AZ-post

Configuration Guide

NCData Services

Manufacturing Software Solutions

www.NCDataServices.com

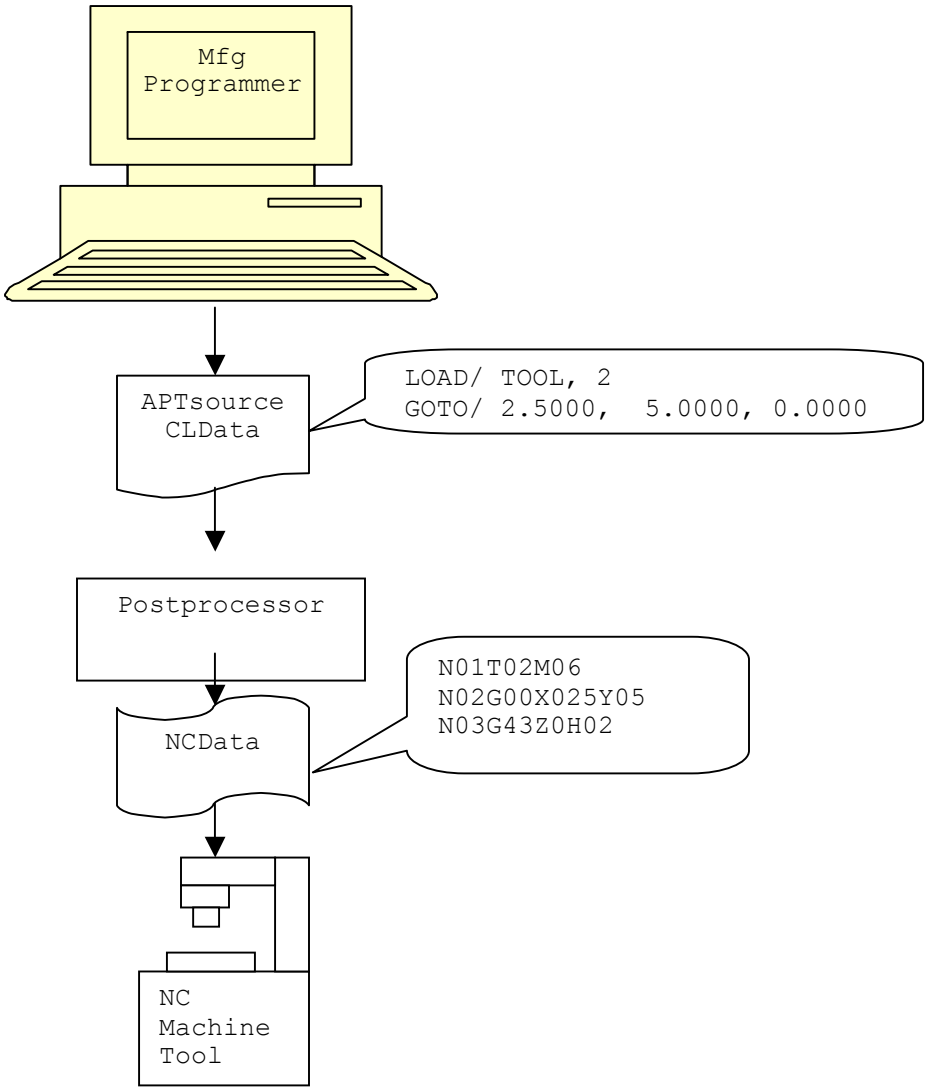
NCData@frontiernet.net

(928) 532-8045

Revised 30-MAR-2011

Introduction

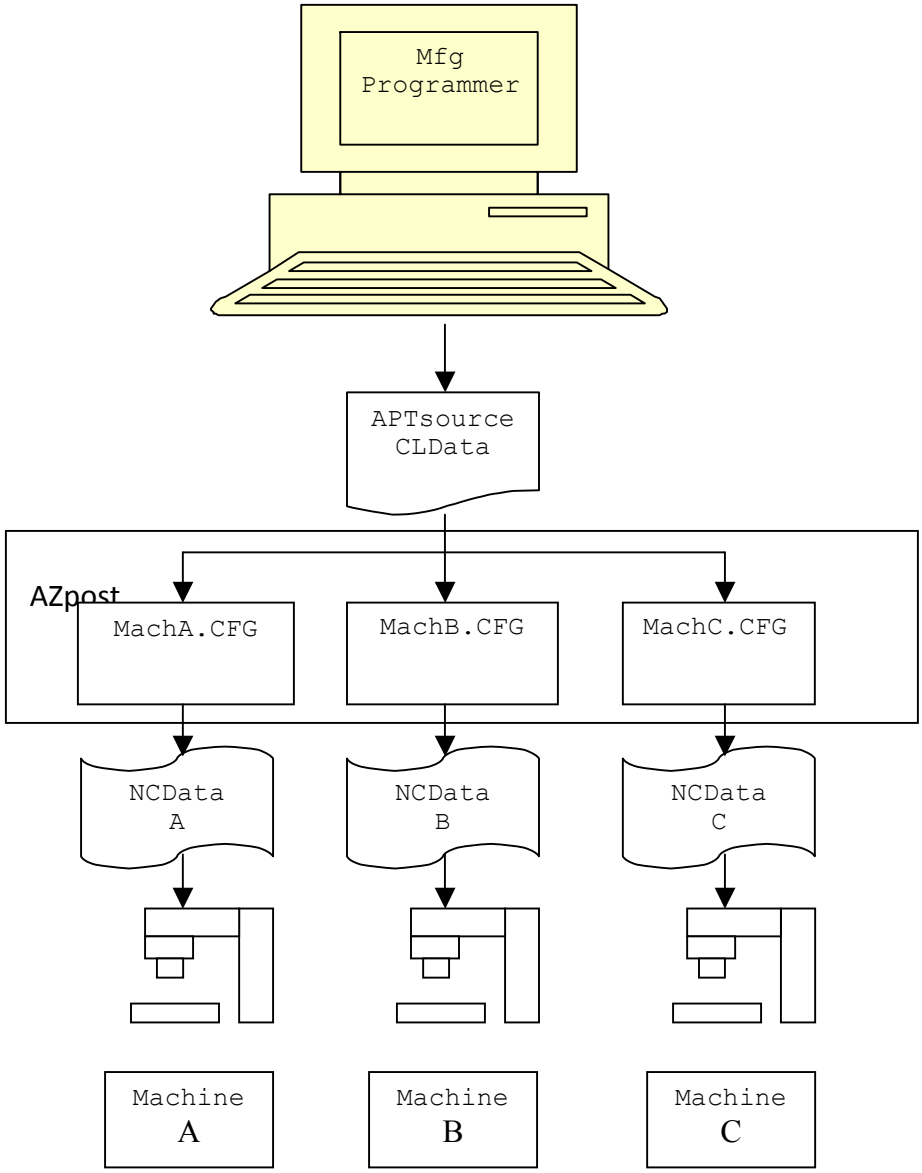
The Challenge



The challenge in postprocessing has always been to make the part program (APT source) as independent of the machine tool as possible. Making the part program independent of the machine tool allows the APT source file to be post-processed for different machine tools (of the same type) with little change to the program. This allows flexibility and efficiency on the shop

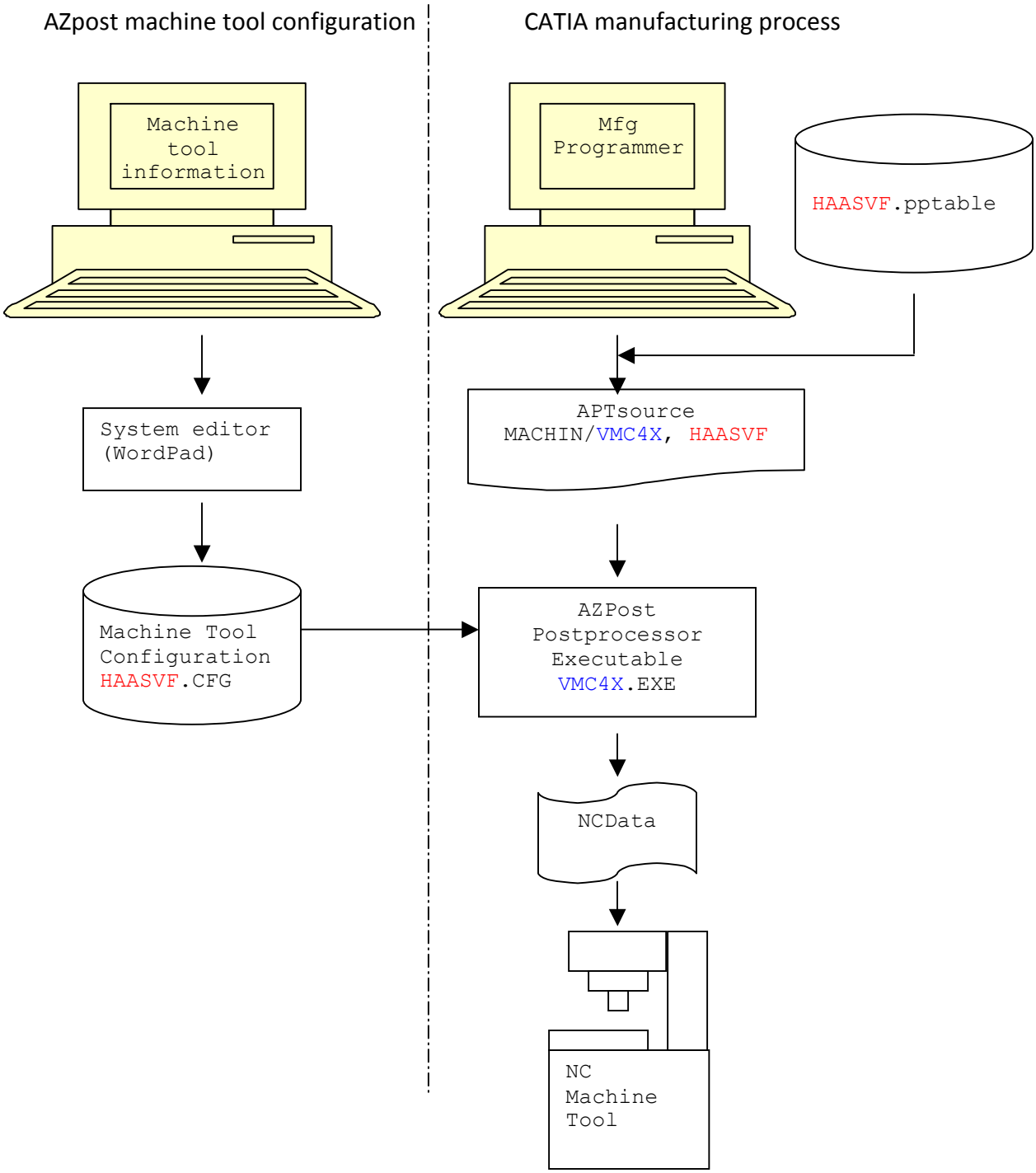
floor when the part is ready for manufacture since re-routing to a different machine tool would be expedited.

The Goal



Remove unique machine tool syntax from the part program (APT source) and put this information in a configuration file whose name can be referenced in the part program. The specific machine tool can then be re-targeted by simply referencing a different configuration file.

The AZpost Concept



Prior to manufacturing process

- Develop a **postprocessor executable** for each customer machining group.
- Create a custom **configuration** file for each specific machine tool.
- Create a CAM **pptable** that targets each specific machine tool configuration using the **MACHIN/** statement (see pptable start macro below).

```
/
*START_NC_INSTRUCTION                    NC_START_MACRO
*START_SEQUENCE
MACHIN/VMC4X,HAASVF,UNITS,INCHES,OUT,INCHES,OFF
PARTNO %MFG_IDENTIFIER %MFG_MACHINE_NAME
*END
*END
/
```

During manufacturing process

- Create a CAM manufacturing process (**CATProcess**) referencing the specific CAM pptable. (Selected from postprocessor words table pull-down in Part Operation).
- Create **APTsource** file using the "Generate NC output" tool.

The APTsource file produced by CATIA will be **targeted** to the correct machine tool configuration and machining group based on the information from the CAM pptable. The part program (APTsource) may now be postprocessed for scheduled machining in the shop or can be held (targeted) in APTsource form until scheduled for machining (see aptsource MACHIN/ below).

```
MACHIN/VMC4X,HAASVF,UNITS,INCHES,OUT,INCHES,OFF
PARTNO 123456 PART OPERATION POSTPROCESSOR TEST
$$ OPERATION NAME : Tool Change.2
$$  Start generation of : Tool Change.2
TLAXIS/ 0.000000, 0.000000, 1.000000
FROM  /    0.00000,    0.00000,    6.00000
$$ TOOLCHANGEBEGINNING
CUTTER/  1.000000,  0.000000,  0.500000,  0.000000, 45.000000,$
         0.000000,  1.500000
TPRINT/T2 - SPOT DRILL 90 DEG
LOAD/TOOL,1,LENGTH,    4.528000
COOLNT/ON
```

Ready for machining

The APTsource can be postprocessed if scheduled for the original targeted machine tool.

< OR >

If scheduled for a different machine tool, the part program can be re-targeted by simply changing the APTsource MACHIN/ reference to the correct machine tool configuration file and then postprocessing.

The Environment

Installation

1. Create directory (folders) structure if initial installation
   (see file & folder structure).
   a. Create \Azpost folder.
   b. Create \Azpost\bin folder.
   c. Create \Azpost\cfg folder.

2. Detach **pp_name.zip** file and unzip into a temporary folder.
3. Move **pp_name.exe** files to \azpost\bin folder.
4. Move **mt_name.cfg** files to \azpost\cfg folder.
5. Move **mt_name.pptable** files to the CATIA pptables folder.

Note: Typical CATIA pptables folder
C:\Program Files\Dassault Systemes\B13\intel_a\startup\Manufacturing\PPTables

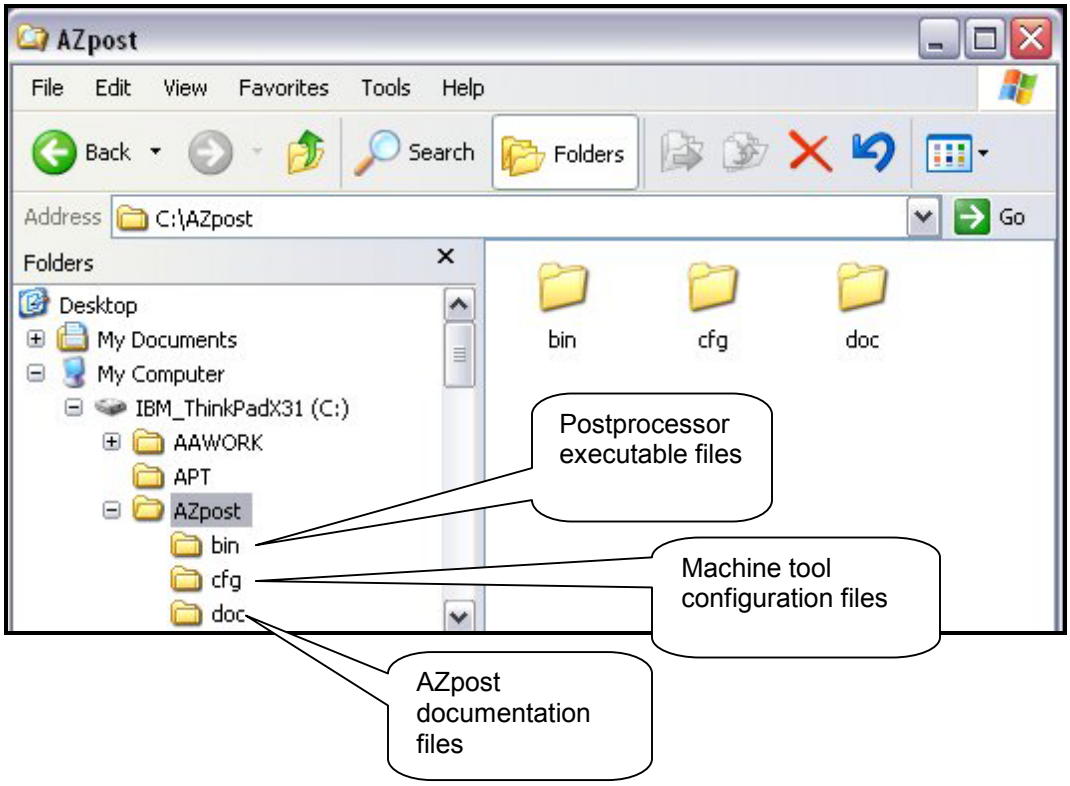
File & Folder Structure

\AZpost

\bin	Contains postprocessor executable files (.exe.)
\cfg	Contains machine tool configuration files (.cfg)

When not using the **Manufacturing Work Bench** a Desk top shortcut can be created for each postprocessor executable in the Azpost \bin folder. Each machine tool should have a configuration file in the Azpost /cfg folder. Each configuration file must have a file extension of .cfg and the file name must not exceed six (6) characters not counting the .cfg extension.

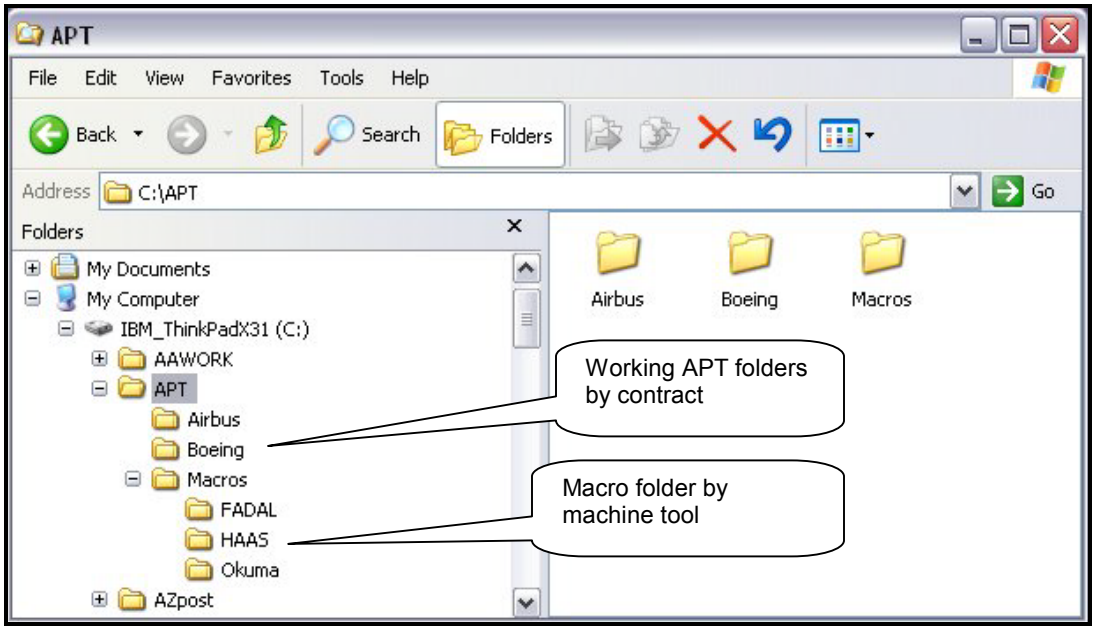
AZpost File Structure



The working AZpost folder contains (3) subfolders as follows:

- The **bin** folder contains all the postprocessor executable files (.**EXE**)
- The **cfg** folder contains all the machine tool configuration files (.**CFG**)
- The **doc** folder contains AZpost documentation files (.**DOC** and .**PDF**)

APT File Structure



The **APT** folder can be sub-divided into working contract folders. These working folders can be where the aptsource, postprocessor listing, and machine code files are created and stored.

A **Macro** folder can be created as a sub-folder for storing APT macros used by the postprocessors. The Macro folder can be further sub-divided by machine tool. This allows different macros with the same name to be created and stored for use with each machine tool. This method can be used to keep the aptsource independent of the machine tool.

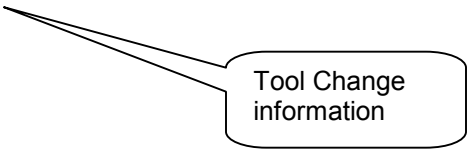
To postprocess an aptsource clfile the aptsource file can be dragged and dropped on the postprocessor desk top icon. Postprocessor output files will be created in the same folder as the aptsource clfile. Typical output files are created as follows:

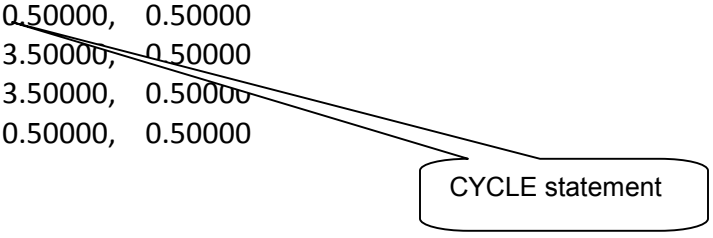
Prg_name.NCD	Machine control readable part program.
Prg_name.NCL	Expanded cross reference listing of machine code and aptsource including and diagnostic information.
Prg_name.NCS	Status file containing diagnostic information and Additional tabulated program information

The APTsource CLFile

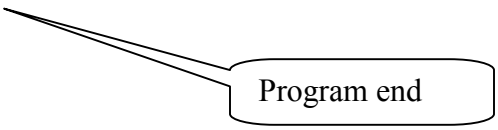
SAMPLE.APTSource CATIA output


```
PPRINT MAK    ROM 8.5" X 4.5" X .5" ALUMINUM STOCK
PPRINT LOCAT  PART ZERO AT:
PPRINT     X = 0   N FROM LEFT EDGE OF STOCK .5"
PPRINT     Y = 0  IN FROM FRONT EDGE OF STOCK .5"
PPRINT     Z = 0  AT BOTTOM OF STOCK

MACHIN/VMC4X,HAASVF,UNITS,INCHES,OUT,INCHES,OFF
PARTNO 1234 PART OPERATION POSTPROCESSOR TEST FANUC GENERIC 3-AXIS

TLAXIS/ 0.000000, 0.000000, 1.000000
FROM /   0.00000,   0.00000,   6.00000

CUTTER/  0.375000,  0.375000,  0.000000,  0.005000,  0.000000,$
      0.000000,  2.000000
TPRINT/ T1 - 3/8 DIA TWIST DRILL LENGTH 4.0
LOAD/TOOL,1,LENGTH,   4.000000
COOLNT/ON

$$ OPERATION NAME : DRILL 4 HOLES
SPINDL/RPM, 1200.0000,CLW
RAPID
GOTO /   0.50000,   0.50000,   4.50000
RAPID
GOTO /   0.50000,   0.50000,   0.70000
CYCLE/DRILL,DEPTH,   0.500000,CLEAR,   0.200000,IPM,   12.000000
GOTO /   0.50000,   0.50000,   0.50000
GOTO /   0.50000,   3.50000,   0.50000
GOTO /   7.50000,   3.50000,   0.50000
GOTO /   7.50000,   0.50000,   0.50000
CYCLE/OFF
RAPID
GOTO /   7.50000,   0.50000,   4.50000

CUTTER/  0.500000,  0.005000,  0.245000,  0.005000,  0.000000,$
      0.000000,  2.000000
```



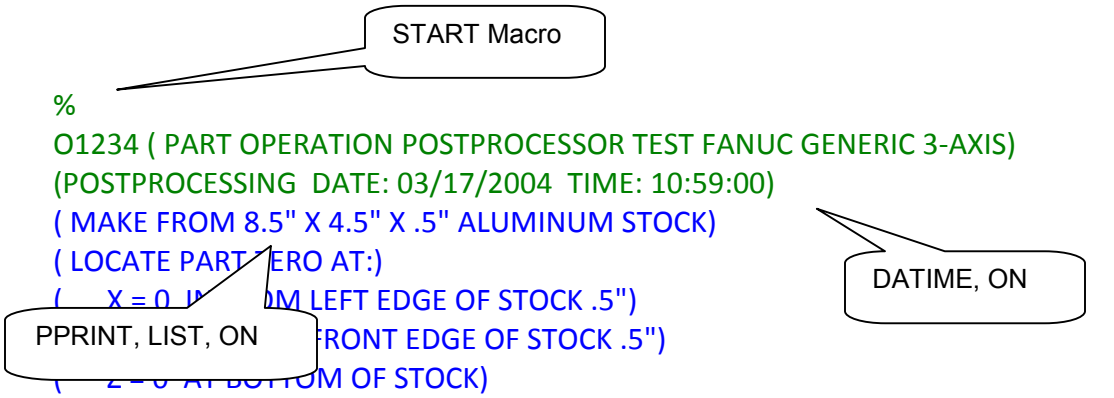


```
TPRINT/ T3 - 1/2 DIA END MILL LENGTH 2.500
LOAD/TOOL,3,LENGTH,    2.500000
COOLNT/ON

$$ OPERATION NAME : PROFILE CONTOUR OUTSIDE
PPRINT PROFILE CONTOUR OUTSIDE
SPINDL/RPM, 1050.0000,CLW
RAPID
GOTO /   1.00000,  -1.00000,   0.50000
FEDRAT/IPM,   30.0000
GOTO /   1.00000,  -1.00000,   0.00000
CUTCOM/LEFT
GOTO /   1.00000,  -0.75000,   0.00000
INTOL /   0.00394
OUTTOL/   0.00000
AUTOPS
INDIRV/   0.00000,   1.00000,   0.00000
TLON,GOFWD/    (CIRCLE/    0.50000,    -0.75000,    0.00000,$
     0.50000),ON,(LINE/    0.50000,    -0.75000,    0.00000,$
                    0.50000,    -0.25000,    0.00000)
FEDRAT/IPM,   5.0000
INDIRV/  -1.00000,   0.00000,   0.00000
TLON,GOFWD/    (CIRCLE/    0.50000,    0.50000,    0.00000,$
     0.75000),ON,(LINE/    0.50000,    0.50000,    0.00000,$
                   -0.25000,    0.50000,    0.00000)
GOTO /  -0.25000,   3.50000,   0.00000
CUTCOM/OFF
GOTO /  -0.30000,   3.55000,   0.00000
GOTO /  -0.30000,   3.55000,   0.50000

COOLNT/OFF
SPINDL/OFF
END
FINI
```

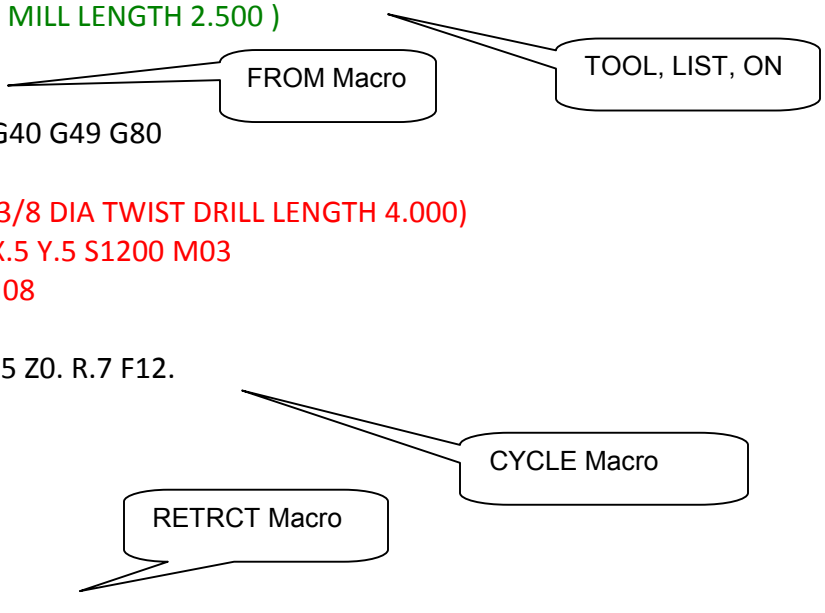


The Machine Code

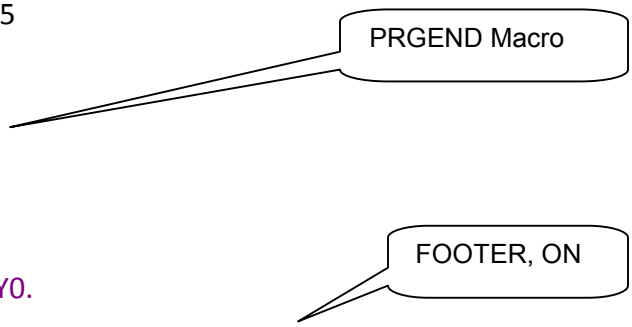
SAMPLE.NCD Machine Code output from postprocessor




```
%
O1234 ( PART OPERATION POSTPROCESSOR TEST FANUC GENERIC 3-AXIS)
(POSTPROCESSING  DATE: 03/17/2004  TIME: 10:59:00)
( MAKE FROM 8.5" X 4.5" X .5" ALUMINUM STOCK)
( LOCATE PART ZERO AT:)
(    X = 0 IN 0M LEFT EDGE OF STOCK .5")
(    Y = 0 IN FRONT EDGE OF STOCK .5")
(    Z = 0 AT BOTTOM OF STOCK)

( T1 - 3/8 DIA TWIST DRILL LENGTH 4.000 )
( T3 - 1/2 DIA END MILL LENGTH 2.500 )

N1 G20
N2 G00 G17 G90 G40 G49 G80

N3 T1 M06  ( T1 - 3/8 DIA TWIST DRILL LENGTH 4.000)
N4 G00 G90 G54 X.5 Y.5 S1200 M03
N5 G43 H3 Z4.5 M08
N6 Z.7
N7 G99 G81 X.5 Y.5 Z0. R.7 F12.
N8 Y3.5
N9 X7.5
N10 Y.5
N11 G80
N12 Z4.5
N13 M05
N14 G91 G28 Z0. M09
N15 G49
N16 M01

N17 T3 M06  ( T3 - 1/2 DIA END MILL LENGTH 2.500)
N18 G05P10000
N19 G00 G90 G54 X1. Y-1. S1050 M03
N20 G43 H3 Z.5 M08
N21 G01 Z0. F30.
N22 G41 D3 Y-.75
N23 G03 X.5 Y-.25 I-.5 J0.
```



```
N24 G02 X-.25 Y.5 I0. J.75 F5.
N25 G01 Y3.5
N26 G40 X-.3 Y3.55
N27 Z.5
N28 M09
N29 M05
N30 G91 G28 Z0.
N31 G05P0
N32 G49
N33 G91 G28 X0. Y0.
N34 M99
(TOTAL MACHINING TIME  =   36716.70)
(PROGRAM SIZE IN BYTES =      604)
%
```



The Configuration File

General Concepts

The configuration file is a plain text file. Most editors can be used to edit the file. The Windows "Word Pad" editor is recommended. The file extension for the configuration file must be .cfg and this extension can be registered with Windows for editing by the Word Pad editor. If other document editors (i.e. Word) are used, fonts and formatting must not be used (ie. turned off) and the file must be saved with extension .CFG.

SECTION ONE – Default Limits and Modes

Section one contains APT predefined postprocessor statements that define default limits and programming modes. The syntax for the statements follow standard APT rules as shown in the following example:

LIMITS/ XAXIS,-40,40, YAXIS,-20,20, ZAXIS,-30,30 $$ AXIS LIMITS

LIMITS/	Major key word defining function (Axis limits) followed by a slash "/"
XAXIS	Minor key word defining feature (X-axis limits) followed by a comma ","
-40,40	Numeric values (with or without decimal point ".") separated by commas.
$$	Text following double dollar is ignored by the postprocessor and is used for commenting statement.

Comments $$ can be used anywhere and all text following the $$ in the statement is ignored by the postprocessor. A comma separator is not used after the last minor word or value in the statement.

Specific syntax is defined by the postprocessor and is documented in the postprocessor The following syntax is basic to all postprocessors:

Note: Major & minor key words must be as shown in upper case. Lower case text must be replaced by appropriate numeric values.

Predefined postprocessor syntaxes

CLEARP/ value

Sets a clearance plane for Z-axis motion based on program coordinates. This value can then be used in the MACROs defined in Section three of the configuration file.

FEDRAT/ MAXUPM, upm

Sets the maximum feed rate in (units per minute) that can be programmed for the machine tool. Any programmed feed rate exceeding this value will create a warning in the listing and status files and the programmed value will be replaced or recomputed using the maximum upm value. See SET/ UNITS for defining units.

INCLUD/ windows_path

This syntax points the postprocessor to the location of the APT Macro folder for this specific machine tool. Example: INCLUDE/C:\APT\MACROS\VMC4X

LIMITS/XAXIS, xmin,xmax, YAXIS, ymin,ymax, ZAXIS, zmin,zmax

Sets the default axis values for postprocessor limit checking. Minor key word for specific axis followed by a minimum and maximum value couplet. This statement also sets the total travel as the difference between the maximum and minimum values for each axis.

MCHTOL/ AAXIS, value

Sets a machine tolerance value when computing A-axis rotation based on tool axis vectors in the aptsource GOTO records. This allows tool axis vectors that are within the tolerance value of the YZ plane to be accepted without errors.

SELECT/ TOOL, AUTO

Sets the selection (ready position) of the next tool for automatic output. See Section two in reference to register T1 for next tool. See Section three for use of T1 in MACROs.

SEQNO/ start-value, INCR, increment-value
OFF

Sets the sequence numbering in the machine code program (i.e. N). Sequence begins with the start value and is incremented using the increment value. Both values should be integer values (non decimal). The sequence numbering can also be turned off using the OFF minor word.

SPINDL/ MAXRPM, rpm

Sets the maximum spindle RPM that can be programmed for the machine tool. Any programmed spindle speed exceeding this value will create a warning in the listing and status files and the programmed value will be replaced or recomputed using the maximum rpm value.

SET/ AAXIS, PLUS, CLW
CCLW

Sets A-axis plus (+) rotation to either CLW or CCLW when looking in the plus X-axis direction. Default ISO standard PLUS = CLW.

SET/ MAXDPM, dpm

Sets maximum degrees per minute for rotary axis, where dpm is a numerical value

SET/ MAXIVT, frn

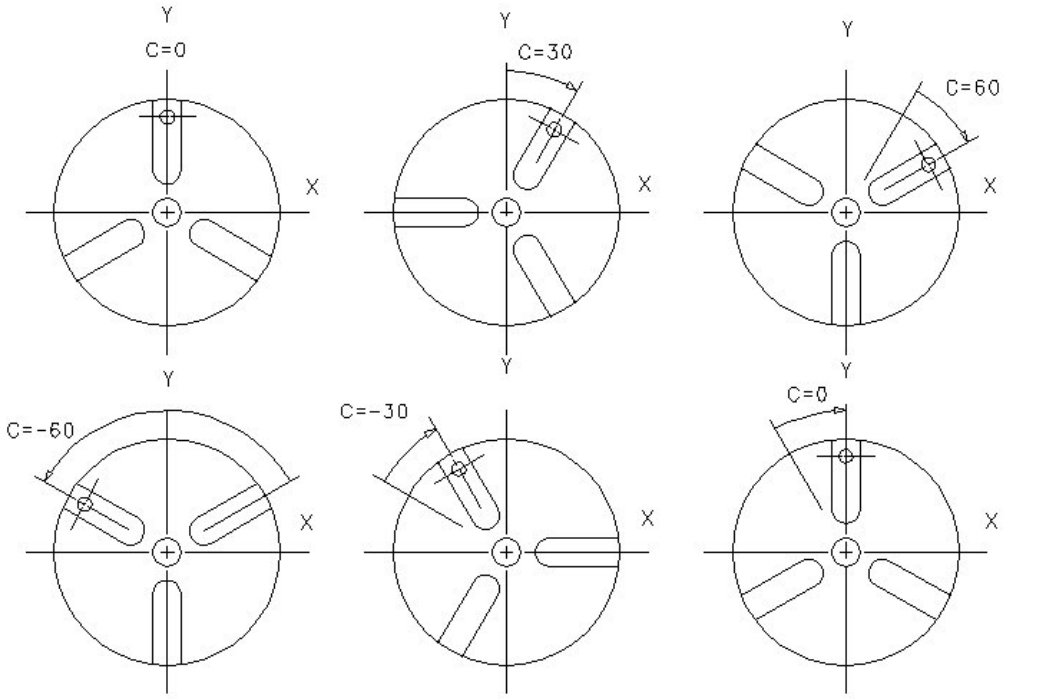
Sets maximum inverse time feed rate number, where frn is a numeric value

SET/ ITUNIT, 1
60

Sets inverse time units, use 1 for minutes and 60 for seconds

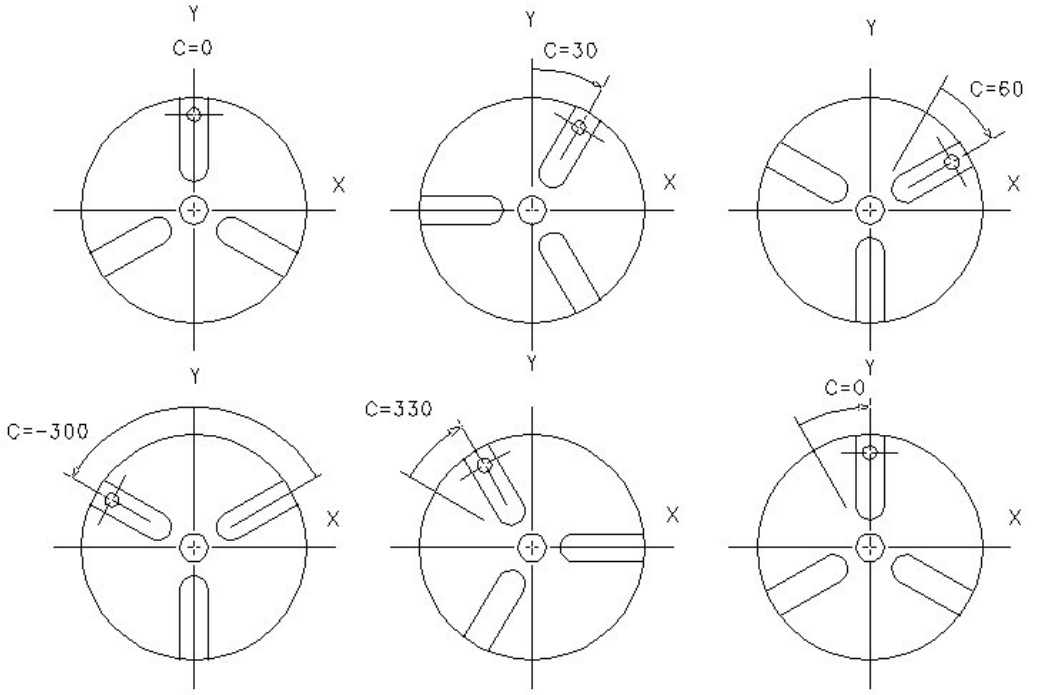
SET/ AAXIS, TYPE, 1
2

Type 1 (default) defines absolute rotary angles signed plus or minus from rotary zero.



ANSI Standard Rotary Positions (Rotating Part +Z Out of Page)
(Sequence Left to Right and Top to Bottom)

Type 2 defines rotary angles absolute (unsigned) with sign determining direction of rotation.



Type 2 (FADAL) Rotary Positions (Rotating Part +Z Out of Page)
(Sequence Left to Right and Top to Bottom)

SET/ CIRCLE, QUADRT, OFFSET, INCR
FULL ABS

Sets the circular interpolation format in the machine code as follows:
QUADRT breaks circles at quadrant points into multiple circle records.
FULL outputs full 360 degree circles in one record.
OFFSET specifies the arc center offset (IJK values) to output as either INCR for incremental or ABS for absolute from part program zero.

SET/ CIRCLE, QUADRT, RADIUS
FULL

Sets the circular interpolation format in the machine code as follows:
QUADRT breaks circles at quadrant points into multiple circle records.
FULL outputs full 360 degree circles in one record.
RADIUS specifies the output of radius (R) instead of arc center offsets.

SET/ CLIPZ, value

Sets a clipping value for the Z-axis. This value is used by the postprocessor to ignore any aptsource GOTO statement that exceeds this Z-axis value. Used to remove automatic CATIA created GOTO statements before and or after a CATIA tool change.

SET/ COMMSG, start_characters, end_characters

Sets the machine code comment message start characters and end characters.
Default SET/COMMSG,(,)

SET/ CYCLE, DEPTH, ABS
INCR

Sets the depth (Z,R) values used in the aptsource CYCLE statement to be either computed in absolute (ABS) from program zero or incremental (INCR) from the cycle reference plane.

SET/ DATIME, ON
OFF

Sets the date and time stamp DATIME when the aptsource was postprocessed to be output in the beginning of the machine code program (as comments). This is can be turned ON or OFF using the corresponding minor key word.

SET/ FOOTER, ON
OFF

Sets the machining time and program size to be output at the end of the machine code program (as comments). This is can be turned ON or OFF using the corresponding minor key word.

SET/ FORMAT, BLANKS, ON
OFF

Sets the machine code format to insert blank spaces before each word address. This function can be turned ON or OFF using the corresponding minor key word.

SET/ FORMAT, PLUS, ON
OFF

Sets the machine code format to insert a plus sign + when dimensional values are positive. This function can be turned ON or OFF using the corresponding minor key word.

SET/ FORMAT, ZEROS, ON
OFF

Sets the machine code format to add a zero after the decimal point on decimal format when values are whole numbers. This function can be turned ON or OFF using the corresponding minor key word.

SET/ LOADPT, x-value, y-value, z-value, a-value, b-value

Sets the machine axis positions at program start in program coordinates. These values are used in computing time for program start, end, and tool changes. This command sets the HOMEX, HOMEY, and HOMEZ variables. These values are overridden by the aptsource FROM statement .

SET/ NCDEXT, string

Sets the file extension characters for the machine code file (default NCD).
Example SET/ NCDEXT, MTF

SET/ ORIGIN, value

Sets the default zero offset (origin) value in the fixture offset register (i.e. G54).

SET/ PIVOTZ, value

Sets distance from the spindle face to the rotary pivot point.

SET/ PPRINT, LIST, ON
OFF

Sets the aptsource PPRINTs before the first cutter statement to be output as comments. This is can be turned ON or OFF using the corresponding minor key word.

SET/ TLCOMP, ON
OFF

Sets the postprocessor tool length compensation ON or OFF.
Default SET/TLCOMP,OFF

SET/ TOOL, MAX, value

Sets the maximum value for the tool that can be programmed in the aptsource using the LOAD/TOOL or LOADTL command. Any programmed TOOL exceeding this value will create a error message in the listing and status files and the postprocessor will output a manual tool change in the machine code file.

SET/ TOOL, LIST, ON
OFF

Sets the TOOL LIST to be output in the beginning of the machine code program (as comments). This is can be turned ON or OFF using the correponding minor key word.

SET/ UNITS, INCHES, OUT, INCHES
MM MM

Sets the program UNITS of the input aptsource and the machine code OUTput units. If the UNITS and OUT are different the postprocessor will convert the input UNITS to the OUT put units for all dimensional data. The UNITS minor key word is followed by its minor key word INCHES or MM. The OUT minor key word is followed by its minor key word INCHES or MM.

SECTION TWO – Register Definitions and Values

Section two contains register definitions similar to machine tool control CNC register groups. The syntax for these statements is shown and is described in the following examples:

```
            Parameters
   1--2—---3----4--5—- 6 thru 48        Comment

REGDEF/ G1, G, 260, 260, T,  0,1,2,3,4,33,34   $$ INTERPOLATION
```

REGDEF/ Required for all register definitions (must be upper case followed by a '/')

1st Parameter Predefined register name (not definable by user).

2nd Parameter Word address user defined per machine tool code. Can be up to six characters

3rd Parameter Inch format specified in 3 digits as follows:

1st Digit	Number of digits left of decimal point
2nd Digit	Format as follows: 1 = Decimal point format output 2 = Leading zeros format output 4 = Trailing zeros format output 6 = Leading and trailing zeros format output
3rd Digit	Number of digits right of decimal point

4th Parameter Metric format specified as 3 digits (same format as inch).

5th Parameter Modality of register (T or F) for true or false.

6 thru 48 All parameters following the modality parameter allow users to define numeric values for fixed features of the register. See values for fixed register features.

$$ Text following double dollar is used to comment function of the predefined postprocessor register.

All parameters are separated by a comma. A comma separator is not used after the last value in the statement.

Registers are predefined in the postprocessor (2nd parameter) and can not be added to or deleted from the configuration file. Word addresses, formats, modality, and numeric values of the predefined register can be changed.

The order that the registers are output in the machine code block is determined by the order that the registers are defined in section two of the configuration file. REGDEF records can be moved using the cut and paste method found in most text editors.
Section two must end with an EOT (End of Table) statement.

Example 1:

REGDEF/ G1, G, 260, 260, T, 0,1,2,3,4,33,34 $$ INTERPOLATION

G1 Predefined register for machine motion interpolation.
G Word address machine code for motion interpolation.
260 Inch format for machine data. Two (2) digits to left of decimal. Zero (0) digits to right of decimal. Format 6 – No decimal point, leading and trailing zeros output.
260 Metric format same as inch
T Modality true. If value same as current then don't output.
0 Value for rapid motion
1 Value for linear motion
2 Value for circular motion CLW
3 Value for circular motion CCLW
4 Value for machine dwell
32 Value for single thread cutting motion
33 Value for multiple thread cutting motion

Example 2:

REGDEF/ X1, X, 214, 413, T, $$ LINEAR ABSCISSA AXIS

X1 Predefined register for machine X-axis (Abscissa)
X Word address machine code for machine X-axis
214 Inch format for machine data. Two (2) digits to left of decimal point. Four (4) digits to right of decimal point. Format (1) = decimal point output.
413 Metric format for machine data. Three (3) digits to left of decimal point. Three (3) deigite to right of decimal

point. Format (1) = decimal point output.

T Modality true. If value same as current then don't output.

Note: No fixed features values are defined for dimensional registers

G Register features:

G0	General	Undefined	
G1	Interpolation	RAPID, LINEAR, CIRCULAR CLW, CIRCULAR CCLW, DWELL, SINGLE THREAD, THREAD CYCLE	
G2	Plane Select	XYPLAN, ZXPLAN, YZPLAN	
G3	Mode Select	ABSOLUTE, INCEMENTAL	
G4	Spindle mode SUM, RPM		
G5	Feed rate mode	INVERSE_TIME, UPM, UPR	
G6	Units mode	INCHES, MM	
G7	Cutter rad comp	OFF, LEFT, RIGHT	
G8	Tool length comp	ADD, SUBTRACT, CANCEL	
G9	Cycle type	OFF	(G80)
		BRKCHP (HS PECK)	(G73)
		TAP (LEFT HAND)	(G74)
		TAP (HEAD)	(G75)
		FBORE (FINE BORE)	(G76)
		DRILL/CSINK	(G81)
		CBORE/ FACE	(G82)
		DEEP	(G83)
		TAP (RIGHT HAND)	(G84)
		BORE/ REAM	(G85)
		BORE6 (SPINDLE OFF)	(G86)
		BORE7 (BACK)	(G87)
		BORE8 (DWELL,SPINDLE OFF)	(G88)
		BORE9 (DWELL)	(G89)
G10	Fixture offset	LOCAL, MACHINE, 1, 2, 3, 4, 5, 6, DATUM	

M Register features:

M0	General	Undefined
M1	Tool Change	LOAD/TOOL
M2	Coolant type	OFF, ON/FLOOD, MIST, THRU, TAP, AIR, AUX
M3	Spindle mode	OFF, CLW, CCLW, LOCK, CLAMP, UNCLAMP, LOW, HIGH
M4	Subroutine mode	MAIN_END, RETRCT, CLEAR
M5	Program control	STOP, OPSTOP, END, REWIND

SECTION THREE – Program MACRO Sequences

Section three contains Macro definitions that contain machine code sequences constructed using the predefined registers and their values enclosed in parentheses. Macros are defined starting with the Major word MACRO followed by a slash followed by the predefined macro name. The body of the macro contains records containing the predefined registers and their values enclosed in parentheses. The end of the macro is identified with the major word END. An EOF record must be the last record in section three after the last macro. The syntax for these records is shown and is described in the following examples:

Example:

```
$$**** RETRACT MACRO SEQUENCE (Tool Change) *****
MACRO/RETRCT
M2(9)$                          $$ Coolant Off
G0(28) G3(91) YY(5.25) ZZ(CLEARP)$      $$ Machine Zero Y & Z-axis
END
```

The $$ can be used for documentation in the macro as shown and is ignored by the postprocessor. The macro starts with the major word MACRO followed by a "/" and the predefined name of the macro. Each predefined macro will be discussed below. The macro must end with the major word END. The body of the macro can contain up to twelve (12) records that represent the machine blocks to be output. Each record must end with a single $ representing the end of block character. Each record can contain registers followed by a pair of parentheses () or text enclosed in a pair of double quotes "". Parentheses and quotes can not be mixed in a record. Key words representing postprocessor values can also be used inside the register parentheses as follows:

() - Empty parentheses specifies that the current value of the register for the specific macro will be output.

(9) – A numeric value specifies that the register is to be output with the given numeric value according to the REGDEF format in section two. The value can be specified with or without a decimal point.

(CLEARP) – A single key word that represents a postprocessor numeric value can be specified.

Each register and parentheses pair must be separated from the next by a space character. The end of block ($ character) is not separated by a space and can be then followed by a comment beginning with a $$.

Key words are predefined by the postprocessor and the number of them can vary based on the specific postprocessor and postprocessor type. The specific key words are defined in the postprocessor documentation, but the basic key words are defined as follows:

Postprocessor Key Words

SEQN Numeric value of sequence number for next (current) block

CURTL Numeric value of the current tool

NEXTL Numeric value of the next tool

GAGEZ Numeric value of the tool gauge length (set length)

PIVOTZ Numeric value of the tool gauge point to tool pivot point or distance from machine table to A-axis center-line.

CLEARP Numeric value of Z-axis at clearance from part zero

CURX Numeric value of the X-axis current position (GOTO)

CURY Numeric value of the Y-axis current position (GOTO)

CURZ Numeric value of the Z-axis current position (GOTO)

CURA Numeric value of the A-axis current position (GOTO)

CURB Numeric value of the B-axis current position (GOTO)

CURC Numeric value of the C-axis current position (GOTO)

HOMEZ Numeric value of Z-axis at its home position (FROM)

HOMEX Numeric value of X-axis at its home position (FROM)

HOMEY Numeric value of Y-axis at its home position (FROM)

HOMEZ Numeric value of Z-axis at its home position (FROM)

FEED Numeric value of current feed rate (upm or upr)

RPM Numeric value of current spindle speed in RPM

FIXTUR Numeric value of current fixture code

COOLNT Numeric value of current coolant code

Postprocessor Key Syntaxes

The following syntaxes can be used as a macro record to control specific postprocessor output. These syntaxes are used without an ending $.

SEQNO/ON Used in macro to turn sequence numbers ON

SEQNO/OFF Used in macro to turn sequence numbers OFF

OPSKIP/ON Used in macro to turn opskip (block delete) ON

OPSKIP/OFF Used in macro to turn opskip (block delete) OFF

TPRINTText value of current tool description (from TPRINT)
If used followed by a > character it will be applied
to the next block as a machine code comment.

PROGID Text value (numeric) of program ID (from PARTNO)

PARTNO Text value of program description (from PARTNO)

Inserting “text” in the output

The use of double quotes allows text to be specified for output in a block. This text is not formatted or checked in any way for correctness. This can be used for outputting special characters including blank records as follows:

“ “ $$ OUTPUTS A BLANK MACHINE BLOCK
“%” $$ OUTPUTS THE REWIND STOP CHARACTER %

Conditional output of blocks

Blocks can conditionally be output based on a predefined key logical variable by using the **IF[…]** syntax followed by the block to conditionally output. The key logical variable is predefined in the specific AZpost postprocessor and is referenced inside the **IF** brackets. If the variable is **TRUE** at the time the macro is called the block will be output. If the variable is **FALSE** at the time the macro is called the block will not be output (skipped).

Example:

IF[XCZ]G0(13.1)$ $$ TURN OFF POLAR INTERPOLATION

Note that spaces are not allowed in the **IF[…]** syntax and that the key logical variables that are referenced in the brackets most be predefined in the specific AZpost executable.

Some examples of key logical variables are:

XYZ - Standard Cartesian coordinate machining mode

XCZ - Virtual Y-axis Cartesian (C-axis) machining mode

POLAR - Polar coordinate input machining mode

Predefined Macros

MACRO/ START
This macro is output at the immediate beginning of the machine code program. It can be used to output any special characters and any program identification information using the PROGID and PARTNO key words.

Example:

Aptsource:

```
PARTNO 1234 PART OPERATION POSTPROCESSOR TEST FANUC GENERIC 3-AXIS
```

Macro Specification:

```
MACRO/START
"%"
O(PROGID) PARTNO
END
```

Machine Code Output:

```
%
O1234 (PART OPERATION POSTPROCESSOR TEST FANUC GENERIC 3-AXIS)
```

MACRO/ FROM

This macro is output at the beginning of the machine code program by the aptsource FROM statement. This macro can be used to output any initial registers and values. The FROM statement sets the HOMEX, HOMEY and HOMEZ values.

Example:

Aptsource:

```
FROM  /   0.00000,   0.00000,   6.00000
```

Macro Specification:

```
MACRO/FROM
G6()$                              $$ INITIALIZE UNITS
G1(0) G2(17) G3(90) G7(40) G8(49) G9(80)$   $$ INITIALIZE G CODES
END
```

Machine Code Output:

```
N1 G20
N2 G00 G17 G90 G40 G49 G80
```

MACRO/ FROM4

Same as MACRO/FROM but is called by the AZpost postprocessor when the program is a MULTAX tool path and has a (6) parameter FROM record.

MACRO/ RETRCT

This macro is output before all tool changes except the first. This macro can be used to cancel any modes, turn of any functions and output any positioning moves prior to the tool change. The aptsource
LOAD/ TOOL statement activates this macro.

Example:

Aptsource:

```
LOAD/TOOL,1,LENGTH,    4.000000
```

Macro Specification:

```
MACRO/RETRCT
M3(5)$                  $$ SPINDLE OFF
G3(91) G8(28) ZZ(0.) M2(9)$   $$ Z-AXIS TO MACHINE ZERO, COOLNT OFF
G8(49)$                 $$ CANCEL TOOL LENGTH
M5(1)$                  $$ MACHINE OPTIONAL STOP
END
```

Machine Code Output:

```
N13 M05
N14 G91 G28 Z0. M09
N15 G49
N16 M01
```

MACRO/ TLCHG1

This macro is output for the first tool change. This macro is used to output any codes for the tool change block including tool description text. The aptsource LOAD/ TOOL statement activates this macro.

Example:

Aptsource:

```
TPRINT/ T1 - 3/8 DIA TWIST DRILL LENGTH 4.0
LOAD/TOOL,1,LENGTH,    4.000000
```

Macro Specification:

```
MACRO/TLCHG1
TPRINT>              $$ Add TPRINT text to next block as comment
TT() M1(6)$           $$ Select and load the requested tool
END
```

Machine Code Output:

```
N17 T3 M06  ( T3 - 1/2 DIA END MILL LENGTH 2.500)
```

MACRO/ TLCHG

This macro is output for all tool changes except the first. This macro is used to output any codes for the tool change block including tool description text. The aptsource LOAD/ TOOL statement activates this macro.

Example:

Aptsource:

```
TPRINT/ T3 - 1/2 DIA END MILL LENGTH 2.500
LOAD/TOOL,3,LENGTH,    2.500000
```

Macro Specification:

```
MACRO/TLCHG
TPRINT>              $$ Add TPRINT text to next block as comment
TT() M1(6)$           $$ Select and load the requested tool
END
```

Machine Code Output:

```
N3 T1 M06  ( T1 - 3/8 DIA TWIST DRILL LENGTH 4.000)
```

MACRO/ GOHOME

This macro is output to position the machine tool axes to the machine home position. This macro can be used to cancel any modes, turn off any functions and output the positioning moves. The aptsource GOHOME
statement activates this macro.

Example:

Aptsource:

```
GOHOME
```

Macro Specification:

```
MACRO/GOHOME
M3(5)$                  $$ SPINDLE OFF
G3(91) G8(28) ZZ(0.) M2(9)$    $$ Z-AXIS TO MACHINE ZERO, COOLNT OFF
G3(91) G8(28) XX(0.) YY(0.)$   $$ X & Y AXIS TO MACHINE ZERO
G8(49)$                 $$ CANCEL TOOL LENGTH
M5(0)$                  $$ MACHINE STOP
END
```

Machine Code Output:

```
N13 M05
N14 G91 G28 Z0. M09
N14 G91 G28 X0. Y0.
N15 G49
N16 M00
```

MACRO/ RESTAR

This macro is output after all tool changes. This macro can be used to restart any modes, turn on any functions and output any positioning moves after the tool change. The first aptsource GOTO statement activates this macro. All values defined in aptsource FEDRAT, SPINDL and ORIGIN statements after LOAD/TOOL are held until this GOTO statement.

Example:

Aptsource:

```
CUTTER/  0.500000,  0.005000,  0.245000,  0.005000,  0.000000,$
       0.000000,  2.000000
TPRINT/ T3 - 1/2 DIA END MILL LENGTH 2.500
LOAD/TOOL,3,LENGTH,    2.500000
COOLNT/ON
SPINDL/RPM, 1050.0000,CLW
RAPID
GOTO  /    1.00000,   -1.00000,    0.50000
```

Macro Specification:

```
MACRO/RESTAR
"G05P10000"                          $$ High speed machining mode
G1(0) G3(90) G10() XX() YY() SS() M3()$  $$ First XY move after tlchg
G8(43) ZZ() HH() M2()$                $$ First Z-axis
END
```

Machine Code Output:

```
N18 G05P10000
N19 G00 G90 G54 X1. Y-1. S1050 M03
N20 G43 H3 Z.5 M08
```

MACRO/ RESTAR4

Same as MACRO/RESTAR but is called by the AZpost postprocessor when the program is a MULTAX tool path and has (6) parameter GOTO records.

MACRO/ PRGEND

This macro is output to end the program, position the machine tool axes to the machine home position. This macro can be used to cancel any modes, turn off any functions and output the positioning moves. The aptsource END statement activates this macro.

Example:

Aptsource:

```
END
```

Macro Specification:

```
MACRO/PRGEND
M2(9)$                      $$ COOLNT OFF
M3(5)$                      $$ SPINDLE OFF
G1(0) G3(91) G8(28) ZZ(0.)$       $$ Z AXIS TO MACHINE ZERO
"G05P0"                      $$ CANCEL High Speed Machining
G8(49)$                     $$ CANCEL TOOL LENGTH
G1(0) G3(91) G8(28) XX(0.) YY(0.)$ $$ X & Y AXIS TO MACHINE ZERO
M5()$                      $$ END PROGRAM
END
```

Machine Code Output:

```
N28 M09
N29 M05
N30 G91 G28 Z0.
N31 G05P0
N32 G49
N33 G91 G28 X0. Y0.
N34 M99
```

MACRO/ PRGEND4

Same as MACRO/PRGEND but is called by the AZpost postprocessor when the program is a MULTAX tool path and has (6) parameter GOTO records.

SECTION FOUR – Additional Syntax Definitions

Section four contains syntax definitions that contain machine code sequences constructed using the predefined registers and their values enclosed in parentheses. Syntaxes are defined starting with the Major word SYNDEF followed by a slash followed by the APT source syntax. The body of the macro contains records containing the predefined registers and their values enclosed in parentheses. An EOF record must be the last record in section four. The syntax for these records is shown and is described in the following examples:

Example:

```
SYNDEF/DEFSUB,&1                    $$ DEFINE APT START OF SUBROUTINE SYNTAX
O1(&1)$                             $$ MACHINE CODE SUBROUTINE START BLOCK
SYNDEF/ENDSUB                       $$ DEFINE APT END OF SUBROUTINE SYNTAX
M0(99)$                             $$ MACHINE CODE SUBROUTINE END BLOCK
SYNDEF/CALSUB,&1,TIMES,&2           $$ DEFINE APT CALL OF SUBROUTINE SYNTAX
G0(98)  P2(&1)  L1(&2)$             $$ MACHINE CODE SUBROUTINE CALL BLOCK
```

The $$ can be used for documentation in the macro as shown and is ignored by the postprocessor. The macro starts with the major word MACRO followed by a "/" and the predefined name of the macro. Each predefined macro will be discussed below. The macro must end with the major word END. The body of the macro can contain up to twelve (12) records that represent the machine blocks to be output. Each record must end with a single $ representing the end of block character. Each record can contain registers followed by a pair of parentheses () or text enclosed in a pair of double quotes "". Parentheses and quotes can not be mixed in a record. Key words representing postprocessor values can also be used inside the register parentheses as follows:

() - Empty parentheses specifies that the current value of the register for the specific macro will be output.

(9) – A numeric value specifies that the register is to be output with the given numeric value according to the REGDEF format in section two. The value can be specified with or without a decimal point.

(CLEARP) – A single key word that represents a postprocessor numeric value can be specified.

Each register and parentheses pair must be separated from the next by a space character. The end of block ($ character) is not separated by a space and can be then followed by a comment beginning with a $$.

Key words are predefined by the postprocessor and the number of them can vary based on the specific postprocessor and postprocessor type. The specific key words are defined in the postprocessor documentation, but the basic key words are defined as follows:

Postprocessor Key Words

SEQN Numeric value of sequence number for next (current) block

CURTL Numeric value of the current tool

NEXTL Numeric value of the next tool

GAGEZ Numeric value of the tool gauge length (set length)

PIVOTZ Numeric value of the tool gauge point to tool pivot point or distance from machine table to A-axis center-line.

CLEARP Numeric value of Z-axis at clearance from part zero

CURX Numeric value of the X-axis current position (GOTO)

CURY Numeric value of the Y-axis current position (GOTO)

CURZ Numeric value of the Z-axis current position (GOTO)

CURA Numeric value of the A-axis current position (GOTO)

CURB Numeric value of the B-axis current position (GOTO)

CURC Numeric value of the C-axis current position (GOTO)

HOMEZ Numeric value of Z-axis at its home position (FROM)

HOMEX Numeric value of X-axis at its home position (FROM)

HOMEY Numeric value of Y-axis at its home position (FROM)

HOMEZ Numeric value of Z-axis at its home position (FROM)

FEED Numeric value of current feed rate (upm or upr)

RPM Numeric value of current spindle speed in RPM

FIXTUR Numeric value of current fixture code

COOLNT Numeric value of current coolant code

Postprocessor Key Syntaxes

The following syntaxes can be used as a macro record to control specific postprocessor output. These syntaxes are used without an ending $.

SEQNO/ON Used in macro to turn sequence numbers ON

SEQNO/OFF Used in macro to turn sequence numbers OFF

OPSKIP/ON Used in macro to turn opskip (block delete) ON

OPSKIP/OFF Used in macro to turn opskip (block delete) OFF

TPRINTText value of current tool description (from TPRINT)
If used followed by a > character it will be applied
to the next block as a machine code comment.

PROGID Text value (numeric) of program ID (from PARTNO)

PARTNO Text value of program description (from PARTNO)

Inserting “text” in the output

The use of double quotes allows text to be specified for output in a block. This text is not formatted or checked in any way for correctness. This can be used for outputting special characters including blank records as follows:

“ “ $$ OUTPUTS A BLANK MACHINE BLOCK
“%” $$ OUTPUTS THE REWIND STOP CHARACTER %

Appendix A

Sample Configuration File

```
$$*************************************************************************
$$   HAASVF.CFG  -  Haas VF 4-Axis Vertical Machining Center / Haas CNC
$$   (C) Copyright NCData Services 2004, 2005   www.NCDataServices.com
$$   LAST CHANGED 17-JAN-2005 (sample)
$$*************************************************************************
$$     SECTION ONE  -  Default Limits and Modes
$$*************************************************************************
LIMITS/XAXIS,-40,40,YAXIS,-20,20,ZAXIS,-30,30  $$ AXIS TRAVEL LIMITS
SPINDL/MAXRPM, 20000                           $$ MAXIMUM SPINDLE RPM
FEDRAT/MAXUPM, 200                             $$ MAXIMUM UNITS FEED RATE
SEQNO/1,INCR,1                                 $$ SEQUENCE NUMBER
CLEARP/10.0                                    $$ CLEARANCE PLANE
MCHTOL/AAXIS,.010                              $$ ROTARY AXIS TOLERANCE
$$SELECT/TOOL,AUTO                             $$ AUTOMATIC PRE-SELECT NEXT TOOL
$$SET/NCDEXT,MTF                               $$ MACHINE CODE FILE EXTENSION 'MTF'
SET/COMMSG,(,)                                 $$ COMMENT MESSAGE START,STOP
SET/TOOL,MAX,18                                $$ MAXIMUM TOOL NUMBER
$$SET/TLCOMP,ON                                $$ TURN ON TOOL LENGTH COMPENSATION
SET/FORMAT,BLANKS,ON                           $$ OUTPUT BLANKS BEFORE WORD ADDRESS
SET/FORMAT,PLUS,ON                             $$ TURN ON PLUS SIGN OUTPUT
SET/ORIGIN,54                                  $$ DEFAULT FIXTURE OFFSET
SET/UNITS,INCHES,OUT,INCHES                    $$ SET UNITS INPUT & OUTPUT TO INCHES
SET/LOADPT,0,20,30,0                           $$ SET MACHINE STARTUP POSITION
SET/CLIPZ,98                                   $$ IGNORE CATIA RAPID GOTO IF Z>98
SET/CIRCLE,FULL,OFFSET,INCR                    $$ SET CIRCLES FULL,OFFSETS INCREMENTAL
SET/CYCLE,DEPTH,ABS                            $$ SET CYCLE DEPTHS ABSOLUTE
SET/TOOL,LIST,ON                               $$ TOOL LISTING IN HEADER
SET/PPRINT,LIST,ON                             $$ DISPLAY PPRINTS BEFORE 1ST CUTTER
SET/DATIME,ON                                  $$ DISPLAYS DATE AND TIME STAMP
SET/FOOTER,ON                                  $$ DISPLAYS TOTAL MACHINING TIME FOOTER
$$*************************************************************************
$$     SECTION TWO  -  Register Definitions, Formats, Sequence and values
$$*************************************************************************
$$           WORD   FORMAT
$$ REGISTER, ADDR,INCH,METRIC,MODAL,VALUES
$$-------------------------------------------------------------------------------
REGDEF/  N1,  N,  540,  540,  F                  $$ SEQUENCE NUMBER
REGDEF/  G0,  G,  260,  260,  F, 98,99           $$ GENERAL PRE-FUNCTION
REGDEF/  G1,  G,  260,  260,  T, 0,1,2,3,4,33,34 $$ INTERPOLATION MODE SELECT
REGDEF/  G2,  G,  260,  260,  T, 17,18,19        $$ INTERPOLATION PLANE SELECT
REGDEF/  G3,  G,  260,  260,  T, 90,91           $$ POSITIONING MODE SELECT
REGDEF/  G4,  G,  260,  260,  T, 96,97           $$ SPINDLE MODE SELECT (SUM,RPM)
REGDEF/  G5,  G,  260,  260,  T, 93,94,95        $$ FEED RATE MODE SELECT
REGDEF/  G6,  G,  260,  260,  T, 20,21      $$ UNITS SELECT (INCHES,MM)
REGDEF/  G7,  G,  260,  260,  F, 40,41,42   $$ CUTTER RADIUS COMPENSATION
REGDEF/  G8,  G,  260,  260,  F, 43,44,49   $$ TOOL LENGTH COMPENSATION
REGDEF/  G9,  G,  260,  260,  F, 80,73,74,75,76,81,82,83,84,85,86,87,88,89    $$ CYCLE
REGDEF/ G10,  G,  260,  260,  F, 52,53,54,55,56,57,58,59,92  $$ FIXTURE OFFSET
REGDEF/  H1,  H,  260,  260,  F         $$ TOOL LENGTH COMPENSATION REGISTER
```

```
REGDEF/  X1,  X,  214,  413,  T         $$ LINEAR ABSCISSA AXIS
REGDEF/  Y1,  Y,  214,  413,  T         $$ LINEAR ORDINATE AXIS
REGDEF/  Z1,  Z,  214,  413,  T         $$ LINEAR NORMAL AXIS
REGDEF/  Z2,  Z,  214,  413,  F         $$ AXIAL DRILL DEPTH
REGDEF/  I1,  I,  214,  413,  F         $$ ARC CENTER OFFSET PARALLEL TO ABSCISSA AXIS
REGDEF/  J1,  J,  214,  413,  F         $$ ARC CENTER OFFSET PARALLEL TO ORDINATE AXIS
REGDEF/  K1,  K,  214,  413,  F         $$ ARC CENTER OFFSET PARALLEL TO NORMAL AXIS
REGDEF/  D1,  D,  260,  260,  F         $$ CUTTER RADIUS COMPENSATION REGISTER
REGDEF/  R1,  R,  214,  413,  F         $$ CLEARANCE PLANE AXIAL DRILL CYCLE
REGDEF/  Q1,  Q,  214,  413,  F         $$ INCREMENT VALUE DEEP & BREAK CHIP CYCLES
REGDEF/  A1,  A,  214,  413,  T         $$ ROTARY AXIS AROUND ABSCISSA
REGDEF/  F1,  F,  311,  412,  T         $$ FEED RATE VALUE
REGDEF/  P1,  P,  244,  244,  F         $$ PROGRAM DWELL VALUE
REGDEF/  S1,  S,  540,  540,  T         $$ SPINDLE SPEED VALUE
REGDEF/  T1,  T,  240,  240,  F         $$ CURRENT TOOL NUMBER
REGDEF/  T2,  T,  240,  240,  F         $$ NEXT TOOL NUMBER
REGDEF/  L1,  L,  240,  240,  F         $$ SUBROUTINE NUMBER
REGDEF/  M0,  M,  260,  260,  T         $$ GENERAL MISCELLANEOUS FUNCTIONS
REGDEF/  M1,  M,  260,  260,  F, 6      $$ TOOL CHANGE FUNCTIONS (LOAD/TOOL)
REGDEF/  M2,  M,  240,  240,  T, 9,8,7,12,50,51,8      $$ COOLANT FUNCTIONS
                                                  (OFF,ON/FLOOD,MIST,THRU,TAP,AIR,AUX)
REGDEF/  M3,  M,  260,  260,  T, 5,3,4,19,21,22,41,42  $$ SPINDLE FUNTIONS
                                               (OFF,CLW,CCLW,LOCK,CLAMP,UNCLAMP,LOW,HIGH)
REGDEF/  M4,  M,  260,  260,  T, 17,98,99              $$ SUBROUTINE FUNCTIONS
                                                      (END,RETRCT,NO_RETRCT)
REGDEF/  M5,  M,  260,  260,  F, 0,1,2,30              $$ PROGRAM CONTROL FUNCTIONS
                                                      (STOP,OPSTOP,END,REWIND)
EOT
$$*************************************************************************
$$     SECTION THREE  -  Program MACRO Sequences
$$*************************************************************************
$$
$$*********** START MACRO SEQUENCE **************
MACRO/START
" "
"%"                                          $$ Rewind stop code
O(PROGID) PARTNO                             $$ Program ID
END
$$*********** FROM MACRO SEQUENCE (3-AXIS) ***************
MACRO/FROM
G3() G6() G7(40) G9(80)$                     $$ Initialization Block
END
$$*********** FROM MACRO SEQUENCE (4-AXIS) ***************
MACRO/FROM4
G3() G6() G7(40) G9(80) A1(0)$               $$ Initialization Block
END
$$**** RETRACT MACRO SEQUENCE (Tool Change) *****
MACRO/RETRCT
M2(9)$                                       $$ Coolant Off
G0(28) G3(91) Y1(HOMEY) Z1(CLEARP)$          $$ Machine Zero Y & Z-axis
END
$$******** 1ST TOOL CHANGE MACRO SEQUENCE *******
MACRO/TLCHG1
TPRINT>                                      $$ Add TPRINT text to next block
T1() M1(6)$                                  $$ Select &  Load programmed tool
END
$$***** ALL OTHER TOOL CHANGE MACRO SEQUENCE ****
```

```
MACRO/TLCHG
" "                                          $$ Blank block before tool change
TPRINT>                                      $$ Add TPRINT text to next block
T1() M1(6)$                                  $$ Select & Load programmed tool
END
$$************ GOHOME MACRO SEQUENCE *************
MACRO/GOHOME
M2(9)$                                       $$ Coolant Off
G0(28) G3(91) Y1(HOMEY) Z1(HOMEZ)$           $$ Machine home Y & Z axis
END
$$************ RESTART MACRO SEQUENCE (3-AXIS) ************
MACRO/RESTAR
G1(0) G3(90) G10() XX() YY() SS() M3()$      $$ 1st XY motion after Tool Change
G8() Z1() H1() M2()$                         $$ 1st Z motion after Tool Change
END
$$************ RESTART MACRO SEQUENCE (4-AXIS) ************
MACRO/RESTAR4
G1(0) G3(90) G10() X1() Y1() A1() S1() M3()$ $$ 1st XY motion after Tool Change
G8() Z1() H1() M2()$                         $$ 1st Z motion after Tool Change
END
$$******** CYCLE ACTIVATE MACRO SEQUENCE *********
MACRO/CYCLE
G0() G9() X1() Y1() Z2() R1() F1()$           $$ CYCLE (DRILL,BORE, etc)Activation
G0() G9() X1() Y1() Z2() R1() Q1() F1()$      $$ CYCLE (DEEP,BRKCHIP) Activation
G0() G9() X1() Y1() Z2() R1() F1()$           $$ CYCLE (TAP)Activation
G9(80)$                                      $$ CYCLE De-activation Block
END
$$************* PROGRAM END SEQUENCE (3-AXIS) ************
MACRO/PRGEND
M3(5)$                                       $$ Spindle Off
G1(0) G2(28) G3(91) M2(9)$                   $$ Z-axis home, Coolant Off
G1(0) G2(28) G3(91) Y1(HOMEY) Z1(HOMEZ)$     $$ Machine home XY Axis
M5(99)$                                      $$ End of Main program code
END
$$************* PROGRAM END SEQUENCE (4-AXIS) ************
MACRO/PRGEND4
M3(5)$                                       $$ Spindle Off
G1(0) G2(28) G3(91) M2(9)$                   $$ Z-axxis home, Coolant Off
G1(0) G2(28) G3(91) Y1(HOMEY) Z1(HOMEZ) A1(0)$ $$ Machine home XY Axis
M5(99)$                                      $$ End of Main program code
END
$$*****************************************************************************
$$     SECTION FOUR  -  Additional Program SYNTAX Definitions
$$*****************************************************************************
SYNDEF/PALLET,IN                             $$ DEFINE LOAD PALLET SYNTAX
M0(77)$                                      $$ DEFINE HYPER MACHINE CODE BLOCK
SYNDEF/PALLET,OUT                            $$ DEFINE UNLOAD PALLET SYNTAX
M0(78)$                                      $$ DEFINE HYPER MACHINE CODE BLOCK
SYNDEF/CALSUB,&1,REP,&2                      $$ DEFINE SUBROUTINE CALL SYNTAX
P2(&1) L1(&2)$                               $$ DEFINE HYPER MACHINE CODE BLOCK
EOF
```