IsoNs – Next Step
Standard M User Guide

www.promax.it

PROMAX Motion & Control

Rev. 1.0.0

1 M6 Linear TOOL CHANGE

All the plc cycle, for M6 management, is writen by IsoNs Gcode.
Following the Gcode

```
//MACRO LINEARTOOL CHANGE REV 2.0.0
//(C) PROMAX SRL
//M6

$APP=$[X7]        // CHECK IF NORMAL RUN OR SIMULATION RUN
IF $APP<>0        //IF THE SYSTEM IT'S NOT IN RUN
        GOTO END          //DOING NOTHING
END_IF

M5                //STOP SPINLDE

IF $[I5]=0        //IF NO "SPINDLE CLOSE WITH TOOL" INPUT
        GOTO  LOAD        //GO DIRECTLY ON CHARGE SECTION
END_IF

G96               //OFFSET SUSPENSION
G98               //ZERO OFFSET SUSPENSION
G87               //HEAD OFFSET SUSPENSION
G44               //TOOL LENGHT COMPENSATION SUSPENSION
G0Z0              //MOVE Z0 IN FREE POSITION
LOAD_VAR TOOL.INF         //LOAD OLD TOOL NUMBER
GET_VAR $OLDUT 0
$ACTUT=$[X6]      //STORE THE TOOL NUMBER TO BE LOADED
$VEL=1            //SET APPROCH SPEED TO 1 MT/MIN

//G81 X2 ENABLE OF THE SECONDARY SOFTWARE LIMIT (IF NECESSARY)
//G81 X3

IF $ACTUT=0       //IF THE ACTUAL TOOL IS 0 - ONLY LEAVE THE TOOL, THAT IS IN THE SPINDLE, IN THE WAREHOUSE
        GOTO DISCHARGE
END_IF
IF $OLDUT=$ACTUT      //IF THE ACTUAL AND THE OLD TOOL ARE THE SAME, ONLY CALCULATION
        GOTO CALCULATE
END_IF
@DISCHARGE
        //LEAVE THE TOOL ALREADY IN THE SPINDLE, IN HIS WAREHOUSE POSITION...
IF $OLDUT=0
        ERROR 3           //IF THE OLD TOOL IS 0 - UNKNOWN TOOL
        END_PROGRAM
END_IF
T[$OLDUT]                 // SET T TO TOOL IN THE SPINDLE
$DELTAZ=$[U19]            //LOAD FROM THE TOOL TABLE, THE POSITION DELTAS FOR CHANGING
$DELTAY=$[U18]
$DELTAX=$[U17]
$POSZ=$[U16]              //LOAD FROM TOOL TABLE, THE TOOL POSITION
$POSY=$[U15]
$POSX=$[U14]
```

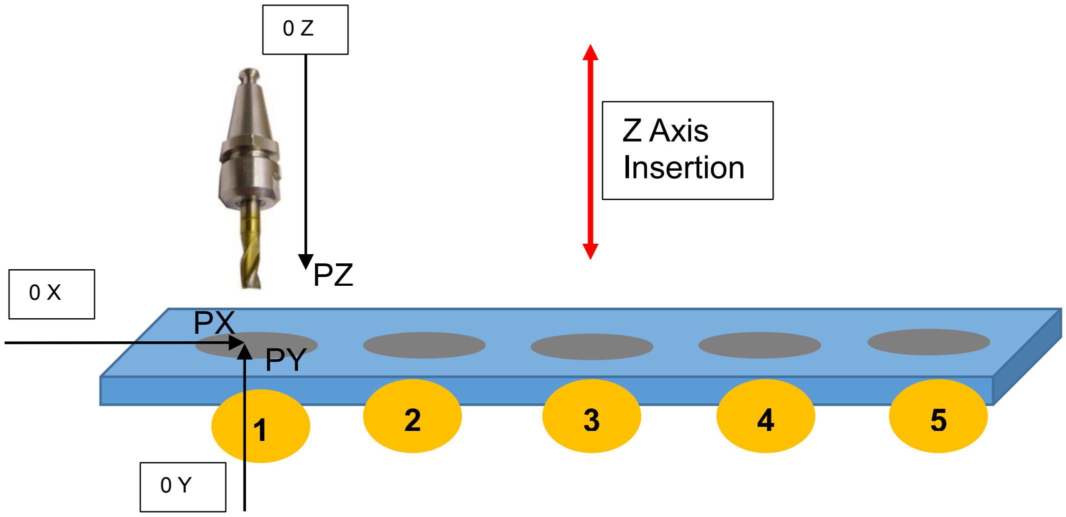
```
$APPX=$POSX+$DELTAX  //LOAD THE POSITIONS
$APPY=$POSY+$DELTAY
$APPZ=$POSZ+$DELTAZ
G0 X[$APPX] Y[$APPY] //GO TO THE DISCHARGE POSITION (WITH DELTA)
G0 Z[$APPZ]
G1 Z[$POSZ] F[$VEL]  //CLAMP ENTRY
G62                  //WAIT END MOVE
G1 X[$POSX] F[$VEL]  //CLAMP ENTRY
G62                  //WAIT END MOVE
G1 Y[$POSY] F[$VEL]  //CLAMP ENTRY

@LOAD //STARTING THE CHARGE SECTION IF NO TOOL IN SPINDLE
G62    //WAIT END MOVE
$[O1]=1       //OPEN THE SPINDLE
$[O2]=1       //CLEANING BLOW ACTIVATION
G0 Z[$APPZ]   //GO TO RELEASE POSITION (ON Z AXIS)

//END OF THE OLD TOOL LEAVING SECTION
T[$ACTUT]     //RELOAD ACTUAL TOOL
IF $ACTUT=0  //IF WE HAVE ONLY TO LEAVE OLD TOOL - GO TO END
       $[O2]=0       //CLOSE CLEANING BLOW
       $[O1]=0       //CLOSE THE SPINDLE
       G4 F0.5       //LITTLE PAUSE
       WAIT_INPUT 4 1 4 1   //WAIT FOR THE SPINDLE CLOSED INPUT (WITHOUT TOOL ,INPUT 4 - 5 PHYSICS)
       G0 Z0         //MOVE Z TO 0
       GOTO END      //GO TO END
END_IF
//OTHERWISE
//LOAD THE NEW TOOL IF IT'S DIFFERENT TO 0
$DELTAZ=$[U19]       //LOAD FROM THE TOOL TABLE, THE POSITION DELTAS FOR CHANGING
$DELTAY=$[U18]
$DELTAX=$[U17]
$POSZ=$[U16]
$POSY=$[U15]
$POSX=$[U14]
$APPX=$POSX+$DELTAX
$APPY=$POSY+$DELTAY
$APPZ=$POSZ+$DELTAZ

G0 X[$POSX] Y[$POSY] //GO TO THE CHARGE POSITION
G1 Z[$POSZ] F[$VEL]  //CLAMP ENTRY
G62    //WAIT END MOVE
$[O2]=0       //CLOSE CLEANING BLOW
$[O1]=0       //CLOSE THE SPINDLE
WAIT_INPUT 5 1 4 1  //WAIT FOR THE SPINDLE CLOSED INPUT (WITH TOOL ,INPUT 5 - 6 PHYSICS)
G4 F0.5
G1 X[$APPX] Y[$APPY] //GO TO THE DISCHARGE POSITION (WITH DELTA)
G0 Z0                //GO UP TO SECURE QUOTA
DIM_VAR 1
WRITE_VAR $ACTUT 0
SAVE_VAR TOOL.INF    //SAVE THE ACTUAL TOOL
//END OF THE NEW TOOL CHARGING
```

```
@CALCULATE              //CALCULATION SECTION

//USE THE FOLLOWING SECTION FOR PRESET Z AXIS BY DIST Z PARAMETER

//------------PRESET Z AXIS  WITH DISTZ PARAMETER----------- (see Chapr.1.6)
//READ_PARMAC "DISTZ" $DISTZ
//$DISTZ=$DISTZ/1000

//$PRESETZ=$[U1]
//$PRESETZ=-$DISTZ+$PRESETZ
//G94 Z[$PRESETZ]
//---------------------END-----------------------------

@END

//G81 X0                              RESTORE NORMAL SOFTWARE LIMIT
//G81 X1

G97     //REACTIVATE OFFSET
G99     //REACTIVATE ZERO OFFSET
G88     //REACTIVATE HEAD OFFSET
```

1.1 Mode of Linear Tool Change

The M6 IsoNs Gcode macro manages the following linear tool change mode:

MODE A (insert tool up)



TOOL Parameters Description

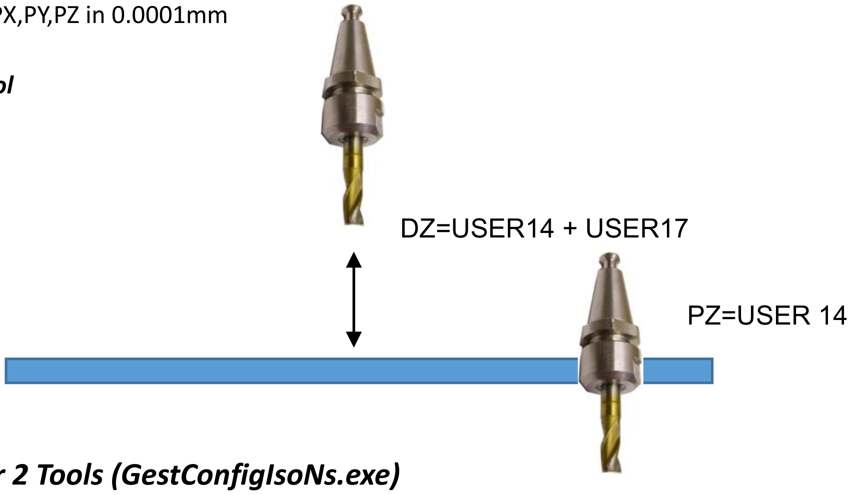
DX=0	($[U17] User 15 in Tool Table)
PX= Abs X pos. refered to tool Nr. **Center Hole** *)	($[U14] User 12 in Tool Table)
DY=0	($[U18] User 16 in Tool Table)
PY= Abs Y pos. refered to tool Nr. **Center Hole** *)	($[U15] User 13 in Tool Table)
DZ=FREE Z POSITION WITH TOOL **)	($[U19] User 17 in Tool Table)
PZ= Abs Z pos. refered to tool Nr. **Center Hole** *)	($[U16] User 14 in Tool Table)

*) About the unit used for PX,PY,PZ parameters, you must use the same unit defined in the RESQUOTE Parameter:

RESQUOTE=1000	PX,PY,PZ in 0.001mm
RESQUOTE=10000	PX,PY,PZ in 0.0001mm

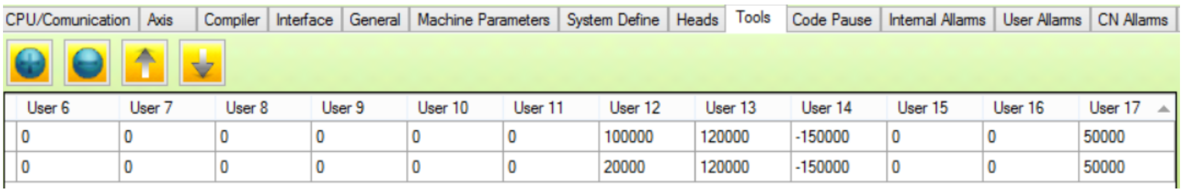
Etc.

*****) Free Z Position with tool***



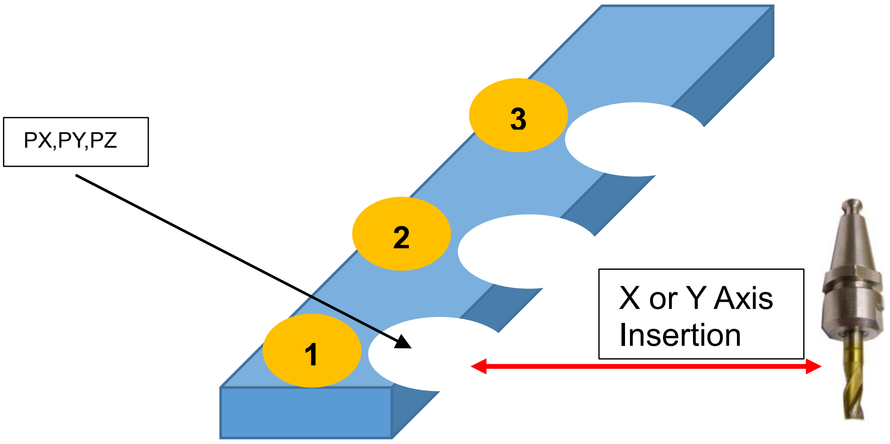
Example Tool Table for 2 Tools (GestConfigIsoNs.exe)

Same position in Z and Y, offset hole in X 100000 um 100 mm. Z negative position in Down direction.



User 6	User 7	User 8	User 9	User 10	User 11	User 12	User 13	User 14	User 15	User 16	User 17
0	0	0	0	0	0	100000	120000	-150000	0	0	50000
0	0	0	0	0	0	20000	120000	-150000	0	0	50000

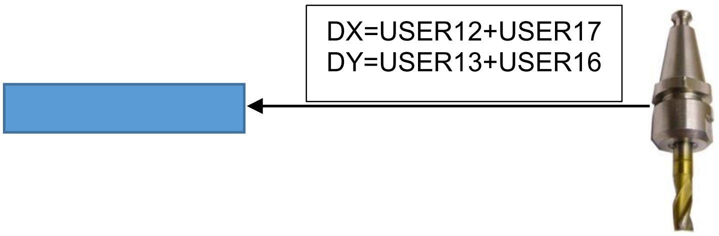
MODE B (insert tool From Side)



TOOL Parameters Description

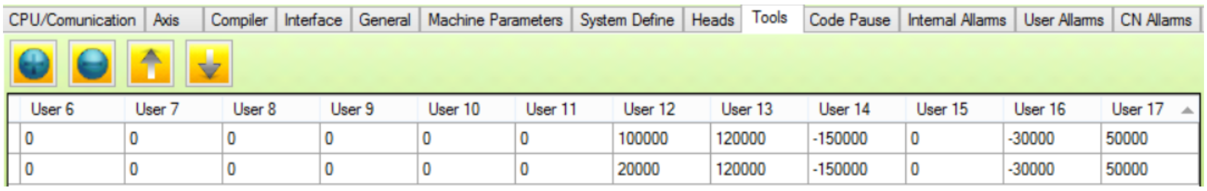
DX=FREE X POSITION ***)**	(\$[U17] User 15 in Tool Table)
PX= Abs X pos. refered to tool Nr. **Center Hole** *)	(\$[U14] User 12 in Tool Table)
DY= FREE Z POSITION ***)**	(\$[U18] User 16 in Tool Table)
PY= Abs Y pos. refered to tool Nr. **Center Hole** *)	(\$[U15] User 13 in Tool Table)
DZ= FREE Z POSITION WITH TOOL **see above**	(\$[U19] User 17 in Tool Table)
PZ= Abs Z pos. refered to tool Nr. **Center Hole** *)	

***)**



Example Tool Table for 2 Tools (GestConfigIsoNs.exe)

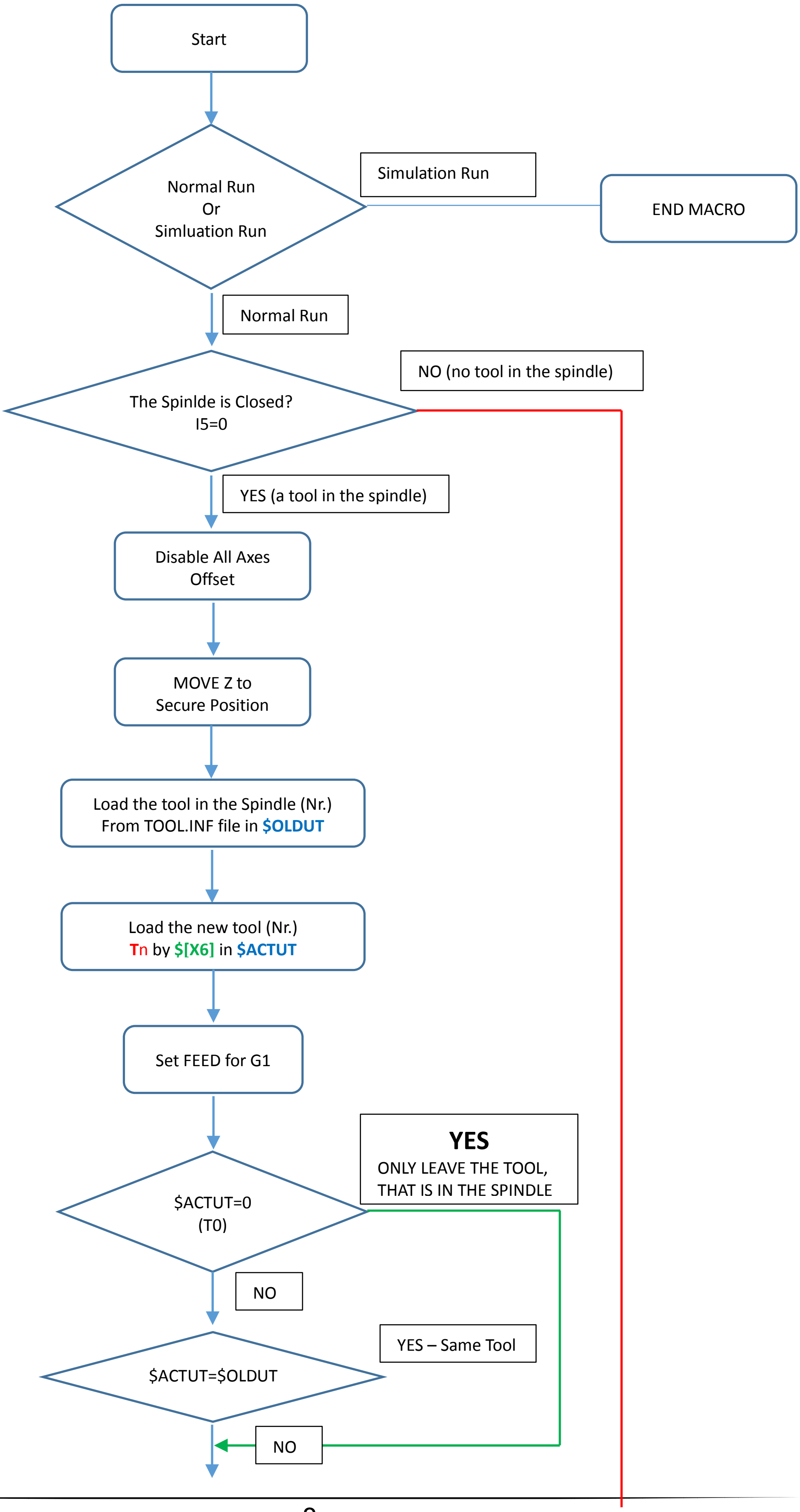
Same position in Z and Y, offset hole in X 100000 um 100 mm. Z negative position in Down direction

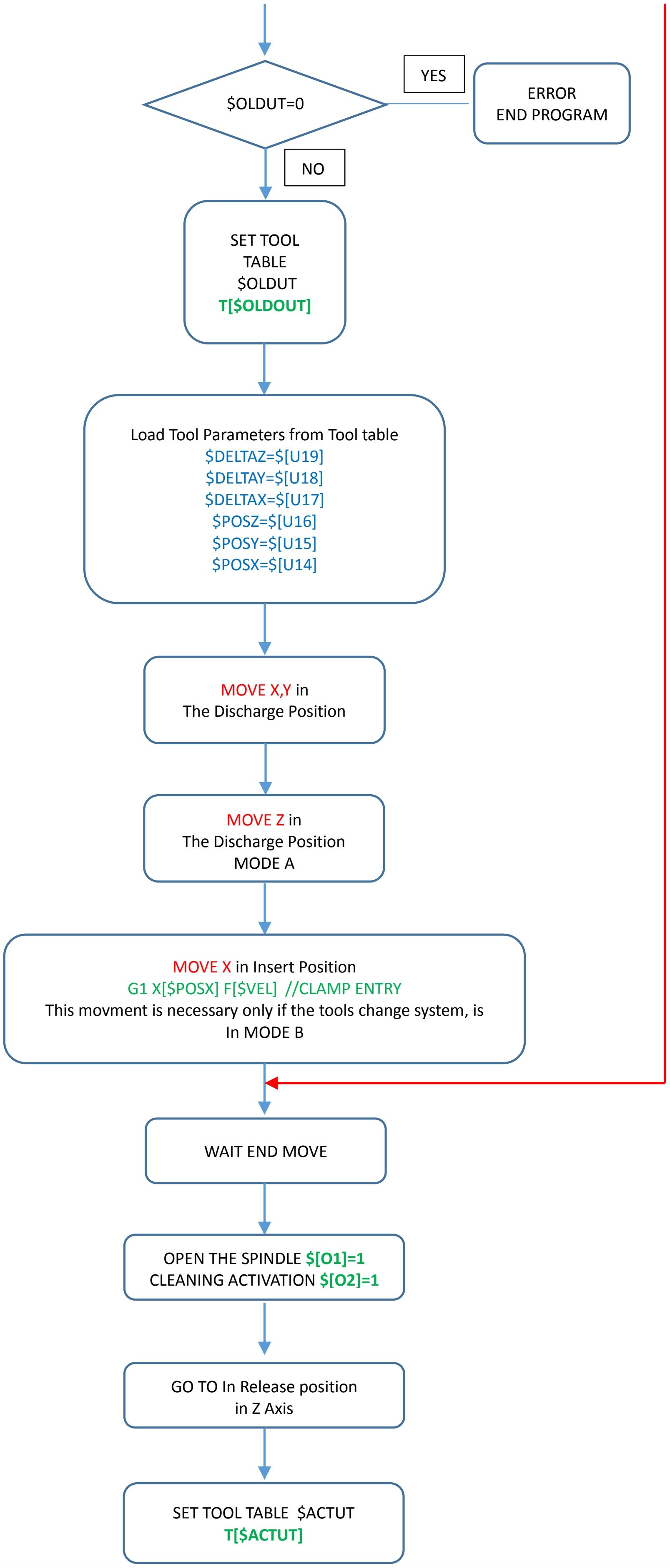


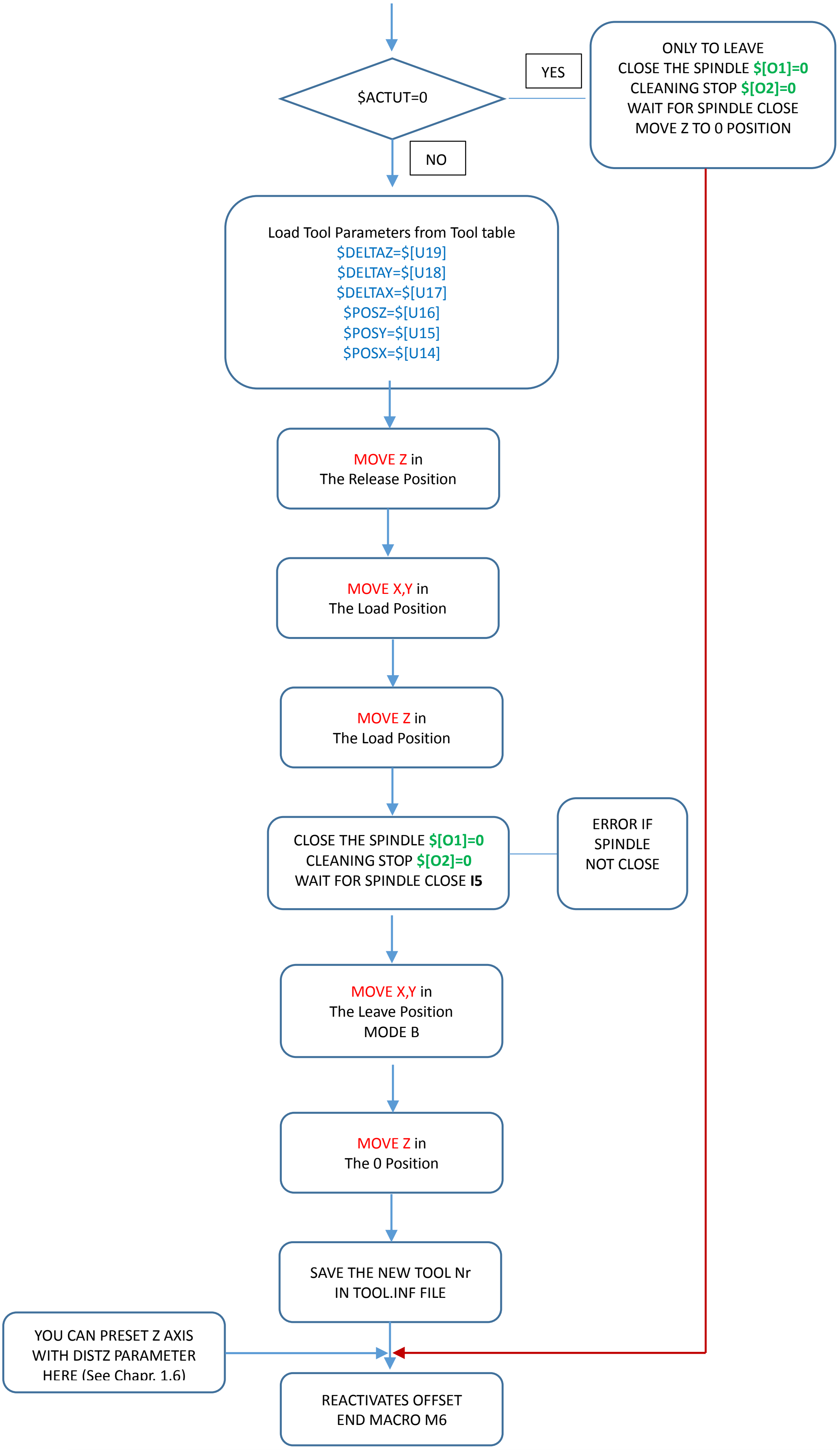
User 6	User 7	User 8	User 9	User 10	User 11	User 12	User 13	User 14	User 15	User 16	User 17
0	0	0	0	0	0	100000	120000	-150000	0	-30000	50000
0	0	0	0	0	0	20000	120000	-150000	0	-30000	50000

1.2 M6 Flow Chart

The M6 IsoNs Gcode macro use the following method:




$OLDUT=0
YES
ERROR
END PROGRAM
NO
SET TOOL
TABLE
$OLDUT
T[$OLDOUT]
Load Tool Parameters from Tool table
$DELTAZ=$[U19]
$DELTAY=$[U18]
$DELTAX=$[U17]
$POSZ=$[U16]
$POSY=$[U15]
$POSX=$[U14]
MOVE X,Y in
The Discharge Position
MOVE Z in
The Discharge Position
MODE A
MOVE X in Insert Position
G1 X[$POSX] F[$VEL] //CLAMP ENTRY
This movment is necessary only if the tools change system, is
In MODE B
WAIT END MOVE
OPEN THE SPINDLE $[O1]=1
CLEANING ACTIVATION $[O2]=1
GO TO In Release position
in Z Axis
SET TOOL TABLE $ACTUT
T[$ACTUT]


$ACTUT=0
YES
ONLY TO LEAVE
CLOSE THE SPINDLE $[O1]=0
CLEANING STOP $[O2]=0
WAIT FOR SPINDLE CLOSE
MOVE Z TO 0 POSITION
NO
Load Tool Parameters from Tool table
$DELTAZ=$[U19]
$DELTAY=$[U18]
$DELTAX=$[U17]
$POSZ=$[U16]
$POSY=$[U15]
$POSX=$[U14]
MOVE Z in
The Release Position
MOVE X,Y in
The Load Position
MOVE Z in
The Load Position
CLOSE THE SPINDLE $[O1]=0
CLEANING STOP $[O2]=0
WAIT FOR SPINDLE CLOSE I5
ERROR IF
SPINDLE
NOT CLOSE
MOVE X,Y in
The Leave Position
MODE B
MOVE Z in
The 0 Position
SAVE THE NEW TOOL Nr
IN TOOL.INF FILE
YOU CAN PRESET Z AXIS
WITH DISTZ PARAMETER
HERE (See Chapr. 1.6)
REACTIVATES OFFSET
END MACRO M6

1.3 CNC Digital Inputs

The inputs are enumerated from 0 (first input is I0)

I4 → Spindle closed without tool

If you not use this input change the following code (See red code):

```
IF $ACTUT=0             //IF WE HAVE ONLY TO LEAVE OLD TOOL - GO TO END
        $[O2]=0                 //CLOSE CLEANING BLOW
        $[O1]=0             //CLOSE THE SPINDLE
        //G4 F0.5                                       //LITTLE PAUSE
        //WAIT_INPUT 4 1 4 1    WAIT FOR THE SPINDLE CLOSED INPUT (WITHOUT TOOL ,INPUT 4 - 5 PHYSICS)
        G4F1 //WAIT 1 SEC FOR SPINDLE OPEN
        G0 Z0                                    //MOVE Z TO 0
        GOTO END                                 //GO TO END
END_IF
```

I5 → Spindle Close with tool

This input is Required to use

1.4 CNC Digital Outputs

The outputs are enumerated from 0 (first output is I0)

O1=1 → Spindle Open
O1=0 → Spindle Close

O2=1 → AIR ON (for cleaning tools)
O2=0 → AIR OFF

If you not use this output change the following code (See red code):

```
@LOAD                                        //STARTING THE CHARGE SECTION IF NO TOOL IN SPINDLE
G62     //WAIT END MOVE
$[O1]=1                 //OPEN THE SPINDLE
//$[O2]=1               //CLEANING BLOW ACTIVATION
.
.
.
IF $ACTUT=0             //IF WE HAVE ONLY TO LEAVE OLD TOOL - GO TO END
        //$[O2]=0               //CLOSE CLEANING BLOW
        $[O1]=0             //CLOSE THE SPINDLE
        .
        .
END_IF
.
.
G62
//$[O2]=0           //CLOSE CLEANING BLOW
$[O1]=0            //CLOSE THE SPINDLE
```

1.5 Tool Table Parameters

The tool table, contains all tool parameters used for M6 tool change and for Gcode. About M6 the parameters meaning are described in the ***Chapr. 1.1 Mode of linear tool change***. These depend by Mode tool used.
The Parameters Table is setted by **Tn** Gcode function.
Below the standard parameter:

Diameter	Tool Diameter - used by G42 G41 ex: 23.2
Len	Tool Len - used by G43 or Z preset
Vrot (rpm)	Rotation max speed – Used by M3 – M4
User 1	Generally used for 2nd clone tool – Reserved
User 2 to User 11	Free
User 12	Absolute Position **X** for Insertion or Extration Tool
User 13	Absolute Position **Y** for Insertion or Extration Tool
User 14	Absolute Position **Z** for Insertion or Extration Tool
User 15	Delta Position **X** for Insertion or Extration Tool
User 16	Delta Position **Y** for Insertion or Extration Tool
User 17	Delta Position **Z** for Insertion or Extration Tool

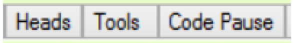
Prepare a tool table

a) Run ***"GestConfigIsoNs.exe"*** in folder ***Utilty→GestConfigIsoNs*** or Run by:

→ **IsoNs Configurator**

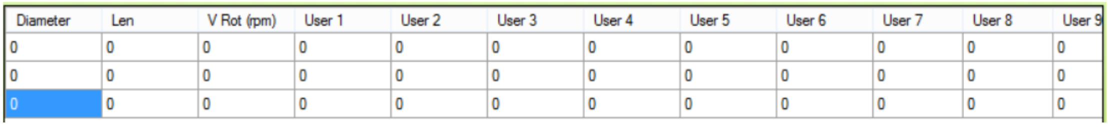
b) Open IsoNs.cfg in IsoNs Folder by ***Load Cfg Button*** (or if you have already run IsoNs, click ***Load default***)

c) Click on ***Tools*** tab



d) Insert the Number of tools available in your machine (ex. 3 tools)

Click on Button ***"+"*** for 3 times



Diameter	Len	V Rot (rpm)	User 1	User 2	User 3	User 4	User 5	User 6	User 7	User 8	User 9
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0

The parameters :

Diameters, Len, V Rot (rpm) can be changed by ***Utiltity→GestTabut→GestTabut.exe*** or:

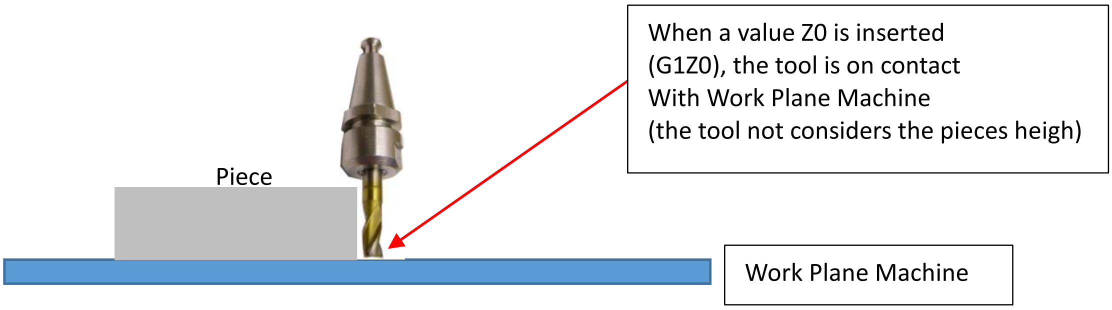
→ Tool Table

e) Insert the parameters User12,User13,User14,User15,User16,User17

f) Insert PassWord and save configuration

1.6 Preset Z Axis with DISTZ Parameter

The M6 code, can preset Z Axis with tool Len.
The preset value considers the following method:



When a value Z0 is inserted (G1Z0), the tool is on contact With Work Plane Machine (the tool not considers the pieces heigh)

Work Plane Machine

For use this method, is necessary the following instructions:

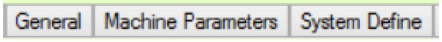
a) Active the code in M6 (remove the remarks)

```
//------------PRESET Z AXIS  WITH DISTZ PARAMETER-----------
READ_PARMAC "DISTZ" $DISTZ
$DISTZ=$DISTZ/1000
$PRESETZ=$[U1]
$PRESETZ=-$DISTZ+$PRESETZ
G94 Z[$PRESETZ]
//--------------------END----------------------------
```

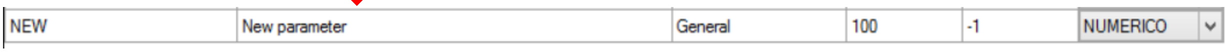
Remove the initial Remarks *"//"*

b) Insert the parameter **DISTZ** in the configuration "IsoNs.cfg" (open the IsoNs.cfg see the Chapr. 1.5)

c) Click on ***Machine Parameters*** tab

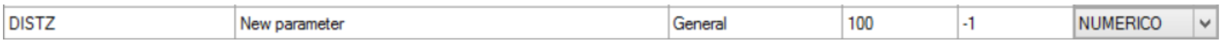


d) Click on Button *"+"*



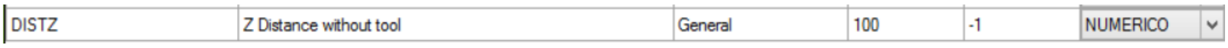
NEW	New parameter	General	100	-1	NUMERICO

e) Change the name in **"DISTZ" (upper case)**



DISTZ	New parameter	General	100	-1	NUMERICO

f) Change the description in "**Z Distance without tool**"



DISTZ	Z Distance without tool	General	100	-1	NUMERICO

g) Save the configuration

1.7 Create a file TOOL.INF

The M6 code, uses a file var TOOL.INF. For create this file use the following code and run it (one times only):

```
$ACTUT=1
DIM_VAR 1
WRITE_VAR $ACTUT 0
SAVE_VAR TOOL.INF   //INIT THE ACTUAL TOOL
```

The above code, writes in the file the nr. 1 tool.
You must insert in $ACTUT variable, the actual tool number in the spindle (ex: **$ACTUT=2 etc.**).
You must charge manually the first tool

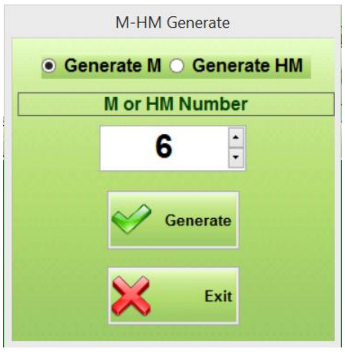
1.8 Create a M6 Function

You can test the M6 code and when it is Ok, you must create the M6 Function in the following mode:

a) Load the M6 code

b) Open the Plug In M HM

c) Set in the Plug In M6 and Generate M



M-HM Generate

● Generate M ○ Generate HM

M or HM Number: 6

Now the M6 is ready to use

2 M3-M4-M5 Spindle management

Following the standard M3,M4,M5 functions for Spindle management.
These functions are developed in two parts:

1) M3 – M4 – M5 in Gcode IsoNs
2) M1003 – M1004 – M1005 (called by M3,M4,M5) in VTB Code on CNC

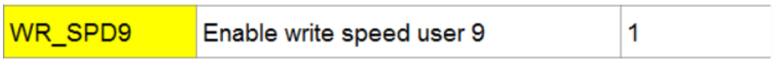
The M1003,M1004,M1005 depends from the CNC type and the analog output type

2.1 GENERATE MACRO M3 M4 M5

The M3,M4 function start the spindle in the CW or CCW direction.
The spindle speed, is set from **Sval** Gcode function (ex: S12000). Generally this function, writes directly the spindle speed in rpm.
For use the **Sval** in the VTB application, is necessary set the IsoNs parameter **WR_SPD9=1**:

a) **Open the Machines Parameters Browser**

b) **Set WR_SPD9 and save the parameters**



WR_SPD9	Enable write speed user 9	1

c) **Write the M3 code**

```
//********************************
//MACRO FOR SPINDLE CW
//(C) PROMAX SRL
//M3
//********************************
M1003   // CALL  M1003 ON CNC
//WAIT_INPUT 6 1 10 1       *)
//G4F2                      **)
```

***) Use this if the Spindle has the VEL REACHED output. In this case uses the INPUT 6 to logical state 1 with time out 10 sec**
****) Use the simple delay**

d) **Open the Plug In M HM**
e) **Set in the Plug In M3 and Generate M**
f) **Write the M4 code and repeat the D and E points (with M4)**

```
//********************************
//MACRO FOR SPINDLE CCW
//(C) PROMAX SRL
//M4
//********************************
M1004   // CALL  M1004 ON CNC
//WAIT_INPUT 6 1 10 1       *)
//G4F2                      **)
```

g) **Write the M5 code and repeat the D and E points (with M5)**

```
//********************************
//MACRO FOR SPINDLE STOP
//(C) PROMAX SRL
//M5
//********************************
M1005   // CALL  M1005 ON CNC
```

2.2 GENERATE MACRO M1003 M1004 M1005

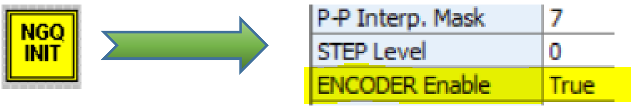
The M1003,M1004 ,M1005 are written in VTB and it manage really the spindle. **The control type, is in Voltage 0-10V.** These are hardware dependent, and the VTB code, is not the same, if the analog output is different.

These Macro read the spindle speed from **ISOV1_Generic(9)** data memory. It is written from Gcode when the **Sval** function is executed In the **ISOV1_Generic(9)** you can read the **Sval**:

Gcode	**VTB**
S12000	ISOV1_Generic(9)=12000
S8000	ISOV1_Generic(9)=8000

M1003,M1004,M1005 for NG35+NGIO, NGMEVO+NGMsX,NGQuark with Analog Output

If the NGQuark board is used, set the **ENCODER ENABLE=true** in the NGQ init object.



P-P Interp. Mask	7
STEP Level	0
ENCODER Enable	True

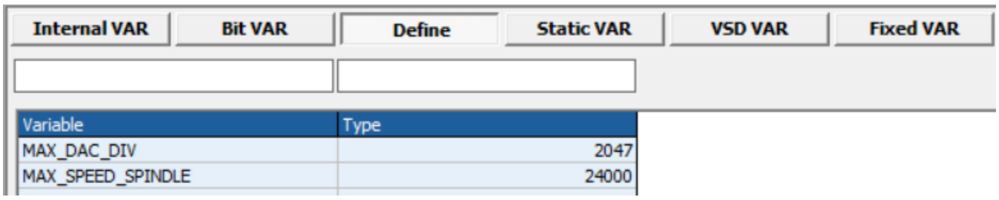
Digital I/O used

Out3→ISOV1.OUT2	**CW Direction**
Out4→ISOV1.OUT3	**CCW Direction**
Out5→ISOV1.OUT4	**START/STOP Spindle**

Analog Output used

Analog0→Ng_Dac(0,val)

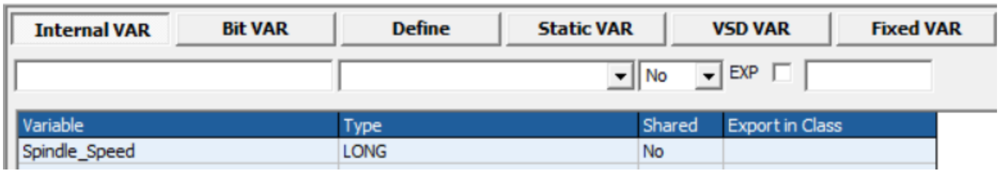
a) Declare the following DEFINE in VTB project



Variable	Type
MAX_DAC_DIV	2047
MAX_SPEED_SPINDLE	24000

MAX_DAC_DIV	Number of Digital Analog Output Divisions (not change)
MAX_SPEED_SPINDLE	Number of Spindle Rpm (set to Rpm at 10 Volt value)

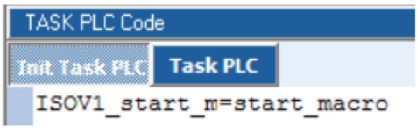
b) Declare the following INTERNAL VAR in VTB Project



Variable	Type	Shared	Export in Class
Spindle_Speed	LONG	No	

SPINDLE_SPEED Long variable

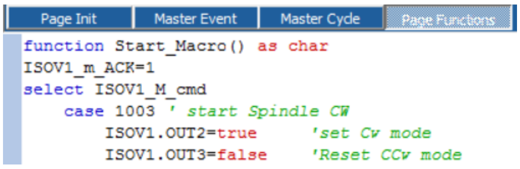
c) Written the following code in the TASK PLC CODE → INIT TASK PLC



```
ISOV1_start_m=start_macro
```

```
ISOV1_Start_m=Start_Macro
```

d) Written the following code in the MAIN → PAGE FUNCTIONS



```
function Start_Macro() as char
ISOV1_m_ACK=1
select ISOV1_M_cmd
    case 1003 ' start Spindle CW
        ISOV1.OUT2=true      'set Cw mode
        ISOV1.OUT3=false     'Reset CCw mode
```

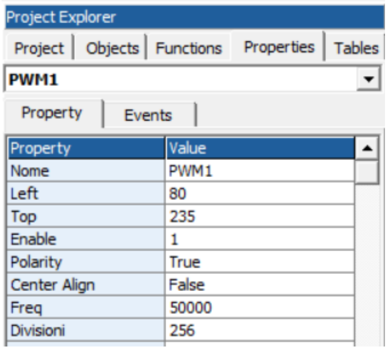
```
function Start_Macro() as char
ISOV1_m_ACK=1
select ISOV1_M_cmd
      case 1003 ' start Spindle CW
            ISOV1.OUT2=true   'set Cw mode
            ISOV1.OUT3=false  'Reset CCw mode
            ' Speed calculation
            Spindle_Spindle=(ISOV1_generic(9)*MAX_DAC_DIV)/MAX_SPEED_SPINDLE
            ng_dac(0, Spindle_Spindle) ' Set analog out
            ISOV1.OUT5=true ' Start Spindle
            ISOV1_status_m_run=0 ' Free IsoNs
      case 1004 ' start Spindle CCW
            ISOV1.OUT2=false  'Reset Cw mode
            ISOV1.OUT3=true   'set CCw mode
            ' Speed calculation
            Spindle_Spindle=(ISOV1_generic(9)*MAX_DAC_DIV)/MAX_SPEED_SPINDLE
            ng_dac(0, Spindle_Spindle) ' Set analog out
            ISOV1.OUT5=true ' Start Spindle
            ISOV1_status_m_run=0 ' Free IsoNs
      case 1005 ' Spindle Stop
            ISOV1.OUT5=false ' Stop Spindle
            Spindle_Spindle =0 ' set Speed to 0
            ng_dac(0,VelSpindle) ' Set analog out
            ISOV1_status_m_run=0 ' Free IsoNs
      case else
            ISOV1_m_ACK=0
endselect
endfunction
```

M1003,M1004,M1005 for NGMEVO+PWM Output

Insert the following object in the VTB Project:

General→ Cpwm.vco → PWM NGM – EVO

And set the following properties



Property	Value
Nome	PWM1
Left	80
Top	235
Enable	1
Polarity	True
Center Align	False
Freq	50000
Divisioni	256

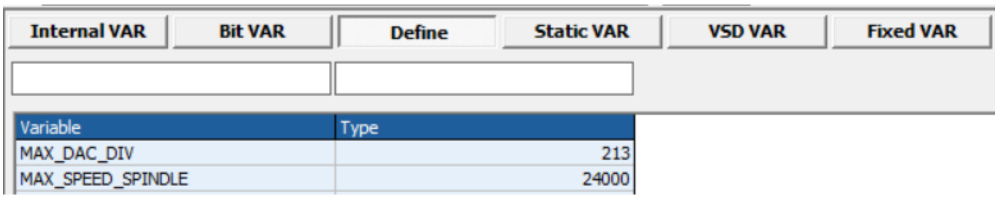
Digital I/O used

Out3→ISOV1.OUT2	**CW Direction**
Out4→ISOV1.OUT3	**CCW Direction**
Out5→ISOV1.OUT4	**START/STOP Spindle**

Analog Output used

Analog0→PWM_Val(0,val)

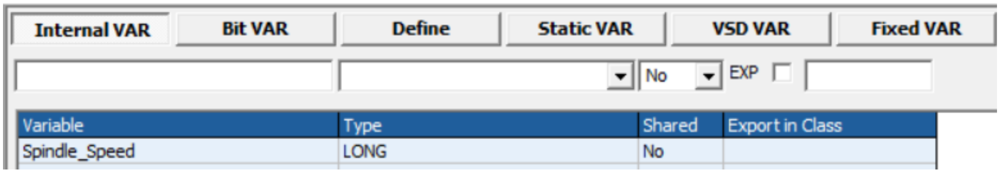
a) Declare the following DEFINE in VTB project



Variable	Type
MAX_DAC_DIV	213
MAX_SPEED_SPINDLE	24000

MAX_DAC_DIV	Number of Digital Analog Output Divisions (not change)
MAX_SPEED_SPINDLE	Number of Spindle Rpm (set to Rpm at 10 Volt value)

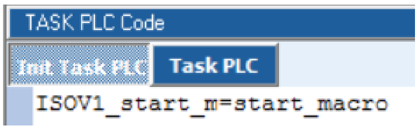
b) Declare the following INTERNAL VAR in VTB Project



Variable	Type	Shared	Export in Class
Spindle_Speed	LONG	No	

SPINDLE_SPEED Long variable

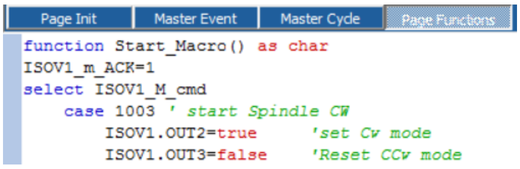
c) Written the following code in the TASK PLC CODE → INIT TASK PLC



```
ISOV1_start_m=start_macro
```

```
ISOV1_Start_m=Start_Macro
```

d) Written the following code in the MAIN → PAGE FUNCTIONS



```
function Start_Macro() as char
ISOV1_m_ACK=1
select ISOV1_M_cmd
    case 1003 ' start Spindle CW
        ISOV1.OUT2=true      'set Cw mode
        ISOV1.OUT3=false     'Reset CCw mode
```

```
function Start_Macro() as char
ISOV1_m_ACK=1
select ISOV1_M_cmd
      case 1003 ' start Spindle CW
            ISOV1.OUT2=true   'set Cw mode
            ISOV1.OUT3=false  'Reset CCw mode
            ' Speed calculation
            Spindle_Spindle=(ISOV1_generic(9)*MAX_DAC_DIV)/MAX_SPEED_SPINDLE
            PWM_Val(0, Spindle_Spindle) ' Set analog out
            ISOV1.OUT5=true ' Start Spindle
            ISOV1_status_m_run=0 ' Free IsoNs
      case 1004 ' start Spindle CCW
            ISOV1.OUT2=false  'Reset Cw mode
            ISOV1.OUT3=true   'set CCw mode
            ' Speed calculation
            Spindle_Spindle=(ISOV1_generic(9)*MAX_DAC_DIV)/MAX_SPEED_SPINDLE
            PWM_Val (0, Spindle_Spindle) ' Set analog out
            ISOV1.OUT5=true ' Start Spindle
            ISOV1_status_m_run=0 ' Free IsoNs
      case 1005 ' Spindle Stop
            ISOV1.OUT5=false ' Stop Spindle
            Spindle_Spindle =0 ' set Speed to 0
            PWM_Val (0,VelSpindle) ' Set analog out
            ISOV1_status_m_run=0 ' Free IsoNs
      case else
            ISOV1_m_ACK=0
endselect
endfunction
```

Index