

VIII.3.3-RES-SNGL-PO3 SINGLE RESERVOIR REGULATION OPERATION IF EXPRESSION STORAGE FORMAT

This Section describes the format that the groups of IF EXPRESSION used with RCL IF and ELSEIF keywords is stored in the PO array for Operation RES-SNGL.

The pointer to the starting location in the PO array of the IF EXPRESSION information is held in word 14 of the PO array. A value of zero indicates that no IF EXPRESSION is used in the RCL.

The first word in this section of the PO array is the number of the IF EXPRESSION groups that are used within the RCL. The remainder of the section is filled with all IF EXPRESSIONS stored in Reverse Polish Notation which will be explained later.

The basic form of an IF-ENDIF block is:

```
IF (expression) THEN (statement_group)
ENDIF
```

where (expression) is a logical expression that has a result of either 'true' or 'false' and
(statement_group) is a DO statement followed by one or more DO statements and/or IF-ENDIF blocks

An evaluation of 'true' for the (expression) results in the statements between the THEN and ENDIF to be executed. A 'false' result ignores those statements and executes the statement following the ENDIF statement.

The checks that can be made in an IF-ENDIF block use two types of comparisons:

- o relational expressions
- o logical variables

A relational expression compares two quantities, whereas a logical variable by itself has a 'true' or 'false' value.

A relational expression takes the form:

```
<system_variable><relational_operator><user_variable>
```

The <system_variable> can be one of the following comparator codes:

1. QO - instantaneous discharge
2. QOM - mean discharge
3. QI - instantaneous inflow
4. QIM - mean flow
5. POOL - pool elevation
6. STORAGE - storage contents
7. DAY - relative Julian date

The <relational_operator> can be one of the following logical

operator codes:

1. .EQ. - left and right sides are equal
2. .NE. - left and right sides are not equal
3. .LT. - left side is less than the right side
4. .GT. - left side is greater than the right side
5. .LE. - left side is less than or equal to the right side
6. .GE. - left side is greater than or equal to the right side

The <user_variable> can be one of the following:

1. Numerical value
2. SET variable name defined prior to the IF-ENDIF block in a SET statement
3. System function name

Numerical values can be either integer or real. Set variables are described in Section V.3.3-RES-SNGL. There are two system functions:

1. RULE - current rule curve elevation
2. MAXQ - maximum discharge that can be released at current elevation

These functions can be used as their actual values or modified by adding, subtracting, multiplying by or dividing by a factor. For example, if the pool elevation is to be checked against the rule curve elevation plus an additional amount, the relational expression would look like:

```
(POOL.LE.RULE+2.0)
```

Examples of relational expressions are:

1. Comparing the instantaneous outflow against exceeding the constant value 50000:

```
(QO.GT.50000.0)
```

2. Comparing the mean discharge for being less than a SET variable named MAXMEAN:

```
(QOM.LT.MAXMEAN)
```

3. Comparing the relative Julian day to the first of February:

```
(DAY.GT.31)
```

4. Comparing the pool elevation against rulecurve elevation using RULE function:

```
(POOL.LE.RULE+2.0)
```

Each relational expression is stored in the PO array in Reverse Polish Notation in numerical code.

```
<system_variable> <user_variable> <relational_operator>
```

where <system_variable> is one of the comparator codes

<user_variable> a negative pointer to the USER VARIABLE section (see Section VIII.3.3-RES-SNGL-PO4) of the PO array where values are to be used to compare against system variables

<relational_operator> is a code representing one of the logical operators

The phrase, A-B-OP is known as Reverse Polish Notation (RPN), where B is compared against A with operator OP. It should be noted that the <user variable> can assume many forms, so that to achieve consistency a separate section of the PO array is set aside to hold information on each occurrence of <user variable>. Within each phrase is just a pointer to the start within the user variable section of the PO array. It should also be noted that this pointer is stored as a negative value and just be made positive to achieve the proper pointer value.

More details on the types and storage formats of the user variables are provided in Section VIII.3.3-RES-SNGL-PO4.

The codes for the system variables are:

<u>System Variable</u>	<u>Numerical Code</u>
QO	501.01
QI	502.01
POOL	503.01
STORAGE	504.01
DAY	505.01
QOM	506.01
QIM	507.01

The numerical codes for the relational operators are:

<u>Relational Operator</u>	<u>Numerical Code</u>
EQ	169.01
NE	168.01
LT	167.01
GE	166.01
LE	165.01
GT	164.01
AND	149.01
OR	139.01

The examples given above are stored as follows:

1. QO.GT.50000.0
501.01 -(user variable section pointer of 50000.0) 164.01
2. QOM.LT.MAXMEAN
506.01 -(user variable section pointer of MAXMGAN) 167.01

3. DAY.GT.31
505.01 -(user variable section pointer of 31) 164.01
4. POOL.LE.RULE+2.0
503.01 -(user variable section pointer of RULE+2.0) 165.01

A logical variable represents the state (either 'on' ('true') or 'off' ('false')) of various hydrologic and timing conditions. The variable by itself constitutes a valid logical expression. The variables available and their 'on' conditions are:

<u>Logical Variable Name</u>	<u>'On' Condition</u>
FORECAST	Run is preceding or on last period of observed data; for ESP runs this variable is always 'on'
OBSERVED	Run is until past last period of observed data
FLOOD	Utility SUMINF has been executed and 'flood' criteria have been exceeded (is 'false' if no SUMINF has been executed prior to test)
SURCHARGE	Utility ENTERISC has been executed and indicates that included surcharge is needed ('false' if ENTERISC not executed)
GOFLASH	Utility GOFLASH has been executed and indicates that flashboard scheme is needed ('false' if GOFLASH not executed)
RISING	Pool elevation is rising from previous period to this period
FALLING	Pool elevation is falling from previous period to this period
NFLOOD	Opposite of FLOOD
NSURCHARGE	Opposite of SURCHARGE
NGOFLASH	Opposite of GOFLASH

The numerical codes for the logical variables are:

<u>Logical Variable</u>	<u>Numerical Code</u>
FLOOD	551.01
SURCHARGE	552.01
FORECAST	553.01
GOFLASH	554.01
NFLOOD	555.01
NSURCHARGE	556.01
NFORECAST	557.01
NGOFLASH	558.01

RISING	559.01
FALLING	560.01

Multiple relational expressions and/or logical variables can be combined in one logical expression by the logical operations .AND. and .OR.. For example,

```
(QO.GT.MAXQ.AND.RISING)
```

is a valid expression.

Subexpressions (less than the entire logical expression) can be isolated by parentheses to ensure proper order of evaluation. For example,

```
((QO.GT.MAXQ.OR.QI.GT.10000.)AND.FALLING)
```

In this example, the combination surrounding the .OR. will be evaluated and that result used with the .AND. operator and the FALLING variable to get the final expression result.

The hierarchy of evaluation (in the order of done first to done last) is:

- o Expression in parentheses
- o .AND. separated expressions
- o .OR. separated expressions

All operators of equal hierarchy are evaluated from left to right.

Phrases can be coupled in the RPN arrangement as follows:

```
<phrase> <phrase> <link>
```

where <link> is one of the logical operators, AND or OR

Within the IF expression section codes are stored for <logical_variables>, <system_variables>, <relational_operators> and <links>.