ON CHANGE

ON CHANGE action

Declaration

```
ON  CHANGE  objIdent  GOTO ProcName [NONE]
```

Parameters

objtIdent	in	Value identifier (or a group of values).
ProcName	in	Procedure name (it must meet the rules for object name).

Description

The action specifies a procedure (using the name ProcName), that is to be called, if a change of the value *objIdent* (or some of the group of values) occurs. A value must be derived from the object name (not from local variable). For example correctly: Sec, SV.Structure[2]^Int, incorrectly: _locVar, ...).

A group of values can represent a row, a column or entire object of Structured variable type.
For example:

- the identifier of the second row: SV.Structure[2] (column name is not stated)
- the identifier of the column Int: SV.Structure[0]^Int (the row index is 0)
- the identifier of all values (entire structure): SV.Structure, or SV.Structure[0] (the row index is 0).

The traced value can be identified through HOBJ or VOBJ.

For example:

```
INT _hbj
...
ON CHANGE (_hbj) GOTO ValueChanged
```

After the action execution, for example:

```
ON  CHANGE Sec GOTO ValueChanged
```

The procedure *ValueChanged* is to be called always, when a value change of the object *Sec* occurs (therefore every second). If the following action is performed within the frame of the script:

```
ON  CHANGE Sec GOTO ValueChanged NONE
```

calling the procedure *ValueChanged* is to be terminated during a value change of the object *Sec*. At the action notation, it is able to state the structure item:

```
ON  CHANGE SV.Structure[2]^Int GOTO ValueChanged
```

We may respond to a change of one value also in several procedures:

```
ON  CHANGE Sec GOTO ValueChanged
```

```
ON  CHANGE Sec GOTO ValueChanged1
```

In this case, the procedures *ValueChanged* and *ValueChanged1* are to be called during a change of the object *Sec*.

A procedure called on the basis of a value change must be declared in the following way:

```
PROCEDURE ProcName(ValueType value, ALIAS _referenceToObject, INT _row,
INT _column)
```

Where:
ProcName is the procedure name, that meets the rules for procedure name.
_value is a parameter, to which a new value of the monitored object (or an item during the monitoring of a structure item change).
ValueType determines the type of the parameter _value. If the type is not selected correctly (in regard to a value changed), an invalid value is to be assigned to the parameter.
_referenceToObject is a parameter of ALIAS type. After the procedure start, it is initialized to refer to the object, which changes the value (in case of monitoring an item, ALIAS is referred to the whole structure, not the item).
_row, _column are the parameters of INT type. They make a difference only if the procedure is called by reasons of a value change of a structured variable item. Otherwise they are set to the value of 0. In case of a value change, the parameter *_row* contains the serial row number and the parameter *_column* contains the serial column number within the frame of the particular structured variable.

The action is usable only within the frame of the script of an Active picture or an object of Event type with the active option Server Event.

Example

Example - the script, within the frame of an active picture, that monitors and reports all value changes of:

- the object *Sec*
- the item *SV.Structure^Int*
- any item in the row *Int* (SV.Struktura[0]^Int)

to an operator on the desktop of the process D2000 HI.

```
; Handling changes of values

 PROCEDURE ValueChanged(REAL _value, ALIAS _obj, INT _row, INT _col)

   ; Is a value of the object Sec changed ?
   IF _obj\HBJ = Sec\HBJ THEN
     MESSAGE "The object Sec changed = " + %IToStr(_value) ON _FROM_HIP

   ; Is a value of the object SV.Structure changed ?
   ELSIF _obj\HBJ = SV.Structure\HBJ THEN

     IF _row = 1 & _col = 1 THEN
       MESSAGE "The item SV.Structure^Int changed = "  + %IToStr(_value)
ON _FROM_HIP
      ELSE
       MESSAGE "In the column SV.Struktura[0]^Int, change of the row = " +
%IToStr(_row)  ON _FROM_HIP
     ENDIF
   ENDIF
 END ValueChanged

 ; Initialization part
 BEGIN
   ON  CHANGE Sec GOTO ValueChanged
   ON  CHANGE SV.Struktura^Int GOTO ValueChanged
   ON  CHANGE SV.Struktura[0]^Int GOTO ValueChanged
 END
```

Related pages:

Script actions