MACRO SHOP

OFFSET RECALCULATION

SIEMENS 840D (version 4.4 and further)

Rev. 0

FIELD OF APPLICATION

The macro-instruction recalculates the offset in relation to the centre of the table, following the rotation of the rotary table. This macro-instruction can be used to create a new rotated or rotated and shifted offset from any basic one by setting the table rotation degrees. The macro can be used in two ways in order to meet various requirements. The first mode creates a new offset recalculated from the table centre, which is typical of machining operations carried out on boring machines where you wish to have new offsets but recalculated from the initial one, so that you can work independently on all the offsets during the finishing phases. A second case of use involves temporarily moving the active offset using the TRANS function to reach the new position of the offset after the table rotation. At the end of the machining, simply deactivate the TRANS function to return to the offset initially set. This second option is typical of machining with table tracking, where the table positions are various and always linked to an offset. The definition of the table centre can be made in the basic frame inside the work offset list or directly inside the macro itself with internal parameters. The macro can be used both in the case of horizontal machines, where the offset recalculation involves the X and Z axes, and in the case of vertical machines, where the offset recalculation involves the X and Y axes.

CYCLE DESCRIPTION

The macro-instruction call will be executed by inserting the call to the subprogram TABLEFRAME in the program. The call must include the parameters necessary for the offset recalculation. Some parameters can be omitted by leaving the space between the commas exactly as for standard SIEMENS cycles.

The macro can be used in two ways:

CASE 1: NEW OFFSET CREATION

TABLEFRAME(TablePos,WO_Dest,X0,Y0,Z0,X1,Y1,Z1,WO_Ori)

PARAMETER DESCRIPTION:

TABLEFRAME
TABLEFRAME is the name of the subprogram with which the macro-instruction is provided.

FOCUS PARAMETERS

CASE1

TABLEFRAME

TABLEPOS

WO_DEST

X0 Y0 Z0

X1 Y1 Z1

WO_ORI

CASE2

TablePos

Table position for which you want to calculate the new rotated offset. Enter both positive and negative values. Example: 15, 270, -35.

WO_Dest

The second parameter WO_Dest is used to indicate the offset to be generated. Indicate the offset without the G function. Example:

54 -> for the G54 work offset
55 -> for the G55 work offset
505 -> for the G505 work offset

X0, Y0, Z0

Parameters X0, Y0 and Z0 are used to indicate the possible translation values of the offset along the respective axes. To be used if you want to absorb any translation directly in the offset. Parameters X0 Y0 Z0 will be used to apply the translation values before rotating the frame.

X1, Y1, Z1

Parameters X1, Y1 and Z1 are used to indicate any translation values of the offset along the respective axes. To be used if you want to absorb any translation directly in the offset. Parameters X1 Y1 Z1 will be used to apply the translation values after the frame has been rotated.

WO_Ori

The last parameter WO_Ori indicates the offset from which the calculation of the new rotated frame should start. If omitted, the offset active at the time of the call will be taken as the reference offset. Indicate the offset without the G function. Example:

54 -> for the G54 work offset
55 -> for the G55 work offset
505 -> for the G505 work offset

CASE 2: ACTIVE OFFSET TRANSLATION

The macro translates the offset using the TRANS function to return the active one to the position calculated after a table rotation. The offset in use will therefore remain active, but simply translated with the TRANS function within the macro itself. At the end of machining, always enter the deactivation of the TRANS function. If it is necessary to use the TRANS function in the machining process, use the additive version ATRANS or, again, use the parameters X0,Y0,Z0 or X1,Y1,Z1. The macro recognises case No. 2 when the target offset WO_Dest is omitted.

So the syntax will be:

G54
TABLEFRAME(45)

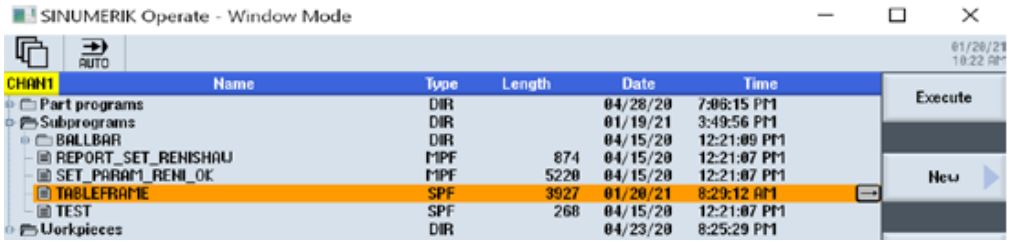
or, using the WO_DEST parameter:

TABLE(45,,,,,,,54)
In this case the macro will recalculate the G54 offset but rotated by 45 degrees. From this point onwards the TRANS function will be active.

PROGRAMMING

MACRO INSTALLATION

1. Copy the TABLEFRAME.SPF program that you will receive when you purchase the macro into the SUBPROGRAMS folder *(see photo below)*.



2. Write the following text inside each program in which you wish to use the macro:
EXTERN TABLEFRAME(REAL,REAL,REAL,REAL,REAL)
This instruction must only be written once before using the macro; it is recommended that you place the instruction at the beginning of your part program.

If you do not want to write the previous definition into each program, you can transform the macro into a canned user cycle. In this case, please contact us at *info@cncofcourse.com* or contact your technical support service.

PROGRAMMING

The macro is programmed by writing the subprogramme call TABLEFRAME followed by the parameters in the parenthesis. The basic rules of Siemens cycles for defining parameters are applied.

- Each parameter has its own order:
TABLEFRAME(TablePos,WO_Dest,X0,Y0,Z0,X1,Y1,Z1,WO_Ori)

If you want to exclude a parameter, you still have to put in commas but no value.

Example:
TABLEFRAME(45,54,,,,,,55)

FOCUS PARAMETERS

- If all the final parameters are omitted, it is possible to omit the commas and close the cycle with the parenthesis.

Example:
TABLEFRAME(45,54)

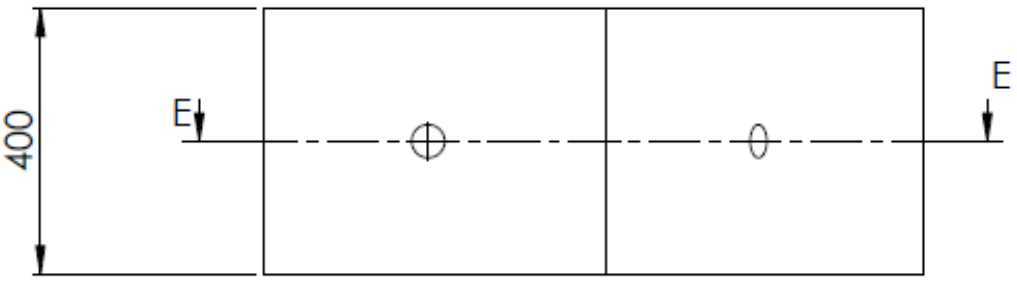
As indicated in the "Cycle description" section, the offset recalculation macro has two different applications: one for the creation of new offsets (case 1) and one for a TRANS function translation of the same offset in use (case 2). Let's now look at examples for both cases.

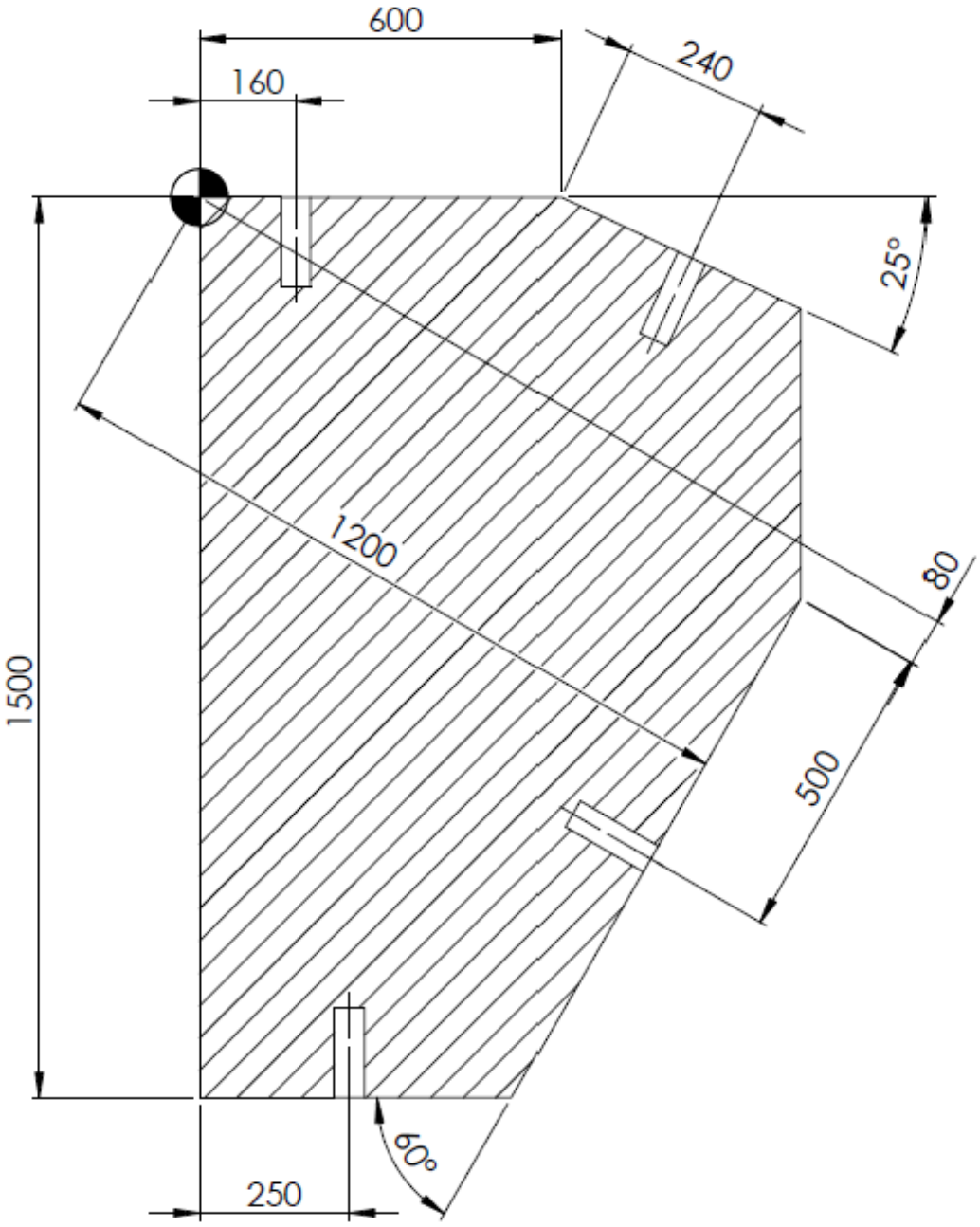
CASE1: Creating a new offset

Case 1 is mainly used when you want to have a separate offset for each table position, which is created in relation to the table centre by a reference. The benefit of case 1 is that you can act on one or the other offset to make corrections independently.

I will therefore have different offsets on which to intervene with modifications during the finishing phase to correct little imperfections linked to the table centre or thermal deformations.

Here is an example of programming the drilling of a workpiece with machining in several angular positions.





The offset marked with the symbol corresponds to the G54 offset manually set by the operator. The G54 offset corresponds to the B0 table position.

EXTERN TABLEFRAME(REAL,REAL,REAL,REAL,REAL,REAL,REAL, REAL,REAL)

G54;STARTING OFFSET
TABLEFRAME(-25,55,-600,-200,0,-240,0,0);G55 B-25
TABLEFRAME(-120,56,0,0,0,-580,-200,1200);G56 B-120
TABLEFRAME(180,57,0,0,0,250,-200,1500);G57 B180

T164
M6
D1 S1500 F450 M3

G54 G0 B0
TRANS X-160 Y-200
REPEAT AA1 BB1

;B-25
G55 B-25
REPEAT AA1 BB1

;B-120
G56 G0 B-120
REPEAT AA1 BB1

;B180
G57 G0 B180
REPEAT AA1 BB1
M30 ;END PROGRAM

AA1:
G0 X0 Y0
G0 Z150
CYCLE82(100,0,5,,50,0,0,1,11)
G0 Z150
G0 SUPA Z0 Y2500
BB1:

In the example, the reference origin was called in the program before the TABLEFRAME subprograms were called. If necessary, it can be inserted directly into the call parameters with the parameter WO_ORI.

EXTERN TABLEFRAME(REAL,REAL,REAL,REAL,REAL,REAL,REAL, REAL,REAL)

TABLEFRAME(-25,55,-600,-200,0,-240,0,0,54); G55 B-25
TABLEFRAME(-120,56,0,0,0,-580,-200,1200,54) ;G56 B-120
TABLEFRAME(180,57,0,0,0,250,-200,1500,54) ;G57 B180

We recommend enclosing the entire part of the program in which the new offsets are generated in a conditional jump or even in a separate subprogram, so that they are not always recalculated and then rewritten with new values during the block search. By creating a separate program, the operator will run this subprogram before the machining and then launch the machining program.
If, on the other hand, you wish to include the offset generation within the main program, it is suggested that you insert conditional jumps.

Example of jump in block search and simulation, using system variables:
If $P_Search AND SP_Sim GOTOF END_CALC
TABLEFRAME(-25,55,-600,-200,0,-240,0,0,54) ;G55 B-25
TABLEFRAME(-120,56,0,0,0,-580,-200,1200,54) ;G56 B-120
TABLEFRAME(180,57,0,0,0,250,-200,1500,54) ;G57 B180
END_CALC:

Example of jumping with user variable:
R99=1
IF R99==1 GOTOF END_CALC
TABLEFRAME(-25,55,-600,-200,0,-240,0,0,54) ;G55 B-25
TABLEFRAME(-120,56,0,0,0,-580,-200,1200,54) ;G56 B-120
TABLEFRAME(180,57,0,0,0,250,-200,1500,54) ;G57 B180
END_CALC:

The operator will only manually set the variable R99=0 the first time or at any time he wishes to recalculate the origins.

CASO2: Traslazione origine TRANS

With case 2, the TABLEFRAME cycle translates the active offset of the values required to follow the table rotation. These values are applied within the cycle using the TRANS function. If it is necessary to use the TRANS function for machining, use the additive version ATRANS. Remember to deactivate the TRANS function used within the cycle at the end of the machining process. To do this, program TRANS without any parameters.

Example 1
T1
M6

D1 S1500 F450 M3
G0 Z300
G54 G0 B0
;Machining

TABLEFRAME(45)
G0 B45
;Machining
TRANS

Example 2: Machining repeated 10 times with an angular pitch of 15°.

R1=0
STEP:
R1=R1+15
TABLEFRAME(R1)
G0 B=R1
;Machining
REPEAT STEP P9
TRANS

WARNINGS

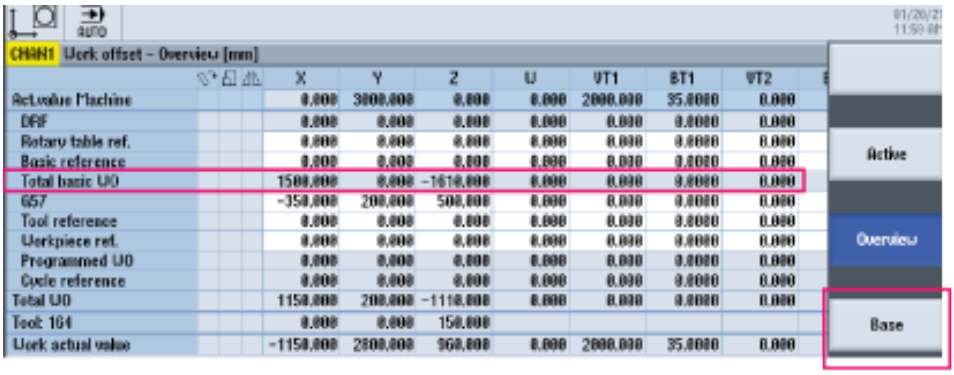
1. INTERNAL PARAMETERS
Within the subprogram TABLEFRAME.SPF that will be sent to you, you will find in the initial blocks the following part of the program, the meaning of which is described in the following points.

```
;*********************************************
;*****INTERNAL PARAMETER*******************

CP127=0        ;TABLE CENTER X
CP128=0        ;TABLE CENTER Y
CP129=0        ;TABLE CENTER Z
CP130=1        ;ROT TABLE CONVENTION 1=ISO -1=NOT ISO
CP131=0        ;MACHINE TYPE 0=ORIZONTAL X-Z   1=VERTICAL X-Y
_ROT_TABLE="BT1"    ;TABLE AXIS NAME

;*****END INTERNAL PARAMETER***************
;*********************************************
```

2. TABLE CENTRE DEFINITION
The TABLEFRAME cycle provides two options for setting the table centre: by entering the table centre positions along X, Y and Z directly inside the subprogram with parameters CP127, CP128 and CP129. The second option is to define the table centre with the basic frame, leaving the previous parameters set to zero.



3. The macro is delivered set up for a machine where the Z axis is horizontal, so the coordinates to be rotated are X and Z. If your machine has a vertical Z-axis, simply set the internal parameter CP131=1 and the macro will count the new offset with X and Y.

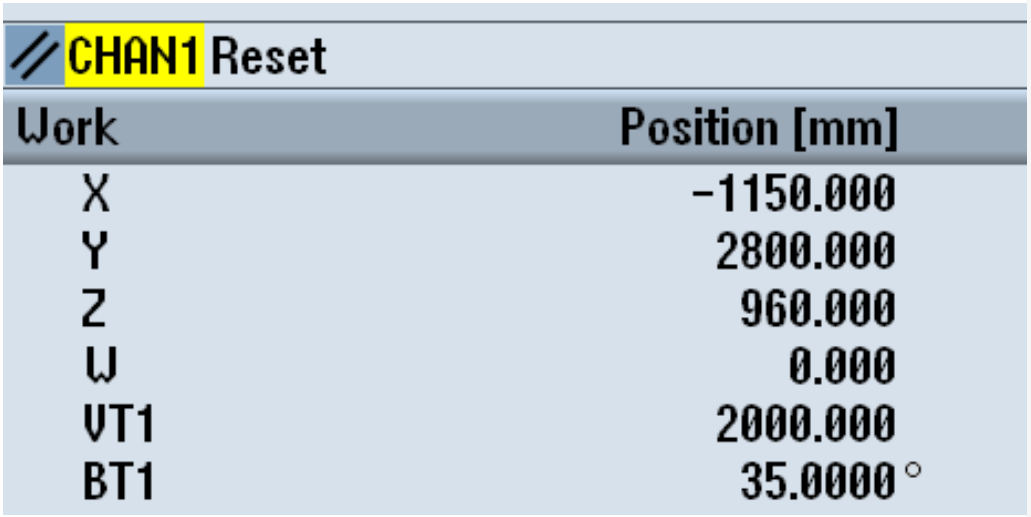
4. The macro is delivered with table degrees that comply with ISO standards; however, it may happen that the table rotation does not match. In this case, set parameter CP130 to -1.

FOCUS PARAMETERS

5. In the macro, define the name of the axis of the rotary table. If, for example, the dimensions displayed for the table correspond to the B axis, set the internal parameter _ROT_TABLE to B by putting the axis address between the quotation marks:

Example 1:
_ROT_TABLE=”B” ;TABLE AXIS NAME

Example 2:



_ROT_TABLE=”BT1” ;TABLE AXIS NAME

6. It is recommended that, when using the macro for the first time, the recalculated origins should always be checked to ensure that the table centre positions and the macro calculations are correct.

7. The cycle is supported by all SIEMENS 840D numerical control versions **after version 4.4.**

www.cncofcourse.com