSE	X	Shift bit 1	Shift bit 2	Shift bit 3	Shift bit 4

⎍ = Rising Edge (FALSE to TRUE transition)
X = Do Not Care

◆ Digital Logic – 4-bit Shift Register Input



Block Name: Shift Cntrl

Block Type: Output

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	LOGIC	Data input for shift register
Input	LOGIC	Clock input for shift register
Input	LOGIC	Reset input for shift register

Block Properties – Shift Cntrl

Only [ID] and Label properties are present.

Description

This block is used in conjunction with the output blocks: Shift Bit 1, Shift Bit 2, Shift Bit 3, and Shift Bit 4. Shift + Ctrl is the input (control) block for a 4-bit serial shift register. The logic state present on the *Data* input is moved into the Shift Bit 1 block at the rising edge of the *Clock* input. At the same time, Shift Bit 2 is moved into Shift Bit 3, and Shift Bit 3 is moved into Shift Bit 4. The *Reset* input when TRUE immediately forces all of the Shift Bit blocks to FALSE.

Truth Table – 4-Bit Shift Register

Reset	Clock	Data	Shift Bit 1	Shift Bit 2	Shift Bit 3	Shift Bit 4
TRUE	X	X	FALSE	FALSE	FALSE	FALSE
FALSE	⎍	TRUE	TRUE	Shift Bit 1	Shift Bit 2	Shift Bit 3
FALSE	⎍	FALSE	FALSE	Shift Bit 1	Shift Bit 2	Shift Bit 3
FALSE	FALSE	X	Shift Bit 1	Shift Bit 2	Shift Bit 3	Shift Bit 4

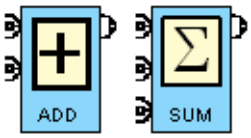
⎍ = Rising Edge (FALSE to TRUE transition)
X = Do Not Care

Numeric Function Tab Function Blocks

This table provides a description for each Numeric Function function block that appears in DriveWorksEZ™ and its location within the manual.

Block Name	Description	Page
ADD	Operation	55
SUM	Operation	55
SUB	Operation	56
MULT	Operation	57
ML10	Operation	57
ML100	Operation	57
MULP	Operation	57
DIV	Operation	58
DV10	Operation	58
DV100	Operation	58
DIVP	Operation	58
ABSV	Operation	60
NEGT	Operation	61
Scale 1	Operation	62
Scale 2	Input	62
DFILT	Input	63
DERV	Input	64
RAMP	Input	65
LIM1	Input	66
LIM2	Input	66
DZONE	Output	66

◆ Addition Functions



Block Names: ADD, SUM

Block Type: Operation

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	output = top input + bottom input (+ middle input)
Input	Numeric	
Input (SUM only)	Numeric	
Output	Numeric	

Block Properties – ADD, SUM

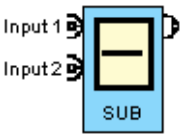
Only [ID] and Label properties are present.

Description

ADD: The numeric output of the ADD is the summation of its two numeric inputs.

SUM: The numeric output of the SUM is the summation of its three numeric inputs.

◆ Subtraction Functions



Block Name: SUB

Block Type: Operation

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Input (top)	Numeric	output = top input - bottom input
Input (bottom)	Numeric	
Output	Numeric	

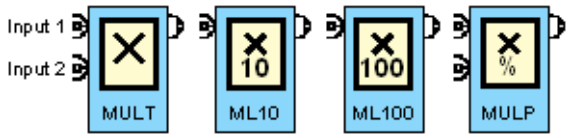
Block Properties – SUB

Only [ID] and Label properties are present.

Description

SUB: The numeric output of the SUB block is produced by subtracting numeric input 2 (bottom input) from numeric input 1 (top input).

◆ Multiplication Functions



Block Names: MULT, ML 10, ML 100, MULP

Block Type: Operation

Use: Multiple times per program

I/O Table – MULT

Connections	Signal Type	Description
Input	Numeric	output = top input * bottom input
Input	Numeric	
Output	Numeric	

I/O Table – ML 10, ML 100

Connections	Signal Type	Description
Input	Numeric	ML 10 output = input * 10
Output	Numeric	ML 100 output = input * 100

I/O Table – MULP

Connections	Signal Type	Description
Input	Numeric	output = (top input * bottom input) / 10000
Input	Numeric	
Output	Numeric	

Block Properties – MULT, ML 10, ML 100, MULP

Only [ID] and Label properties are present.

Description

MULT: The numeric output of the MULT is the product of its two numeric inputs. The calculation is done in terms of integer values. Therefore, 1.00% x 1.00% = 100.00%. or (100 x 100 = 10000).

ML 10: Numeric output = input x 10

ML 100: Numeric output = input x 100

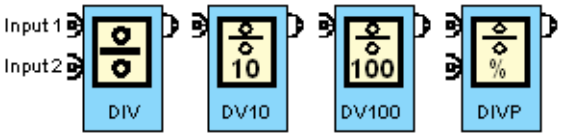
MULP: The numeric output of the MULP is the product of its two numeric inputs divided by 10000 (100.00%).

Example

If input 1 is 10000 (100.00%) and input 2 is 5000 (50.00%), the output of the MULP would be 5000 (50.00%).

The largest decimal number allowed is 2,147,483, 647.5 signed or hex (FFFF FFFF).

◆ Division Functions



Block Names: DIV, DV 10, DV 100, DIVP

Block Type: Operation

Use: Multiple times per program

I/O Table – DIV

Connections	Signal Type	Description
Input (top)	Numeric	output = top input / bottom input
Input (bottom)	Numeric	
Output	Numeric	

I/O Table – DV 10, DV 100

Connections	Signal Type	Description
Input	Numeric	DV 10 output = input / 10
Output	Numeric	DV 100 output = input / 100

I/O Table – DIVP

Connections	Signal Type	Description
Input (top)	Numeric	output = (top input / bottom input) * 10000
Input (bottom)	Numeric	
Output	Numeric	

Block Properties – DIV, DV 10, DV 100, DIVP

Only [ID] and Label properties are present.

Description

DIV: The numeric output of the DIV is produced by dividing the numeric value of the upper input (numerator) by the numeric value of the lower input (denominator). The calculation is done in terms of integer values.

Therefore, 100.00% / 100.00% = 0.01% or (10000/10000 = 1).

If the numeric value of the lower input (denominator) is 0, the output will equal the numeric value of the upper input.

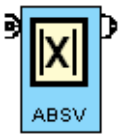
DIV10: Numeric output = input /10

DIV100: Numeric output = input /100

DIVP: The numeric output of the DIVP is produced by dividing the numeric value of the upper input by the numeric value of the lower input and then multiplying the result by 10000 (100.00%). If the numeric value of the upper input is 10000 (100.00%) and the numeric value of the lower input is 5000 (50.00%), the output of MULP would be 20000 (200.00%).

If the numeric value of the lower input (denominator) is zero, the output will equal the numeric value of the upper input * 10000.

◆ Absolute Value Function



Block Name: ABSV

Block Type: Operation

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	output = | input |
Output	Numeric	

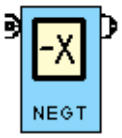
Block Properties – ABSV

Only [ID] and Label properties are present.

Description

The numeric output of the ABSV block is the absolute value of its numeric input.

◆ Inverse Sign Function



Block Name: NEGT

Block Type: Operation

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	output = input * (-1)
Output	Numeric	

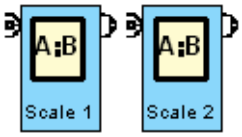
Block Properties – NEGT

Only [ID] and Label properties are present.

Description

The numeric output of the NEGT block is the negative inverse sign of the numeric input (change of sign).

◆ Scaling Function



Block Names: Scale 1, Scale 2

Block Type: Operation

Use: Only once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	output = (input * multiplier / divisor) + bias
Output	Numeric	

Block Properties – Scale 1

Name	Default	Min.	Max.	Parameter	Description
Scale Multiplier	1	-9999	9999	Q4-03	Used to multiply an input by a ratio Then allows a bias to be added in See description below
Scale Divisor	1	-9999	9999	Q4-04	
Scale Bias	0.00%	-99.99	99.99	Q4-05	

Block Properties – Scale 2

Name	Default	Min.	Max.	Parameter	Description
Scale 2 Multiplier	1	-9999	9999	Q4-12	Used to multiply an input by a ratio Then allows a bias to be added in See description below
Scale 2 Divisor	1	-9999	9999	Q4-13	
Scale 2 Bias	0.00%	-99.99	99.99	Q4-14	

Description

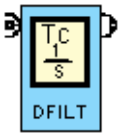
Scale 1 and Scale 2 have one numeric input and one numeric output. The Scale 1 output is calculated based on the Scale Multiplier (Q4-03), Scale Divisor (Q4-04), and Scale Bias (Q4-05) as follows:

Scale 1 Output = ((Input x Q4-03) / Q4-04) + Q4-05.

Scale 2 functions the same as Scale 1 using Q4-12 to Q4-14 instead of Q4-03 to Q4-05, respectively.

Scale 2 Output = ((Input x Q4-12) / Q4-13) + Q4-14.

◆ Delay Filter Function



Block Name: DFILT

Block Type: Operation

Use: Only once per program

I/O Table

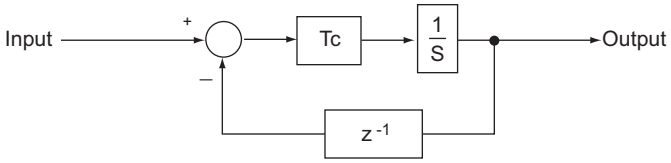
Connections	Signal Type	Description
Input	Numeric	1st order lag filter
Output	Numeric	

Block Properties – DFILT

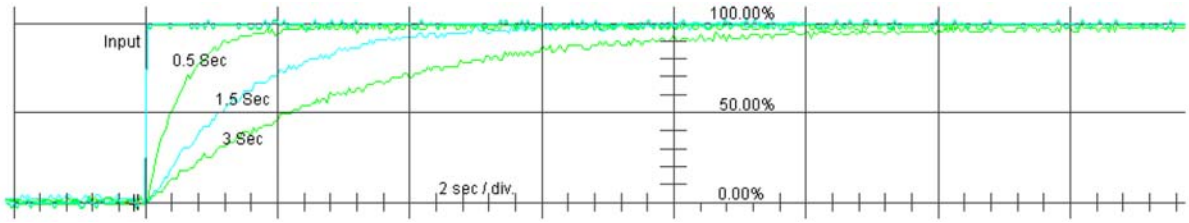
Name	Default	Min.	Max.	Parameter	Description
Delay Filter Time	0.5sec	0.0	3.0	Q4-02	Filter time constant

Description

The numeric output of the DFILT block filters the numeric input based on the time constant programmed in Delay Filter Time (Q4-02). This is a first order lag filter. The block diagram is shown below:



Output of DFILT block for various Q1-02 Settings



◆ Derivative Function

Block Name: DERV

Block Type: Operation

Use: Only once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	output = input (this scan) – input (last scan) (5ms scan time) or (10ms if A1-02 = Open Loop Vector 2)
Output	Numeric	

Block Properties – DERV

Only [ID] and Label properties are present.

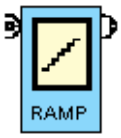
Description

DERV creates the derivative (change) of its input per each DriveWorksEZ™ scan.

Example

Scan 1 input is 35.45% and the next scan (scan 2) the value is 30.45%. The output of the DERV block at scan 2 will be: –5.00%.

◆ Ramp Function



Block Name: RAMP

Block Type: Operation

Use: Only once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Linear ramp function
Output	Numeric	

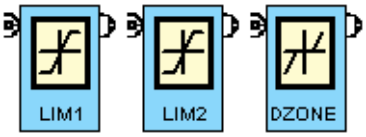
Block Properties – RAMP

Name	Default	Min.	Max.	Parameter	Description
Ramp Time	0.0sec	0.0	600.0	Q4-01	Time to ramp output signal from 0% to 100%

Description

RAMP functions as a soft-starter. An input step change from 0 to 100.00% will produce an output that ramps linearly from 0 to 100.00% in the time set by Q4-01. The rate is the same when the output is decreasing.

◆ Limit Functions



Block Names: LIM1, LIM2, DZONE

Block Type: Operation

Use: Only once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Output is limited per the parameters Q4-06~Q4-10
Output	Numeric	Configuration of limit type is determined by Q4-11

Block Properties – LIM 1

Name	Default	Min.	Max.	Parameter	Description
Upper Limit 1	0.00%	0.00	655.35	Q4-06	Upper limit for the LIM 1 block
Lower Limit 1	0.00%	0.00	655.35	Q4-07	Lower limit for the LIM 1 block
Limit / Dead Zone Properties	0h	0h	7Fh	Q4-11	Bits “0 0 0 0 0 0 X X” determine the properties for the LIM 1 block

Block Properties – LIM 2

Name	Default	Min.	Max.	Parameter	Description
Upper Limit 2	0.00%	0.00	655.35	Q4-08	Upper limit for the LIM 2 block
Lower Limit 2	0.00%	0.00	655.35	Q4-09	Lower limit for the LIM 2 block
Limit / Dead Zone Properties	0h	0h	7Fh	Q4-11	Bits “0 0 0 0 X X 0 0” determine the properties for the LIM 2 block

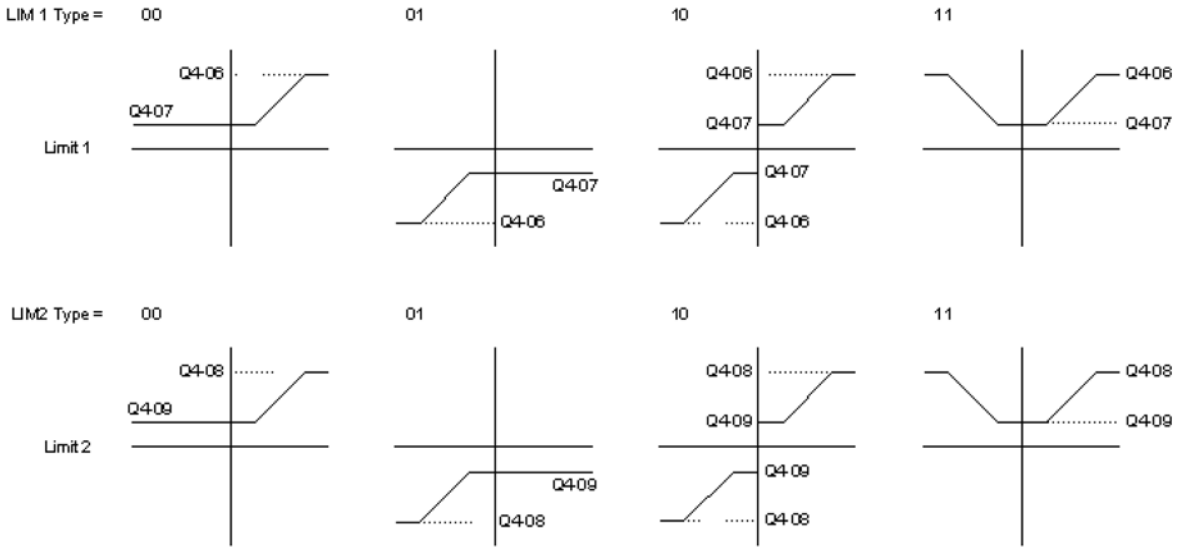
Block Properties – DZONE

Name	Default	Min.	Max.	Parameter	Description
Dead Zone	0.00%	0.00	655.35	Q4-10	Dead zone used with the DZONE block
Limit / Dead Zone Properties	0h	0h	7Fh	Q4-11	Bits “0 X X X 0 0 0 0” determine the properties for the DZONE block

Description

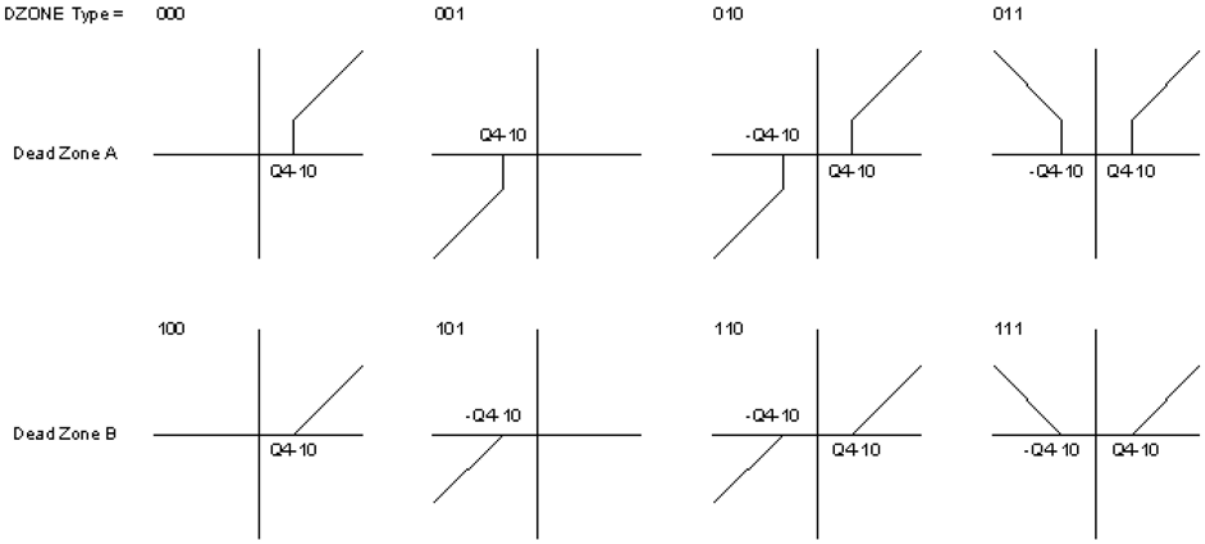
LIM1 and LIM2 are flexible upper and/or lower limit functions, which are configured by the setting of Limit/Dead Zone Properties (Q4-11) as shown below. Q4-06 and Q4-07 set the Upper and Lower limit for LIM1. Q4-08 and Q4-09 set the Upper and Lower limit for LIM2. Both LIM1 and LIM2 can either be positive, negative, bipolar, or absolute value type limits.

LIM 1/LIM 2 Configurations (x-axis = input, y-axis = output)



DZONE is used to zero data when it is below the set Dead Zone (Q4-10) value. DZONE can be positive, negative, bipolar, or absolute value and of type A or B, as shown below.

DZONE Configurations (x-axis = input, y-axis = output)



Q4-11 Limiter Set-up Description	Bits	Hex
Limiter 1 Positive only	xxxxxx00	00h
Limiter 1 Negative only	xxxxxx01	01h
Limiter 1 Bipolar	xxxxxx10	02h
Limiter 1 Absolute value	xxxxxx11	03h
Limiter 2 Positive only	xxxx00xx	00h
Limiter 2 Negative only	xxxx01xx	04h
Limiter 2 Bipolar	xxxx10xx	08h
Limiter 2 Absolute value	xxxx11xx	0Ch
D Zone A Positive Only	x000xxxx	00h
D Zone A Negative Only	x001xxxx	10h
D Zone A Bipolar	x010xxxx	20h
D Zone A Absolute Value	x011xxxx	30h
D Zone B Positive Only	x100xxxx	40h
D Zone B Negative Only	x101xxxx	50h
D Zone B Bipolar	x110xxxx	60h
D Zone B Absolute Value	x111xxxx	70h

Parameter Q4-11 can be set via the Properties window for each block (LIM 1, LIM 2, DZONE), or it can be set manually. To calculate the proper for Q4-11, choose one configuration for each block, then add up their hexadecimal values.

Example

Configure LIM 1 as a negative only, configure LIM 2 as a bi-polar, and configure DZONE as a *B* type Bipolar.

Q4-11 = (Lim 1, Neg Only) + (Lim 2, Bi Polar) + DZone, B-bi-polar)

Therefore: Q4-11 = 1hex + 8hex +60hex

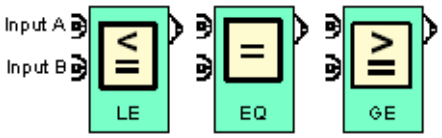
Answer: Q4-11 = 69h

Compound Function Tab Function Blocks

This table provides a description for each Compound Function function block that appears in DriveWorksEZ™ and its location within the manual.

Block Name	Description	Page
LE	Operation	70
EQ	Operation	70
GE	Operation	70
MOP	Operation	72
SMOP	Operation	73
CNTR	Operation	74
NUMS	Operation	75
NUML1	Operation	76
NUML2	Operation	76
NUML3	Operation	76
TMR4	Operation	77
PI	Operation	78
DIA	Operation	80
DWEZ ParW1	Operation	83
DWEZ ParW2	Operation	83

◆ Compare Functions



Block Names: LE, EQ, GE

Block Type: Operation

Use: Multiple times per program (see note below)

I/O Table

Connections	Signal Type	Description
Input	Numeric	Compare the two numeric inputs then provide a logic output
Input	Numeric	
Output	LOGIC	

Block Properties – LE

Name	Default	Min.	Max.	Parameter	Description
Compare (<=) Hysteresis	0.00%	0.00	10.0	Q5-09	Hysteresis amount for the LE compare function

Block Properties – EQ

Name	Default	Min.	Max.	Parameter	Description
Compare (=) Bandwidth	0.00%	0.00	10.0	Q5-10	Hysteresis amount for the EQ compare function

Block Properties – GE

Name	Default	Min.	Max.	Parameter	Description
Compare (>=) Hysteresis	0.00%	0.00	10.0	Q5-11	Hysteresis amount for the GE compare function

Description

LE, EQ, and GE are used to compare numeric data sources. Each has two numeric inputs and one logic output.

The logic output of LE will be TRUE if numeric Input A < numeric Input B when Q5-09 = 0.

The logic output of EQ will be TRUE if numeric Input A = numeric Input B when Q5-10 = 0.

The logic output of GE will be TRUE if numeric Input A => numeric Input B when Q5-11 = 0.

Parameters Q5-09, Q5-10, and Q5-11 set the hysteresis levels for LE, EQ, and GE, respectively. Hysteresis affects the reset or transition of TRUE to FALSE only. It does not affect the FALSE to TRUE transition. When a value is set in the hysteresis parameters Q5-09 to Q5-11, the blocks will function as below.

Hysteresis Examples

Input B = 10.00%

Q5-09, Q5-10, & Q5-11 all have 1.00% value.

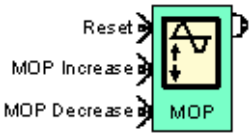
GE: Output is TRUE when Input A => 10.00%. To return to FALSE after TRUE, Input A < 9.00%.

EQ: Output is TRUE when Input A = 10.00%. To return to FALSE after TRUE, Input A < 9.00% or Input A > 11.00%.

LE: Output is TRUE when Input A =< 10.00%. To return to FALSE after TRUE, Input A > 11.00%.

If Q5-09 > 0.00%, LE can only be used once in a program. If Q5-11 > 0.00%, GE can only be used once in a program.

◆ Motor Operated Pot (MOP) Function



Block Name: MOP

Block Type: Operation

Use: Only once per program

I/O Table

Connections	Signal Type	Description
Input (Reset)	LOGIC	Provides a Motor Operated Pot functionality
Input (MOP Inc.)	LOGIC	
Input (MOP Dec.)	LOGIC	
Output	Numeric	

Block Properties – MOP

Name	Default	Min.	Max.	Parameter	Description
MOP Time	0.0sec	0.0	600.0	Q5-01	MOP ramp rate. The time it takes the output to go from 0% to 100%
MOP Minimum Value	0.0%	-999.9	999.9	Q5-02	Minimum allowed MOP output
MOP Maximum Value	0.0%	-999.9	999.9	Q5-03	Maximum allowed MOP output
MOP Reset Value	0.0%	-999.9	999.9	Q5-04	Output value when the Reset LOGIC input is active

The following must be true for the MOP block to work properly:
Q5-02 < = Q5-04 < = Q5-03.

Description

The operation of MOP is analogous to a motor operated pot. It has three logic inputs (Reset, Up, and Down) and one numeric output. When the *Up* input is TRUE, the output will increase by a rate based on the programmed time. When the *Down* input is TRUE, the output will decrease by a rate based on the programmed time. If the *Up* and *Down* inputs are either both TRUE or both FALSE the output value will not change. When the *Reset* input is TRUE, the output will be set to the MOP block Reset Value (Q5-04).

The MOP Time (Q5-01) sets the amount of time for the output to change from the min value to the max value, and vice versa. MOP Min Value (Q5-02) and MOP Max Value (Q5-03) set the upper and lower limits of the output, respectively.

◆ Step Motor Operated Pot (SMOP) Function

Block Name: SMOP

Block Type: Operation

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input (Reset)	LOGIC	Resets output to Q5-04 Value
Input (Step Up)	LOGIC	Increases output by Q5-05
Input (Step Down)	LOGIC	Decreases output by Q5-05
Output	Numeric	Step MOP output

Block Properties – SMOP

Name	Default	Min.	Max.	Parameter	Description
Step MOP Amount	0.0%	0.0	999.9	Q5-05	Sets the increment / decrement value for each step
Step MOP Min Value	0.0%	-999.9	999.9	Q5-06	Sets the minimum output of the block
Step MOP Max Value	0.0%	-999.9	999.9	Q5-07	Sets the maximum output of the block
Step MOP Reset Value	0.0%	-999.9	999.9	Q5-08	Sets the value of the output when the Reset input is true

IMPORTANT

The following must be true for the SMOP block to work properly:
Q5-06 < Q5-08 < Q5-07.

Description

SMOP is a combination MOP/counter. It has three logic inputs (Reset, Step Up, and Step Down) and one numeric output. A transition between scans from logic FALSE to a logic TRUE on the *Step Up* input results in the numeric output being incremented by the Step MOP Amount (Q5-05). A transition between scans from logic FALSE to a logic TRUE on the *Step Down* input results in the numeric output being decremented by the Step MOP amount (Q5-05).

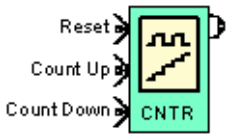
If both the *Step Up* and *Step Down* inputs transition from FALSE to TRUE between scans, the output will not change. When the *Reset* input is TRUE, the output will be set to the Step MOP Reset Value (Q5-08). The output will also be set to the Q5-08 setting upon power-up or when any of the connection blocks are changed.

Step MOP Min Value (Q5-06) and Step MOP Max Value (Q5-07) set the upper and lower limits of the output, respectively.

INFO

The output will also be set to the Q5-08 setting upon power-up or when any of the connection blocks are changed.

◆ Up/Down Counter (CNTR) Function



Block Name: CNTR

Block Type: Operation

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input (Reset)	LOGIC	Resets output to zero
Input (Count Up)	LOGIC	Increases output by 1
Input (Count Down)	LOGIC	Decreases output by 1
Output	Numeric	Counter output

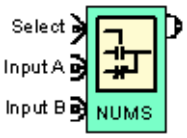
Block Properties – CNTR

Only [ID] and Label properties are present.

Description

CNTR functions as an Up/Down counter. A transition between scans from logic FALSE to a logic TRUE on the *Count Up* input results in the numeric output being incremented by 1. A transition between scans from logic FALSE to a logic TRUE on the *Count Down* input results in the numeric output being decremented by 1. If both the *Count Up* and *Count Down* inputs transition from FALSE to TRUE between scans, the output will not change. When the *Reset* input is TRUE, the output will be reset to 0. The numeric output can range from -2147483648 to 2147483647. If this range is exceeded, the counter will roll over to 0.

◆ Numerical Select (NUMS) Function



Block Name: NUMS

Block Type: Operation

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input (Select)	LOGIC	Switches output between Input A and Input B
Input (Input A)	Numeric	Moved to output when *Select* is TRUE
Input (Input B)	Numeric	Moved to output when *Select* is FALSE
Output	Numeric	Numeric Output

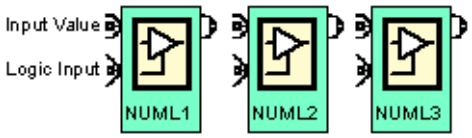
Block Properties – NUMS

Only [ID] and Label properties are present.

Description

NUMS is used to switch between two numeric inputs. This function operates as a two-channel analog selector switch. When the *Select* input is TRUE, numeric output is equal to numeric input A. When logic input A is FALSE, numeric output is equal to numeric input B.

◆ Numerical Latch Function



Block Names: NUML1, NUML2, NUML3

Block Type: Operation

Use: Once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	See Description Below
Input (Hold)	LOGIC	
Output	Numeric	

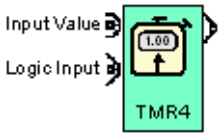
Block Properties – NUML1, NUML2, NUML3

Only [ID] and Label properties are present.

Description

NUML operates as a sample-hold function. It has one numeric input, one logic input (hold), and one numeric output. When the *Hold* input is FALSE, numeric output = numeric input. While the hold input is TRUE, the numeric output stays at the value that was present on the numeric input during the first scan that the hold input became TRUE. All three numeric latches function identically and each can only be used once in a program.

◆ On-Delay Timer (Numeric Adjustable)



Block Name: TMR4

Block Type: Operation

Use: Once per program

I/O Table

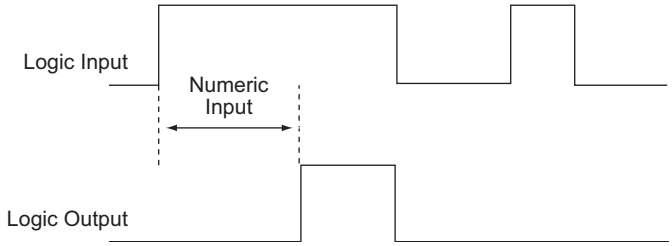
Connections	Signal Type	Description
Input	Numeric	Scaling 0.01% = 0.1sec
Input	LOGIC	Timer input
Output	LOGIC	Timer output

Block Properties – TMR4

Only [ID] and Label properties are present.

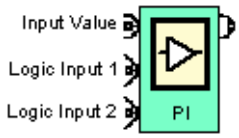
Description

TMR4 has one numeric input, one logic input, and one logic output. The function of the TMR4 is identical to TMR1, TMR2, and TMR3, except the delay time is set via the numeric input rather than by parameter. The numeric input scaling is 0.01% = 0.1sec, so a data value of 1.00% would set a 10.0sec delay. The delay time input must be limited to 65535 (6553.5sec) to ensure proper functioning of TMR4.



The scaling on the time setting of TMR4 is multiplied by ten so that a numeric input of 1.00% is not one second but ten seconds.

◆ Process Controller (PI) Function



Block Name: PI

Block Type: Operation

Use: Only once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	PI input (generally connected to *setpoint minus feedback*)
Input (Logic Input 1)	LOGIC	The function of these two inputs is determined by the setting of parameter Q6-02.
Input (Logic Input 2)	LOGIC	
Output	Numeric	PI output

Block Properties – PI

Name	Default	Min.	Max.	Parameter	Description
PI Properties	0	0	7	Q6-01	Selects how the PI block is configured 0: Normal acting uni-polar output 1: Inverse acting uni-polar output 2: Normal acting bi-polar output 3: Inverse acting bi-polar output 4 or 6: Normal acting absolute value output 5 or 7: Inverse acting absolute value output
PI Logic Input Select	16 (10 hex)	0	255 (FF hex)	Q6-02	Defines the action of the two logic inputs. See the table below for details
PI Proportional Gain	1.00	0.0	25.00	Q6-03	Proportional gain for the PI process controller
PI Integral Time	1.0sec	0.0	360.0	Q6-04	Integral time for the PI process controller
PI Integral Limit	100.0%	0.0	300.0	Q6-05	Limit for the PI process controller's integrator
PI Output Limit	100.0%	0.0	300.0	Q6-06	Overall output limit for the PI process controller

Parameter Q6-02 Description

	PI Controller Set-up Description	Decimal	Hexadecimal
Logic Input 1 Configuration	PI Disable	0	0h
	PI Integral Reset	1	1h
	PI Integral Hold	2	2h
	PI Inverse Act Enabled	3	3h
	PI Bypass	4	4h
Logic Input 2 Configuration	PI Disable	0	0h
	PI Integral Reset	16	10h
	PI Integral Hold	32	20h
	PI Inverse Act Enabled	48	30h
	PI Bypass	64	40h

Example

Configure Logic Input 1 to *Integral Hold* and configure Logic Input 2 to *Bypass*.

(Value for the DWEZ™ PC Tool) Q6-02 = (Logic Input 1 = Integral Hold) + (Logic Input 2 = PI Bypass)

Solution: 66 (decimal) = 2 + 64

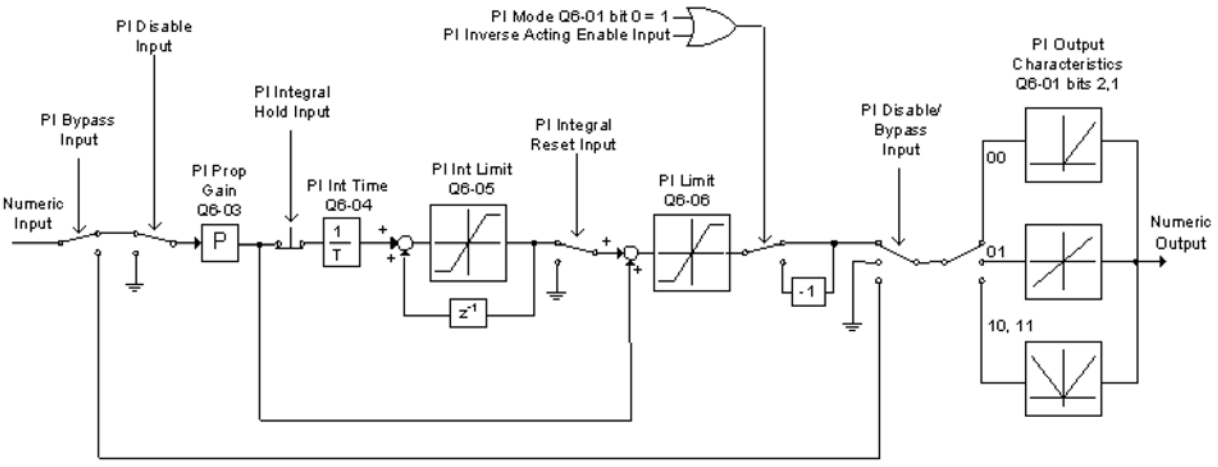
Convert 66 to hexadecimal and manually enter results via the keypad.

(drive keypad value) **Q4-11 = 42(hex)**

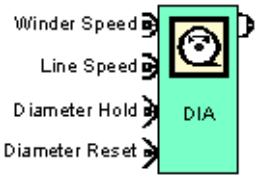
INFO

If parameter Q6-01 is programmed as “inverse acting”, the LOGIC input *PI Inverse Act Enabled* has no effect. Also, if both LOGIC inputs are programmed to the same function, the input will be ORed together.

Description



◆ Diameter Calculator Function



Block Names: DIA

Block Type: Operation

Use: Only once per program

I/O Table

Connections	Signal Type	Description
Input (Winder Spd)	Numeric	Winder spindle speed
Input (Line Spd)	Numeric	Line Speed (winder surface speed)
Input (Dia. Hold)	LOGIC	Holds the output at its present value
Input (Dia. Reset)	LOGIC	Resets the output to the value stored in Q6-10
Output [1]	Numeric	Ratio of core diameter over calculated diameter (multiplied by 100.00%)

1. All diameter values are entered and displayed as a ratio of "(core diameter / full roll diameter) * 100.00%."

Care must be taken in applying this block to a specific unwind or rewind application. This block is intended for flux vector, closed-loop only and should only be applied following all of the limits of application (inclusive) indicated below.

Limits of Application:

- Maximum line speed = 2000 FPM.
- Maximum diameter build/unbuild ratio = 20:1.
- Maximum wind/unwind drive dynamic range = 200:1 (minimum - maximum controller speed).
- Maximum Torque control range (CTCW) = 10:1.

Block Properties – DIA

Name	Default	Min.	Max.	Parameter	Description
Diameter Integral Time	20.0sec	0.0	600.0	Q6-07	Time it takes for the output of the DIA block to ramp from 0 to 100%
Max. Build Ratio	20.00	1.00	20.00	Q6-09	Maximum build ratio, determined by the following formula: $\frac{\textit{Full Roll Diameter}}{\textit{Core Diameter}}$
Diameter Reset Value [1]	100.0%	5.0	100.0	Q6-10	Value loaded into DIA output integrator when the *Dia Reset* logic input is TRUE. Calculated by the following formula: $\frac{\textit{Core Diameter}}{\textit{Reset Diameter}} * 100.0\%$
Diameter Integrator Deadband	2.00%	0.00	10.00	Q6-11	Amount of diameter change before the output of the DIA block integrates that change
Diameter Filter Time	0.5sec	0.0	3.0	Q6-12	1st order lag filter on the output of the internal divide calculation
Diameter Properties	3	1	3	Q6-13	Determines diameter calculation mode 1: Increase Only (Unwind) 2: Decrease Only (Wind) 3: Increase and Decrease allowed

1. Diameter reset value is entered and displayed as a ratio of "(core diameter / diameter) * 100.00%."

Example

Winder Core Diameter = 3", Full Roll Diameter = 15", Desired reset value = 3".

Maximum Build Ratio - (15" / 3"), then **Q6-09 = 5.00**

Diameter Reset Value - (3" / 3") * 100.0%, then **Q6-10 =100.0%**

The Diameter Calculator output value is reset to the value of Q6-10 when the drive is powered down or a program connection block changes.

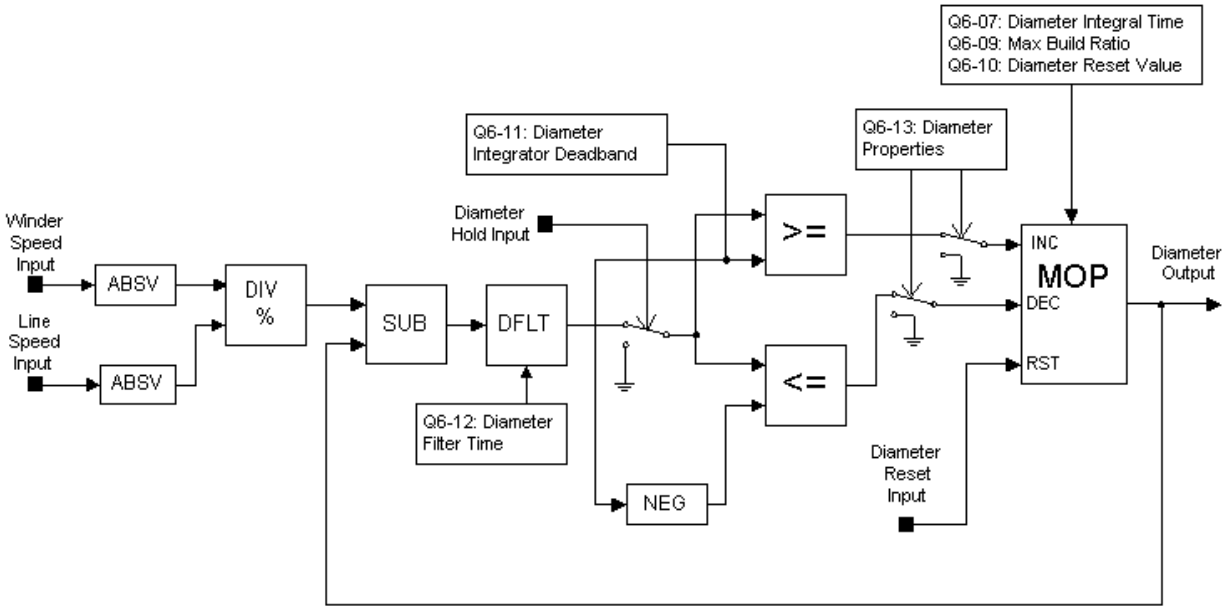
Description

The DIA block is a diameter calculator block that is used to calculate the diameter of an unwind or a rewind roll on a center driven winder. The block basically functions in this manner:

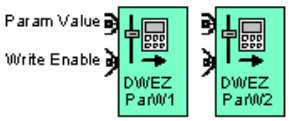
1. The absolute value of the winder (motor) speed is divided by the absolute value of the line speed in a divide by percent block.
2. A delay filter that operates the same as DFILT and filters the output of the divide result. The time delay setting of the delay filter is in parameter Q6-12.
3. A MOP then integrates the output plus a comparison circuit that compare the division output to the diameter integrator current value. As the division value increases or decreases, the diameter integrator will change its value to match the output of the division block. The output of the diameter integrator becomes the DIA block output. The minimum value of the diameter integrator (largest diameter roll) is controlled by parameter Q6-09 (Maximum Build Ratio). Internally, the minimum output value is calculated as follows: (1 / Q6-09) * 100.00%.
4. The diameter integrator can be reset to a preset value stored in parameter Q6-10.

5. The comparator circuit will move the diameter integrator up or down based on the when the divide value is greater or less than 0 (+/- 2.00%). The +/-2.00% will reduce the tendency of the circuit to follow small variation in motor speed due to tuning and process changes. This value is set in a parameter Q6-11.
6. It is possible to select the MOP to be able to move in one direction only (either UP/Increase or Down/decrease) or in both directions. This mode of operation is set by Diameter Properties parameter Q6-13.
7. When the winder speed matches the Line speed, the block considers this to be at the core diameter and the block output is at 100.00%. When the winder speed is a 50% of the line speed, the output is 50.00% and so on.
8. Multiplying the diameter calculated output and multiplying the Line speed plus trim in a % multiply block will yield the correct Winder speed or frequency. The internal block diagram of the DIA block is shown below.

Internal Block Diagram of the DIA Block



◆ Parameter Overwrite Functions



Block Names: DWEZ ParW1, DWEZ ParW2

Block Type: Operation

Use: Only once per program

I/O Table

Connections	Signal Type	Description
Input (Param Value)	Numeric	Numeric value overwrites the parameter as specified by parameter Q5-12 / Q5-13
Input (Write Enable)	LOGIC	Enables the parameter overwrite

Block Properties – DWEZ ParW1

Name	Default	Min.	Max.	Parameter	Description
DWEZ Parameter Write 1	0	0	22	Q5-12	Selects which parameter the DWEZ ParW1 block will overwrite. See table below for selections

Block Properties – DWEZ ParW2

Name	Default	Min.	Max.	Parameter	Description
DWEZ Parameter Write 2	0	0	22	Q5-13	Selects which parameter the DWEZ ParW2 block will overwrite. See table below for selections

Q5-12 / Q5-13 Setting	Overwritten Parameter	Description	Param Range	Numeric Range	Scale
0	None	Not Used	-	-	-
1	Q2-04	AI -14B Channel 1 Gain	0.0–100.00%	0.0–100.00%	100.00% = 100.0
2	Q2-05	AI -14B Channel 1 Bias	-100.0–100.0%	-100.0–100.0%	100.00% = 100.0
3	Q2-06	AI -14B Channel 2 Gain	0.0–100.00%	0.0–100.00%	100.00% = 100.0
4	Q2-07	AI -14B Channel 2 Bias	-100.0–100.0%	-100.0–100.0%	100.00% = 100.0
5	Q2-08	AI -14B Channel 3 Gain	0.0–100.00%	0.0–100.00%	100.00% = 100.0
6	Q2-09	AI -14B Channel 3 Bias	-100.0–100.0%	-100.0–100.0%	100.00% = 100.0
7	Q3-01	Timer 1 Value	0.0–600.0sec	0.0–100.00%	100.00% = 600.0
8	Q3-02	Timer 2 Value	0.0–600.0sec	0.0–100.00%	100.00%=600.0sec
9	Q3-03	Timer 3 Value	0.0–600.0sec	0.0–100.00%	100.00%=600.0sec
10	Q3-04	1 shot time	0.0–600.0sec	0.0–100.00%	100.00%=600.0sec
11	Q3-05	Int Timer On Delay	0.0–600.0sec	0.0–100.00%	100.00%=600.0sec
12	Q3-06	Int Timer Off Delay	0.0–600.0sec	0.0–100.00%	100.00%=600.0sec
13	Q4-01	Ramp Time	0.0–600.0sec	0.0–100.00%	100.00%=600.0sec
14	Q4-02	Delay Filter	0.0–3.0sec	0.0–100.00%	100.00%=3.0sec
15	Q5-01	MOP Time	0.0–600.0sec	0.0–100.00%	100.00%=600.0sec
16	Q5-05	Step MOP Amount	0.0–999.9%	0.0–100.00%	100.00% = 999.9%
17	Q6-03	PI Prop Gain	0.00–25.00	0.00–100.00%	100.00% = 25.00
18	Q6-04	PI Int. Time	0.0–360.0sec	0.00–100.00%	100.00%=360.0sec
19	Q6-05	PI Int. Limit	0.0–300.0%	0.00–100.00%	100.00%=300.0%
20	Q6-06	PI Limit	0.0–300.0%	0.00–100.00%	100.00%=300.0%
21	Q6-07	Dia Int Time	0.0–600.0sec	0.00–100.00%	100.00%=600.0sec
22	Q6-10	Diameter Reset Value	5.00–100.00%	5.0–100.00%	100.00%=100.00%

Description

The numeric input of the DWEZ ParW1 and DWEZ ParW2 blocks are used to overwrite the selected parameter with the numeric value present on the input. The DWEZ ParW1 and DWEZ ParW2 blocks will only perform this overwrite when the Write Enable LOGIC input is TRUE.

When the drive is first powered up, the selected parameter reverts back to its QX-XX setting, until the Write Enable LOGIC input is switched to TRUE. If the logic input is switched back to FALSE, the parameter retains the last value present on the input. See the example below for clarification.

Example

Block DWEZ ParW1 is set to overwrite Q3-01, the Timer 1 Value (Q5-12 = 7), and the stored value of the Timer 1 is 30.0 seconds **(Q3-01 = 30.0)**.

DWEZ Par W1 Block

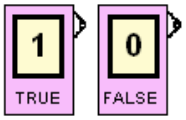
Description	LOGIC (Write Enable) Input	Numeric (Param Value) Input	Timer 1 actual time delay.
Drive is first powered up	FALSE	(don't care)	30.0sec
Because the logic input is false, the Timer 1 value remains at its Q3-01 value	FALSE	(don't care)	30.0sec
When the logic input is changed to TRUE, Timer 1's internal value changes	TRUE	10.00%	60.0sec
When the numeric input changes, Timer 1's internal value immediately changes	TRUE	20.00%	120.0sec
When the logic input is changed to FALSE, Timer 1's internal value retains the last valid input	FALSE	(don't care)	120.0sec
Drive is powered down, then powered back up, the Timer 1 value is reset to the Q3-01 value	FALSE	(don't care)	30.0sec

Constants Function Tab Function Blocks

This table provides a description for each Constants Function function block that appears in DriveWorksEZ™ and its location within the manual.

Block Name	Description	Page
TRUE	Input	87
FALSE	Input	87
100P	Input	88
50P	Input	88
10P	Input	88
0.1P	Input	88
0.09P	Input	88
0.08P	Input	88
0.07P	Input	88
0.06P	Input	88
0.05P	Input	88
0.04P	Input	88
0.03P	Input	88
0.02P	Input	88
0.01P	Input	88
0P	Input	88

◆ Constants – LOGIC



Block Names: TRUE, FALSE

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	LOGIC	Used where a different block requires a fixed logic input

Block Properties – TRUE, FALSE

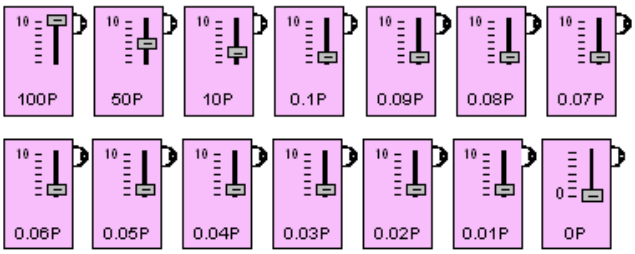
Only [ID] and Label properties are present.

BlockName	Output Value
TRUE	TRUE
FALSE	FALSE

Description

These blocks provide a constant logic state on the output.

◆ Constants – Numeric



Block Names: 100P, 50P, 10P, 0.1P, 0.09P, 0.08P, 0.07P, 0.06P, 0.05P, 0.04P, 0.03P, 0.02P, 0.01P, 0P

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	Numeric	Used where a different block requires a fixed numeric value

Block Properties – 100P, 50P, 10P, 0.1P, 0.09P, 0.08P, 0.07P, 0.06P, 0.05P, 0.04P, 0.03P, 0.02P, 0.01P, 0P

Only [ID] and Label properties are present.

The number in parenthesis is the actual integer value at the output.

Block Name	Output Value
100P	100.00% (10000)
50P	50.00% (5000)
10P	10.00% (1000)
0.1P	0.1% (10)
0.09P	0.09% (9)
0.08P	0.08% (8)
0.07P	0.07% (7)
0.06P	0.06% (6)
0.05P	0.05% (5)
0.04P	0.04% (4)
0.03P	0.03% (3)
0.02P	0.02% (2)
0.01P	0.01% (1)
0.00P	0.00% (0)

Description

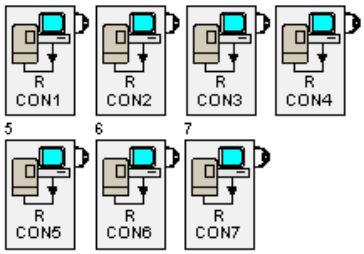
These blocks provide a constant value on their output.

Communications Tab Function Blocks

This table provides a description for each Communications function block that appears in DriveWorksEZ™ and its location within the manual.

Block Name	Description	Page
RCON1	Input	90
RCON2	Input	90
RCON3	Input	90
RCON4	Input	90
RCON5	Input	90
RCON6	Input	90
RCON7	Input	90
RCOL1	Input	91
RCOL2	Input	91
RCOL3	Input	91
RCOL4	Input	91
RCOL5	Input	91
RCOL6	Input	91
RCOL7	Input	91
RCOL8	Input	91
WCON1	Output	92
WCON2	Output	92
WCON3	Output	92
WCON4	Output	92
WCON5	Output	92
WCON6	Output	92
WCON7	Output	92
WCOL1	Output	93
WCOL2	Output	93
WCOL3	Output	93
WCOL4	Output	93
WCOL5	Output	93
WCOL6	Output	93
WCOL7	Output	93
WCOL8	Output	93

◆ Communications – Numeric Read



Block Names: RCON1, RCON2, RCON3, RCON4, RCON5, RCON6, RCON7

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	Numeric	Numeric output value is read from a modbus register (see table below)

Block Properties – RCON1, RCON2, RCON3, RCON4, RCON5, RCON6, RCON7

Only [ID] and Label properties are present.

Block Name	Modbus Address	Unit	Range
RCON1	06F0h	0.01%	0–635.35%
RCON2	06F1h	0.01%	0–635.35%
RCON3	06F2h	0.01%	0–635.35%
RCON4	06F3h	0.01%	0–635.35%
RCON5	06F4h	0.01%	+/- 327.67%
RCON6	06F5h	0.01%	+/- 327.67%
RCON7	06F6h	0.01%	+/- 327.67%

The Modbus Register values are reset to zero when the drive is powered down or the program changes (a connection block changes).

Description

These function blocks are used to read Modbus communication data. RCON1–RCON7 have one numeric output corresponding to Modbus registers 06F0h–06Fh, respectively. Data output from RCON1–RCON4 is interpreted as unsigned (0–65536). Data output from RCON5–RCON7 is interpreted as signed (-32768–32767).

Example

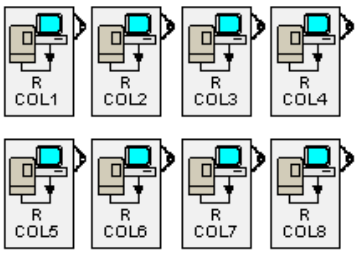
If a PLC or other Modbus controller were to write a value of 12345 into address 06F0h, the numeric output of the RCON1 block would be 123.45%.

RCON1–RCON7 and WCON1–WCON 7 blocks share the same Modbus addresses.

Example

Both RCON1 block and WCON1 block are Modbus 06F0h.

◆ Communications – Logic Read



Block Names: RCOL1, RCOL2, RCOL3, RCOL4, RCOL5, RCOL6, RCOL7, RCOL8

Block Type: Input

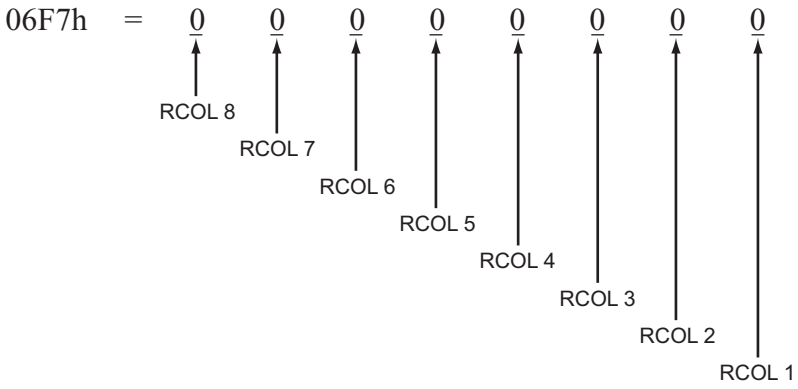
Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	LOGIC	LOGIC state is read from modbus register 06F7h (see diagram below)

Block Properties – RCOL1, RCOL2, RCOL3, RCOL4, RCOL5, RCOL6, RCOL7, RCOL8

Only [ID] and Label properties are present.



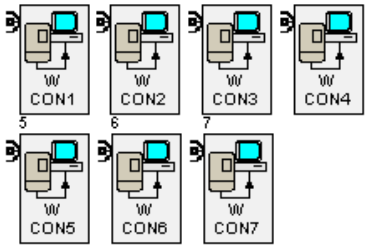
The Modbus Register values are reset to zero when the drive is powered down or the program changes (a connection block changes).

Description

RCOL1–RCOL8 blocks have one logic output corresponding to Modbus register 06F7h bits 0–7, respectively. When the corresponding bit is a **1** in the Modbus register 06F7h, the logic output of the respective RCOLx block will be TRUE.

The RCOLx and WCOLx blocks both use the same Modbus register (06F7h) and the same bits.

◆ Communications – Numeric Write



Block Names: WCON1, WCON2, WCON3, WCON4, WCON5, WCON6, WCON7

Block Type: Output

Use: Once

I/O Table

Connections	Signal Type	Description
Input	Numeric	Numeric input value is written to a modbus register (see table below)

Block Properties – WCON1, WCON2, WCON3, WCON4, WCON5, WCON6, WCON7

Only [ID] and Label properties are present.

Block Name	Modbus Address	Unit	Range
WCON1	06F0h	0.01%	0–635.35%
WCON2	06F1h	0.01%	0–635.35%
WCON3	06F2h	0.01%	0–635.35%
WCON4	06F3h	0.01%	0–635.35%
WCON5	06F4h	0.01%	+/- 327.67%
WCON6	06F5h	0.01%	+/- 327.67%
WCON7	06F6h	0.01%	+/- 327.67%

The Modbus Register values are reset to zero when the drive is powered down or the program changes (a connection block changes).

Description

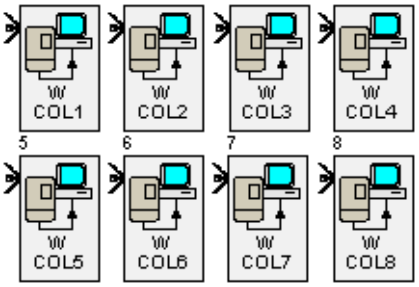
These function blocks are used to write Modbus communication data. WCON1–WCON7 blocks have one numeric input corresponding to Modbus registers 06F0h–06F6h, respectively. Data on the input of WCON1–WCON4 blocks is interpreted as unsigned (0–65536). Data on the input of WCON5–WCON7 blocks is interpreted as signed (-32768–32767).

Example

The numeric input to the WCON2 block is 23.45%. A PLC or other Modbus controller reading address 06F1h would read 2345.

RCON1–RCON7 and WCON1 to WCON 7 blocks share the same Modbus addresses.
Example
Both RCON1 block and WCON1 block are Modbus 06F0h.

◆ Communications – Logic Write



Block Names: WCOL1, WCOL2, WCOL3, WCOL4, WCOL5, WCOL6, WCOL7, WCOL8

Block Type: Output

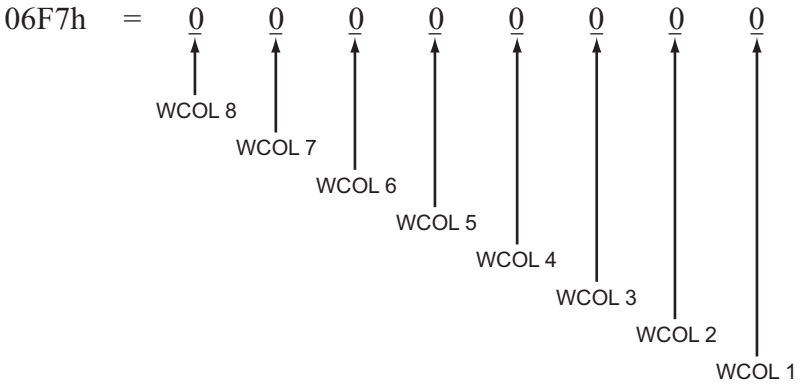
Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Input	LOGIC	LOGIC state is written to modbus register 06F7h (see diagram below)

Block Properties – WCOL1, WCOL2, WCOL3, WCOL4, WCOL5, WCOL6, WCOL7, WCOL8

Only [ID] and Label properties are present.



INFO

The Modbus Register values are reset to zero when the drive is powered down or the program changes (a connection block changes).

Description

WCOL1–WCOL8 blocks have one logic input corresponding to Memobus register 06F7h bits 0–7, respectively. When the corresponding WCOLx block input is TRUE, the respective bit in Modbus register 06F7h is set to 1.

IMPORTANT

The RCOLx and WCOLx blocks both use the same Modbus register (06F7h) and the same bits.

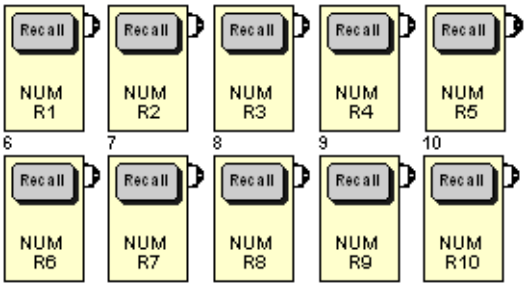
Temporary Register Tab Function Blocks

This table provides a description for each Temporary Register function block that appears in DriveWorksEZ™ and its location within the manual.

Block Name	Description	Page
NUMR1	Input	95
NUMR2	Input	95
NUMR3	Input	95
NUMR4	Input	95
NUMR5	Input	95
NUMR6	Input	95
NUMR7	Input	95
NUMR8	Input	95
NUMR9	Input	95
NUMR10	Input	95
LOGR1	Input	96
LOGR2	Input	96
LOGR3	Input	96
LOGR4	Input	96
LOGR5	Input	96
LOGR6	Input	96
LOGR7	Input	96
LOGR8	Input	96
LOGR9	Input	96
LOGR10	Input	96

Block Name	Description	Page
NUMW1	Output	97
NUMW2	Output	97
NUMW3	Output	97
NUMW4	Output	97
NUMW5	Output	97
NUMW6	Output	97
NUMW7	Output	97
NUMW8	Output	97
NUMW9	Output	97
NUMW10	Output	97
LOGW1	Output	98
LOGW2	Output	98
LOGW3	Output	98
LOGW4	Output	98
LOGW5	Output	98
LOGW6	Output	98
LOGW7	Output	98
LOGW8	Output	98
LOGW9	Output	98
LOGW10	Output	98

◆ Temporary Registers – Numeric Read



Block Names: NUMR1, NUMR2, NUMR3, NUMR4, NUMR5, NUMR6, NUMR7, NUMR8, NUMR9, NUMR10

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	Numeric	Numeric output value is associated with the NUMWX blocks

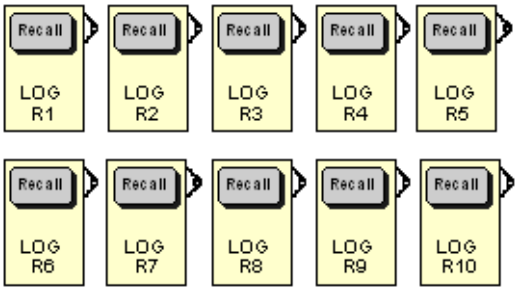
Block Properties – NUMR1, NUMR2, NUMR3, NUMR4, NUMR5, NUMR6, NUMR7, NUMR8, NUMR9, NUMR10

Only [ID] and Label properties are present.

Description

Numeric Read Blocks 1–10 are used to read numeric data previously stored by a Numeric Write block. These blocks have one numeric output which produces the data from the Numeric Write. NUMR1 to NUMR10 blocks correspond to NUMW1 to NUMW10 blocks. These blocks are often used to connect multiple inputs to one output, or to move a numeric value from one page of a project to another.

◆ Temporary Registers – Logic Read



Block Names: LOGR1, LOGR2, LOGR3, LOGR4, LOGR5, LOGR6, LOGR7, LOGR8, LOGR9, LOGR10

Block Type: Input

Use: Multiple times per program

I/O Table

Connections	Signal Type	Description
Output	LOGIC	LOGIC output state is associated with the LOGWX blocks

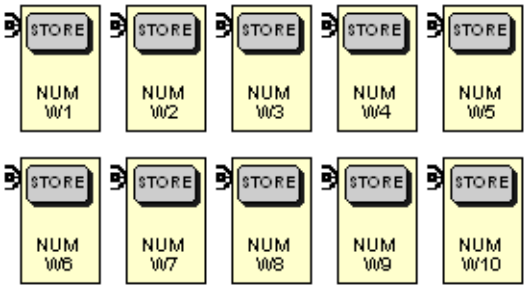
Block Properties – LOGR1, LOGR2, LOGR3, LOGR4, LOGR5, LOGR6, LOGR7, LOGR8, LOGR9, LOGR10

Only [ID] and Label properties are present.

Description

Logic Read Blocks 1–10 are used to read logic data previously stored by a Logic Write block. These blocks have one logic output which produces the data from the Logic Write. LOGR1 to LOGR10 blocks correspond to LOGW1 to LOGW10 blocks. These blocks are often used to connect a LOGIC output to multiple LOGIC inputs, or to move a LOGIC value from one page of a project to another.

◆ Temporary Registers – Numeric Write



Block Names: NUMW1, NUMW2, NUMW3, NUMW4, NUMW5, NUMW6, NUMW7, NUMW8, NUMW9, NUMW10

Block Type: Output

Use: Only once per program

I/O Table

Connections	Signal Type	Description
Input	Numeric	Sends the value of the numeric input to the associated NUMRX block

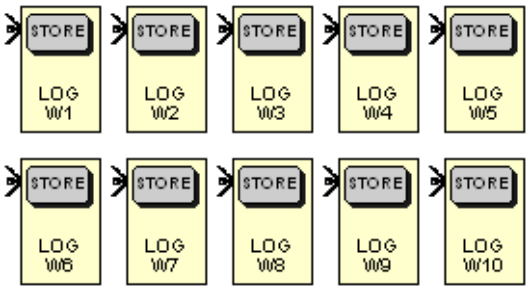
Block Properties – NUMW1, NUMW2, NUMW3, NUMW4, NUMW5, NUMW6, NUMW7, NUMW8, NUMW9, NUMW10

Only [ID] and Label properties are present.

Description

Numeric Write Blocks 1–10 are used to write numeric data that can subsequently be read by a Numeric Read block. These blocks have one numeric input which stores the data. NUMW1 to NUMW10 blocks correspond to NUMR1 to NUMR10 blocks. These blocks are often used to connect multiple inputs to one output, or to move a numeric value from one page of a project to another.

◆ Temporary Registers – Logic Write



Block Names: LOGW1, LOGW2, LOGW3, LOGW4, LOGW5, LOGW6, LOGW7, LOGW8, LOGW9, LOGW10

Block Type: Output

Use: Only once per program

I/O Table

Connections	Signal Type	Description
Input	LOGIC	Sends the state of the LOGIC input to the associated NUMWX block

Block Properties – LOGW1, LOGW2, LOGW3, LOGW4, LOGW5, LOGW6, LOGW7, LOGW8, LOGW9, LOGW10

Only [ID] and Label properties are present.

Description

Logic Write Blocks 1–10 are used to write logic data that can subsequently be read by a Logic Read block. These blocks have one logic input which stores the data. LOGW1 to LOGW10 blocks correspond to LOGR1 to LOGR10 blocks. These blocks are often used to connect multiple inputs to one output, or to move a LOGIC value from one page of a project to another.

4

Appendix

This chapter provides a definition of terminology used throughout the manual, information on EtherNet/IP, EtherNet Modbus TCP/IP, and sample programs.

Definition of Terminology

The following terms are used throughout this manual.

◆ General Terminology

Application program – The program created by the developer using the DriveWorksEZ™ PC tool. This program is downloaded to the G7 drive to customize it for specific application functionality.

DriveWorksEZ™ Object – The software inside the G7 drive that is separate from the normal drive software that provides the functionality of DriveWorksEZ™.

Firmware – The custom software residing in the G7 drive (ROM area) that contains the DriveWorksEZ™ function blocks.

Function blocks – The building blocks available to create a DriveWorksEZ™ application program. These provide access to drive I/O points, math functions, logic, etc.

PC tool – The software that runs on the developer's PC and provides a graphical interface to the DriveWorksEZ™ function blocks used for creating, saving, monitoring, and editing an application program.

Project – A complete set of files used to create a DriveWorksEZ™ application program.

Project Navigator – A window in the file folder format that allows access and displays information in the project.

Properties – Changeable settings for a specific function block, a project, or the DriveWorksEZ™ PC tool.

◆ Function Block Terminology

Compile – An automatic process to convert the graphical block diagrams into a parameter connection list to be downloaded to the drive.

Input block – A block that is designed to provide input to an application program. This type of block has no inputs and one output.

Logic input – An input to a function block that is designed to receive a logical (True or False) value.

Logic output – A function block output that generates a logical (True or False) value.

Numeric input – An input to a function block that is designed to receive a numeric value.

Numeric output – A function block output that generates a numeric value.

Operation block – A block with one output that performs some operation on one or more inputs.

Output block – A block that is designed to provide output from the application program. This type of block has no outputs and one input.

Page – A container for a set of function blocks properly connected together. A project can have up to eight pages and unlimited subroutine pages.

User Parameter Tables

◆ DriveWorksEZ™ Keypad Input: Q1

Parameter Number	Name	Description	Factory Setting	Minimum	Maximum
Q1-01	Customer Parameter 1	User-settable values for use inside of a DWEZ program	0.00%	0.00	655.35
Q1-02	Customer Parameter 2		0.00%	0.00	655.35
Q1-03	Customer Parameter 3		0.00%	0.00	655.35
Q1-04	Customer Parameter 4		0.00%	0.00	655.35
Q1-05	Customer Parameter 5		0.00%	0.00	655.35
Q1-06	Customer Parameter 6		0.00%	0.00	655.35
Q1-07	Customer Parameter 7		0.00%	-327.67	327.67
Q1-08	Customer Parameter 8		0.00%	-327.67	327.67
Q1-09	Customer Parameter 9		0.00%	-327.67	327.67
Q1-10	Customer Parameter 10	User-settable LOGIC values for use inside of a DWEZ program	00h	0h	FFh

◆ DriveWorksEZ™ Drive I/O: Q2

Parameter Number	Name	Description	Factory Setting	Minimum	Maximum
Q2-01	Max Pulses CH1	Input frequency on PG card that will result in 100.00% output	32,020 Hz	0	65,535
Q2-02	Max Pulses CH2	Input frequency on PG card that will result in 100.00% output	32,020 Hz	0	65,535
Q2-03	Maximum Torque Reference	Limits the maximum torque reference or torque command supplied by this block	300.00%	0.00	300.00
Q2-04	AI-14B Ch1 Gain	Sets the gain level for Terminal TC1 input	100.0%	0.0	1000.0
Q2-05	AI-14B Ch1 Gain	Sets the bias level for the Terminal TC1 input	0.0%	-100.0	100.0
Q2-06	AI-14B Ch 2 Gain	Sets the gain level for Terminal TC2 input	100.0%	0.0	1000.0
Q2-07	AI-14B Ch2 Gain	Sets the bias level for the Terminal TC2 input	0.0%	-100.0	100.0
Q2-08	AI-14B Ch 3 Gain	Sets the gain level for Terminal TC3 input	100.0%	0.0	1000.0
Q2-09	AI-14B Ch3 Gain	Sets the bias level for the Terminal TC3 input	0.0%	-100.0	100.0
Q2-10	AI-14B Signal Level Select	Selects the signal level type for all option board terminals 0: 0 – 10VDC 1: +/- 10VDC 2: 4 – 20 mA (move all jumpers on option board to the “I” position)	0	0	2
Q2-11	Monitor Sel 1	Selects which U1 Monitor value is output	0	0	17
Q2-12	Monitor Sel 2	Selects which U1 Monitor value is output	0	0	17

Parameter Number	Name	Description	Factory Setting	Minimum	Maximum
Q2-13	ASR P Ramp Time	Sets the time it will take for the ASR P Gain to change from the minimum (0.00%) value to the maximum value (100.00%)	0.50sec	0.00	10.00
Q2-14	ASR I Ramp Time	Sets the time it will take for the ASR I Time to change from the minimum (0.00%) value to the maximum value (100.00%)	0.50sec	0.00	10.00

◆ DriveWorksEZ™ Logic Functions: Q3

Parameter Number	Name	Description	Factory Setting	Minimum	Maximum
Q3-01	Timer 1 Delay	On-delay time for the TMR 1 block	0.0sec	0.0	600.0
Q3-02	Timer 2 Delay	On-delay time for the TMR 2 block	0.0sec	0.0	600.0
Q3-03	Timer 3 Delay	On-delay time for the TMR 3 block	0.0sec	0.0	600.0
Q3-04	1 Shot On Time	On time for the 1-Shot block	0.0sec	0.0	600.0
Q3-05	Interval Off Time	Interval timer “Off” time	0.0sec	0.0	600.0
Q3-06	Interval On Time	Interval timer “On” time	0.0sec	0.0	600.0

◆ DriveWorksEZ™ Numeric Functions: Q4

Parameter Number	Name	Description	Factory Setting	Minimum	Maximum
Q4-01	Ramp Time	Time to ramp output signal from 0% to 100%	0.0sec	0.0	600.0
Q4-02	Delay Filter Time	Filter time constant	0.5sec	0.0	3.0
Q4-03	Scale Multiplier	Used to multiply an input by a ratio then allow a bias to be added in.	1	-9999	9999
Q4-04	Scale Divisor		1	-9999	9999
Q4-05	Scale Bias		0.00%	-99.99	99.99
Q4-06	Upper Limit 1	Upper limit for the LIM 1 block	0.00%	0.00	655.35
Q4-07	Lower Limit 1	Lower limit for the LIM 1 block	0.00%	0.00	655.35
Q4-08	Upper Limit 2	Upper limit for the LIM 2 block	0.00%	0.00	655.35
Q4-09	Lower Limit 2	Lower limit for the LIM 2 block	0.00%	0.00	655.35
Q4-10	Dead Zone	Dead zone used with the DZONE block	0.00%	0.00	655.35
Q4-11	Limit/Dead Zone Properties	Determines the properties for the LIM 1, LIM 2, and DZONE blocks	0h	0h	7Fh
Q4-12	Scale 2 Multiplier	Used to multiply an input by a ratio then allow a bias to be added in	1	-9999	9999
Q4-13	Scale 2 Divisor		1	-9999	9999
Q4-14	Scale 2 Bias		0.00%	-99.99	99.99

◆ DriveWorksEZ™ Compound Functions: Q5

Parameter Number	Name	Description	Factory Setting	Minimum	Maximum
Q5-01	MOP Time	MOP ramp rate. The time it takes the output to go from 0% to 100%	0.0 sec	0.0	600.0
Q5-02	MOP Minimum Value	Minimum allowed MOP output	0.0%	-999.9	999.9
Q5-03	MOP Maximum Value	Maximum allowed MOP output	0.0%	-999.9	999.9
Q5-04	MOP Reset Value	Output value when the Reset LOGIC input is active	0.0%	-999.9	999.9
Q5-05	Step MOP Amount	Sets the increment/decrement value for each step	0.0%	0.0	999.9
Q5-06	Step MOP Min Value	Sets the minimum output of the block	0.0%	-999.9	999.9
Q5-07	Step MOP Max Value	Sets the maximum output of the block	0.0%	-999.9	999.9
Q5-08	Step MOP Reset Value	Sets the value of the output when the Reset input is true	0.0%	-999.9	999.9
Q5-09	Compare (<=) Hysteresis	Hysteresis amount for the LE compare function	0.00%	0.00	10.0
Q5-10	Compare (=) Bandwidth	Hysteresis amount for the EQ compare function	0.00%	0.00	10.0
Q5-11	Compare (>=) Hysteresis	Hysteresis amount for the GE compare function	0.00%	0.00	10.0
Q5-12	DWEZ Parameter Write 1	Selects which parameter the DWEZ ParW1 block will overwrite	0	0	22
Q5-13	DWEZ Parameter Write 2	Selects which parameter the DWEZ ParW2 block will overwrite	0	0	22

◆ DriveWorksEZ™ Application Functions: Q6

Parameter Number	Name	Description	Factory Setting	Minimum	Maximum
Q6-01	PI Properties	Selects how the PI block is configured 0: Normal acting uni-polar output 1: Inverse acting uni-polar output 2: Normal acting bi-polar output 3: Inverse acting bi-polar output 4 or 6: Normal acting absolute value output 5 or 7: Inverse acting absolute value output	0	0	7
Q6-02	PI Logic Input Select	Defines the action of the two logic inputs	16 (10 hex)	0	255 (FF hex)
Q6-03	PI Proportional Gain	Proportional gain for the PI process controller	1.00	0.0	25.00
Q6-04	PI Integral Time	Integral time for the PI process controller	1.0sec	0.0	360.0
Q6-05	PI Integral Limit	Limit for the PI process controller's integrator	100.0%	0.0	300.0
Q6-06	Diameter Integral Time	Time it takes for the output of the DIA block to ramp from 0 to 100%	20.0sec	0.0	600.0

Parameter Number	Name	Description	Factory Setting	Minimum	Maximum
Q6-07	Max. Build Ratio	Maximum build ratio, determined by the following formula $\frac{Full\ Roll\ Diameter}{Core\ Diameter}$	20.00	1.00	20.00
Q6-08	Diameter Reset Value	Value loaded into DIA output integrator when the "Dia Reset" logic input is TRUE. Calculated by the following formula $\frac{Core\ Diameter}{Reset\ Diameter} * 100.0\%$	100.00%	5.0	100.0
Q6-09	Diameter Integrator Deadband	Amount of diameter change before the output of the DIA block integrates that change	2.00%	0.00	10.00
Q6-10	Diameter Filter Time	1st order lag filter on the output of the internal divide calculation	0.5sec	0.0	3.0
Q6-11	Diameter Properties	Determines diameter calculation mode 1: Increase Only (Unwind) 2: Decrease Only (Wind) 3: Increase and Decrease allowed	3	1	3

◆ DriveWorksEZ™ Monitors: U1

Parameter Number	Name	Description	Factory Setting	Minimum	Maximum
U1-60	Customer Monitor 1	Allows DWEZ program values to be displayed on the keypad	-	0.00	635.35
U1-61	Customer Monitor 2		-	0.00	635.35
U1-62	Customer Monitor 3		-	0.00	635.35
U1-63	Customer Monitor 4		-	0.00	635.35
U1-64	Customer Monitor 5		-	0.00	635.35
U1-65	Customer Monitor 6		-	0.00	635.35
U1-66	Customer Monitor 7		-	-327.67	327.67
U1-67	Customer Monitor 8		-	-327.67	327.67
U1-68	Customer Monitor 9		-	-327.67	327.67
U1-69	Customer Monitor 10	Allows DWEZ LOGIC values to be displayed on the keypad	-	-	-

Using DriveWorksEZ™ With Ethernet Communication

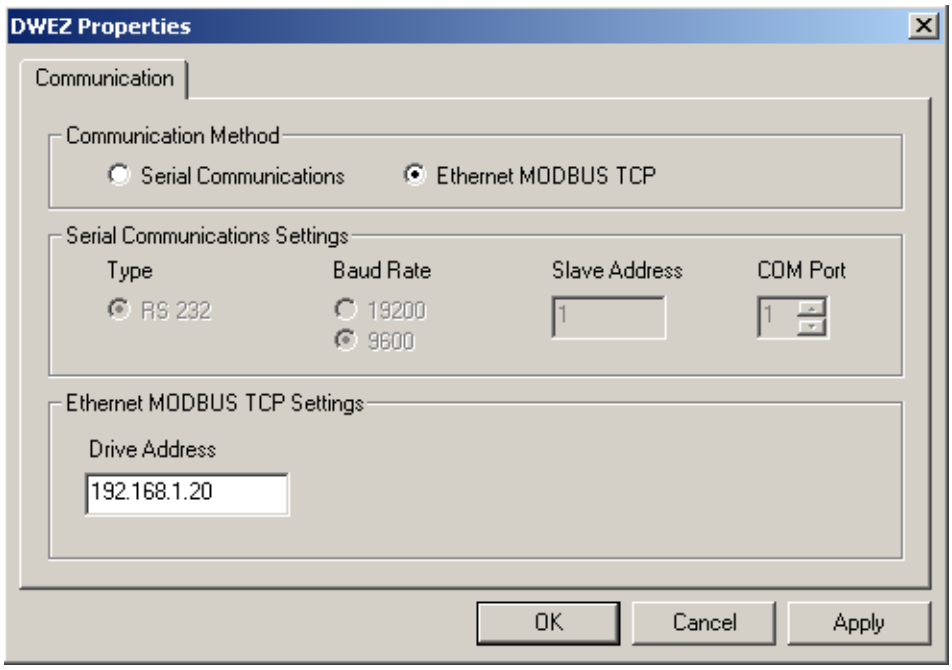
Ethernet communication can be used to connect to the G7 drive when either the CM090 (Modbus TCP/IP) or CM092 (EtherNet/IP) option kits are installed.

Before following the instructions below, first install and set up the (CM090 or CM092) option card according to their respective installation guides IG.AFD.25 and IG.AFD.26.

◆ Connecting to the Drive

To configure the DWEZ PC tool for use with the (CM090 or CM092) option card, there first must be an open project.

Select *File* > *DWEZ Properties*. The Properties page will appear.



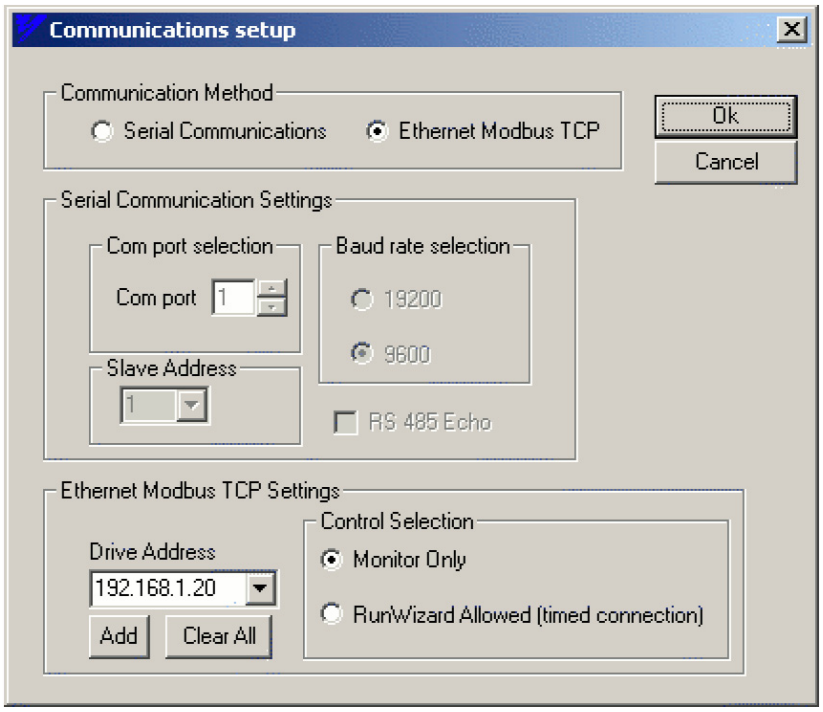
Change *Communication Method* to *Ethernet MODBUS TCP* and type in the IP address of the G7 drive in the *Drive Address* box. Click *OK*. Use this setting for the CM092 EtherNet/IP kit as well.

It should now be possible to connect to the drive using the *Connect* button on the toolbar or by selecting *Tools Connect.* If a *Failed to Connect to Drive* error message occurs, verify the IP address entered in the DWEZ Properties.

◆ Using DriveWorksEZ™ in Conjunction with DriveWizard™ and CM090 or CM092

It is very useful to have the capability to perform general drive parameter management, monitoring, and oscilloscope plotting while developing and tuning a DWEZ application program. When the (CM090 or CM092) option kit is installed on the G7 drive, DriveWorksEZ™ and DriveWizard™ can be connected to the drive simultaneously using Ethernet Communication.

To configure DriveWizard™ for Ethernet Modbus TCP/IP, select *Communications* > *Setup*.



Change *Communication Method* to *Ethernet Modbus TCP* and type in the IP address of the G7 drive in the *Drive Address* box. Click *OK*. Use this setting for the CM092 EtherNet/IP kit as well.

It should now be possible to connect to the drive using the *Connect* button on the toolbar or by selecting *Inverter Connect*. If an *Operation timed out* error message occurs, verify the IP address entered in the communications setup.

INFO

Please be aware of the following limitations and precautions when using DriveWizard™ with the DWEZ firmware.

- When the G7 drive is enabled for DWEZ operation, the Access Level A1-01 will be set to “423” (DWEZ Open Access). The DriveWizard™ assumes this to be an invalid setting and it will appear in red in the working value column for A1-01.
- The DWEZ parameters and monitors cannot be accessed by DriveWizard™. Q1-XX, Q2-XX, Q3-XX, Q4-XX, Q5-XX, Q6-XX, and U1-60 to U1-69 must be accessed through the DWEZ PC tool or G7 drive digital operator.
- The standard drive parameters shown below can be changed by the DWEZ PC tool. If the settings are overwritten by DriveWizard™, unexpected operation of the DWEZ application program may occur.

Standard Drive Parameters that exist within Function Blocks in DWEZ

Drive Parameter	DWEZ Block Name	Description
C1-01	SFS	Accel Time 1
C1-02	SFS	Decel Time 1
F3-01	DI16H	DI Input
H1-01	S3	Terminal S3 Selection
H1-02	S4	Terminal S4 Selection
H1-03	S5	Terminal S5 Selection
H1-04	S6	Terminal S6 Selection
H1-05	S7	Terminal S7 Selection
H1-06	S8	Terminal S8 Selection
H1-07	S9	Terminal S9 Selection
H1-08	S10	Terminal S10 Selection
H1-09	S11	Terminal S11 Selection
H1-10	S12	Terminal S12 Selection
H2-01	M1	Terminal M1-M2 Sel
H2-02	M3	Terminal M3-M4 Sel
H2-03	M5	Terminal M5-M6 Sel
H2-04	P3	Terminal P3 Sel
H2-05	P4	Terminal P4 Sel
H3-01	A1	Terminal A1 Signal
H3-02	A1	Terminal A1 Gain
H3-03	A1	Terminal A1 Bias
H3-04	A3	Terminal A3 Signal
H3-05	A3	Terminal A3 Selection

Drive Parameter	DWEZ Block Name	Description
H3-06	A3	Terminal A3 Gain
H3-07	A3	Terminal A3 Bias
H3-08	A2	Terminal A2 Signal
H3-09	A2	Terminal A2 Selection
H3-10	A2	Terminal A2 Gain
H3-11	A2	Terminal A2 Bias
H3-12	A3	Filter Average Time
H3-12	A2	Filter Average Time
H3-12	A1	Filter Average Time
H4-01	FM	Terminal FM Sel
H4-02	FM	Terminal FM Gain
H4-03	FM	Terminal FM Bias
H4-04	AM	Terminal AM Sel
H4-05	AM	Terminal AM Gain
H4-06	AM	Terminal AM Bias
H4-07	FM	AO Level Select 1
H4-08	AM	AO Level Select 2
H6-02	RP	Pulse Input Scaling
H6-03	RP	Pulse Input Gain
H6-04	RP	Pulse Input Bias
H6-05	RP	Pulse Input Filter Time
H6-06	MP	Pulse Moni Sel
H6-07	MP	Pulse Output Scaling

DriveWorksEZ™ Program Examples

These program examples are included with the DriveWorksEZ™ PC Tool installation. These programs have been fully tested at Yaskawa Electric America and are intended to show how an application can be solved by utilizing the various function blocks with the DriveWorksEZ™ program. These program examples are not intended to be a final solution to any application. These programs must be tailored to the specific application and proven on the actual machinery.

Below is a brief summary of each program included. See the details of the respective program.

■Valve Controller

This program shows how a process can be controlled by simple off/on control. A process variable is compared to a setpoint, and the error goes through a gain setting. The error then goes to a comparator circuit that either runs the drive (or another fixed speed motor) forward or reverse. The drive runs at a speed that is proportional to the amount of process error within a minimum and maximum speed range.

■Simple Pump Controller

This program shows how a generic duplex pump control can be accomplished. The program is modeled after a duplex wet well control where the speed and number of pumps “on” is determined by the level of liquid in a well or tank. As the level of liquid rises in the well or tank, first one pump and then the second (if required) will come on to steady or lower the level of liquid. The speed of the pumps is dependent on the level of liquid. The program has an alternator to switch which pump goes on first to even the wear on the pumps.

■Traverse Drive (P-Jump)

The program will generate a disturbed-sawtooth wave on top of the drive’s set speed. The amplitude, ramp time, and P-jump at transition from positive slope to negative and vice versa can be set by the user.

■Digital Ratio Follower

The program is used with the PG-W2 dual encoder feedback option card. The G7 drive with DWEZ is the slave drive and can follow a master encoder. A +/- draw setting allows the user to adjust the speed difference between the master and slave as a percent of master speed. There is a PI controller that can be switched off and on to provide dancer position trim control to the master reference. The PI controller output gain can be varied by master reference or a fixed user parameter setting.

■Simple Winder Control

This program shows how a simple speed regulated winder with dancer position trim can be accomplished. The winder will follow a master encoder line speed reference from channel 2 of a PG-W2 feedback option card. A diameter calculator block is used to derive the winding roll diameter. A PI controller is used to trim winder speed for proper dancer position.

Valve Controller – DriveWorksEZ™ Program Example

This program shows how a process can be controlled by simple off/on control. A process variable is compared to a setpoint and the error goes through a gain setting. The error then goes to a comparator circuit that either runs the drive (or another fixed speed motor) forward or reverse. The drive runs at a speed that is proportional to the amount of process error within a minimum and maximum speed range.

■Purpose of Program

The purpose of this program is to show the following principles:

- The ability to create an off /on type controller for closed process loop control

■Drive Physical Control Connections

- Process or flow reference signal to terminal A1
- Process or flow feedback signal to terminal A2
- S1 digital input for controller enable
- S8 digital input for forward over travel limit switch
- S9 digital input for reverse over travel limit switch
- M5-M6 contact output for over travel alarm indication

■Changes From Standard G7 Drive Software Functions

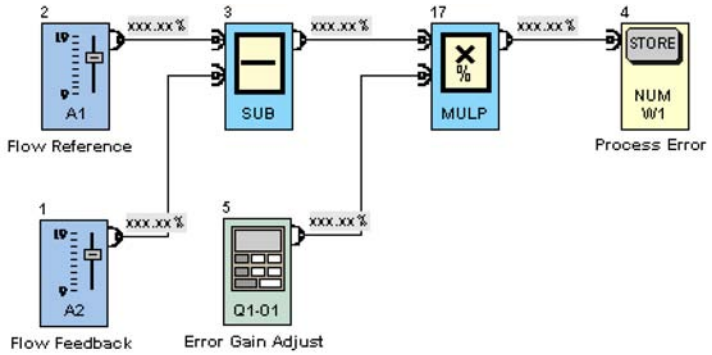
- Drive speed reference is from the DWEZ program only.
- Drive forward and reverse commands are from the DWEZ program only.

■Program Added Functions

- Proportional control with gain
- Forward reverse control based on process error
- Drive speed control based on process error
- End limit over travel alarm detection

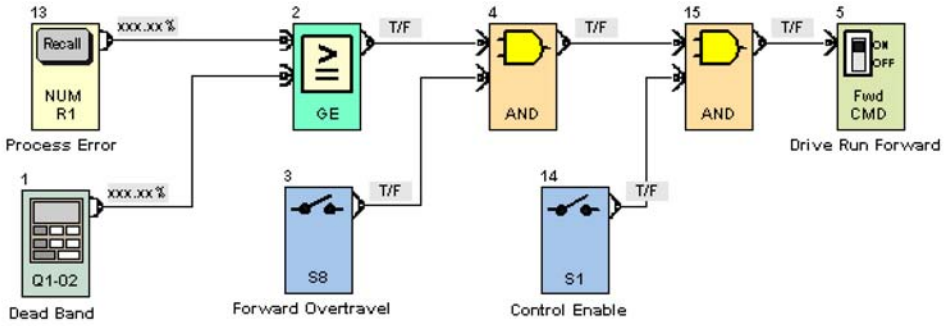
■Description of Operation

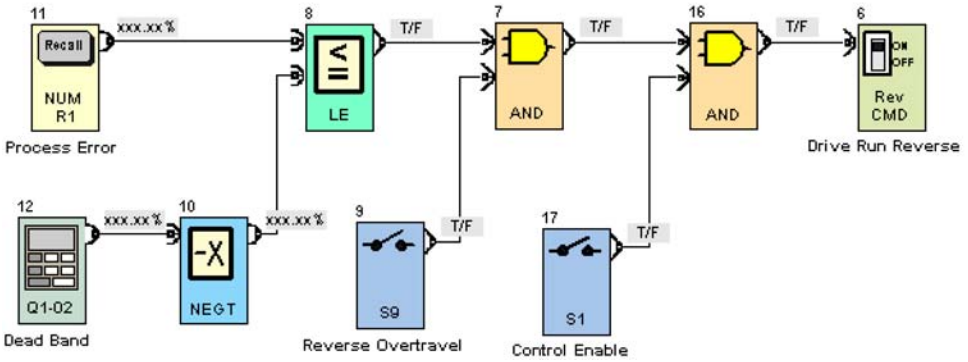
1. Process Error Page



Circuit name	Process error circuit
Function block name(s)	1(A2), 2(A1), 3(SUB), 4(NUMW1), 5(Q1-01), and 17(MULP)
Functional description	The value of the A1 block is subtracted from the value of the A2 block and multiplied (in percent) by the value of Q1-01. The resultant process error output is the input to the NUMW1 block, which passes the process error value to other pages.

2. Drive Direction Control Page

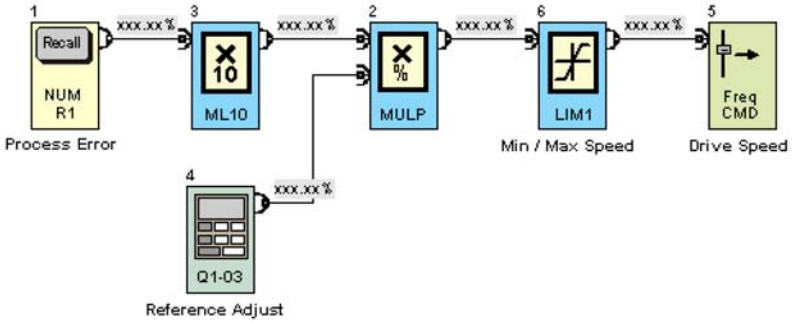




Circuit name	Drive run forward circuit
Function block name(s)	1(Q1-02), 13(NUMR1), 2(GE), 3(S8), 4(AND), 14(S1), 15(AND), and 5(Fwd CMD)
Functional description	When the process error is greater than or equal to the Q1-02 block setting, the GE block is TRUE. When the GE block is TRUE and the S8 block is TRUE (overtravel is NC, open on reaching the limit) and the control is enabled (S1=TRUE), then the input to the Fwd CMD block is TRUE.

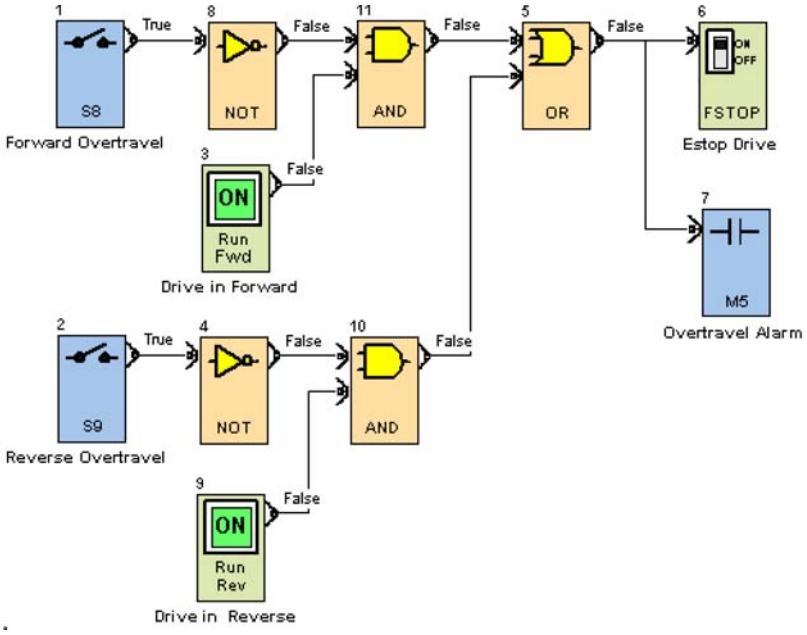
Circuit name	Drive run reverse circuit
Function block name(s)	12(Q1-02), 10(NEGT), 11(NUMR1), 8(LE), 9(S9), 7(AND), 17(S1), 16(AND), and 6(Rev CMD)
Functional description	When the process error is less than or equal to the negative Q1-02 block setting, the LE block is TRUE. When the GE block is TRUE and the S9 block is TRUE (overtravel is NC, open on reaching the limit) and the control is enabled (S1=TRUE), then the input to the Rev CMD block is TRUE.

3. Drive Speed Control Page



Circuit name	Drive speed control circuit
Function block name(s)	1(NUMR1), 3(ML10), 4(Q1-03), 2(MULP), 6(LIM1), and 5(Freq CMD)
Functional description	The process error is multiplied by 10 and then multiplied (in percent) by the value of Q1-03. Then the value goes through the LIM1 block which sets the minimum and maximum frequency reference. The output of the limit block is sent to the input of the Freq CMD block, which sets the drive frequency reference.

4. Over Travel Alarm Page



Circuit name	Forward end over travel alarm circuit
Function block name(s)	1(S8), 8(NOT), 3(RUNFwd), and 11(AND)
Functional description	If the S8 block goes FALSE while the drive is running forward, then the AND block will become TRUE.

Circuit name	Reverse end over travel alarm circuit
Function block name(s)	2(S9), 4(NOT), 9(RUNRev), and 10(AND)
Functional description	If the S9 block goes FALSE while the drive is running in reverse, the AND block will become TRUE.

Circuit name	Drive Estop circuit
Function block name(s)	5(OR), 6(FSTOP), and 7(M5)
Functional description	If either input to the OR block is TRUE, the drive will Estop by the FSTOP (Fast Stop) function block. and the M5-M6 contact is closed, indicating a over travel limit has been reached.

Simple Pump Controller – DriveWorksEZ™ Program Example

This program shows how a generic duplex pump control can be accomplished. The program is modeled after a duplex wet well control where the speed and number of pumps turned on is determined by the level of liquid in a well or tank. As the level of liquid rises in the well or tank, first one pump and then the second (if required) will come on to steady or lower the level of liquid. The speed of the pumps is dependent on the level of liquid. The program has an alternator to switch which pump goes on first to even the wear on the pumps.

■Purpose of Program

The purpose of this program is to show the following principles:

- The ability to control the run/stop and direction of the drive
- The ability to create state logic in a program
- The ability to sequence two drives (or drive and motor) in a controlled manner

■Drive Physical Control Connections

- Process feedback signal is connected to terminal A2 (4-20mA input).
- Digital input S8 is used to switch the lead pump assignment.
- Open contact output M5–M6 is used to start the other motor or drive.
- Analog output FM is used to supply a speed reference to the other drive.

■Changes From Standard G7 Drive Software Functions

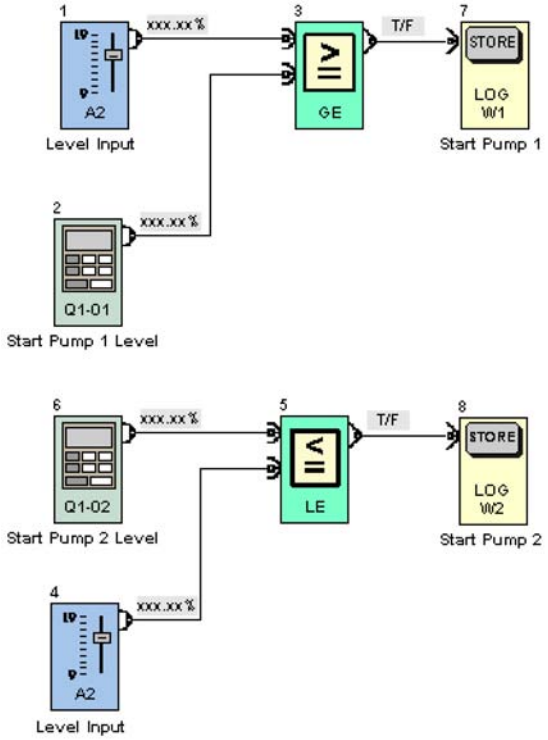
- Drive speed reference is from the DWEZ program only.
- Drive forward and reverse run commands are from the DWEZ program only.
- Drive I/O S8, M5, and FM are assigned to the DWEZ program only.

■Program Added Functions

- Ability to monitor a process signal input and turn on two different drives at pre-determined set points.
- Pump lead/lag control that switches the lead pump on every lead pump start or by digital input (S8).
- Speed reference set to allow one or two pumps, in parallel, to run at a value determined by the process variable input.

■Description of Operation

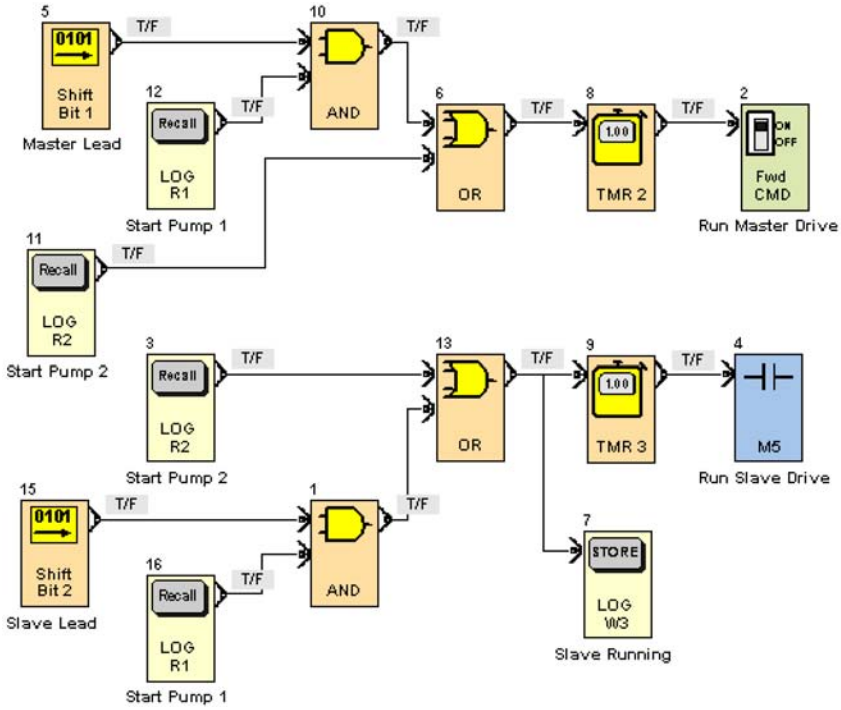
1. Level Detect Page



Circuit name	Start pump 1 circuit
Function block name(s)	1(A2), 2(Q1-01), 3(GE), and 7(LOGW1)
Functional description	When the A2 block input is greater than the Q1-01 block setting, logic store bit 1 is set to TRUE. Pump 1 turns on the designated lead pump on page 2.

Circuit name	Start pump 2 circuit
Function block name(s)	4(A2), 6(Q1-02), 5(LE), and 8(LOGW2)
Functional description	When the A2 block input is greater than the Q1-02 setting, logic store bit 2 is set to TRUE. Pump 2 turns on the designated lag pump on page 2.

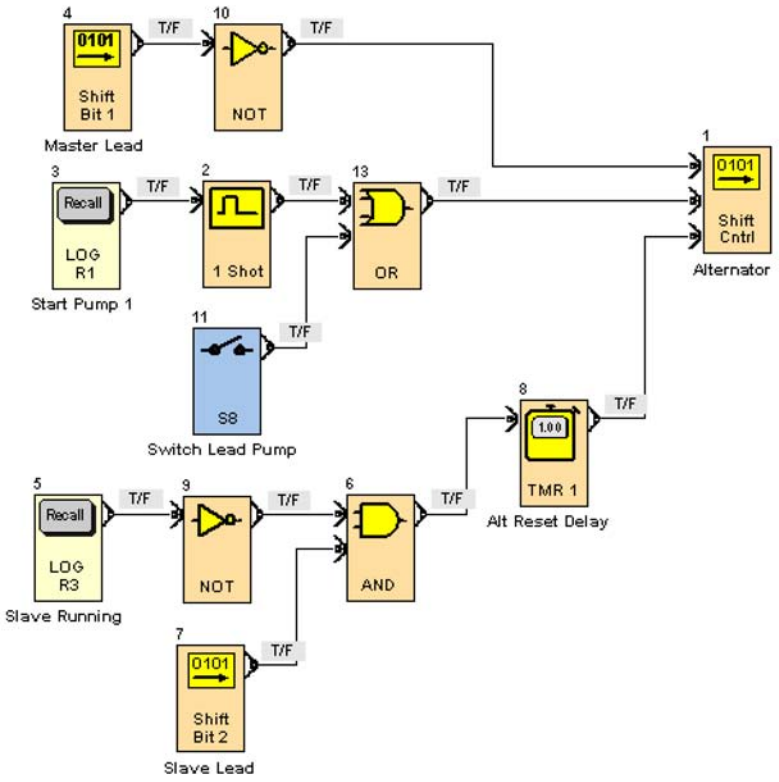
2. VFD Start Page



Circuit name	Master VFD start circuit
Function block name(s)	5(Shift Bit 1), 10(AND), 12(LOG R1), 11(LOG R2), 6(OR), 8(TMR 2), and 2(Fwd CMD).
Functional description	When Shift Bit 1 block is TRUE, the master drive is the lead. When in lead (Shift Bit 1 = TRUE) and LOG R1 block turns on when pump 1 is called for (page 1), the AND output is set to TRUE. Master drive can also start when pump 2 is called for (if lag) by the OR of LOG R2 (start pump 2 from page 1). TMR 2 block is a start delay to increase system stability. The Fwd CMD block sets the G7 drive (Master Drive) to the forward run mode.

Circuit name	Slave VFD start circuit
Function block name(s)	15(Shift Bit 2), 1(AND), 16(LOG R1), 3(LOG R2), 13(OR), 7(LOG R3), 9(TMR 3), and 4(M5).
Functional description	When Shift Bit 2 block is TRUE, the slave drive is the lead. When in lead (Shift Bit 2 = TRUE) and the LOGR1 block turns on when pump 1 is called for (page 1), the AND output is set to TRUE. The slave drive can also start when pump 2 is called for (if lag) by the OR of LOG R2 block (start pump 2 from page 1). The TMR 3 block is a start delay to increase system stability. The M5 closes when its input is TRUE to run the slave Drive. LOG W3 block transmits the slave running state to page 3.

3. Sequence Page



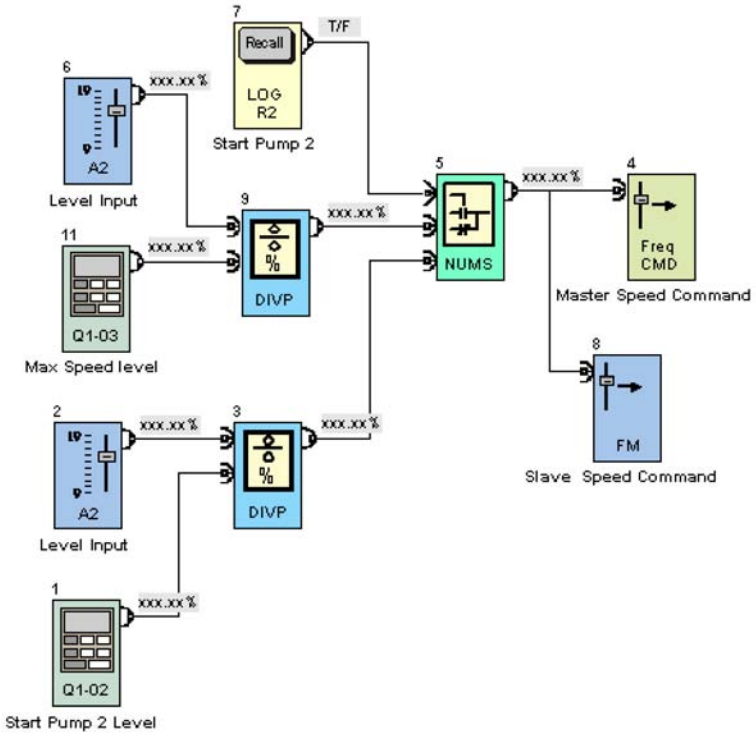
Circuit name	State machine to determine which drive is lead and which is lag
Function block name(s)	1(Shift Cntrl)
Functional description	When the bit in position 1 is TRUE, then the master is the lead. When the bit in position 2 is TRUE, then the slave drive is the lead. When the slave is not running and the slave is the lead, the shift register is reset to all bits = FALSE.

Circuit name	Data input to the shift register circuit
Function block name(s)	4(Shift Bit 1) and 10(NOT)
Functional description	This circuit puts a single 1 bit (TRUE bit) that moves from bit position 1 to bit position 2.

Circuit name	Shift input to the shift register circuit
Function block name(s)	3(LOG R1), 2(1-Shot), 11(S8), and 13(OR)
Functional description	When LOGR1 block is TRUE (Pump 1 start), the 1-Shot block outputs a TRUE for only 0.1 seconds. This logic signal causes the shift register to shift or move the bits from position 1 to position 2, etc. When the S8 block is TRUE, the shift register will also shift.

Circuit name	Shift register reset circuit
Function block name(s)	5(LOG R3), 9(NOT), 6(AND), 7(Shift Bit 2), and 8(TMR 1)
Functional description	When the slave is not running (NOT = TRUE) and the bit in shift register position 2 is set to TRUE, then all of the shift register bits are set to FALSE. The TMR 1 block is used to compensate for a logic race as the AND output can be TRUE momentarily when the slave starts and it is the lead.

4. Drive Reference Page



The value of the process variable or wet well level is used to create the drive speed reference. When pump 1 is running, its speed reference is scaled so that it is at 100% speed output at the level pump 2 is set to start. When pump 2 is started, both pumps are running in parallel so the speed reference must be reduced to a value that results in the same amount of flow level or pressure that pump 1 output at the same level. The maximum speed level setting sets the slope of the speed reference that is output with both pumps running. When the process variable or wet well level reaches the Max Speed Level, the speed reference for both pumps will be 100.00%.

Circuit name	Lead speed reference circuit
Function block name(s)	1(Q1-02), 2(A2), and 3(DIVP)
Functional description	The A2 block value is divided in percent by the setting of the Q1-02 block which is the start pump 2 setting. When the A2 block output value equals the setting of Q1-02, then the output of the DIVP block will be 100.00%.

Circuit name	Lead + lag speed reference circuit
Function block name(s)	11(Q1-03), 6(A2), and 9(DIVP)
Functional description	The A2 block value is divided in percent by the setting of the Q1-03 block which is the Max Speed Level setting. When the A2 block output value equals the setting of Q1-03, then the output of the DIVP block will be 100.00%.

Circuit name	NUMS
Function block name(s)	5(NUMS)
Functional description	Selects which calculated speed reference value is used by the state of LOG R2. When pump 2 is started (LOGR2 = TRUE), the NUMS block selects the output of block 9 DIVP.

Circuit name	Master speed command
Function block name(s)	4(Freq CMD) and 8(FM)
Functional description	Block 4(Freq CMD) sets the speed reference of the master drive (G7 drive) and block 8(FM) outputs the speed reference value to the slave drive.

Traverse Drive (P-Jump) – DriveWorksEZ™ Program Example

The program will generate a disturbed-sawtooth wave on top of the drive's set speed. The amplitude, ramp time, and P-Jump at transition from positive to negative slope and vice versa can be set by the user.

■Purpose of Program

The purpose of this program is to show the following principles:

- The ability to create a sawtooth waveform
- The ability to create a P-Jump at the change of slope of the sawtooth waveform
- The ability to add the sawtooth waveform to the drive frequency reference
- The ability to change the value of the RAMP time parameter (on-the-fly) by the DWEZ parameter write block

■Drive Physical Control Connections

- Speed reference via the standard A1 analog input
- Disturb or sawtooth waveform enabled by the selection of the S8 terminal input

■Changes From Standard G7 Drive Software Functions

Drive speed reference is from the DWEZ program only.

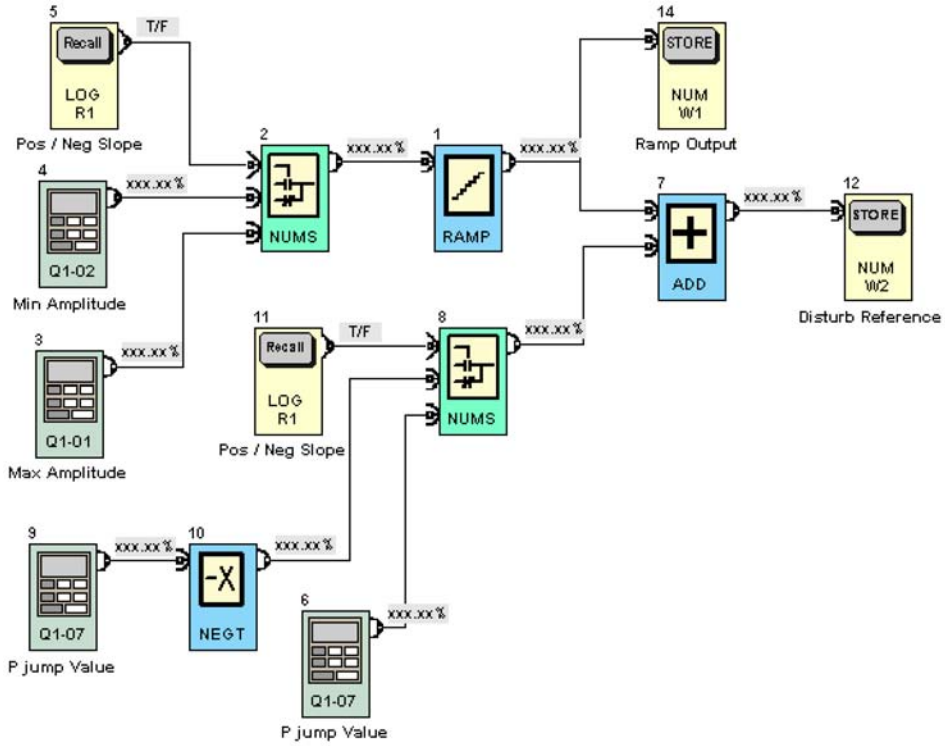
■Program Added Functions

P-Jump sawtooth generator adjustments:

- Minimum Amplitude (Q1-02)
- Maximum Amplitude (Q1-01)
- P-Jump Value (Q1-07)
- Positive Slope Time (Q1-03)
- Negative Slope Time (Q1-04)

■ Description of Operation

1. Ramp Generator Page

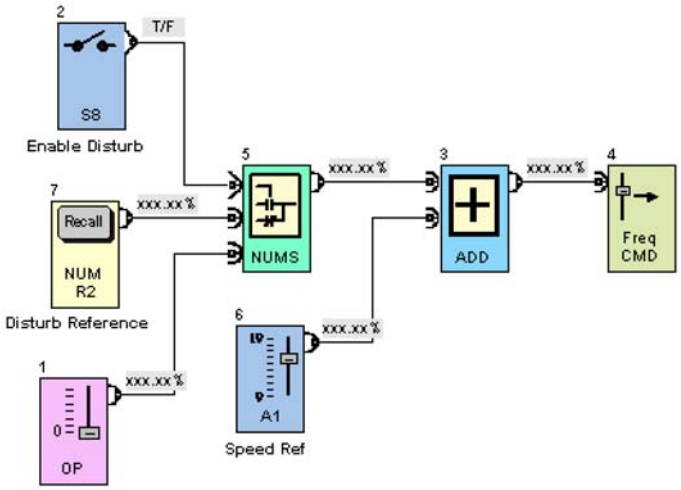


Circuit name	Basic sawtooth generator
Function block name(s)	1(RAMP), 2(NUMS), 3(Q1-01), 4(Q1-02), 5(LOG R1), and 14(NUM W1)
Functional description	The RAMP block input is switched between the Q1-01 and Q1-02 values by the NUMS block. This generates a sawtooth set of values at the output of the RAMP block. NUMS block switches whenever the RAMP block output reaches the Q1-01 or Q1-02 block values. The state of the LOG R1 block changes from FALSE to TRUE. This RAMP block set of values is sent to the Ramp Generator Logic Page by the NUM W1 block.

Circuit name	P-jump
Function block name(s)	8(NUMS), 11(LOG R1), 10(NEGT), 6(Q1-07), and 9(Q1-07)
Functional description	The value of the Q1-07 block is added or subtracted from the RAMP output which shifts the waveform, thus creating the P-Jump function. When the slope of the sawtooth is changed, the LOGR1 block changes state from FALSE to TRUE and vice versa which changes which input NUMS selects to output. This will be either the value of block Q1-07 or the negative value of block Q1-07.

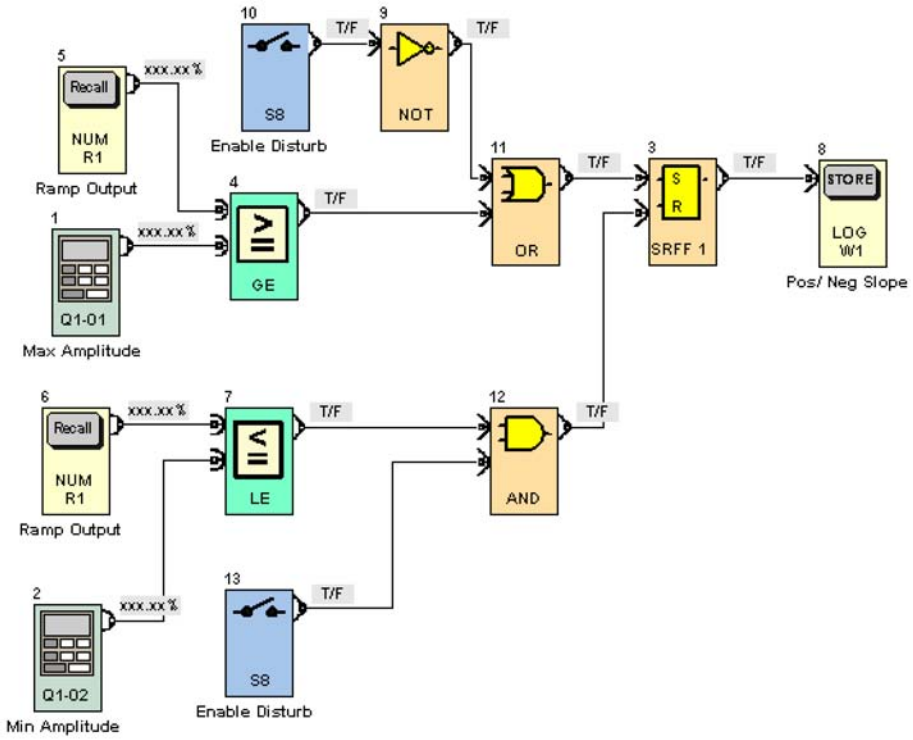
Circuit name	Disturb reference circuit
Function block name(s)	7(ADD) and 12(NUM W2)
Functional description	The output of the sawtooth generator is added to the P-Jump circuit. The combined output is sent to the NUM W2 block for use on other pages.

2. Drive Reference Page



Circuit name	Drive reference circuit
Function block name(s)	1(0P), 2(S8), 3(ADD), 4(Freq CMD), 5(NUMS), 6(A1), and 7(NUMR2)
Functional description	The drive speed reference is added to either a zero value or the output of the sawtooth generator depending on the state of the S8 block. When the S8 block is TRUE, then the value of the NUMR2 block is the output of NUMS and added to the A1 value. The combined value is input to the Freq CMD block which sets the drive frequency reference.

3. Ramp Generator Logic Page

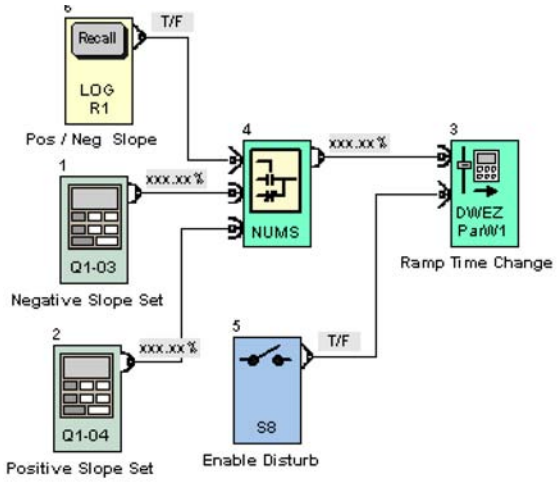


Circuit name	Sawtooth positive negative slope circuit
Function block name(s)	3(SRFF 1) and 8(LOGW1)
Functional description	When the flip-flop SRFF 1 block output is FALSE, the ramp slope is positive or increasing. When it is TRUE, the ramp slope is negative or decreasing. The output of the SRFF 1 block is sent to other pages via the LOGW1 block.

Circuit name	Switch to negative slope or SRFF 1 set circuit
Function block name(s)	1(Q1-01), 5(NUMR1), 4(GE), 10(S8), 9(NOT), and 11(OR)
Functional description	The sawtooth generator output from the NUMR1 block is compared to the Q1-01 Max Amplitude. If it is equal or greater than the Q1-01 block, the SRFF 1 set input is TRUE. Also, if S8, enable disturb is FALSE, then the SRFF 1 set input is TRUE to have the ramp output go to the minimum value in the disabled state.

Circuit name	Switch to positive slope or SRFF 1 reset circuit
Function block name(s)	6(NUMR1), 2(Q1-02), 7(LE), 12(AND), and 13(S8)
Functional description	The sawtooth generator output from the NUMR1 block is compared to the Q1-02 Min Amplitude. If it is less than or equal to Q1-02 and the S8, enable disturb input is TRUE, the SRFF 1 reset input is TRUE. This sets the SRFF 1 output to FALSE.

4. Slope Time Set Page



Circuit name	Slope time set circuit
Function block name(s)	1(Q1-03), 2(Q1-04), 3(DWEZ ParW1), 4(NUMS), 5(S8), and 6(LOGR1)
Functional description	When the slope is positive (LOGR1 = FALSE), the value of the Q1-04 block is set to the RAMP block time set parameter Q4-01 via the DWEZ ParW1 block. When the slope is negative (LOGR1 = TRUE), the value of the Q1-03 block is set to the RAMP block time set parameter Q4-01 via the DWEZ ParW1 block.

Digital Ratio Follower – DriveWorksEZ™ Program Example

This program closely resembles the F7 CASE program called "Enhanced PID with Digital Velocity Follower." The program is used with the PG-W2 dual encoder feedback option card. The G7 drive w/ DWEZ is the slave drive and can follow a master encoder. A +/- draw setting allows the user to adjust the speed difference between the master and slave as a percent of master speed. There is a PI controller that can be switched off and on to provide dancer position control trim to the master reference. The PI controller output gain can be varied by master reference or a fixed user parameter setting.

■Purpose of Program

The purpose of this program is to show the following principles:

- The ability of the drive to precisely follow the speed from another drive or machine function
- The ability to account for gear ratios to get per unit speeds of 100%
- The ability to position regulate a dancer with the drive following a master reference
- The ability to change the authority or gain of the dancer position regulator in relation to the master speed to keep the position regulator gain constant
- The ability to precisely adjust the speed ratio between the master and slave as a percent

■Drive Physical Control Connections

- Slave motor speed feedback encoder is connected to Channel 1 of PG-W2
- Master encoder is connected to Channel 2 of PG-W2
- Dancer position feedback is connected to analog input A2
- Dancer position control enable (closed = enabled) is connected to terminal S8
- Fixed/variable dancer gain select (closed = variable) is connected to terminal S7

■Changes From Standard G7 Drive Software Functions

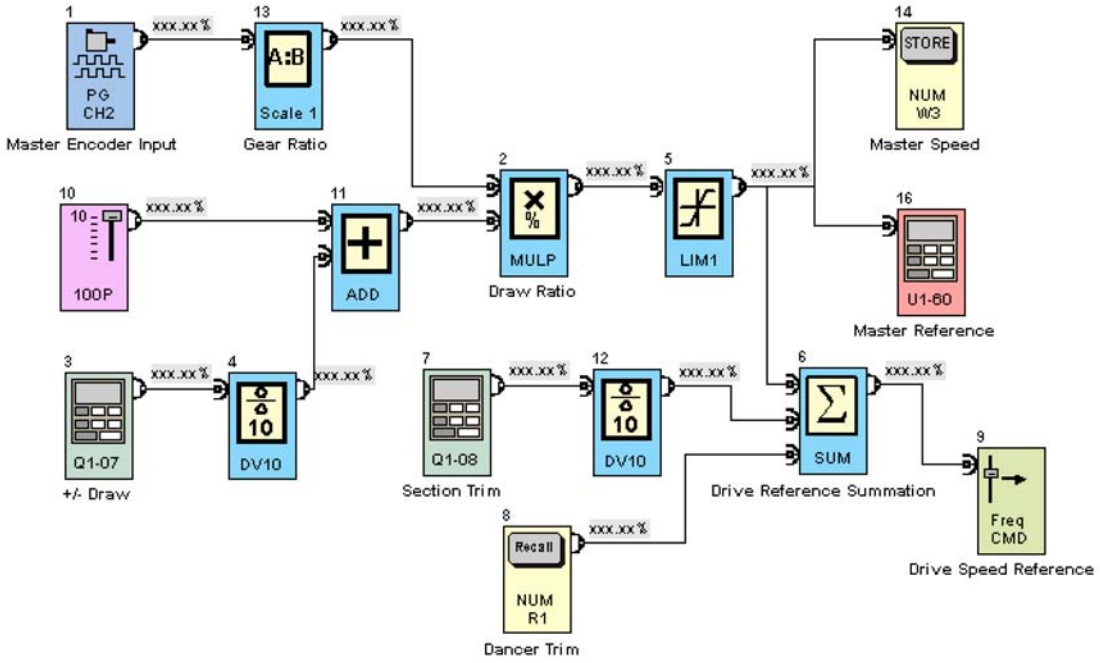
Drive speed reference is from the DWEZ program only.

■Program Added Functions

- Ability to follow external encoder signal with PG-W2
- Ability to adjust for gear ratio of master encoder to match to machine/process speed
- Ability to increase decrease the ratio of master speed to slave speed as a percent of master speed
- Ability to select master following mode as a positive speed only, negative speed only, positive and negative speed, or the absolute value of the master speed
- Ability to add a fixed amount of positive or negative speed reference to the slave drive
- Ability to turn the dancer position control on or off while running
- Ability to have the dancer position control output gain a fixed value or select it to be variable, based on master speed

■Description of Operation

1. Drive Reference Page



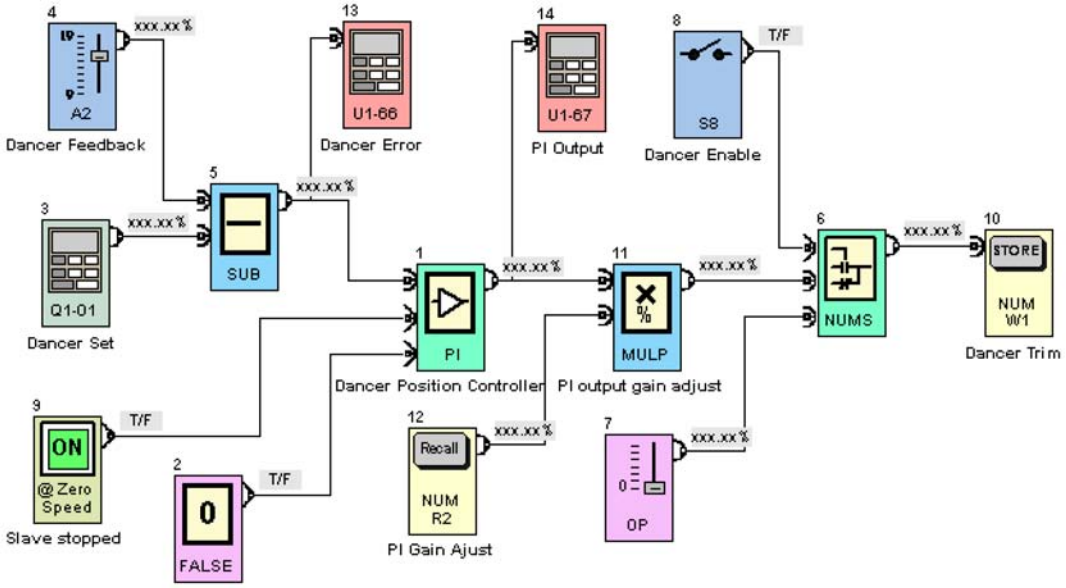
The master encoder speed comes into the program in block 1(PG CH2). The parameter setting of Q2-02 along with the settings of block 13(Scale 1) that are parameters Q4-03, 04, 05 are adjusted to derive a value of 100.00% at the output of block 13(Scale 1) when the master device is at 100% of its speed.

Circuit name	Percent draw function circuit
Function block name(s)	2(MULP), 3(Q1-07), 4(DIV10), 10(100P), and 11(ADD)
Functional description	The value of keypad parameter Q1-07 is the percent draw of master speed. Since the value of Q1-07 can only be set in 0.0%, the DIV10 block is used to allow setting draw in 0.01% increments. The output of DIV10 block is added to the constant 100P to create the percent that the Scale 1 output is multiplied by. Therefore, if the output Scale 1 is 50.00% and Q1-07 = 10.0 (1.00%), then the DIV10 block output will be 1.00%, the ADD block output will be 101.00%, and the MULP block output will be 50.5%. We can say then that with a draw of 1.00%, our slave will go 1.00% faster than the master. If we were to input -1.00%, then we would multiply the output of Scale 1 by 99.00% and the MULP block output would be 49.50%.

Circuit name	Sets the range of master speed input the slave will follow with parameters Q4-06 and Q4-07. Selects how the slave will follow the master by the setting of Q4-11.
Function block name(s)	5(L1M1)
Functional description	Block 5(LIM1) performs two functions. First, it sets the range of master speed input the slave will follow with parameters Q4-06 and Q4-07. Second, it selects how the slave will follow the master by the setting of Q4-11. • Q4-11 = 00, the slave will only follow a positive master reference. • Q4-11 = 01, the slave will only follow a negative master reference. • Q4-11 = 10, the slave will follow the master forward and reverse (positive and negative master reference). • Q4-11 = 11, the slave will follow the positive and negative master reference in one direction only.

Circuit name	Drive reference summation
Function block name(s)	6(SUM)
Functional description	Block 6(SUM) algebraically adds the outputs of block 5(LIM1), block 12(DV10), and block 8(NUMR1), which is the value from the dancer control page. The output of the SUM block becomes the slave (G7 drive) speed reference.

2. Dancer Control Page



Circuit name	Dancer position error circuit
Function block name(s)	3(Q1-01), 4(A2), 5(SUB), and 13(U1-66)
Functional description	Block 3(Q1-01) sets the dancer position which is subtracted from the dancer analog feedback signal connected to the drive terminal A2. Monitor U1-66 shows the dancer error on the slave drive keypad.

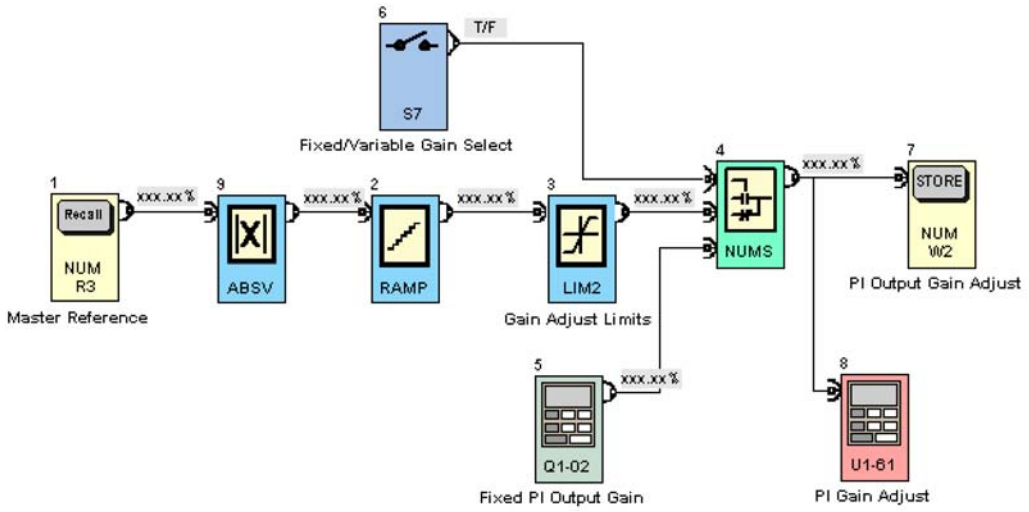
Circuit name	Dancer position control
Function block name(s)	1(PI) and 14(U1-67)
Functional description	The PI control input is the output of Block 5(SUB) which is dancer error. The logical inputs to the PI control are set to defaults (Q6-02) which means that both inputs are PI control disabled when true. One input is connected to Block 9 (@ Zero Speed) which is true when the slave is stopped. This is done to disable and reset the PI control so any dancer movement will not cause any PI integrator wind-up as well as move any material when the line is stopped. Block 14(U1-67) allows the PI or dancer controller output to be displayed on the slave drive keypad.

Circuit name	PI output gain adjust circuit
Function block name(s)	11(MULP) and 12(NUMR2)
Functional description	This circuit is used to adjust the amount of speed change that is allowed to the dancer controller for system stability. The output of the PI controller is multiplied in percent by the value of block 12(NUMR2). This value comes from block NUMW2 on page 3, PI Output Gain Control.

Circuit name	Dancer enable circuit
Function block name(s)	8(S8), 6(NUMS), and 7(0P)
Functional description	This circuit turns on or off the output of the dancer control. Block 6(NUMS) selects either 0.00% or the output of block 11(MULP) based on the state of S8. When block S8 is closed (TRUE), the NO numerical input to block 6(NUMS) will output to block 10(NUMW1)

Circuit name	Input block to transfer the dancer trim value to page 1 block 8(NUMR1).
Function block name(s)	10(NUMW1) and 8(NUMR1)
Functional description	Block 10(NUMW1) is the input block to transfer the dancer trim value to page 1 block 8(NUMR1).

3. PI Output Gain Page

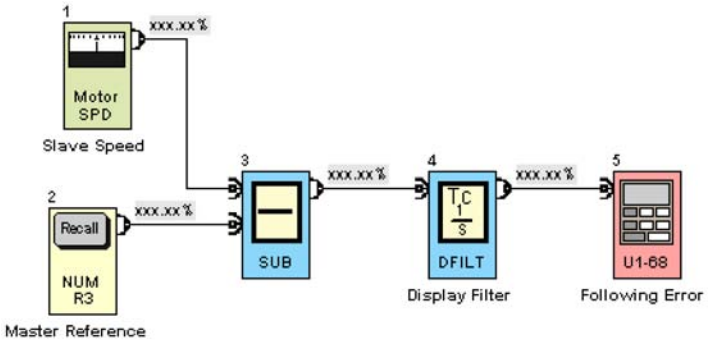


Circuit name	Master reference PI gain value generate
Function block name(s)	1(NUMR3), 9(ABSV), 2(RAMP), and 3(LIM2)
Functional description	The scaled master reference value from page 1 is input via block 1(NUMR3). Since the value of the master reference can be bipolar, block 9(ABSV) provides an absolute or positive value of the master reference. Block 2(RAMP) is used to provide a ramp that prevents the PI output gain from changing rapidly for system stability reasons. Block 3(LIM2) is used to set the minimum and maximum PI output gain values.

Circuit name	Allow the selection of a PI output variable gain or a fixed gain value.
Function block name(s)	4(NUMS), 6(S7), and 5(Q1-02)
Functional description	When block S7 is closed (TRUE) the PI output gain is a function of master line speed. The reason we would need a variable gain on the output of the PI controller is to keep the ratio of PI control output to line speed (and thus slave speed) constant. This improves system stability, especially over a wider operating speed range. When block S7 is open (FALSE), the PI control output gain is set by the value of Q1-02.

Circuit name	PI Output Gain Adjust
Function block name(s)	7(NUMW2)
Functional description	Transfer the PI output gain value to page 2. Monitor U1-61 displays the value of the PI output gain adjust on the slave drive keypad.

4. Following Error Page



Circuit name	Following error
Function block name(s)	2(NUMR3), 1(MotorSPD), 4(DFILT)
Functional description	This page is used only to derive the value to be displayed on U1-68 which is the following error between the master drive and the slave drive. The slave drive speed, block 1 (MotorSPD) is subtracted from the scaled master reference on page 1 via block 2(NUMR3). Block 4(DFILT) is a filter block which is used to smooth out the display of U1-68. U1-68 is used because the value is bipolar.

Simple Winder Control – DriveWorksEZ™ Program Example

This program shows how a simple speed regulated winder with dancer position trim can be accomplished. The winder will follow a master encoder line speed reference from channel 2 of a PG-W2 feedback option card. A diameter calculator block is used to derive the winding roll diameter. A PI controller is used to trim winder speed for proper dancer position.

■Purpose of Program

The purpose of this program is to show the following principles:

- The ability of the drive to precisely follow the speed from another drive or machine function
- The ability to compensate for the diameter of the material being wound
- The ability to position regulate a dancer with the drive following a master reference
- The ability to change the authority or gain of the dancer position regulator in relation to the master speed to keep the position regulator gain constant

■ Drive Physical Control Connections

- Slave motor speed feedback encoder is connected to Channel 1 of PG-W2.
- Master encoder is connected to Channel 2 of PG-W2.
- Dancer position feedback is connected to analog input A1.
- Diameter reset is connected to S8. When S8 = TRUE, then diameter calculator resets to reset value.

■ Changes From Standard G7 Drive Software Functions

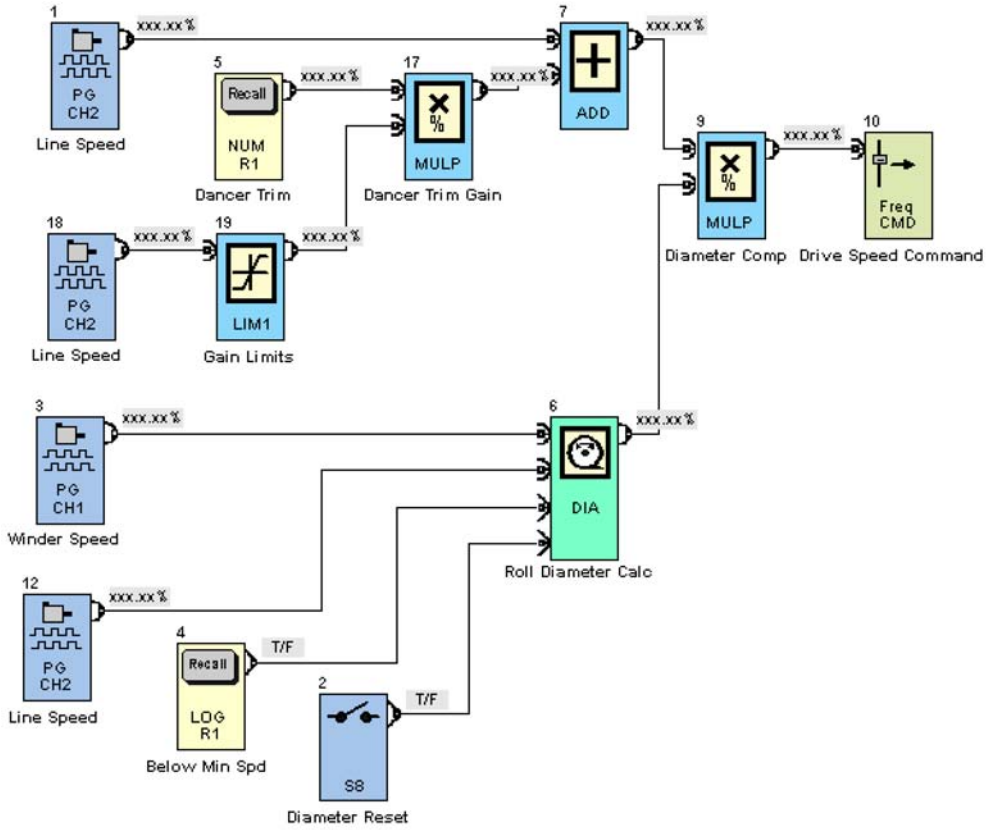
Drive speed reference is from the DWEZ program only.

■ Program Added Functions

- Ability to follow external encoder signal with PG-W2
- Ability to adjust for diameter build up of material
- Ability to control the position of a dancer
- Ability to turn the dancer position control on or off while running
- Ability to have the dancer position control output gain variable, based on master speed

■Description of Operation

1. Winder Reference Page



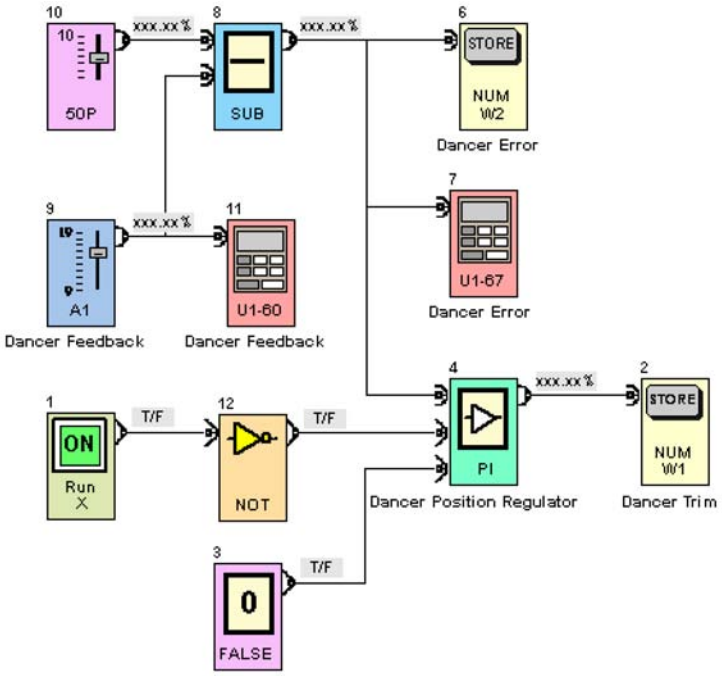
Circuit name	Winder speed reference circuit
Function block name(s)	1(PGCH2), 7(ADD), 5(NUMR1), 18(PGCH2),19(LIM1), 9(MULP),17(MULP),and 10(Freq CMD)
Functional description	The master or line speed is input from the PGW2 dual channel encoder feedback option card channel 2 input. The output of the dancer position controller circuit (page 2) comes in via the NUMR1 block.

Circuit name	Gain limits
Function block name(s)	19(LIM1)
Functional description	The amount of dancer position controller output is varied by master line speed to make the dancer position controller influence constant. This is done by having the dancer position controller output (Dancer Trim – NUMR1) multiplied in percent by a value proportional to master line speed. The LIM1 block sets the maximum and minimum gain limits. The master line speed and gain adjusted dancer position that control output are added together and multiplied in percent by the output of the diameter calculator.

Circuit name	Drive speed command
Function block name(s)	10(Freq CMD)
Functional description	The diameter calculator is set for rewinding so at winder core its output is 100.00%. As the winding bundle builds up, the diameter calculator output reduces, reducing the speed reference to the winder drive. The diameter compensated speed reference value is output to the Freq CMD block which sets the drive speed reference.

Circuit name	Diameter calculator circuit
Function block name(s)	2(S8), 4,(LOGR1), 12(PGCH2), 3(PGCH1), and 6(DIA)
Functional description	Winder speed and line speed are input into the diameter block. The LOGR1 block is TRUE when the winder is below the minimum speed setting, holding the diameter calculator block at its current value. Switch S8 is connected to the diameter reset input to reset the diameter calculator to its reset value. Since this is a rewinder, it is set to 100.00%.

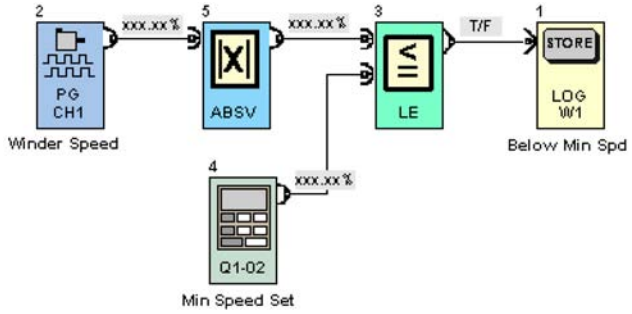
2. Dancer Control Page



Circuit name	Dancer error circuit
Function block name(s)	10(50P), 8(SUB), 9(A1), 11(U1-60), 7(U1-67), and 6(NUMW2)
Functional description	The dancer feedback signal is subtracted from 50.00% to get a dancer position error value. The U1-60 block displays the dancer feedback value on the drive keypad. The U1-67 block displays the value of dancer position error on the drive keypad. The NUMW2 block is used to send the dancer position error to the dancer limits page.

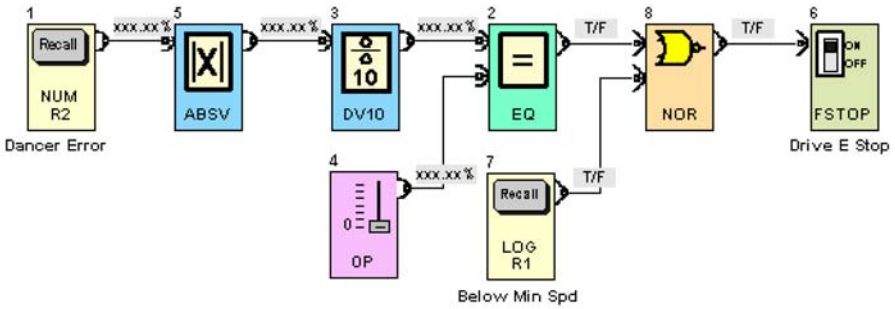
Circuit name	Dancer controller circuit
Function block name(s)	1(Run X), 12(NOT), 3(FALSE), 4(PI), and 2(NUMW1)
Functional description	The PI controller acts on the dancer position error signal and outputs to the NUMW1 block a position control value. Run X and NOT blocks create a not running = TRUE value, which is input to the PI controllers disable input.

3. Minimum Speed Detect Page



Circuit name	Minimum speed detection
Function block name(s)	2(PGCH1), 5(ABSV), 4(Q1-02), 3(LE), and 1(LOGW1)
Functional description	The speed of the winder drive is converted to an absolute value and compared to the setting of block Q1-02. When the winder speed is less than or equal to block Q1-02, then the LE block is TRUE. This value is sent to other parts of the program via the LOGW1 block.

4. Dancer Limits Page



Circuit name	Dancer limits
Function block name(s)	6(FSTOP)
Functional description	The dancer limits circuit works on the value of dancer error. When the error is greater than +/- 37.50%, then a dancer limit is declared if the winder is above minimum speed. When a dancer limit is reached, the drive is e-stopped by the FSTOP block.

Circuit name	Dancer error valve
Function block name(s)	1(NUMR2), 5(ABSV), 2(EQ)
Functional description	The dancer error value comes from page 2 via NUMR2. The value is converted to an absolute value by the ABSV block. The error is divided by 10 and then compared to see if it is equal to 0. Since the deadband of the EQ block is set to 3.75%, its output will not switch to TRUE until the input is greater than 3.75%. Since the dancer error is divided by 10, this translates to a dancer error value of 37.50%.

This page intentionally left blank.

Index

A

B

C

D

YASKAWA ELECTRIC AMERICA, INC.
Drives Division
16555 W. Ryerson Rd., New Berlin, WI 53151, U.S.A.
Phone: (800) YASKAWA (800-927-5292) Fax: (262) 782-3418
Internet: http://www.drives.com

YASKAWA ELECTRIC AMERICA, INC.
Chicago-Corporate Headquarters
2121 Norman Drive South, Waukegan, IL 60085, U.S.A.
Phone: (800) YASKAWA (800-927-5292) Fax: (847) 887-7310
Internet: http://www.yaskawa.com

YASKAWA ELECTRIC CORPORATION
New Pier Takeshiba South Tower, 1-16-1, Kaigan, Minatoku, Tokyo, 105-0022, Japan
Phone: 81-3-5402-4511 Fax: 81-3-5402-4580
Internet: http://www.yaskawa.co.jp

YASKAWA ELECTRIC EUROPE GmbH
Am Kronberger Hang 2, 65824 Schwalbach, Germany
Phone: 49-6196-569-300 Fax: 49-6196-888-301

YEA Document Number: TM.DWEZ.01 **12/1/2005**