QuickStart Guide

This guide will show the steps required to create a postprocessor utilizing the standard IMSpost™ controller files and the IMSmodel™ machine configuration files.

This guide provides an initial overview of standard development of a postprocessor. Each explanation is followed by a task to be accomplished following the Lets Do It! statement.

This Quickstart Guide is to introduce IMSpost users to the method of defining a postprocessor and is not intended to replace training.

Consult the IMSpost™ Users Guide for additional information on the various sections for development of a postprocessor.

The intention of this guide is not to introduce or create macro programming, but utilize IMSpost™ standards. Macro programming courses are available and can be scheduled by contacting your IMS Dealer.
IMSpost Quickstart
Window Environment ......................................................................................... 1
IMS Window Interface ..................................................................................... 1
IMS Dialog Boxes .......................................................................................... 2
Dialog Box Definition ..................................................................................... 3
Starting the IMSpost Interface Environment ................................................. 4
Creation of the IMSpost Processor ................................................................. 5
Explanation of Select Postprocessor Dialog Box ........................................... 6
Use Default Machine Description ................................................................. 6
New Machine Description ............................................................................ 7
Select New Machine Description ................................................................. 8
Customize the Post Processor ....................................................................... 9
IMSpost Checklist ......................................................................................... 9
Start of Program ............................................................................................ 10
Partno Definition ........................................................................................ 11
Test Commands – Program, Part No ............................................................ 12
  Test Commands - Completed ................................................................... 13
Coolant Definition ......................................................................................... 14
Test Commands – Coolant Definition .......................................................... 15
  Test Commands - Completed ................................................................... 16
Spindle Definition ........................................................................................ 17
Register Definition ....................................................................................... 18
Tool Change .................................................................................................. 19
  Tool Change – Tool Definition ................................................................ 19
  Tool Change – Tool Definition ................................................................ 19
  Tool Change – Data Options ................................................................... 20
  Tool Change – Before/After Output ......................................................... 21
Test Commands – Program, Part No, Coolant, Spindle, Tool CHG ............. 23
Feedrate Definition ....................................................................................... 24
Cutter Compensation .................................................................................... 25
Motion Functions ........................................................................................ 26
  Motion Function Data Tabs ..................................................................... 26
IMSpost Quickstart

SORT OUTPUT ........................................................................................................... 51
SORT OUTPUT ........................................................................................................... 52
TEST COMMANDS – SEQUENCE BLOCKS, SORT OUTPUT .......................... 53
TAPE FORMAT ....................................................................................................... 54
POSTPROCESS ....................................................................................................... 55
POSTPROCESSOR – TEXT FILE OUTPUT ......................................................... 56
Window Environment

**IMS Window Interface**

The IMSpost uses a Windows interface for the creation of the postprocessor. This guide will introduce a few Icon and Pull-down Toolbar definitions to get you started. IMS instructional training will expand on the use and purpose of additional Icon and Pull-down toolbar options.
IMSpost Quickstart

IMS Dialog Boxes
Standard postprocessor functions can be customized through dialog boxes.

Dialog boxes are launched by selecting the desired task from the menus, the toolbars, or the checklist.

Most dialog boxes have an **OK** or **Cancel** to complete the task.

Part Number Dialog Box
Initiated by the IMSpost Checklist

IMS Checklist and Dialog Box
Dialog Box Definition

The dialog box defines the choices or options for the specific postprocessor function.

- **Radio buttons**: Choose from various options.
- **Check box**: Confirm choices or cancel choices.
- **Default input**: Pre-filled text that can be changed.
- **Pull-down list**: SELECT ITEM
- **Optional input**: Text that is not required to be filled.
- **On-line documentation**: Detailed instructions on using the dialog box.

**Example Dialog Box**

```
Part Number Options
- Ignore Part Number processing
- Read from APT/CL PARTNO statement
- Read from APT/CL, if exists; else prompt user
- Prompt user at runtime

Prefix string for output with Part number
Part number register
Default Part number value
Sequence numbers off with part number
```

**Ok**, **Cancel**, **Help** buttons are available for further actions.
IMSpost Quickstart

Starting the IMSpost Interface Environment

This is your opportunity to start a post processor - **Let's Do It!**

1. Select **Start** icon ↓
2. Select **Program** ↓
3. Select **IMSpost 70** ↓
4. Select **IMSpost**

5. The execution of the program **IMSpost** will display the IMS Window Interface.
Creation of the IMSpost Processor

Default machine description.

The first step to create a postprocessor is the selection a default machine description - **Let's Do It!**

1. Select **File** from toolbar ↓
2. Select **New** from pulldown.
3. Displays **Select Postprocessor** dialog box.
4. Select **milling.def** from listing.
5. Select **OK** standard post is loaded.
Explanation of **Select Postprocessor** Dialog Box

**Use default machine description**
This example uses *milling.def* from the Postprocessor Listing.

Execution of dialog box:  1. Activates default parameters
2. Displays IMSpost Checklist

**List of library posts**
- Project File – complete description of parameters defines filename with extension .prj
- Default parameters of machine selected or new machine format requirements
- Sample output from this library postprocessor

**Note:**
Controller and Model Number determine default filename. Select **OK** to load postprocessor parameters.
The project file saves as any windows file. - **Save** or **Save As**...
New Machine Description
Standard library files do not support multi-axis machines. Various machine configurations are available by selecting a new machine description.

Lets Do It!
This is similar to a previous selection of a new machine description.
1. Select File from toolbar ↓
2. Select New from pulldown.
3. Displays Select Postprocessor dialog box.
4. Select milling.def from listing.
5. Identify the radio button Select new machine description
6. Select OK standard post is loaded.
Select new machine description
Select machine dialog box display various machine configurations. The dialog box has two portions:
- Selection Tree
- Machine configuration

Let's Do It!
1. Open selection tree to define machine. Select plus sign to open each level or minus sign to close:
   - Milling → Vertical → 4-axis → C-table
2. Observe the change in machine configuration.
3. Complete the selection with Load Machine.

The IMSpost will link the machine configuration with a standard controller file. The IMSpost Checklist will be displayed allowing custom modifications to occur.

The Quickstart Guide will introduce the Program Codes from the IMSpost Checklist to get you started with a 3-axis description.

The definition of multi-axis will require an expanded use of the IMSpost Checklist that includes motion axes, machine comments and reference points.

Additional training instruction is available to satisfy the multi-axis description.
Customize the post processor

Customizing the post processor is achieved by displaying the IMSpost Checklist, identify a parameter value and executing change.

Currently you have an unsaved project file named `milling_def.prj`.

**IMSpost Checklist**

The IMSpost Checklist is a sheet of steps suggesting a set of tasks necessary to complete the postprocessor.

- Program Codes – predefined parameters
- Controller Format - predefined parameters
- Machine Format - predefined parameters

First column of checkmarks indicates assigned parameter values. **Optional Choice:** Double click checkmark to remove checkmark; a parameter change will redisplay the checkmark, a visual verification.

Modifications will occur on the Program Codes on the proceeding pages. Remove checkmarks to better identify changes using the Quickstart Guide.

**Lets Do It!**  Remove all checkmarks from **Program Codes** Category.

---

**Default IMSpost Checklist**

**Before**

**Program Codes**
- Start of Program
- Partno Definition
- Coolant Definition
- Spindle Definition
- Tool Change
- Feedrate Definition
- Cutter Compensation
- Motion Functions
- Drill Cycles
- End of Program

**Controller Format**
- Sequence Numbers
- Comment Blocks
- Register Format
- Function Codes (G/M)
- Sort Output

**Machine Format**
- Motion Axes
- Machine Components
- Reference Points

---

**Checkmarks removed**

**After**

**Program Codes**
- Start of Program
- Partno Definition
- Coolant Definition
- Spindle Definition
- Tool Change
- Feedrate Definition
- Cutter Compensation
- Motion Functions
- Drill Cycles
- End of Program

**Controller Format**
- Sequence Numbers
- Comment Blocks
- Register Format
- Function Codes (G/M)
- Sort Output

**Machine Format**
- Motion Axes
- Machine Components
- Reference Points
Start of Program

The *start of program* dialog controls what codes are output at the start of every program. These codes typically set up the initial state of the machine – what units to use, initial locations, absolute or incremental mode, etc.

**Lets Do It!**
1. Double click **Start of Program** in checklist.
2. Dialog box appears with default parameters.
3. Edit text as shown in dialog box, no spaces required. Multiple lines are allowed.

**Note:** Codes can be formatted according to Number Formats, Sort Order, Upper/Lower Case and more…. Adding double quotes to the start and end of line will cause the line to output exactly as is. No number formatting, sorting, or any other conversions will be applied.

“G17G20G40”
“G49G54G80”
“G90G94G98”
Partno Definition

The *partnumber definition* dialog provides control over part number processor when postprocessing files. Part numbers can be read from the input file, input by the operator, or ignored.

Dialog box has two areas:

A. Part Number Options – prompt or no prompt sequence (see below).
   - Take no action involving part number, no prompt.
   - Process part number as APT statement or no action taken.
   - Process part number as APT statement or prompts for value.
   - Will prompt for part number, even with APT statement.

B. Relationship of values associated with part number

Let's Do It!

1. Double click **Partno Definition** in checklist.
2. Identify prefix string input, type **Quickstart**.
3. Identify part number pull down, select **A**.
4. Identify default part number value, edit to **100**.
5. Select **OK** – Red checkmark will appear in checklist.
Test Commands – Program, Part No

On-line testing. This will allow a test of parameter changes without executing the entire program. This test is to verify the program start and part name choices.

This page will setup the input within Text Command dialog box. The output result is displayed on the following page.

Let’s Do It!

1. Select **Execute** from pulldown.
2. Select **Test Commands**.
3. Displays Test Commands dialog box
   Identify input block, type **GOTO/100,200,350**.
4. Verify input data type – **General APT** is default, otherwise change CAM system.
5. Select **Go**.
6. Displays dialog box, due to prompt request.
   Observe number equals prior input.
   Select **Continue**, result on following page.

The result is shown on following page.
Test Commands - Completed
This test is to verify the program start and part name choices.

Reminder:
Start of Program (from page 10) – G17G20G40 – see result below
G49G54G80
G90G94G98

Partno Definition (from page 11) – Prompts for value - see result below
Prefix string = QuickStart
Part register = A
Default value = 100

Note:
1. Output lines are defined as “N” with line numbers.
   Lines N1, N2 and N3 are start program codes.
2. The “%” character is associated with tape format.
Coolant Definition

The *coolant definition* dialog defines the codes for controlling the various coolant operations (mist, flood, thru, air, off).

**Lets Do It!**
1. Double click **Coolant Definition** in checklist.
2. Dialog box appears with default parameters.
3. Identify mist coolant, edit to **M8@MIST** (see special note).
4. Identify mist coolant, edit to **M8@FLOOD** (see special note).
5. Select **OK** – Red checkmark will appear in checklist.
Test Commands – Coolant Definition

On-line testing. This will allow a test of parameter changes without executing the entire program. This test is to verify the @ usage in the coolant definition.

This page will setup the input within Text Command dialog box. The output result is displayed on the following page.

**Lets Do It!**

1. Select **Execute** from pulldown.
2. Select **Test Commands**.
3. Displays Test Commands dialog box
   Identify input block, type **COOLANT/MIST**
   **GOTO/1,2,3**
   **COOLANT/OFF**
4. Verify input data type – **General APT** is default, otherwise change CAM system.
5. Select **Go**.
6. Displays dialog box, due to prompt request.
   Observe number equals prior input.
   Select **Continue**, result on following page.

The result is shown on following page.
Test Commands - Completed
This test is to verify the @ usage in the coolant definition.

Reminder:
Coolant Definition (from page 15) – COOLNT/MIST
GOTO/1,2,3
COOLNT/OFF

Note:
Output lines are defined as “N” with line numbers.
Lines N1, N2 and N3 are start program codes.
Lines N4 and N5 are associated with coolant definition.
Spindle Definition

The *spindle definition* dialog defines the codes used to control the spindle, and the default value for spindle speed.

**Lets Do It!**
1. Double click **Spindle Definition** in checklist.
2. Dialog box appears with default parameters.
3. Identify spindle speed register, **S** (register definition on next page).

Registers contain lines of control parameters.
Lines are associated with a register character (S, G, X, N, F...)
Control parameters – specify number format – decimal output – startup values, etc.

**Letter S Register** – Header definitions on page 18
Register Definition

<table>
<thead>
<tr>
<th>Name</th>
<th>Units Out</th>
<th>Units In</th>
<th>Initial Value</th>
<th>Digits</th>
<th>Dec</th>
<th>Lead</th>
<th>Trail</th>
<th>Sign</th>
<th>Modal</th>
<th>Incr</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>CONST</td>
<td>UNIT</td>
<td>0.000000</td>
<td>5.0</td>
<td>☑️</td>
<td>☑️</td>
<td>☑️</td>
<td>☑️</td>
<td>☑️</td>
<td>☑️</td>
</tr>
</tbody>
</table>

- ☑️ = OFF
- ☑️ = ON

- **Name**: Displays the output name of the register.
- **Units Out**: Displays the output units (INCH, MM, DEG, CONST). This parameter does not affect the output format except in the case of rotary axis registers.
- **Units In**: Displays the input units (INCH, MM, DEG, CONST).
- **Initial Value**: Displays the initial value of the register.
- **Digits**: Specifies the number of digits output before and after the decimal point. For example, 2.4 means output two places to the left and four places to the right of the decimal.
- **Dec**: Indicates whether a decimal point is required.
- **Lead**: Indicates whether there will be leading zero output. Deselect the checkbox to suppress leading zeros.
- **Trail**: Indicates whether there will be trailing zero output. Deselect the checkbox to suppress trailing zeros.
- **Sign**: Indicates whether to include a "+" sign before positive values.
- **Modal**: Indicates whether the register is modal or nonmodal.
- **Incr**: Indicates whether the register is incremental. Deselect the checkbox for absolute.
Tool Change

The *tool change* dialog provides a number of options for defining codes used for tool change operations, and functions that occur when tools are changed. In addition, the operator may provide additional codes before or after the first, and subsequent tool changes.

**Lets Do It!**

1. Double click **Tool Change** in checklist.
2. Dialog box appears with default parameters.
   - Dialog box defines three sections
     - A. Tool Definition
     - B. Data Options
     - C. Before/After Output

Registers contain lines of control parameters.
Lines are associated with a register character (T, G, X, N, F...)
Control parameters – specify number format – decimal output – startup values, etc.

Letter **T** Register – Header definition on page 18
Tool Change – Tool Definition
Customize the tool change output for your postprocessor.

A Portion of Tool Change

Parameters choices.
- Auto tool change function
- Tool number register
- Length compensation register
- Time (mins) to change tools

Let's Do It!
1. Verify or change tool definitions as shown.

<table>
<thead>
<tr>
<th>Auto tool change function:</th>
<th>M6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tool number register:</td>
<td>T</td>
</tr>
<tr>
<td>Length compensation register:</td>
<td>H</td>
</tr>
<tr>
<td>Time (mins) to change tools:</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Default parameters 1
**Tool Change – Data Options**
Customize the tool change output for your postprocessor.

### Portions of Data Options
Parameters choices.

<table>
<thead>
<tr>
<th>Output codes on a separate block</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output codes on the next motion block</td>
</tr>
<tr>
<td>Cancel coolant before tool change</td>
</tr>
<tr>
<td>Cancel spindle before tool change</td>
</tr>
<tr>
<td>Auto coolant after tool change</td>
</tr>
<tr>
<td>Enable tool pre-selection</td>
</tr>
<tr>
<td>Reload 1st tool at the end</td>
</tr>
<tr>
<td>Tool Number Incremental Start:</td>
</tr>
</tbody>
</table>

**Default parameters**

**Lets Do It!**
1. Dialog box appears with default parameters.
2. Identify parameter changes:
Tool Change – Before/After Output
Customize the tool change output for your postprocessor.

C Portion of Before/After Output

Parameters choices.

Lets Do It
1. Dialog box appears with default parameters.
2. Identify first tool and other tools output values.
Test Commands – Program, Part No, Coolant, Spindle, Tool Chg

On-line testing of parameter changes without executing the entire program.

**Lets Do It!**
1. Select **Execute** from pulldown.
2. Select **Test Commands**.
3. Displays Test Commands dialog box
   Edit input block as shown, no spaces required.
4. Verify input data type – **General APT**.
5. Select **Go**.
6. Displays dialog prompt, select **Continue**.
7. Output block displays result.

Output Block

```plaintext
% Quickstart A100. ← Part No Definition
N1 G17 G20 G40
N2 G49 G54 G80
N3 G90 G94 G98
N4 G1 X100. Y200. Z350. T0005 ← go to position x y z, tool register
N5 X110. Y21.5 Z47.375 M8 ← position flood coolant
N6 G28 G91 X Y ← incremental output before tool change
N7 M5 ← spindle off
N8 M9 ← coolant off
N9 T0005 M6 ← tool register, tool change
N10 G90 G55 ← output after tool change
N11 G0 X100. Y120. ← go to position x y plane
N12 G43 Z250. H11 ← cut length, depth, length register
N13 G1 Z200. F1500. ← linear position, feedrate register and value
N14 M30 ← machine rewind
%
```
Feedrate Definition

The *federate definition* dialog controls the codes used for feedrate output, and formatting options for various feedrate types.

**Let's Do It!**

1. Double click **Feedrate Definition** in checklist.
2. Dialog box appears with default parameters.
3. Identify **Feed per minute**: edit to:
   
   Function G94  Format (mm) 5.3  Format (inch) 4.4

4. Identify **Feed per revolution**: edit to:
   
   Function G95  Format (mm) 3.3  Format (inch) 2.4

5. Select **OK** – Red checkmark will appear in checklist.

**Feedrate Note:**
- Enter the feederate only for the related machine specifying INCH and MM output.
- Format values – number to the left are integer places, number to the right are decimal places (2.3 = 00.000).
- The feedrate register value (F) in this dialog box supercedes Register Format definition.
Cutter Compensation

The cutter compensation dialog defines the codes for outputting cutter compensation commands, and the register for the compensation value.

Lets Do It!

1. Double click **Cutter Compensation** in checklist.
2. Dialog box appears with default parameters.
   No modification required.
IMSpost Quickstart

Motion Functions

The *motion functions* dialog has several tabs which control the output of various motion types (lines, arcs/circles, and curves). Support is provided for different output formats for arcs and circles; support for NURBS output is also included. Options for line and arc fitting (including restricting arcs to certain planes) for reduced file size and improved surface finish are also available.

Double click **Motion Functions** from IMSpost Checklist. The following pages will identify motion function uses.

**Motion Function Data Tabs**

The dialog box defines six data tabs.
Motion Functions – Rapid/From
Output codes to control tool movement.

Explanation of Rapid/From Dialog Box

The dialog box has four sections.
- **Rapid Area**
  - A Setting rapid motion
  - B Positioning move
- **From Area**
  - C Handling start point
  - D Point from program

The following page defines sections A B C D
Motion Functions - Rapid Area
Output codes to control tool movement.

Section A and B
Rapid Area

A Setting rapid motion
B Positioning move

A Setting rapid motion options available:

1. **Rapid motion code.** G or M code, that specifies positioning motion.

2. **Rapid block special codes.** Function code, register, or string value using the positioning function mode.

B Positioning move options available:

3. **Break up rapid moves.** Output positioning moves using:
   a. The tool is moving away from the part, motion is made first along the tool axis, retracting from the part, and then to the final position.
   b. The tool moving toward the part, motion is first made to the tool axis, and then along the tool axis toward the final position

4. **Restore feed after rapid move.** Feedrate value output on the next non-rapid move following a rapid motion.
Motion Functions - From Area
Output codes to control tool movement

Section C and D
From Area

C  Handling starting point
D  Point from program

C  Handling starting point. (only one may be selected)
  o  Output the point as a rapid positioning move.
  o  Output the point as a linear/feed move
  o  Save the point as a home position (no output)
  o  Ignore from command.

D  Point from program.
  o  Apply to all from points
  o  Apply to the first from point encountered in the program
**Test Commands** – Load tool

On-line testing. This will allow a test of parameter changes without executing the entire program. This test is to verify the output on a load tool.

This page will setup the input within Text Command dialog box. The output result is displayed on the following page.

**Lets Do It!**

1. Select **Execute** from pulldown.
2. Select **Test Commands**.
3. Displays Test Commands dialog box
   - Identify input block, type **FROM/100,200,300**
   - **RAPID**
   - **GOTO/0.0,0.0,100.0**
   - **FEDRAT/100.0,IPM**
   - **GOTO/10.0,20.0,100.0**

4. Verify input data type – **General APT** is default, otherwise change CAM system.
5. Select **Go**.
6. Displays dialog box, due to prompt request.
   - Observe number equals prior input.
   - Select **Continue**, result on following page.
Test Commands - Completed
This test is to verify the output on a load tool.

Reminder: (from page 30) - FROM/100,200,300
RAPID
GOTO/0.0,0.0,100.0
FEDRAT/100.0,IPM
GOTO/10.0,20.0,100.0

Note:
Output lines are defined as “N” with line numbers.
Lines N1, N2 and N3 are start program codes.
Lines N4 and N5 are associated with the load tool output.

Note: Verify input data type
Important Quickstart change to checklist

The preceding pages will identify motion function parameters. The ability to isolate specific output results will better recognize the use of these parameters.

Remove *Start of Program* codes from the Program Codes category. This will isolate output from the *Test Commands* relating to specific motion parameters.
**Motion Functions - Linear**
Output codes to control tool movement.

**Explanation of Linear Dialog Box**
Incorporates slow down span conditions for the linear motion.

**Quickstart Note:**
The use of Linear by example is beyond the intention of Quickstart and satisfies a more advanced requirement.

<table>
<thead>
<tr>
<th>Count</th>
<th>Linear equation</th>
<th>Feed equation</th>
<th>Speed equation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Variable description:
- L1, L2: Length of move
- X0, Y0, Z0: Last position
- X1, Y1, Z1: Required (current) position
- X2, Y2, Z2: Next position
- A: Angle to next position
- N: Iteration number for each expression
- PF: Programmed feedrate
- CF: Current feedrate
- CS: Current spindle speed
- PS: Programmed spindle speed

Enable/Disable linear break up

Linear motion function code: [G1]  Process motion internally (not with G010 macro)  Help...
Motion Functions - Circle
Output codes to control tool movement.

Explanation of Circle Dialog Box – Radius format
Specify the type of circular interpolation controller will support.
Radius format defines six different circle results.

The Test Command method will identify the results of a different radius format.

Purpose of input values:
- Circle center = X0, Y0
- Circle start = X1, Y0
- Circle end = X0, Y1
- Motion = Counterclockwise

Explanation of input text: 
CIRCLE/0,0,0,0,0,1,0,1,0

XYZ Circle center
XYZ Circle start
XYZ Circle end
IMSpost Quickstart

Motion Functions - Circle - Test Commands

This page will setup the input within Text Commands dialog box. The output result will be the result of radius format displayed on the following page.

**Lets Do It!**

1. Select **Execute** from pulldown.
2. Select **Test Commands**.
3. Displays dialog box, remove all text from Input Block
4. Identify input block, type `GOTO/1,0,0`  
   `CIRCLE/0,0,0,0,0,1,0,1,0`
5. Verify input data type – **General APT** is default, otherwise change CAM system.
6. Select **Close**, input will be used on following pages.

![Diagram](image-url)
Radius Format – Absolute center coordinates

1st Phase qualify radius format

Let's Do It!

1. Double click Motion Functions from IMS Checklist.
2. Select Circle Tab from Motion Parameters dialog box.
5. Select OK, radius format defined.

2nd Phase output result with Test Commands

Let's Do It!

1. Select Execute from pulldown.
2. Select Test Commands, displays previous input.
3. Select Go, see results
**Radius Format** – Distance from start to center

**1st Phase** qualify radius format

**Lets Do It!**
1. Double click **Motion Functions** from IMS Checklist.
2. Select **Circle** Tab from Motion Parameters dialog box.
3. Select **I,J,K: distance from circle start to circle center**.
4. Select **OK**, radius format defined.

![Radius Format Image]

**2nd Phase output result with Test Commands**

**Lets Do It!**
1. Select **Execute** from pulldown.
2. Select **Test Commands**, displays previous input.
3. Select **Go**, see results

![Test Commands Image]
**Radius Format** – Distance from center to start

1st Phase qualify radius format

**Let's Do It!**

1. Double click **Motion Functions** from IMS Checklist.
2. Select **Circle** Tab from Motion Parameters dialog box.
3. Select **I,J,K: distance from circle center to circle start**
4. Select **OK**, radius format defined.

**2nd Phase output result with Test Commands**

**Let's Do It!**

1. Select **Execute** from pulldown.
2. Select **Test Commands**, displays previous input.
3. Select **Go**, see results
IMSpost Quickstart

Radius Format – Unsigned distance from center to start

1st Phase qualify radius format

Lets Do It!

1. Double click Motion Functions from IMS Checklist.
2. Select Circle Tab from Motion Parameters dialog box.
4. Select OK, radius format defined.

2nd Phase output result with Test Commands

Lets Do It!

1. Select Execute from pulldown.
2. Select Test Commands, displays previous input.
3. Select Go, see results
Radius Format – Circle radius
1st Phase qualify radius format

**Let's Do It!**

1. Double click **Motion Functions** from IMS Checklist.
2. Select **Circle** Tab from Motion Parameters dialog box.
3. Select **R : circle radius**.
4. Select **OK**, radius format defined.

---

2nd Phase output result with Test Commands

**Let's Do It!**

1. Select **Execute** from pulldown.
2. Select **Test Commands**, displays previous input.
3. Select **Go**, see results

---

![Test Commands](image)
Radius Format – Linear moves

1st Phase qualify radius format

**Let's Do It!**

1. Double click **Motion Functions** from IMS Checklist.
2. Select **Circle** Tab from Motion Parameters dialog box.
7. Select **Break into linear moves**.
8. Select **OK**, radius format defined.

![Radius Format Diagram]

**2nd Phase output result with Test Commands**

**Let's Do It!**

1. Select **Execute** from pulldown.
2. Select **Test Commands**, displays previous input.
3. Select **Go**, see results

![Test Commands Window]
**IMSpost Quickstart**

**Motion Function - Nurbs**
Output codes to control tool movement.

**Explanation of Nurbs Dialog Box**
Nurbs properties will be displayed.

**Quickstart Note:**
The use of Nurbs by example is beyond the intention of Quickstart and satisfies a more advanced requirement.
Motion Functions – Arc-Fit
Output codes to control tool movement.

Explanation of Arc-Fit Dialog Box
Fit an arc from a series of points

An interpolation of incoming points using a tolerance value defining points used for the arc fit result.

1. Minimum fit an arc points. Four point minimum.
2. Minimum required arc angle.
3. Minimum required arc radius.
4. Maximum required arc radius.
5. Tolerance value for plane fluctuation.
6. Tolerance value for radius fluctuation.
7. Plane selection, single major plane, all three planes.
8. Exact start point, exact end point to fit an arc.
9. On/Off will active/deactivate optimization results for postprocessor.
Motion Functions – Line-Fit
Output codes to control tool movement.

Explanation of Line-Fit Dialog Box
Fit a line from a series of points

1. Tolerance value for points to fit a line. Within tolerance create single continuous line, exceed tolerance will create smaller line length.
2. On/Off will active/deactivate optimization results for postprocessor.

An interpolation of incoming points using a tolerance value defining points used for the line fit result.
Drill Cycles

The *drill cycles* dialog allows the operator to specify required codes for standard drilling cycles. Cycles may also be simulated by GOTO moves by IMSpost if the cycle are not available in the machine control.

**Lets Do It!**

1. Double click **Drill Cycles** in checklist.
2. Dialog box appears with default parameters.

**Drill Cycles Note:**
- A blanked canned cycle will automatically simulate the cycle with linear and rapid motions.
- To simulate all drilling cycles check the Simulated Cycle checkbox.

**Canned cycle codes**

- **Registers for cycle depth, incremental depth, clearance plane, dwell**
End of Programs

The end of program dialog controls what codes are output at the end of every program. These codes typically set up the final state of the machine – turning off coolant, moving the machine to a home position, and the like.

Lets Do It!

1. Double click End of Program in checklist.
2. Enter text as shown in dialog box, no spaces required. Multiple lines are allowed.
Program Codes - Save project file

Customizing the post processor is achieved by editing the Program Codes in the IMSpost Checklist. Currently you have an unsaved project file named `milling_def.prj`. Now is the time to Save As `MyQuickstart.prj`

**IMSpost Checklist status**

The checklist define three categories

- Program Codes – modified parameters
- Controller Format - unchanged parameters
- Machine Format - unchanged parameters

Checkmarks indicates default or assigned parameter values.

IMSpost Checklist

<table>
<thead>
<tr>
<th>Program Codes</th>
<th>Controller Format</th>
<th>Machine Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓ Start of Program</td>
<td>✓ Sequence Numbers</td>
<td>✓ Motion Axes</td>
</tr>
<tr>
<td>✓ Partno Definition</td>
<td>✓ Comment Blocks</td>
<td>✓ Machine Components</td>
</tr>
<tr>
<td>✓ Coolant Definition</td>
<td>✓ Register Format</td>
<td>✓ Reference Points</td>
</tr>
<tr>
<td>✓ Spindle Definition</td>
<td>✓ Function Codes (G/M)</td>
<td></td>
</tr>
<tr>
<td>✓ Tool Change</td>
<td>✓ Sort Output</td>
<td></td>
</tr>
</tbody>
</table>

Completed
Continue Customizing the post processor
The program codes have been defined and the project file named MyQuickstart.prj is saved.

Controller Format

**Reminder:** The checkmarks in the IMSpost Checklist are displayed by default. The checkmarks indicate predefined parameters for the selected machine. Double click checkmark to remove checkmark; a parameter change will redisplay the checkmark, a visual verification.

**Lets Do It!** Remove checkmarks from **Controller Format** Category (not all).
- Sequence Numbers
- Comment Blocks
- Register Format
- Sort Output

<table>
<thead>
<tr>
<th>Default IMSpost Checklist Before</th>
<th>Checkmarks removed After</th>
</tr>
</thead>
</table>

![Checklist Diagram](image-url)
Sequence Numbers

The *sequence number* dialog controls the output of block numbers for the posted program – what code is to be used, the start and increment values for the numbering, whether to use block numbers at all, etc.

**Lets Do It!**

1. Double click **Sequence Numbers** in checklist.
2. Dialog box appears with default parameters.
3. Edit **Start value** to 5.
4. Edit **Increment value** to 10
5. Edit **Maximum value** to 40
6. Select **OK** – Red checkmark will appear in checklist

**Block Number Note:** The program code will start with line N5
Each line will increment by 10; N15, N25...
Maximum line number is 40
Comment Blocks

The *comment blocks* dialog specifies what characters are interpreted by the control as the start of a comment, and various comment formatting options. Comments may also be excluded from the final output using this dialog.

**Lets Do It!**
1. Double click **Comment Blocks** in checklist.
2. Dialog box appears with default parameters.
3. Edit **Prefix string** to `<!>`.
4. Edit **Suffix string** to `!+!`
5. Select **OK** – Red checkmark will appear in checklist

---

**Partial result from NC file**

- Quickstart A100.
- N1 G17 G20 G40
- N2 G49 G54 G80
- N3 G90 G94 G98
- `<!>TOOLEND !+!`
- `<!>ROUGH CUTTING !+!`
- `<!>TOOL DIAMETER:25.4!+!`
Register Format

The register format dialog provides numerous options for formatting the various registers, including decimal format, modality, leading & trailing zeros, and more.

Lets Do It!
1. Double click **Sequence Numbers** in checklist.
2. Dialog box appears with multiple registers, select **Basic**.
3. Edit **Digits** to **4.5**.
4. Edit **Dec** to **ON**.
5. Edit **Trail** to **ON**.
6. Identify **K** register, observe **Test Output**, verify register results.
7. Select **OK** – Red checkmark will appear in checklist.
Sort Output

The *sort output* dialog allows the operator to control the order of function code output on a given block.

**Lets Do It!**
1. Double click **Sort Output** in checklist.
2. Dialog box appears with default parameters.
3. Edit *Sort Order* column, move M before T and G
4. Select **OK** – Red checkmark will appear in checklist
Test Commands – Sequence Blocks, Sort Output

On-line testing of parameter changes without executing the entire program.

** Lets Do It! **
1. Select **Execute** from pulldown.
2. Select **Test Commands**.
3. Displays Test Commands dialog box
   - Edit input block as shown, no spaces required.
   - Add to previous **LOADTL/10**
4. Verify input data type – **General APT**.
5. Select **Go**.
6. Displays dialog prompt, select **Continue**.
7. Output block displays result.

IMSpost Checklist change are Quickstart complete - Save project file.
** File -> Save -> MyQuickstart.prj **
Tape Format

The *Tape Format* dialog allows for specific formatting of the NC output file, such as start and end blocks, file size limits, and end block characters.

**Lets Do It!**

1. Select **Controller** from toolbar ↓.
2. Select **Tape Format** from pulldown.
3. Dialog box appears with default parameters.
4. Select **OK**.

![Tape Format dialog](image)
Postprocess
On-Line Test of codes and parameters.

**Let's Do It!**

1. Select **Execute** from pulldown.
2. Select **Postprocess**.
3. Displays **Postprocessor Execution** dialog box. Identify .apt source input file.
4. **Browse**, navigate to select .apt source file, **Open**.
5. Output file defaults to input file name with .nc extension, or enter another filename.
7. Select **Go**.
Observing and verifying input, output and list filename. Prompts for part name.

Select **Continue**, result is notepad file (see following page).
Postprocessor – Text File Output

The file name is the default product name with the extension .nc.
This output file starts at N1 and finishes with N974
File will be stored in the /outfiles folder

Tape format definition, page 53
Comment Block definition, page 49
Tape format definition, page 53