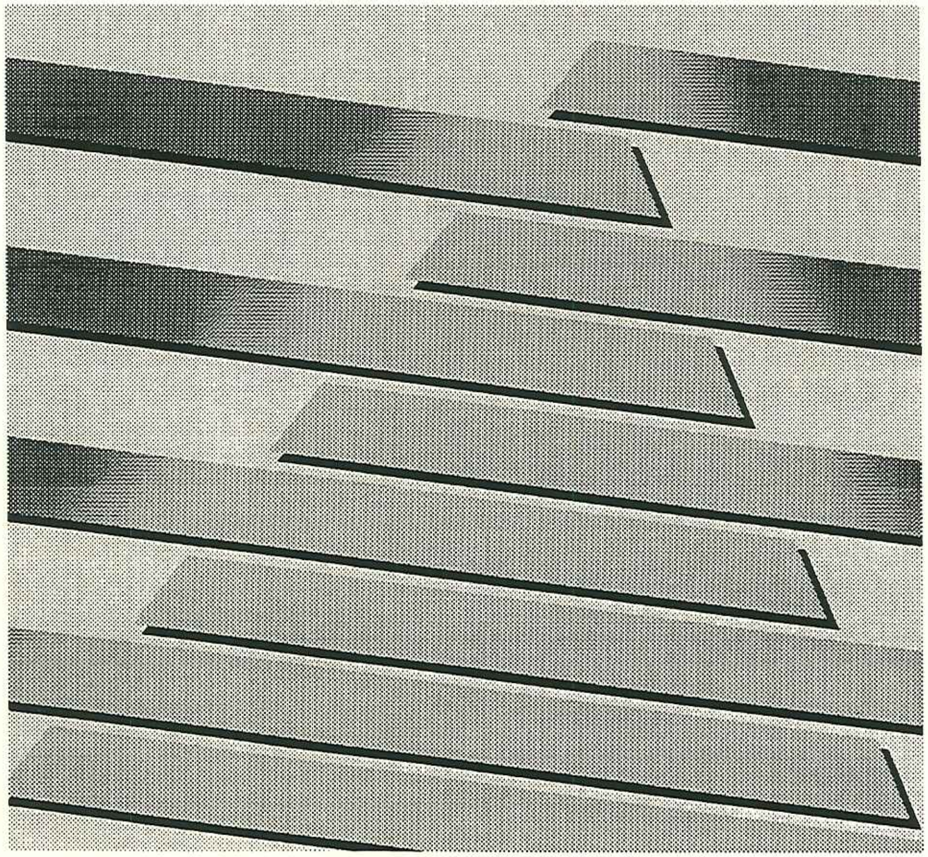
ALLEN-BRADLEY

GML™
Graphical Motion Control Language

GML Programmer's Workshop
User Manual

Software Version 3.4
for the Windows™ Operating System



DISCLAIMER The information contained in this document is subject to change without notice and should not be construed as a commitment by Allen-Bradley Co. Allen-Bradley assumes no responsibility for any errors that may appear.

Printed in USA

Contents

Introduction 1
A Word to the Wise... 3
PC Configuration 4

Getting Started 5
Back-Up Your Disk 5
Install GML 5
Run GML 6

Learning GML 7
The Diagram Window 7
Getting Help 8
Setting GML Preferences 10
Working with Blocks 13
Adding Blocks 13
Connecting Blocks or Modules 14
Moving Blocks 15
Disconnecting Blocks 15
Moving Connection Lines 15
Encapsulating Blocks into Modules 16
Cutting, Copying, and Pasting Blocks 17
Using Change, Insert Between, and Swap 18
Spacing and Aligning Blocks 20
Snap To Grid 21
Entering Block Parameters 21
Expressions 22
Variables, Constants, and I/O 22
Testing Diagrams 23
Saving Diagrams 23
Printing Diagrams 24
Navigating Complex Diagrams 25
Using the Windows Menu 25
Searching a Diagram Using Find and Find Again 26
Converting a Diagram to a Script 27
Downloading a Diagram 28
Quitting GML 28

GML Tutorial 29

Introduction

GML—the exclusive Graphical Motion Control Language from Allen-Bradley—is a revolutionary tool which reduces the learning curve of motion control programming and makes problem-solving easier for non-programmers. GML Programmer's Workshop provides an integrated software development and debugging environment for using GML to solve your motion control problem. Development of the GML diagram is carried out using the GML Programmer's Workshop software within the Windows™ operating system environment on your PC.

GML uses a natural, flowchart-oriented approach to motion control programming. A diagram of the desired application solution is produced by placing function blocks representing the desired actions on the screen and connecting them in the proper order to achieve the desired sequence of operations. Motion and process details are entered later using a fill-in-the-form approach. After the diagram is completed, it is translated into a program or *script* in the native language of the motion controller. This application program—which gives the motion controller its personality for each different application—is then downloaded to the motion controller. While the PC is required for application program development, it is not needed for the final turn-key application.

GML Programmer's Workshop is available as a separate software package for each family of Allen-Bradley motion controllers. The available GML features depend on the motion controller family. A brief description of each of the available GML Programmer's Workshop packages follows.

GML for IMC S Class

GML for IMC S Class (GML-D-S) is the GML Programmer's Workshop package for all IMC S Class motion controllers. It includes a fully functional Online Manager for real-time communication with the motion controller. Upload and download capabilities for GML diagrams as well as setup parameter, user variable, and cam table values are provided. GML diagrams can be traced and debugged graphically (blocks are highlighted as they are executed in the motion controller) allowing you to visually follow execution of the program. In addition, breakpoints may be easily inserted and monitored while the program executes. The program may also be paused and resumed or executed step-by-step to verify proper operation. Individual GML icons may be executed directly—sending immediate commands to the motion controller—for testing and troubleshooting application problems. Variable values may also be individually read at any time while the program is executing for monitoring purposes.

GML for the IMC-201

GML for the IMC-201 (GML-D-A) is the GML Programmer's Workshop package for the IMC-201 and the MAX/CONTROL. It includes an Online Manager for real-time communication with the motion controller. Download capability for GML diagrams as well as direct access to the motion controller's setup menus are provided. The program may be paused and resumed or executed step-by-step to verify proper operation. Individual GML icons may be executed directly—sending immediate commands to the motion controller—for testing and troubleshooting application problems. Variable values may also be individually read at any time while the program is executing for monitoring purposes.

GML for the IMC Classics

GML for the IMC Classics (GML-D-M) is the GML Programmer's Workshop package for the IMC 110, 120, 120AR, 121, 123, and 123CR. It requires the appropriate ODS (Offline Development System) software for your motion controller to compile, download, and debug the MML program generated from a GML diagram. No Online Manager is provided.

A Word to the Wise...

GML is designed to reduce the learning curve of motion control programming and make problem solving easier for non-programmers. Make no mistake, however, you still need a basic understanding of the problem you are trying to solve and how to use GML to solve it. Advanced planning is necessary—GML cannot divine the functions and operations you need to solve you particular application problem. Examples of how some problems can be solved are provided, but they do not offer a specific solution for every application.

PC Configuration

To use GML, you need an MS-DOS compatible PC with the following configuration:

- 386 class CPU (486 class recommended)
- 2 Mbyte RAM (4+ Mbyte recommended)
- Microsoft® Windows™ 3.0 (3.1 recommended)
- MS-DOS 5.0 (6.0 recommended)
- EGA Screen (VGA or Super VGA recommended)
- Windows compatible mouse and driver
- 3.5 inch High Density (1.44 Mbyte) floppy disk drive
- Two serial ports (COM1 and COM2)

You should be familiar with the operation of Microsoft Windows and your motion controller before using GML Programmer's Workshop. See your Microsoft Windows User's Guide, MS-DOS Operating System Guide, and the documentation supplied with the specific motion controller you are using.

Getting Started

To use GML, you must first *install* it on your hard drive, following the instructions in this section. After installing GML (and optionally connecting your motion controller to the PC), use the remainder of this manual to learn about GML. For a quick introduction to building a GML diagram, go to the Learning GML section. If you would like to create an example application step by step, do the GML Tutorial.

Back-Up Your Disk

We recommend that you make a back-up copy of the GML distribution disk. Use the MS-DOS DISKCOPY command (see the User Reference section of your MS-DOS Manual) to do this. After you have made a copy of the GML for Windows disk, put the original in a safe, dry place and use it only if the copy is damaged or destroyed.

> **WARNING:** *It is a violation of Federal Copyright Law to copy your GML Programmer's Workshop disk except for back-up purposes to guard against accidental loss or damage. No part of the software contained on your GML Programmer's Workshop disk may be reproduced, transmitted, or transferred without the prior written permission of Allen-Bradley Company.*

Install GML

GML is supplied on a single 3.5 inch high density (1.4MB) floppy disk as a self-installing executable file. The GML installation program must be run from within Windows. To run the Install program,

1. Insert the copy of the GML for Windows distribution diskette in the disk drive (for example drive A:).
2. Run the SETUP.EXE program on this disk using the Windows File Manager application, or choose Run from the File menu in the Windows Program Manager application and enter A:SETUP.

3. Follow the instructions on the installation program screens to complete the installation process.

The installation process places all GML files in the specified GML directory path and creates a Windows Program item called GML. A Read Me file is included which includes the latest up-to-date information about GML. Use the GML Script Editor—or any other DOS or Windows text editor—to read this document.

Run GML

If you have not already done so, install GML on the hard drive in your computer as explained in Installing GML earlier in this section. Start GML by double-clicking on the GML icon, as shown below.

GML

Click **OK** to dismiss the start-up screen, and momentarily the GML diagram window will appear.

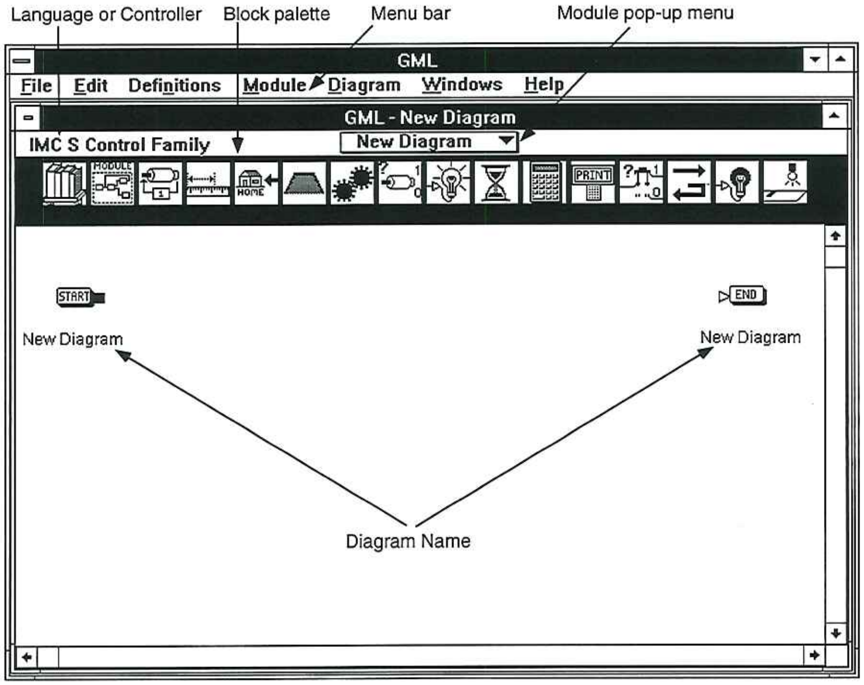
The *diagram window* is used to create or edit a GML diagram which shows a motion control program as a sequential flow of events. Special icons called *blocks* represent actions and decisions. Connections link blocks in a specified order to show relationships and program flow. Double-clicking on a block opens it and displays its parameter definition dialog. The parameter definition dialogs allow you to view and set parameter values for the block.

Learning GML

This section explains how to use the features and tools in the GML Programmer's Workshop to create, edit, translate, and download GML diagrams. This information is presented in roughly the order you are likely to need it, but feel free to jump around to find what you need.

The Diagram Window

When you first run GML, your diagram window looks like this.

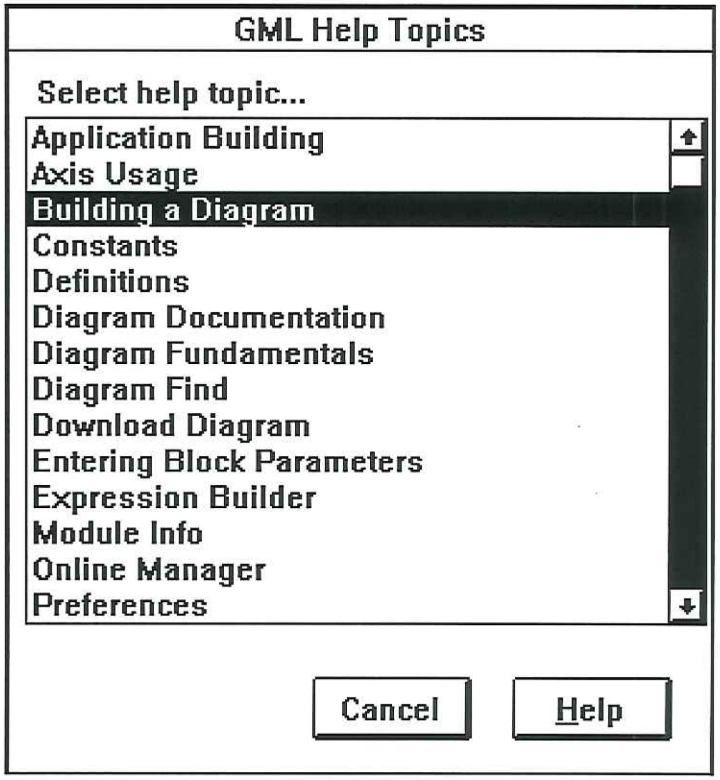


The diagram window includes a *menu bar* for defining and drawing diagrams, the standard Windows close box, grow boxes, and scroll bars.

At the top of the diagram window is the *Module pop-up menu* and, to the left, the currently selected motion control family. Just below is the *block palette* containing frequently used blocks. If you are using a PC with a color monitor, notice that the blocks are color-coded according to related functions (motion blocks are green, etc.). Below the block palette is the empty diagram itself containing only the START and END blocks and the diagram name.

Getting Help

Choose GML Help from the Help menu to display the Help Topics.

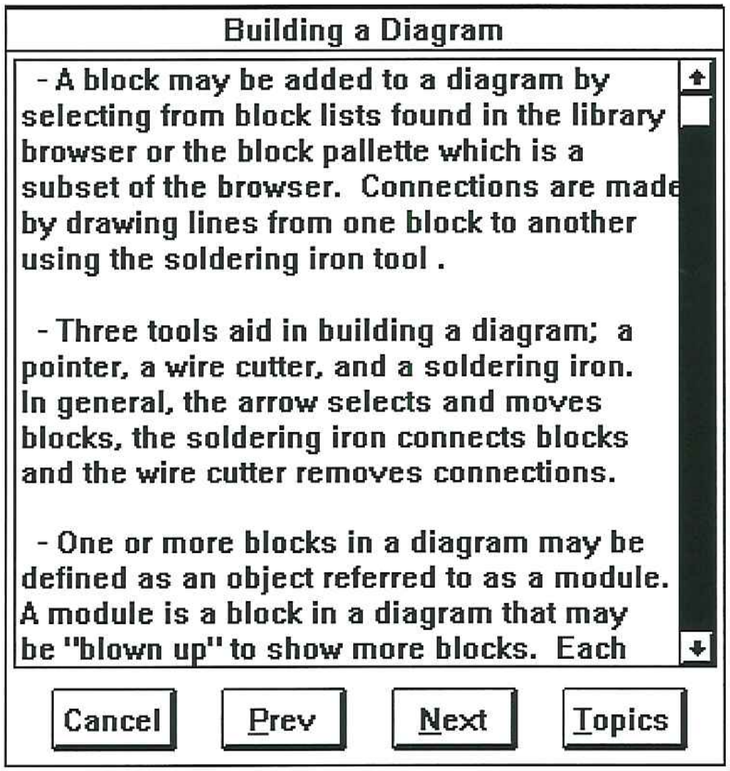


GML Help Topics

Select help topic...

- Application Building
- Axis Usage
- Building a Diagram
- Constants
- Definitions
- Diagram Documentation
- Diagram Fundamentals
- Diagram Find
- Download Diagram
- Entering Block Parameters
- Expression Builder
- Module Info
- Online Manager
- Preferences

Select the desired topic and click **Help**. Information on the selected topic is displayed, as shown below.



Building a Diagram

- A block may be added to a diagram by selecting from block lists found in the library browser or the block pallette which is a subset of the browser. Connections are made by drawing lines from one block to another using the soldering iron tool .

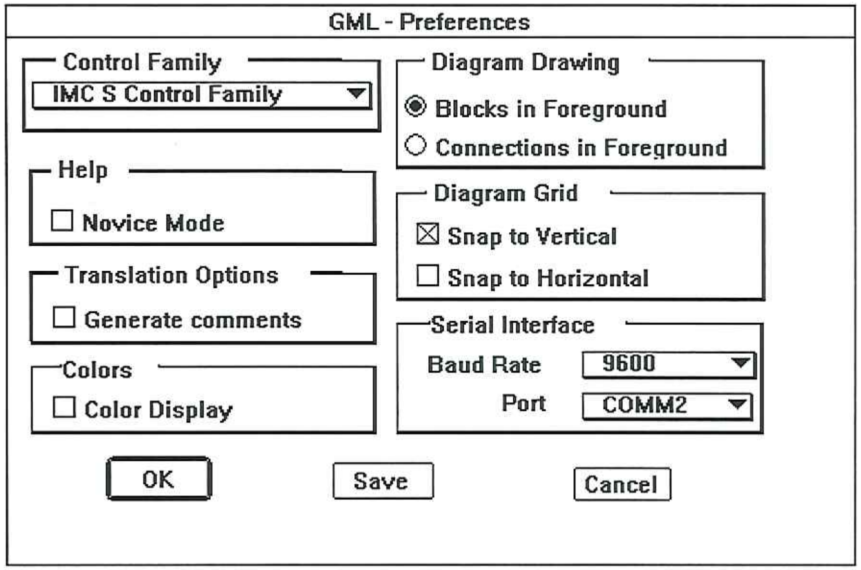
- Three tools aid in building a diagram; a pointer, a wire cutter, and a soldering iron. In general, the arrow selects and moves blocks, the soldering iron connects blocks and the wire cutter removes connections.

- One or more blocks in a diagram may be defined as an object referred to as a module. A module is a block in a diagram that may be "blown up" to show more blocks. Each

The **Next** and **Prev**ious buttons move forward or backward through the information. To get help on another topic, click **Topics**. To leave the help system, click **Cancel**.

Setting GML Preferences

GML allows you to turn certain features on and off, and to define how others are to function. These *preferences* may be set for a specific diagram, a specific session, or saved for future sessions. Choose Preferences from the File menu. The Preferences dialog box is displayed, as shown below.



GML - Preferences

Control Family
IMC S Control Family

Diagram Drawing
● Blocks in Foreground
○ Connections in Foreground

Help
□ Novice Mode

Diagram Grid
☒ Snap to Vertical
□ Snap to Horizontal

Translation Options
□ Generate comments

Serial Interface
Baud Rate 9600
Port COMM2

Colors
□ Color Display

Make sure the appropriate Control Family is selected from the Control Family pop-up menu, set any other preferences to your liking as explained below and then click **OK** or **Save** to close the Preferences dialog. **OK** stores your preferences as part of the current GML diagram only, while **Save** stores them in a separate GML Preferences file as well.

The Preferences dialog also remembers the location and size of any open Diagram, Script Editor, or Online Manager window and uses these locations and sizes as the defaults for new windows. For example, if you want new diagrams to always be located on the top half of the screen and scripts to be located on the bottom half, locate the Diagram Window on the top half of the screen and a new Script

Window (choose New Script from the File menu) on the bottom half before choosing Preferences from the File menu.

The available preferences are explained below.

Control Family

Select the motion controller family of which the motion controller you are using is a member. This setting determines the language used for translating diagrams to programs, as shown below.

Allen-Bradley GMC Motion Controller Families		
To Use This Motion Controller...	*Use This GML...*	*and Select this Control Family*
IMC-S/202...	GML-D-S	IMC S Control
IMC-S/204...	GML-D-S	IMC S Control
IMC-S/212...	GML-D-S	IMC S Control
IMC-S/214...	GML-D-S	IMC S Control
IMC-201	GML-D-A	MAX/IMC-201
MAX/CONTROL	GML-D-A	MAX/IMC-201
MAX4/CONTROL	GML-D-A	MAX/IMC-201
SA2	GML-D-A	MAX/IMC-201
IMC-110	GML-D-M	IMC Classic
IMC-120	GML-D-M	IMC Classic
IMC-120AR	GML-D-M	IMC Classic
IMC-121	GML-D-M	IMC Classic
IMC-123	GML-D-M	IMC Classic
IMC-123CR	GML-D-M	IMC Classic

You *must* choose the appropriate motion controller family *before* creating a diagram. Some of the programming tools change according to the capabilities of the motion controller family. The specific motion controller within the family is chosen in the Hardware Options definition.

Diagram Grid The GML diagram is divided into an invisible grid. If either of the Snap to Grid options is selected, blocks placed in the diagram snap to the nearest corresponding grid location to help keep them aligned.

Colors On color monitors, GML displays blocks in color. Turn color off for black and white monitors.

Translation Options When a diagram is translated to a program, comments describing each action are automatically inserted if Generate Comments is selected.

Help When Novice Mode is selected, the help text associated with each block is displayed automatically when the block is selected.

Diagram Drawing This setting determines if blocks are drawn on top of connecting lines or vice versa. The normal setting is Blocks in Foreground which hides overlapping connecting lines behind blocks.

Serial Interface Select the desired serial channel for motion controller communications. The default selection is COM2.

Working with Blocks

Blocks representing program functions are placed on the diagram between the

START and ▷END

blocks. These blocks are always present in the diagram and cannot be deleted. Blocks are generally added left to right on the page. They are then interconnected to show the flow of program functions. Blocks can be moved around, and their connections can be changed to change program flow. Complex diagrams can be simplified by grouping (encapsulating) several blocks together.

For easy comprehension and editing, try to have block connections flow from left to right or up to down. Limit the number of blocks in the work space to 10 or less, and avoid forcing the use of scroll bars by making the diagram larger than your computer screen.

Adding Blocks

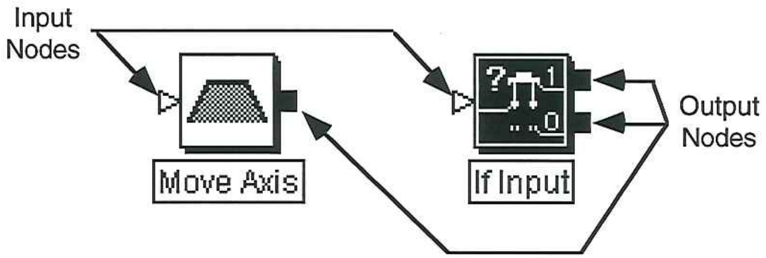
Blocks are added to the diagram from the *block palette* or from the *library* which contains all of the available blocks for the selected motion controller. To add a block from the block palette, click on the desired block. It is highlighted and its title appears immediately below it. Place the block by clicking at the desired location in the diagram.

If the desired block is not on the block palette, double-click any of the blocks there, or click on the library block at the far left of the block palette. Either method takes you to the library, but double-clicking a block palette block brings up the part of the library related to that block. The *library* shows all available blocks, including those on the block palette. Scroll to locate the desired block, click on it and then click the **Select** button to return to the diagram. Alternatively, you can double-click on a library block to both select it and return to the diagram. Place the block by clicking at the desired location.

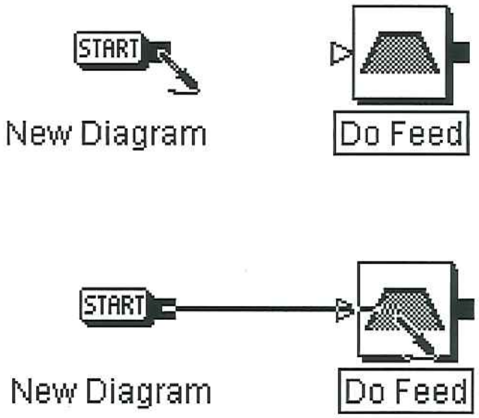
When the block is placed, the description below it is highlighted. Type in a new description if desired.

Connecting Blocks or Modules

Most blocks and modules—including the START and END blocks—have input and output nodes. The diagram must start at the START block with connections drawn from an output node of one block to the input node of another. The diagram usually ends at the END block.



Move the cursor to the desired output node. The cursor changes from a pointer to a soldering iron. Press the mouse button, drag to the desired input node and release. A line representing the connection is drawn, as shown below.

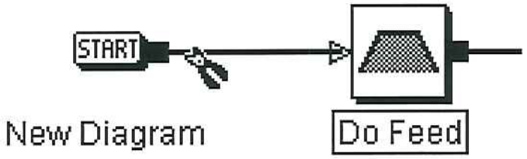


Moving Blocks

Blocks—including the START and END blocks—can be dragged anywhere on the diagram. Wherever they are moved, their connections remain as before the move. The Diagram Drawing preference (choose Preferences from the File menu) can be set to show connecting lines on top of blocks to help keep track of connections or underneath to show blocks. See Setting GML Preferences earlier in this manual for more information.

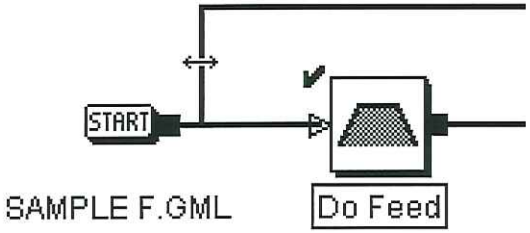
Disconnecting Blocks

To cut a connection, move the pointer to a connecting line. The cursor changes to a wire cutter. Click to cut the connection, as shown.



Moving Connection Lines

In some cases, when you have a connection that loops back to an earlier block in the diagram, you may want to move a vertical connection to avoid other blocks. Place the pointer on the vertical connection. A double arrow replaces the pointer. Press the mouse button, drag the connection to the desired position and release.

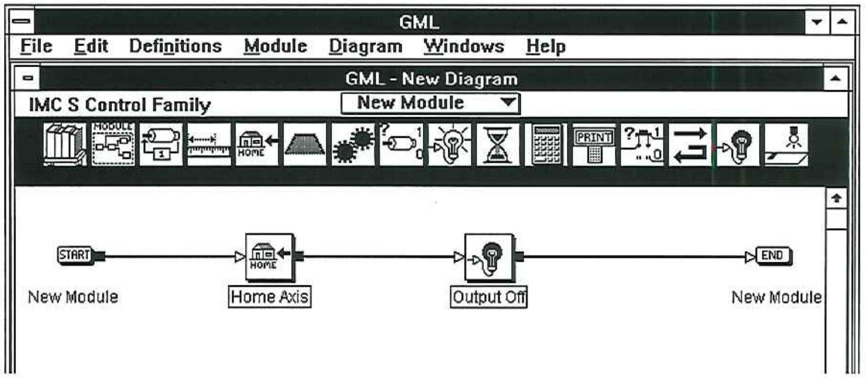


Encapsulating Blocks into Modules

Several blocks that together comprise a larger function or related functions can be grouped together in one block called a *module* to hide detail and save space. To select the desired blocks, position the arrow tool above and to the left of them. Press and hold the mouse button down. Drag the arrow to a position below and to the right of the blocks and release the button. Choose Encapsulate from the Module menu to encapsulate the selected blocks in a module.

When the module is created, the description below it is highlighted. Type in a new description of the combined function represented by the module. You can also document it more extensively by selecting it, choosing Documentation from the Module menu, and entering the desired information in the Documentation dialog.

The individual blocks encapsulated in a module are always accessible. To see them, select the module and choose Show Details from the Module menu or double-click the module. The resulting view is exactly like the diagram window except that the name under the START and END blocks is that of the module, and only the blocks encapsulated in the module appear.



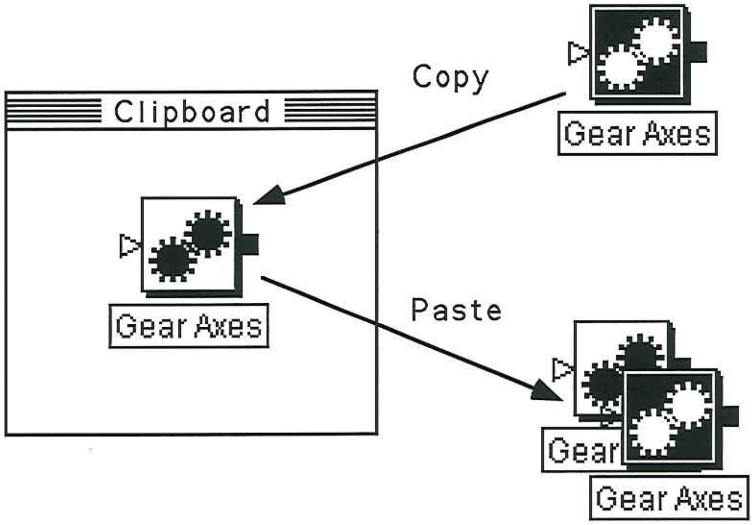
A module can be duplicated, copied, moved, connected, and encapsulated with other blocks or modules just like other blocks in the diagram. Modules can be unencapsulated to return the blocks they contain to the main diagram. Select the module and choose Unencapsulate from the Module menu. To return to the main diagram window, select it from the Module pop-up menu at the top of the screen or choose Show Overview from the Module menu.

Cutting, Copying, and Pasting Blocks

GML provides a set of tools which use the clipboard for editing diagrams. The clipboard is a temporary buffer for storing and retrieving data. The clipboard tools, located on the Edit menu, are used to copy blocks or modules from one module to another or between diagrams.

Any selected block or group of blocks with associated parameters can be copied to the clipboard. Copying has no effect on the diagram itself, leaving the selected blocks exactly as they were. When modules are copied, the information placed on the clipboard includes all of the blocks and other modules contained in the module.

Cut operates exactly like Copy except that the selected blocks are deleted from the diagram after they are copied to the clipboard. Paste puts the contents of the clipboard back on the diagram in the current view. Clear deletes the selected blocks, but does not put them on the clipboard. You can undo all of these actions if you make a mistake (choose Undo from the Edit menu).



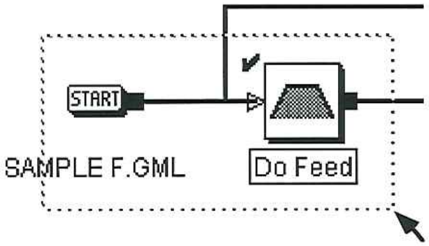
Using Change, Insert Between, and Swap

GML provides tools which make it easy to edit blocks that are already connected. These functions are also located on the Edit menu.

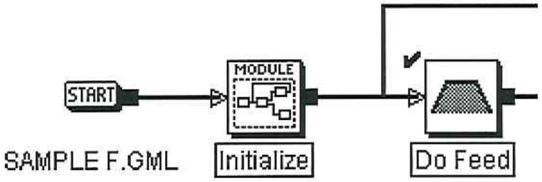
Change allows you to change the type of a block without affecting its connections. Select a single block and then choose Change from the Edit menu. The library appears. Choose the desired block and click **Select**. The original block is replaced by the block selected from the library.

Insert Between requires selection of two connected blocks. Select two blocks such as Initialize and Do Feed as shown opposite, and then choose Insert Between from the Edit menu. Select the desired block to insert (a pre-defined Initialize module in this case) from the library and click **Select**. The selected block is then placed between and connected to the originally selected blocks, as shown.

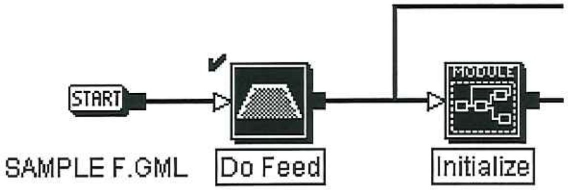
The Swap function swaps the location and connections of two selected blocks. The blocks do not have to be connected. Select two blocks such as Initialize and Do Feed, as shown opposite, and then choose Swap Blocks from the Edit menu. The two blocks change places but the connections do not change, as shown.



Select two



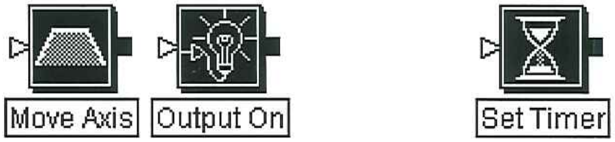
Insert Between



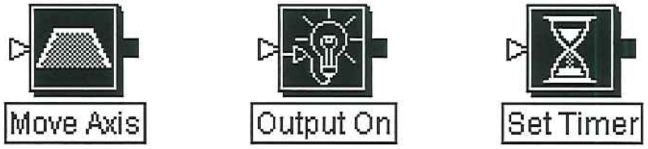
Swap Blocks

Spacing and Aligning Blocks

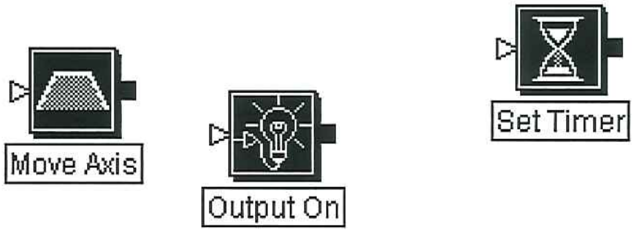
You can *align* selected blocks in a row and evenly *space* them horizontally to make your diagrams neat. To automatically space out the selected blocks evenly in a horizontal line, select the desired blocks



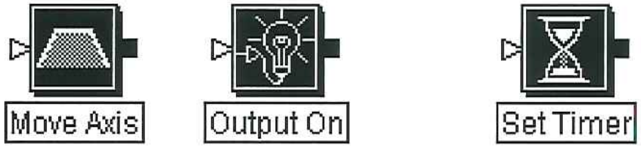
and then choose Space Blocks from the Edit menu. The blocks are evenly spaced, as shown below.



To automatically align blocks in a row, select the desired blocks



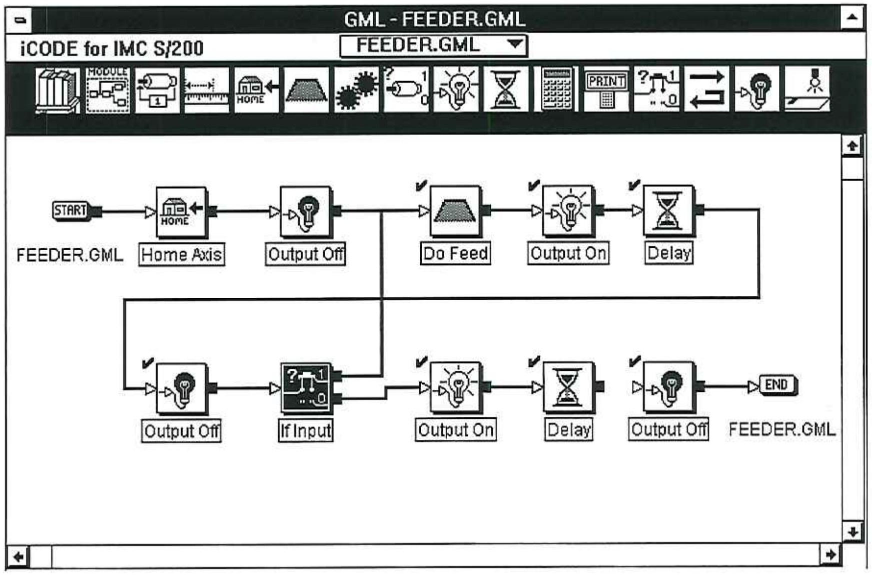
and choose Align Blocks from the Edit menu. The blocks are aligned to a common horizontal line, as shown below.



Snap To Grid

The Snap to Grid option in the Preferences dialog defines how blocks are placed in the diagram. GML divides the diagram into invisible horizontal and vertical lines. The horizontal grid is set for ¼ the size of a block and the vertical spacing is set for ½ the size of a block. You have the option of specifying that blocks be aligned to the nearest horizontal and/or vertical grid line when placed on the diagram. The Snap to Grid option sets the grid and defines the desired type of alignment. See Setting GML Preferences earlier in this manual for more information on grid snap.

Below is an example of a diagram aligned with grid snap. Note that functions and operations can flow to an additional row (or rows) of blocks if required.



Entering Block Parameters

Once a block has been added to a diagram, its parameters must be entered. Do this by double-clicking on the block and entering information in the displayed dialog box using the check boxes, buttons, scrolling lists, data entry boxes, and pop-up menus provided. Click

Save to save them and return to the diagram window. Help is provided in a scroll box at the bottom of the dialog. More detailed information on each block may be found in the programming manual for your motion controller.

Expressions

Although you can type a numeric parameter directly into a data entry box, you can also use the Expression Builder to build an expression. Expressions are formulas of variables, constants and functions that can be used in place of a numeric parameter. The Expression Builder includes a calculator keypad, a list of special operators, and easy access to lists of all variables, constants, and functions that can be used in expressions. Choose Build Expression from the Edit menu, enter the desired expression, and click **Save** to copy it to the appropriate data entry box.

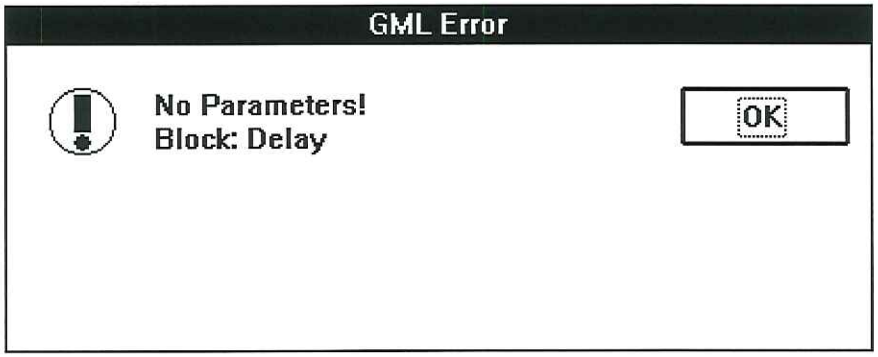
The items available as elements of an expression vary with the selected motion controller. See the programming manual for your motion controller for details about these elements.

Variables, Constants, and I/O

User variables, constants, discrete inputs and outputs, etc. must each be *defined* before they can be used as block parameters. These items are typically defined as needed. Choose the appropriate item from the Definitions menu and enter the data required. Click **Save** to save the definition.

Testing Diagrams

Choose Test Diagram from the Diagram menu to translate the current diagram into a *script* in the appropriate native language for the selected motion controller. The GML translator checks for errors or potential errors. If an error is found, the block and error are reported as shown below.


GML Error

No Parameters!
Block: Delay

Click **OK** in the error dialog and GML automatically zooms to the appropriate module in the diagram and highlights the offending block. If the error is with a parameter within a block, GML also automatically opens the block.

Note that a successful test only indicates that the diagram translates to a valid program, not that the motion controller will operate your machine flawlessly. Remember, the motion controller only does what you *tell* it, which may not necessarily be what you *want*.

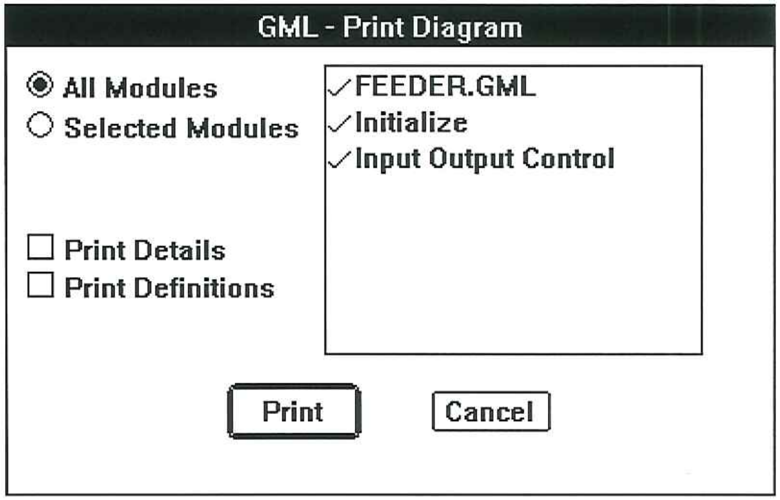
Saving Diagrams

Save your GML diagram by choosing Save from the File menu. If this is a new file, a dialog appears giving you the opportunity to enter an appropriate filename. Save As may also be chosen from the File menu to rename a file or save it to a different name or location.

Printing Diagrams

Both diagrams and scripts (programs) can be printed. However, the Windows Print Manager needs to be appropriately configured before printing. Use the Page Setup dialog (which will be different depending on your specific printer) to define the page layout.

Choose Print from the File menu to print the current diagram or script. The print dialog for the Simple Feeder diagram created in the tutorial is shown below.



GML - Print Diagram

● All Modules
○ Selected Modules

☐ Print Details
☐ Print Definitions

✓FEEDER.GML
✓Initialize
✓Input Output Control

This diagram has two modules (Initialize and Input Output Control) as shown in the scrolling list. You may print the diagram including all modules or selected modules as desired. A module is selected for printing if it is checked.

When Print Details is selected, the selected modules are printed with numeric identifiers assigned to each block in the module. On a separate page, each block's parameter information is printed beneath its assigned numeric identifier.

A Module Definitions page is printed when Print Definitions is selected. This page lists each module definition title with a description of the definition.

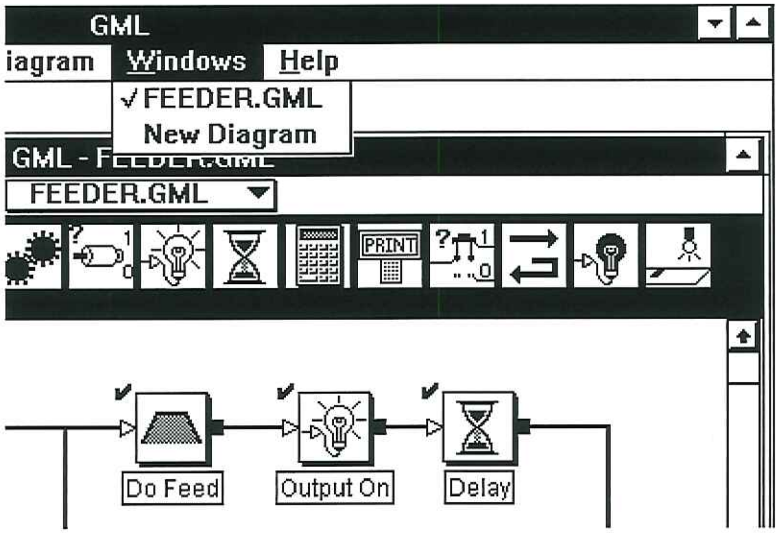
Navigating Complex Diagrams

GML provides several special methods for quickly moving around diagrams when they become large. These are explained following.

Using the Windows Menu

The Windows menu provides a way to quickly move between all open windows. This includes all diagrams and programs, as well as the Online Manager window. This is helpful if you have a small screen and cannot see all open windows because they are hidden.

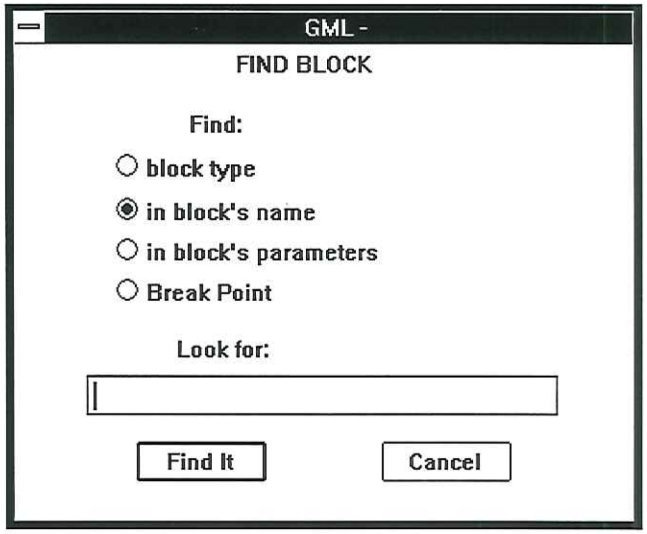
The Windows menu keeps track of all GML windows that are currently open and displays them by name when clicked. Choosing any item in the Windows menu immediately brings that window to the front and it becomes the active window.



Note that the currently active window is checked.

Searching a Diagram Using Find and Find Again

It is often useful to quickly locate a particular type of block, block name, or a specific parameter variable or value. Choosing Find from the Diagram menu brings up the Find dialog as shown below. There are three searching techniques.



GML -

FIND BLOCK

Find:

○ block type

● in block's name

○ in block's parameters

○ Break Point

Look for:

To search for occurrences of a particular block type, select block type and click **Find It**. The library is displayed so you may select the type of block to search for. Once a block type has been selected, the Find function searches the current diagram for the first occurrence of the block. When found, the block is highlighted. To search for the next occurrence of the specified block type, choose Find Again from the Diagram menu.

If the in block's name button is clicked, the Find function searches all block description fields for the first occurrence of the specified text. When the text is found, the block is highlighted. To find a particular parameter variable or value, select in block's parameters and enter the value to search for in the edit box. When the variable or value is found, the Find function goes directly to the parameter dialog containing the search text.

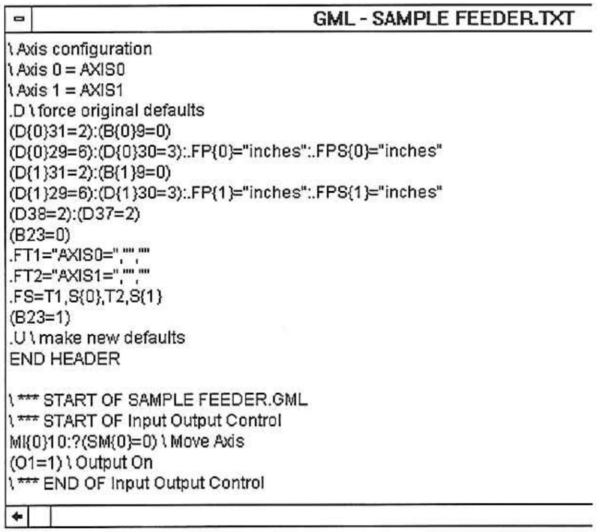
Both text searches are case insensitive and find all occurrences of the text including instances partially within longer words. To find the next occurrence of the specified text, choose Find Again from the Diagram menu.

Converting a Diagram to a Script

The actual language used for the motion controller program, or *script*, generated from a GML diagram is determined by the selected motion controller family. Regardless of the selection, however, the program may be viewed using the Script Editor. The script editor opens a program document.

1. Choose Open Diagram from the File menu. Find and open the Feeder diagram created in the tutorial.

2. Choose Translate to Script from the File menu. GML translates the Feeder diagram and displays the resulting iCODE program as shown below.



```
\ Axis configuration
\ Axis 0 = AXIS0
\ Axis 1 = AXIS1
.D \ force original defaults
(D{0}31=2):(B{0}9=0)
(D{0}29=6):(D{0}30=3):.FP{0}="inches":.FPS{0}="inches"
(D{1}31=2):(B{1}9=0)
(D{1}29=6):(D{1}30=3):.FP{1}="inches":.FPS{1}="inches"
(D38=2):(D37=2)
(B23=0)
.FT1="AXIS0=","",""
.FT2="AXIS1=","",""
.FS=T1,S{0},T2,S{1}
(B23=1)
.U \ make new defaults
END HEADER

\ *** START OF SAMPLE FEEDER.GML
\ *** START OF Input Output Control
MI{0}10:?(SM{0}=0) \ Move Axis
(O1=1) \ Output On
\ *** END OF Input Output Control
```

3. Save the program by choosing Save from the File menu and specifying the desired file name and directory.

Downloading a Diagram

After building a diagram, it must be transferred to the motion controller. If you are using an IMC S Control or MAX/IMC-201 family motion controller, the diagram may be downloaded directly to the motion controller. See the GML programming manual for your motion controller for more information.

If you are using an IMC Classic family motion controller, the diagram must first be converted to a script (MML program) as explained previ ously and then transferred to the ODS (Offline Development System) software for compiling and downloading to the motion controller.

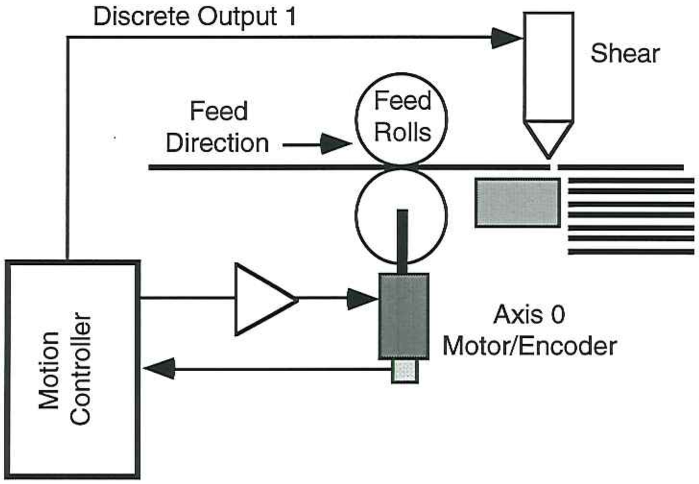
Quitting GML

To leave GML, choose Quit from the File menu. GML checks all open documents for changes and asks if each changed document should be saved before quitting.

GML Tutorial

The following tutorial illustrates the features and operation of GML Programmer's Workshop by taking you step-by-step through the process of creating a complete GML diagram. Before using this tutorial, you should be familiar with the basic GML concepts discussed in Learning GML earlier in this manual.

For this tutorial, the machine to be controlled is a feeder, used to cut a continuous material (such as paper, plastic, sheet metal, etc.) into pieces of a specific length, as diagrammed below.

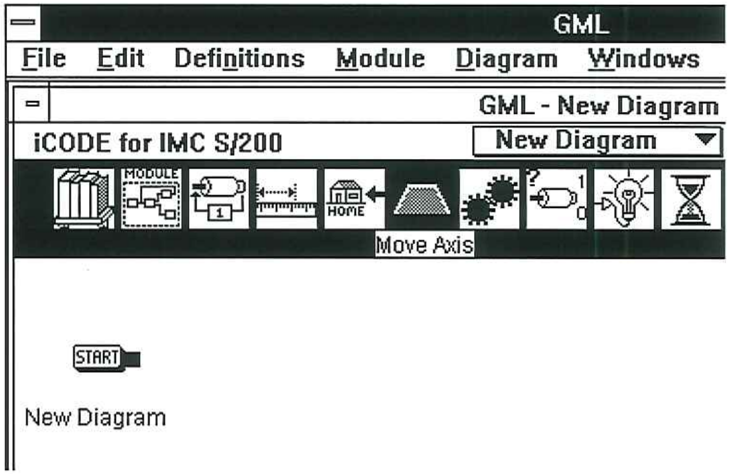


In operation, a length of material is fed via the feed rolls by commanding the motor to move the specified distance and then stop. The material is cut by a shear activated by a discrete output from the motion controller. This cycle is repeated until the necessary number of pieces have been cut.

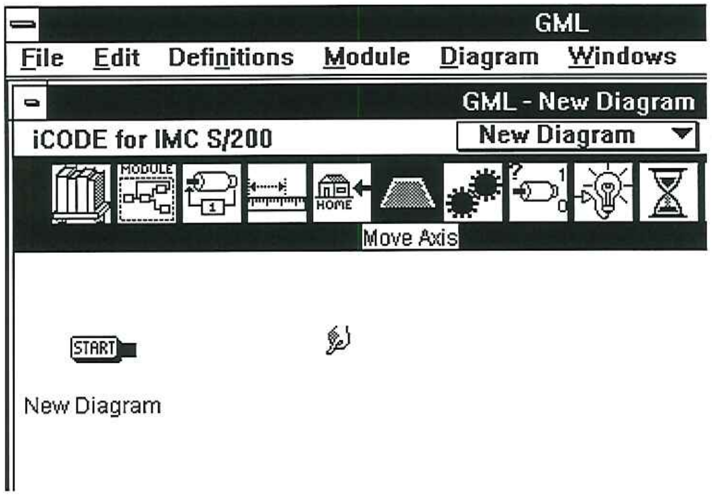
Place a Move Axis Block

The first step in building the Feeder application is adding a Move Axis block.

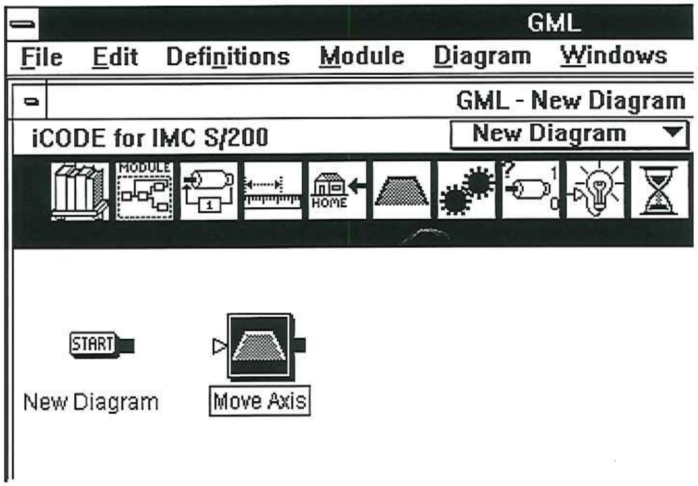
1. Click on the sixth icon in the block palette. GML highlights the icon and displays the block name, in this case Move Axis.



2. Move the pointer down onto the field of the diagram. Notice that the pointer changes to a hand indicating that it is ready to place the selected block.



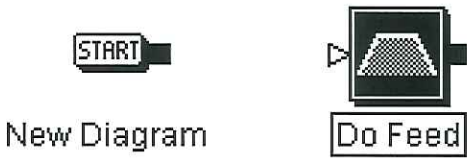
3. Move the hand to a position to the right of the START block and click. This places a Move Axis block at the hand location.



When a block is added, the description field beneath the block is highlighted. To change the block description to something meaningful in your application, type in a new description. When the new description is complete, move the pointer away from the description box and click. To keep the original description, do not type in a new description, just move the pointer away from the description box and click.

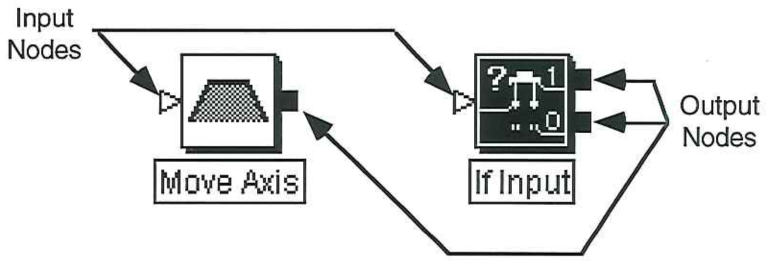
If an error is made, correct it using standard editing procedures. If the description is entirely erased and the insertion point lost, click exactly in the center beneath the block and type in the desired description.

4. Type "Do Feed", move the pointer away from the description, and click .

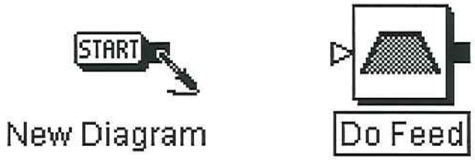


Connect the Start and Move Axis Blocks

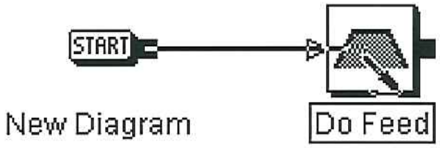
Connections define the sequential flow of a diagram. Connections always start from the START block. In this example, the Move Axis block is the first action and must be connected to the START block.



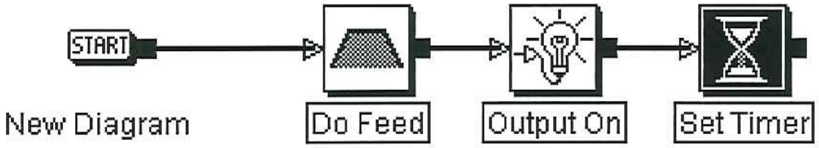
1. Move the pointer to the output node of the START block. Notice that the pointer changes to a soldering iron. Press and hold down the mouse button.



2. Drag the soldering iron into the Do Feed block and release the mouse button to connect them. Notice the soldering iron disappears and the pointer is back.



3. Before continuing, fill out the diagram by adding two more blocks and making the connections shown below.



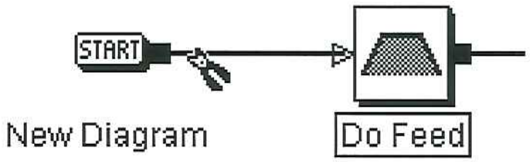
Move the Do Feed Block

Position the pointer over the Do Feed block, then press and hold down the mouse button. Move the pointer to another location and release the mouse button. Notice that while you are moving the block, it appears as a dotted box with a hand in it, following the pointer. After releasing, notice that the connections follow the position of the block.

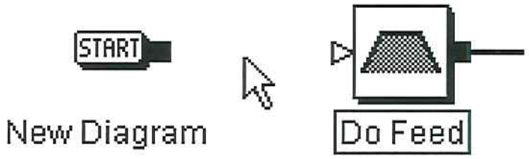
Disconnect the Do Feed Block

Connections between blocks are removed using the wire cutter tool.

1. Position the pointer over the connecting line between the START block and the Do Feed block, notice that the pointer disappears and a wire cutter appears, allowing you to cut the connection between the blocks.



2. Click and notice that the connection is removed and the pointer is back.

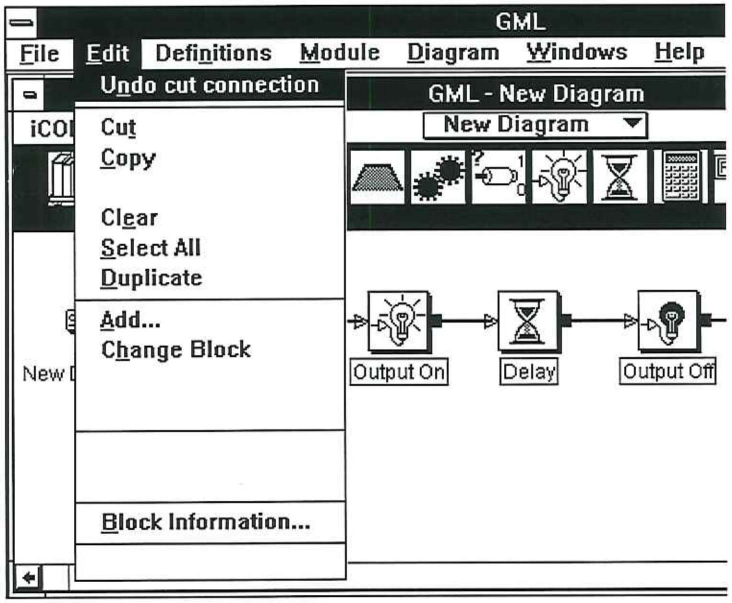


RA v. AMS
Ex. 1021

Undo the Disconnect

Most actions that change a diagram can be reversed using the Undo function.

1. Choose Undo from the Edit menu. Notice that the disconnect action from the last section is removed and the START/Do Feed connection has been restored.

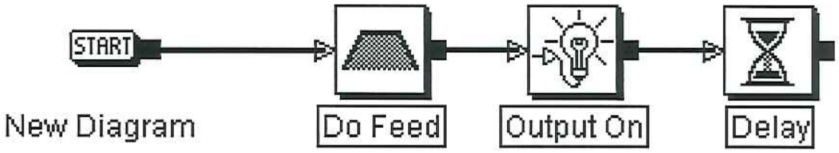


Change the Set Timer Block Description

Block descriptions may be changed at any time.

1. Position the pointer on the description text for the Set Timer block and click.

2. The description field should be highlighted. Type "Delay", move the pointer away from the description field and click.



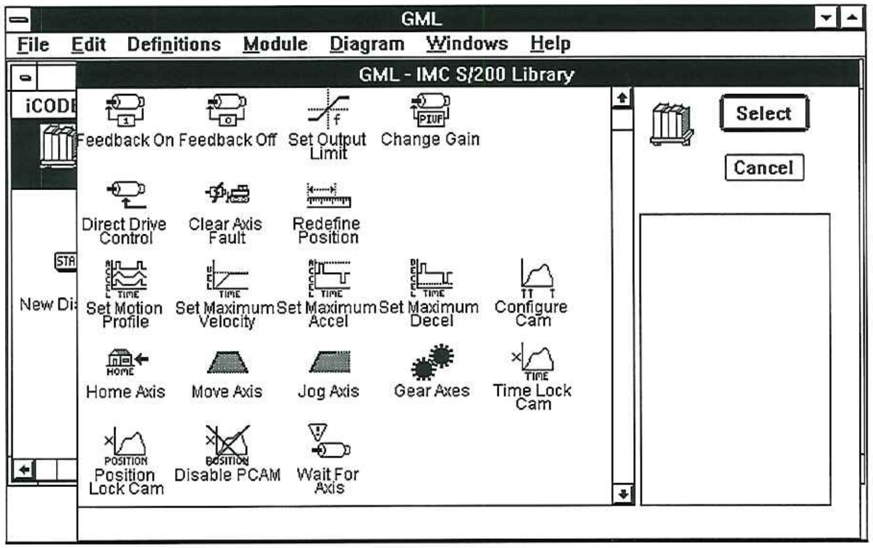
Add an Output Off Block from the GML Library

The block palette at the top of the diagram holds the most frequently used blocks. To view and select from the complete list of blocks, click the leftmost block in the block palette (the library block) to open the GML *library*. The library contains all the available blocks for your motion controller *including* those shown on the palette. The library displays block icons and descriptions of each. If you are using a PC with a color monitor, the blocks are color coded according to related functions (motion blocks are green, etc.).

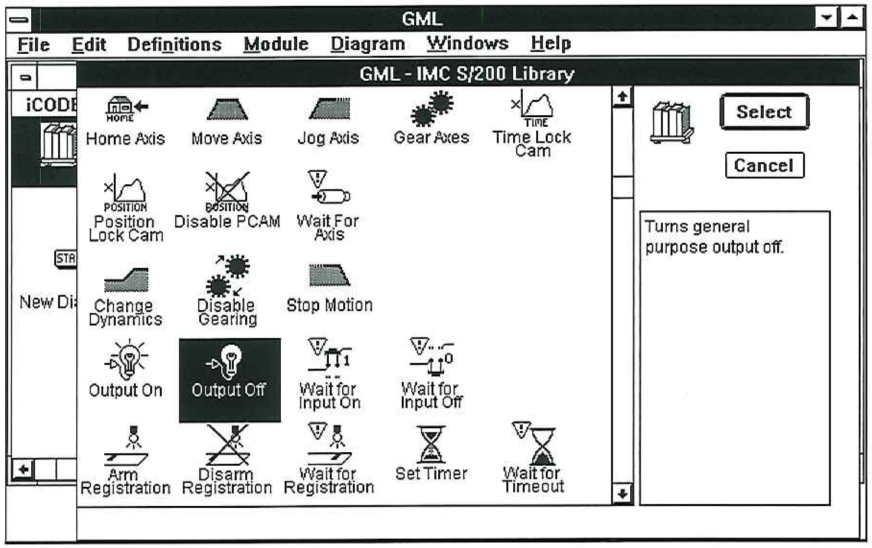
The library may also be called by choosing Add from the Edit menu or by double clicking on any icon in the palette. Clicking on a block icon in the library highlights the icon and displays a short description of the block at the right side of the library window. Double-clicking on a block selects the block and exits the library.

RA v. AMS
Ex. 1021

1. Choose Add from the Edit menu. The library is displayed.



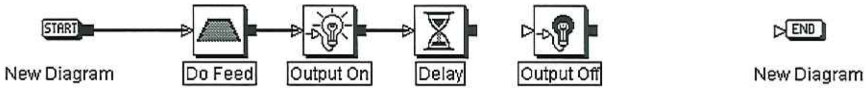
2. Scroll down through the list until you see the Output Off block and click on it. Notice the description displayed on the right side of the library.



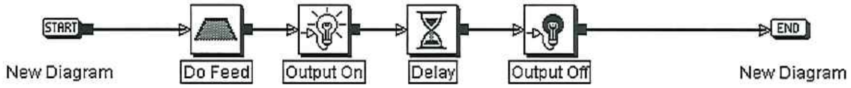
Turns general purpose output off.

3. With the Output Off block highlighted, click Select to exit the library and return to the diagram.

4. Position the hand cursor between the Delay block and the END block and click to insert the Output Off block.



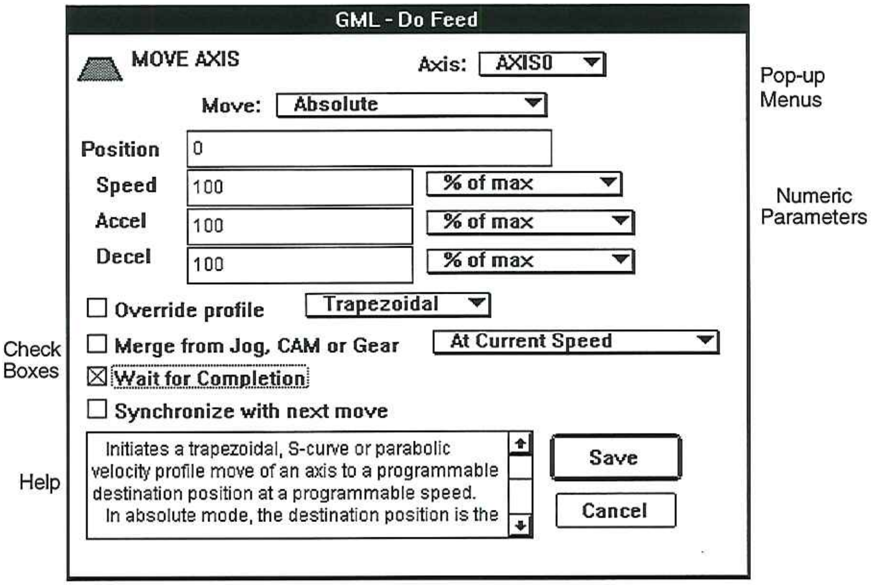
5. Connect the Delay block to the Output Off block and the Output Off block to the END block.



Enter the Block Parameters

Before a GML diagram can be translated into a motion control program, the parameters for each block must be entered. Do this by double-clicking on the block icon and setting parameter values using the check boxes, buttons, scrolling lists, edit text boxes, and pop-up menus provided. Help is provided in a scroll box at the bottom of the dialog.

1. Double-click on the Do Feed block.



GML - Do Feed

MOVE AXIS Axis: AXIS0

Move: Absolute

Position	0	
Speed	100	% of max
Accel	100	% of max
Decel	100	% of max

□ Override profile Trapezoidal

□ Merge from Jog, CAM or Gear At Current Speed

☒ Wait for Completion

□ Synchronize with next move

Initiates a trapezoidal, S-curve or parabolic velocity profile move of an axis to a programmable destination position at a programmable speed.
In absolute mode, the destination position is the

Save

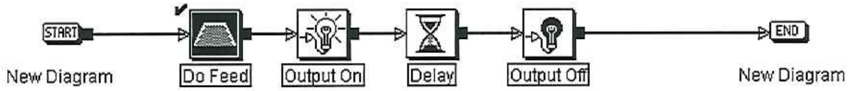
Cancel

2. Click in the box to the left of the text "Wait for Completion". Notice the X in the box. Click again. Notice the X is removed. Before finishing with the dialog, make sure you leave it checked for this diagram.

3. Choose Incremental from the Move pop-up menu. Notice that the type of move shown changes to Incremental.

4. Double-click on the 0 in the edit box to the right of Distance. Type the value 10.

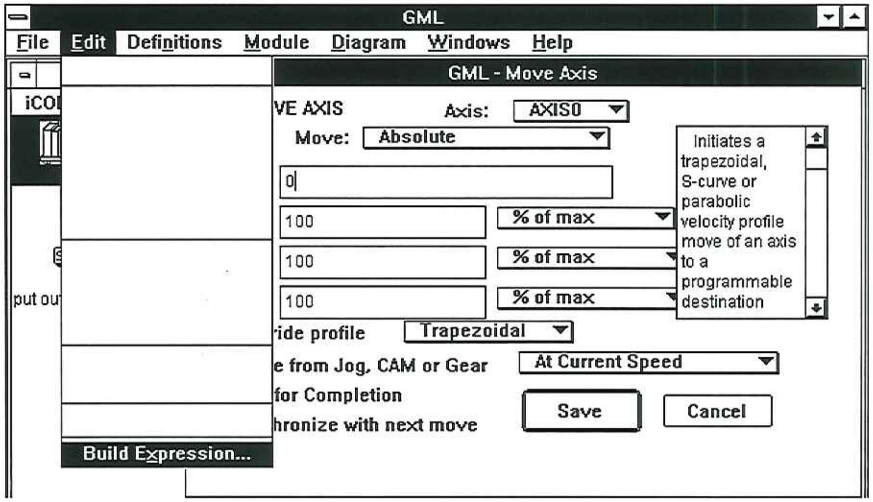
5. Click Save to exit the dialog and save the parameters. GML checks the parameters to make sure they are valid and then returns to the diagram. A check mark to the upper-left of the block indicates that parameters have been entered.



Enter an Expression for the Move Distance Using the Expression Builder

Expressions are formulas of variables, constants and functions that can be used in place of a numeric parameter. Although you can type a numeric parameter directly into an edit box as in step 4 above, you can also use the Expression Builder to build an expression when required.

1. Return to the Do Feed block Parameter dialog and choose Build Expression from the Edit menu.



Axis: AXIS0

Move: Absolute

0

100 % of max

100 % of max

100 % of max

Initiates a trapezoidal, S-curve or parabolic velocity profile move of an axis to a programmable destination

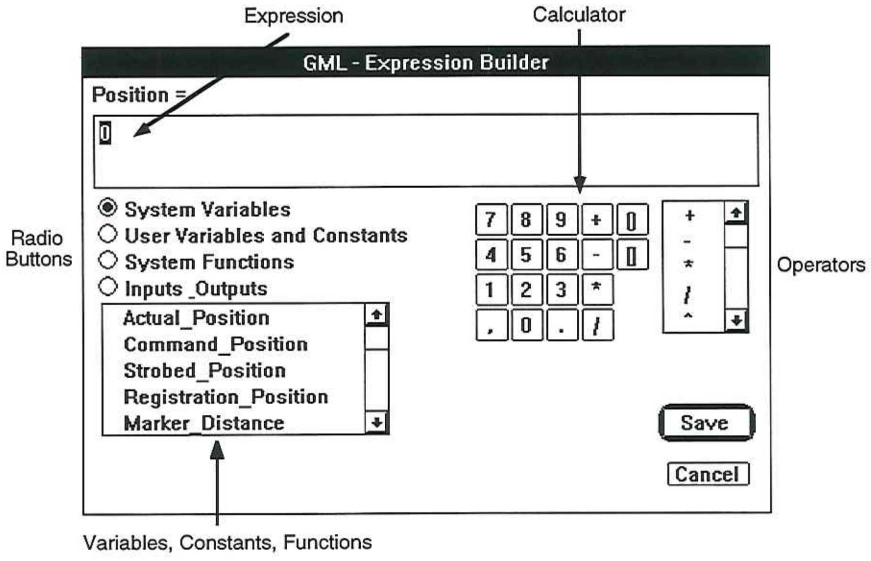
ride profile Trapezoidal

e from Jog, CAM or Gear At Current Speed

for Completion

hronize with next move

The Expression Builder is displayed as shown below.



2. Click on the calculator buttons "1", "0", and "+" in sequence. Notice the expression being built in the box at the top of the Expression Builder.

3. Click the System Variables button. A list of system variables is displayed in the list window.

4. Click the list item `Actual_Position`. Notice that an Axis pop-up menu and an **Insert** button appear.

5. Choose `AXIS1` from the Axis pop-up menu.

6. Now click **Insert**. Notice that the expression is updated.

7. Click **Save**. This copies the expression

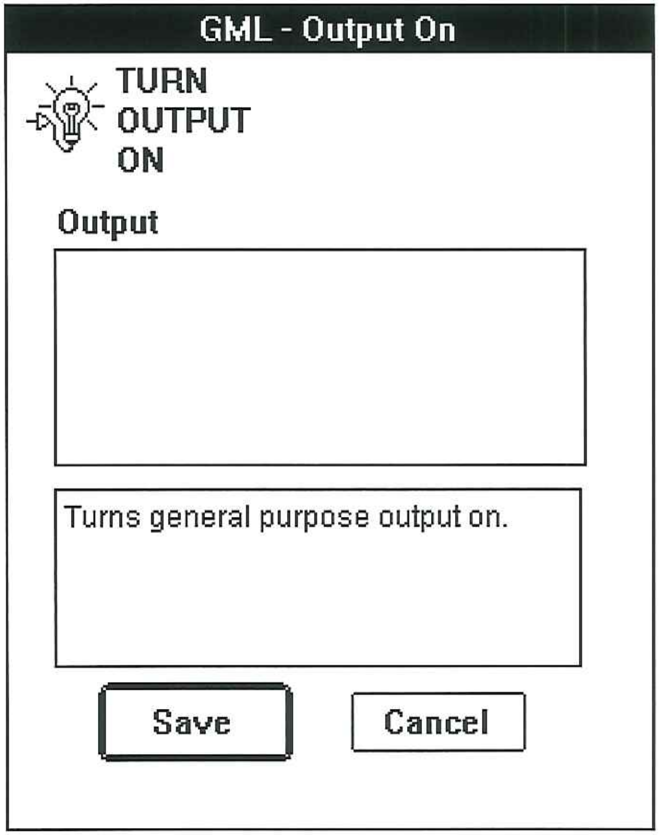
```
10 + Actual_Position_AXIS1
```

to the Position edit box in the Do Feed dialog. Click **Save** to exit the Do Feed dialog.

Define the Motion Done Output

The Feeder program turns on a discrete output after completing each feed. This output must be defined.

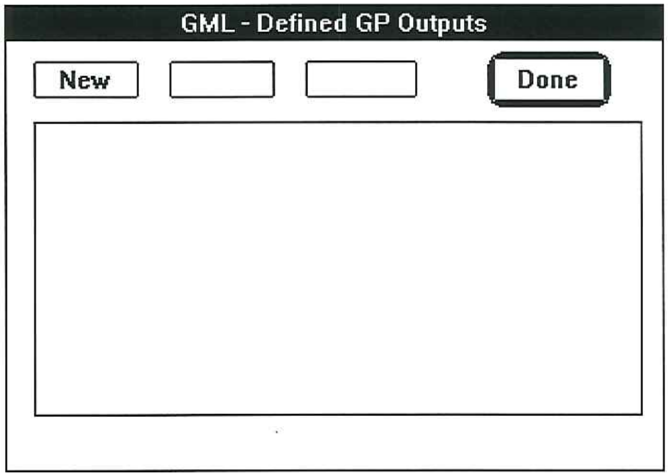
1. Double-click on the Output On block. The Output list of this block is empty because no signals have been defined.



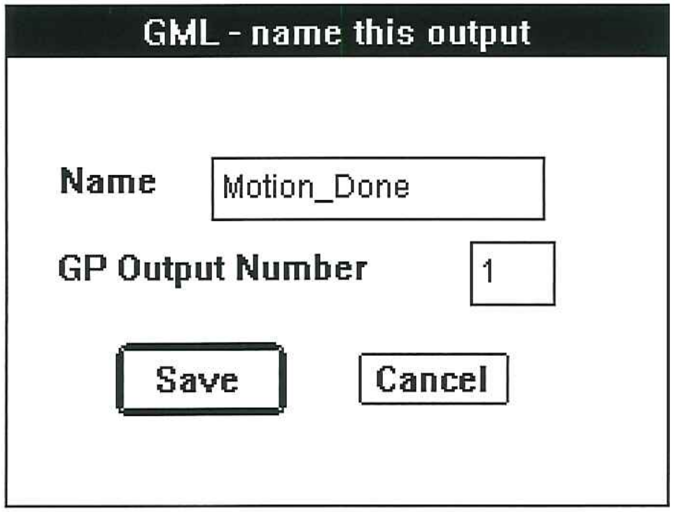
Turns general purpose output on.

2. To define a general purpose output, choose General Purpose Outputs from the Definitions menu.

3. Click on the New button to define a new output.



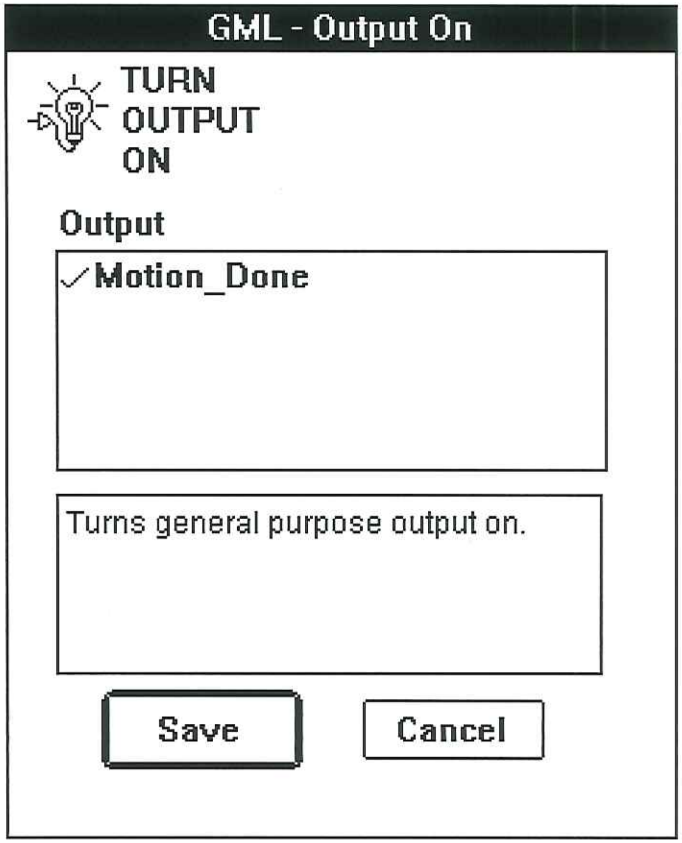
4. Define the output signal as shown below. Type the name Motion Done. Position the arrow on the GP Output Number edit box, double-click and type 1 to define the signal as Output 1. Click **Save**.



Name Motion_Done

GP Output Number 1

5. Click **Done** to return to the Output On dialog. Since only one output is defined, GML automatically selects it in the list with a check mark.



✓Motion_Done

Turns general purpose output on.

6. Click **Save** to save the parameters.

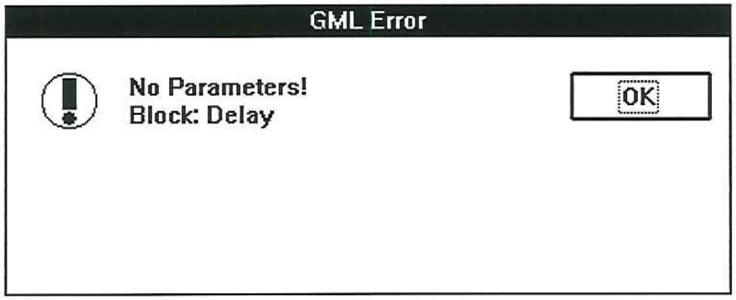
7. Select the same `Motion_Done` signal in the Output Off block parameters dialog. Verify that the `Motion_Done` signal is checked. Click **Save** to save the parameters.

Test the Diagram

Choose Test Diagram from the Diagram menu to translate the current diagram to a script and check for errors or potential errors. Note that a successful test only indicates that the diagram translates to a valid program, not that the program will control the machine flawlessly. Remember, the motion controller only does what you *tell* it, which may not necessarily be what you *want*.

1. Test Feeder by choosing Test Diagram from the Diagram menu.

2. A translator error message explains that the Delay block parameters are missing.



GML Error

No Parameters!
Block: Delay

3. Click **OK**. GML automatically finds and opens the Delay block so that the parameters may be defined.

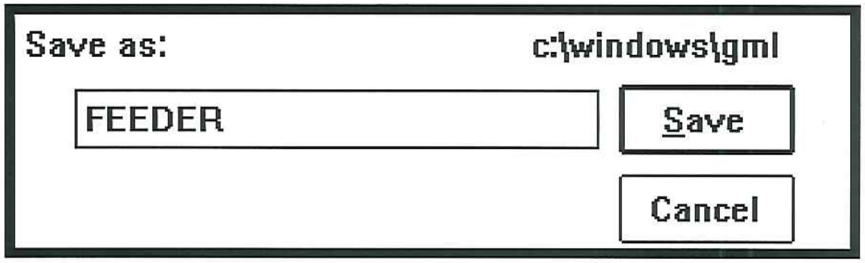
4. Set a 0.5 second delay and check the Wait for Timeout option. Click **Save**.

5. Repeat Test Diagram. This time there should be no errors.

Save the Diagram

At this point, the Feeder diagram should be saved to disk.

1. Choose Save from the File menu.
2. Currently, the application is still named the default diagram name, New Diagram. Change the name by typing "Feeder".
3. Click **Save** to save the Feeder diagram.



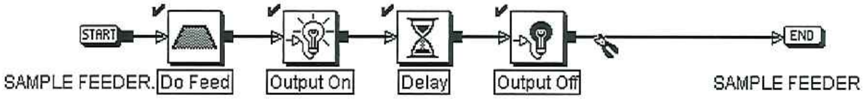
Save as: c:\windows\gml

FEEDER

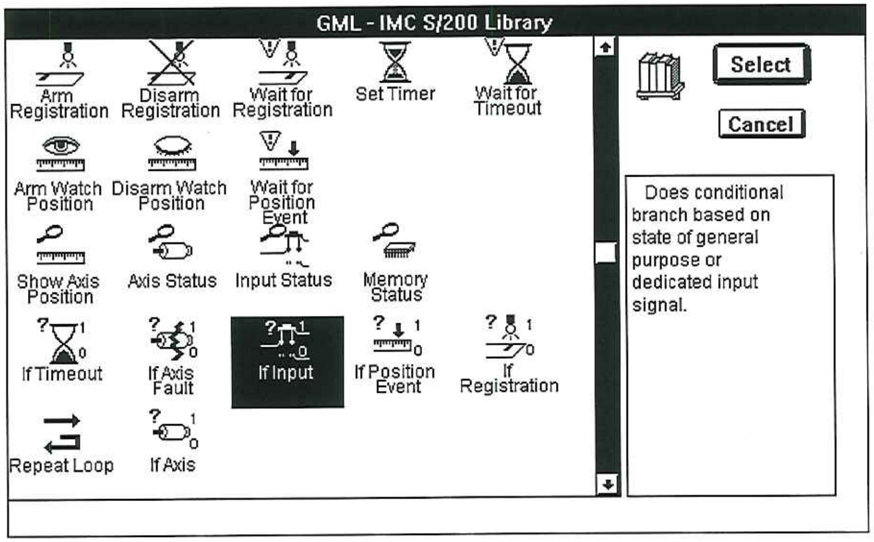
Add a Decision-Making Block

Decision-making blocks that define conditional sequences have two output nodes. If a defined condition is true, the diagram continues from the 1 node; if the condition is false, the diagram continues from the 0 node. In this diagram, we need to decide whether to feed another piece or not based on the state of a discrete input.

1. Disconnect the Output Off block from the END block.

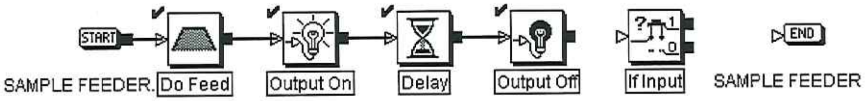


2. Click the library icon in the palette to access the library. Find and select the If Input block.

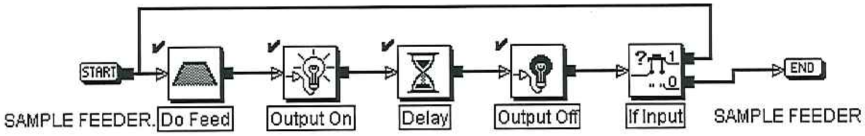


Does conditional branch based on state of general purpose or dedicated input signal.

3. Place the If Input block between the Output Off block and the END block. If necessary, move the existing blocks to make room.



4. Connect the Output Off block to the If Input Block and the If Input block to both Do Feed and END as shown below.

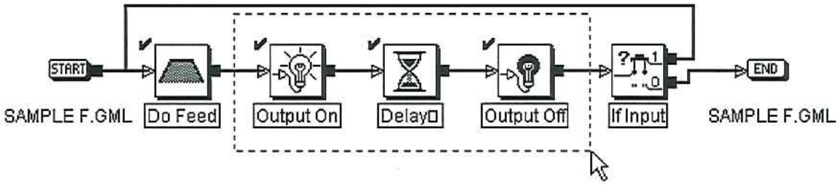


Define a discrete input to be used to determine whether to cut another piece to complete the parameters for the If Input block.

Encapsulate the I/O Blocks into a Module

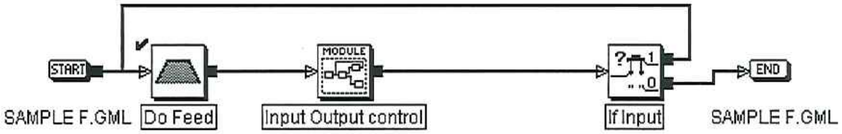
Modules are blocks which contain other blocks or modules. They appear on a diagram as a single block, but represent the functions of all the blocks and modules contained within them. Besides saving space, they hide confusing detail and allow grouping of blocks that comprise a single function or several related functions into a single block. The contents of any module can be accessed by clicking on it and choosing Show Details from the Module menu.

1. Position the arrow tool above and to the left of the Output On block. Press and hold the mouse button. Drag the arrow to a position below and to the right of the Output Off block. Release the button to select the Output On, Delay and Output Off blocks.

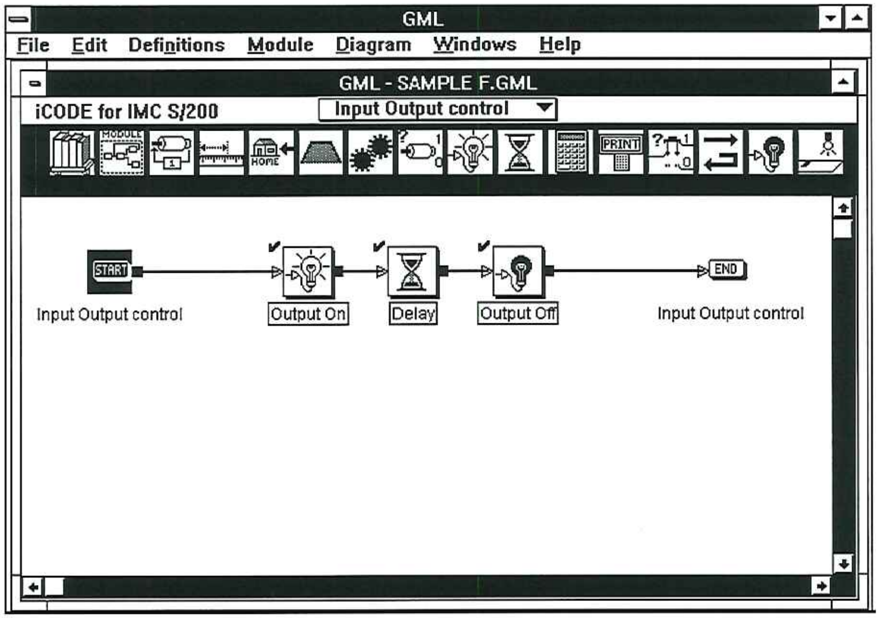


2. Choose Encapsulate from the Module menu.

3. A Module block replaces the three selected blocks. Type Input Output Control, then move the pointer away from the description field, and click.



4. Click on the Input Output Control module and choose Show Details from the Module menu.



Notice that the diagram window now shows the Input Output Control module. The module has its own START and END blocks and below each of them is the module description. The Output On, Delay, Output Off sequence is unchanged. The module currently displayed is also indicated in the Module pop-up menu located at the top of the diagram window just under the diagram title.

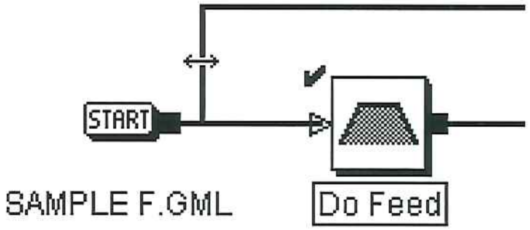
5. Choose Show Overview from the Module menu to return to the top level of the diagram.

You can also use the Module pop-up menu to go directly from a module to the next level module or to the top or overview level of the diagram window.

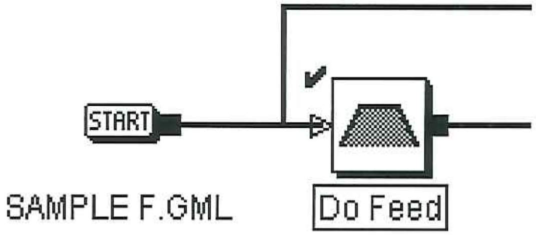
Add an Initialization Module

You can use modules to represent a group of blocks you have already created via encapsulation as illustrated above, or you can place blank modules directly into your diagram and later fill in their detailed operations. This method is particularly convenient for sketching out the process at a high level before having to think about detailed functionality. An "empty" module can be added to a diagram by selecting the module icon in the palette and placing it like any other block. As an example, we'll add an initialization module to the sample application.

1. Move the Do Feed block, and the Input Output Control Module to the right to make room for the new module. To slide the "loopback" connection to the Do Feed block over, position the pointer over the vertical section of the connecting line. The pointer becomes a double arrow.



2. Press the mouse button down and hold it. Drag the mouse to the right, moving the connection over to the right. Release the mouse button when the desired position is reached.

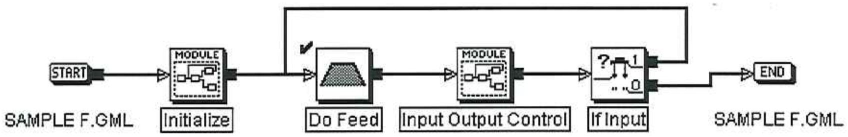


3. Select both the START and Do Feed blocks by dragging a box around them. They should both become highlighted.

4. Choose Insert Between from the Edit menu. The library appears. Any library block selected is inserted between the two selected blocks on the diagram, and all connections are automatically fixed.

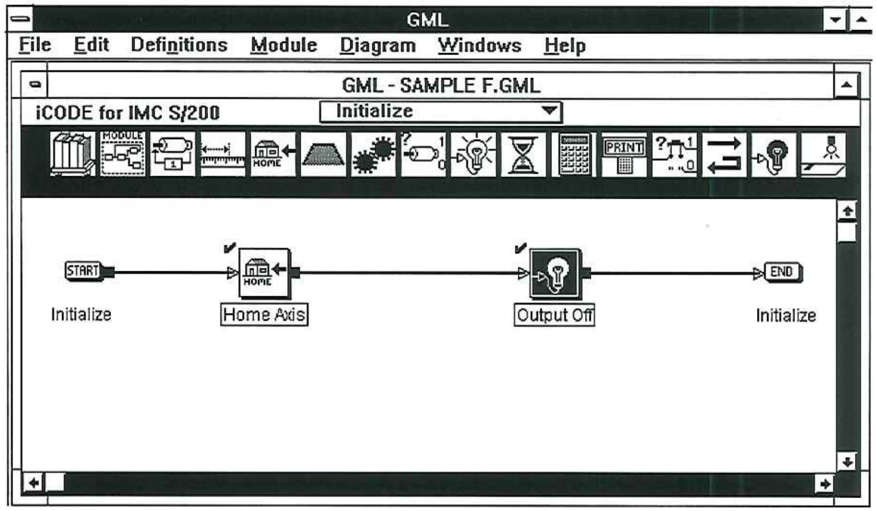
5. Scroll down through the library until the “New Module” icon shows (it’s near the end). Double-click on it and it is automatically inserted in the correct position.

6. Type “Initialize” into the description field.



7. Define the new Initialize Module by double clicking on it. Place a Home Axis and an Output Off block onto the diagram. Connect the module START to the Home block. Connect Home to Output Off and then Output Off to the module END with the soldering iron.

8. Double click on the Home block to enter its parameters. Choose AXIS1 and Normal mode from their pop-up menus, and select Wait for Completion. In the Output Off block parameters, select the Motion_Done output.



9. Go back up to the top level by choosing Feeder from the Module pop-up menu.

Once a module has been created, that module becomes part of the library. From the library, you can select your previously defined module and place it anywhere in the diagram. The New Module module in the library is an empty module.

1. Position the pointer above and to the left of the If Input block. Press and hold the mouse down. Drag the pointer to a position below and to the right of the END block. Release the mouse button. The If Input and END blocks should be highlighted.

2. Access the library by choosing Insert Between from the Edit menu.

3. Scroll down to the Input Output Control module block (near the end), click on it and click **Select**. The new block is placed between the If Input block and the END block.

4. Place the arrow on the new Input Output Control module block and click.

5. Choose Show Details from the Module menu.

6. Notice that this module is an exact copy of the original Input Output Control module.

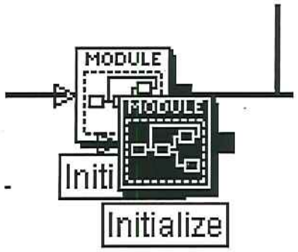
GML is very flexible about module creation. If you decide that you no longer need to represent a group of blocks as a module, you can bring them up one level in the hierarchy by using Unencapsulate from the Module menu. To "unpackage" a module in this way, select it by clicking on it once and then choose Unencapsulate from the Module menu. Any blocks or modules contained within this module are brought up to the current level with all their connections intact. It may be necessary to do a little rearrangement of the unencapsulated blocks to make the diagram look better.

Copy and Duplicate the Initialize Module

Modules can be copied and duplicated. These two functions are very different. A copied module can be modified and the modifications do not effect the original module. A duplicated module is really just another instance of the original module, modifications to either instantly effect both modules.

1. Select the Initialize module by clicking on it.

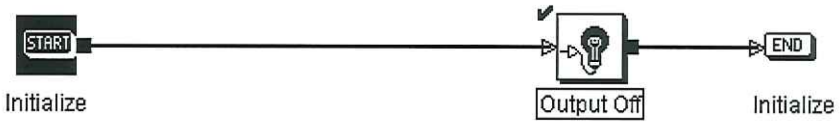
2. Choose the Duplicate option from the Edit menu.



Notice that the names of the two modules are identical.

3. Double click on the new Initialize module to see the details of the module.

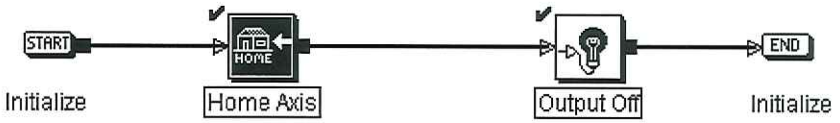
4. Select the Home block, then choose Clear from the Edit menu—this removes the Home block from the module.



5. Return to the Feeder level via the Module pop-up menu.

6. Double click on the original Initialize module, notice that the Home block is not present.

7. Choose Undo clear from the Edit menu to return the Home block to the module.



8. Return to the Feeder level and repeat steps 1 through 6, but instead of duplicating the Initialize module, copy it. Notice that the modification to the new module does not effect the original module.

After experimenting with copying and duplicating the Initialize module, be sure to delete all but the original module from the Feeder diagram. At this point, you should test the diagram by choosing Test Diagram from the Diagram menu to ensure you have a working program. Fix any mistakes or errors and then save the completed Feeder diagram.

This completes the GML tutorial.