

Unit 7. Control Building SCMs

- **Lesson 1. Sequential Control Modules**
- **Lesson 2. SCM Activation**

Lesson 1. Sequential Control Modules (SCMs)

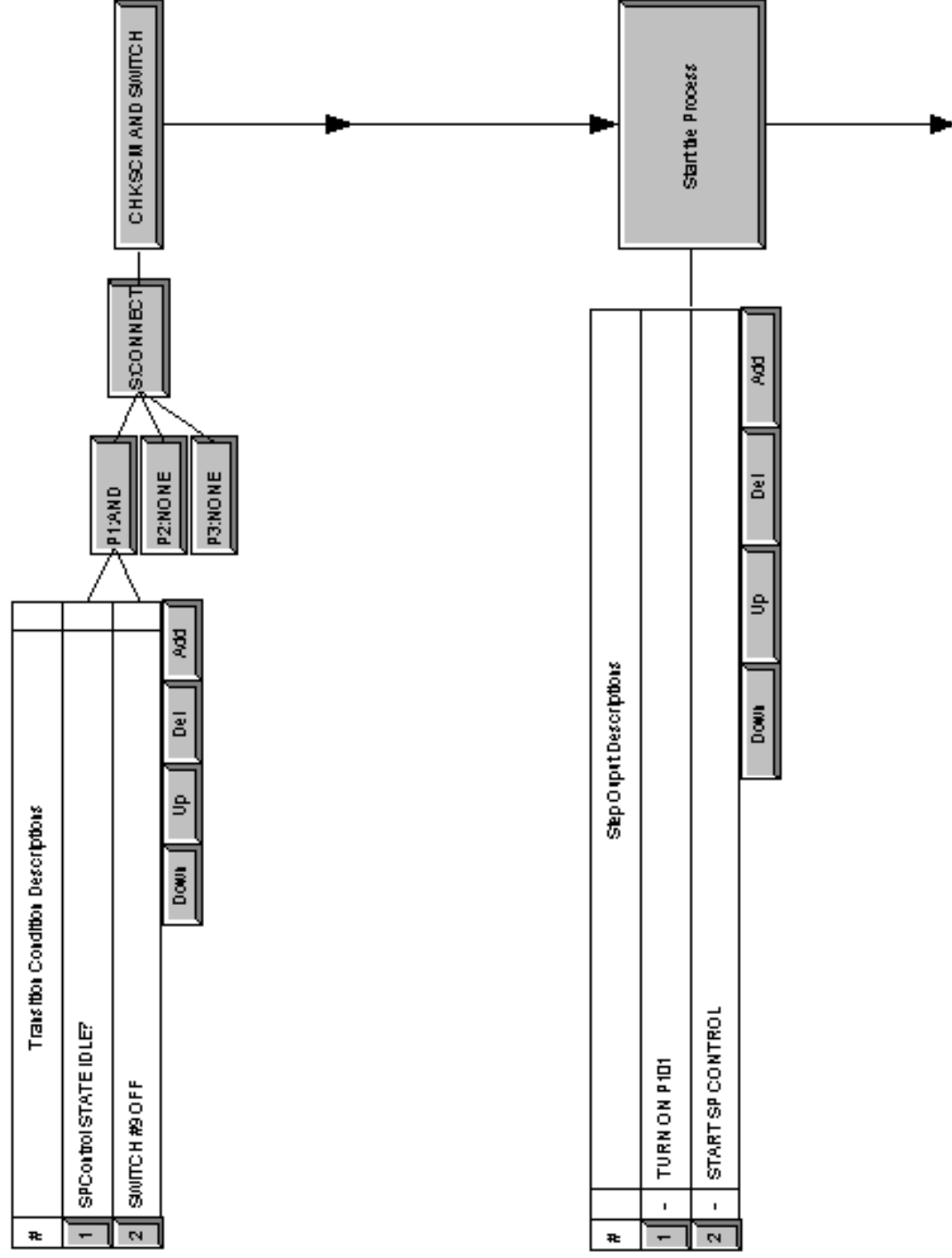
Objectives:

- **Build and edit SCMs**
- **Configure step and transition function blocks**
- **Configure step timing**
- **Add looping and branching functionality**
- **Create exception handler**

SCMs

- provide phase-level batch functionality
- command regulatory and discrete function blocks contained within CMs in a defined order to accomplish higher level tasks
 - for example, boiler start up
- contain the following function blocks:
 - transitions that determine when to proceed
 - steps that take action
 - handlers for specified exceptions

SCM Example



SCM Transitions

- evaluate if the SCM is ready to advance to next step
- define up to 10 conditions
- define up to 4 logic gates
 - 3 primary
 - 1 secondary

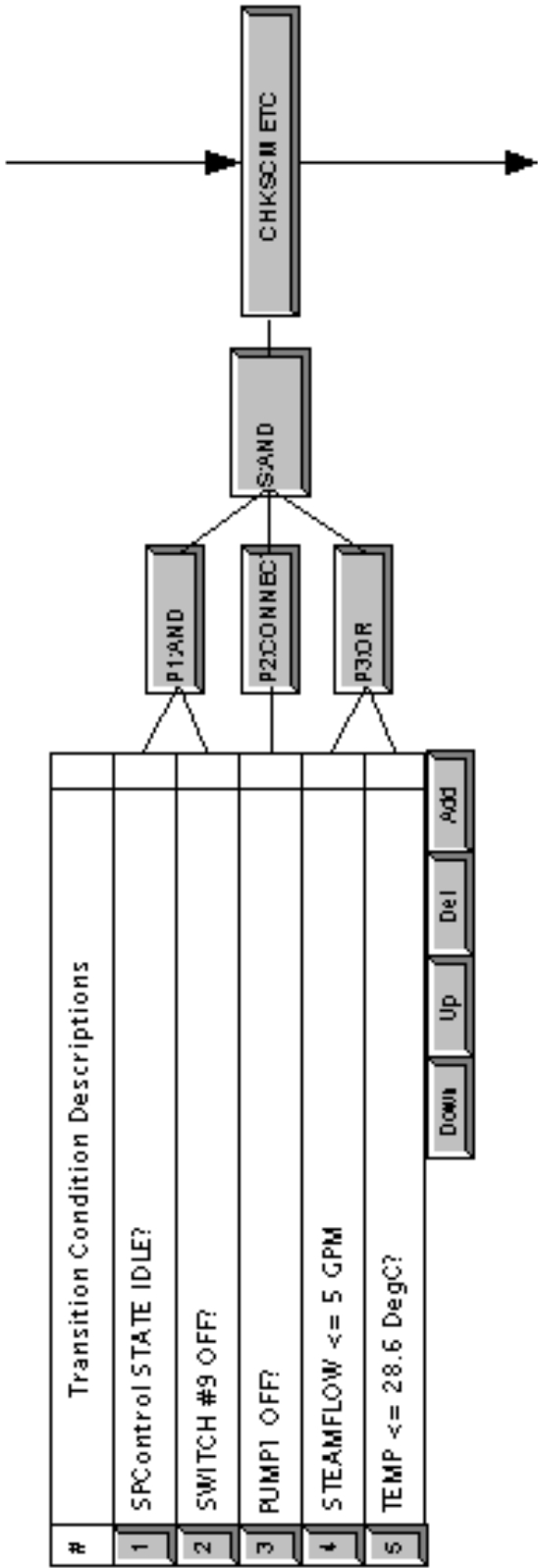
Transition Conditions

- true=on=1 false=off=0
- can be forced on or off from Monitor tab
- compound statements with OR, AND logic permitted

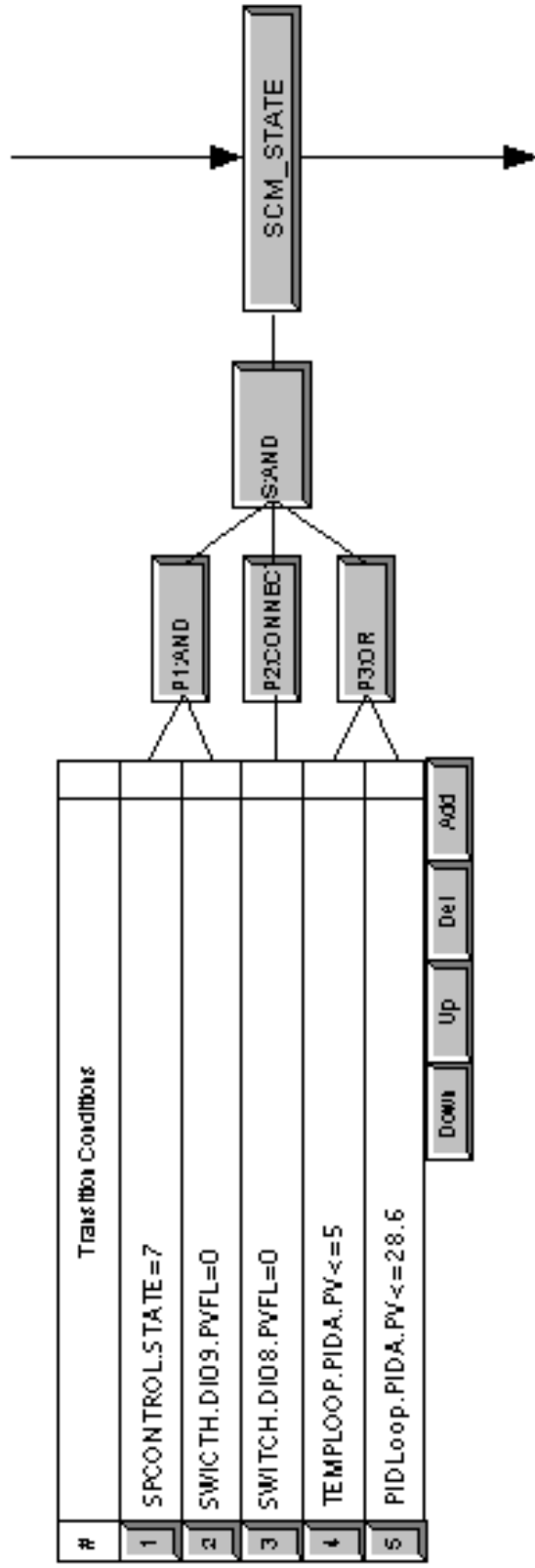
Examples:

- PIDLOOP.PIDA.OP >= 15.
- SWITCH.DI07.PVFL = 0
- 1
- SWITCH.DI07.PVFL=0 and PIDLOOP.PIDA.SP <= 5.

SCM Transition Conditions -- Descriptions



SCM Transition Conditions -- Code



SCM Steps

- send output/actions to CM function blocks (CM Mode attribute must be Program)
- can define up to 16 individual outputs/actions
for example, PIDLOOP.PIDA.MODE:=1
 - instructs PIDA to go to AUTO mode
 - PIDA is a PID function block contained in the PIDLOOP CM
- support min and max wait times
- indicate whether the output action is allowable

Step Outputs

- must have assignment
- only one assignment per output
- cannot leave blank
- assignment designator is :=

Examples:

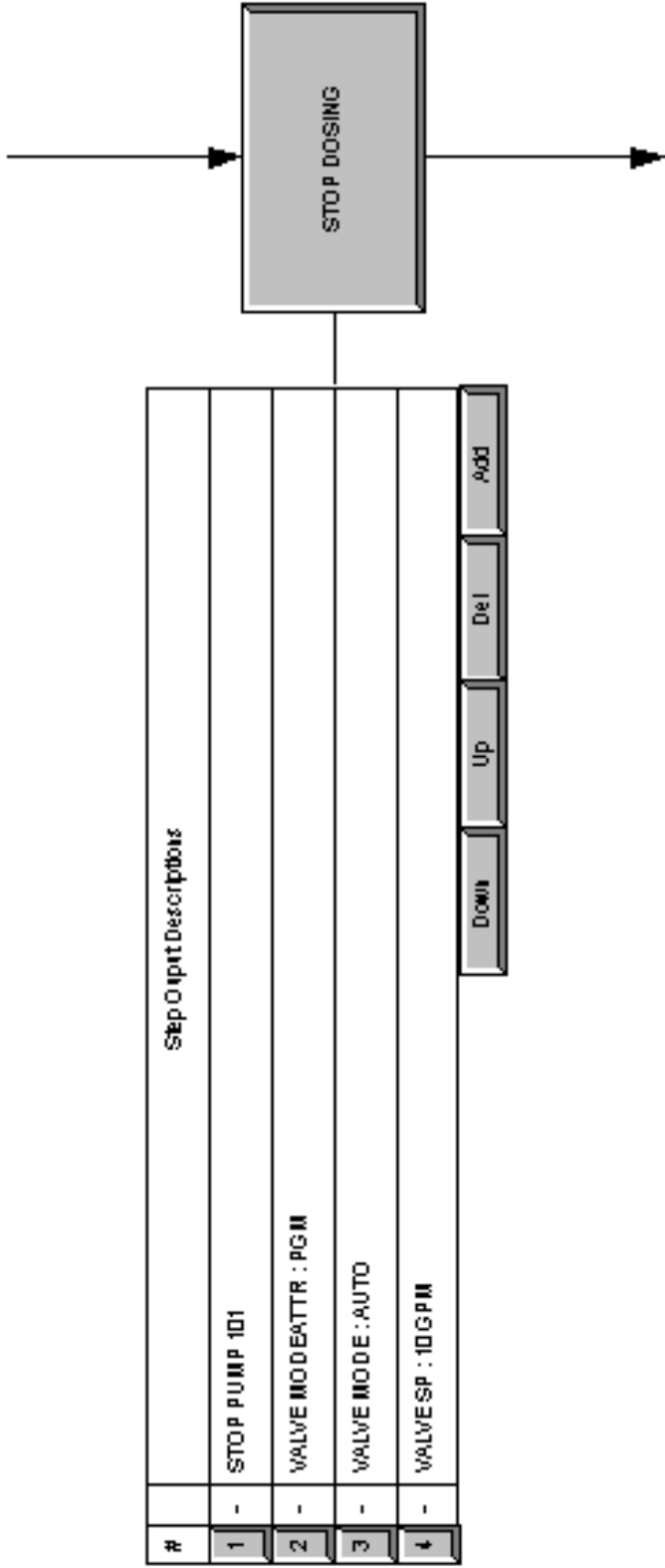
PIDLOOP.PIDA.MODE := 1

PIDLOOP.PIDA.SP := 7.

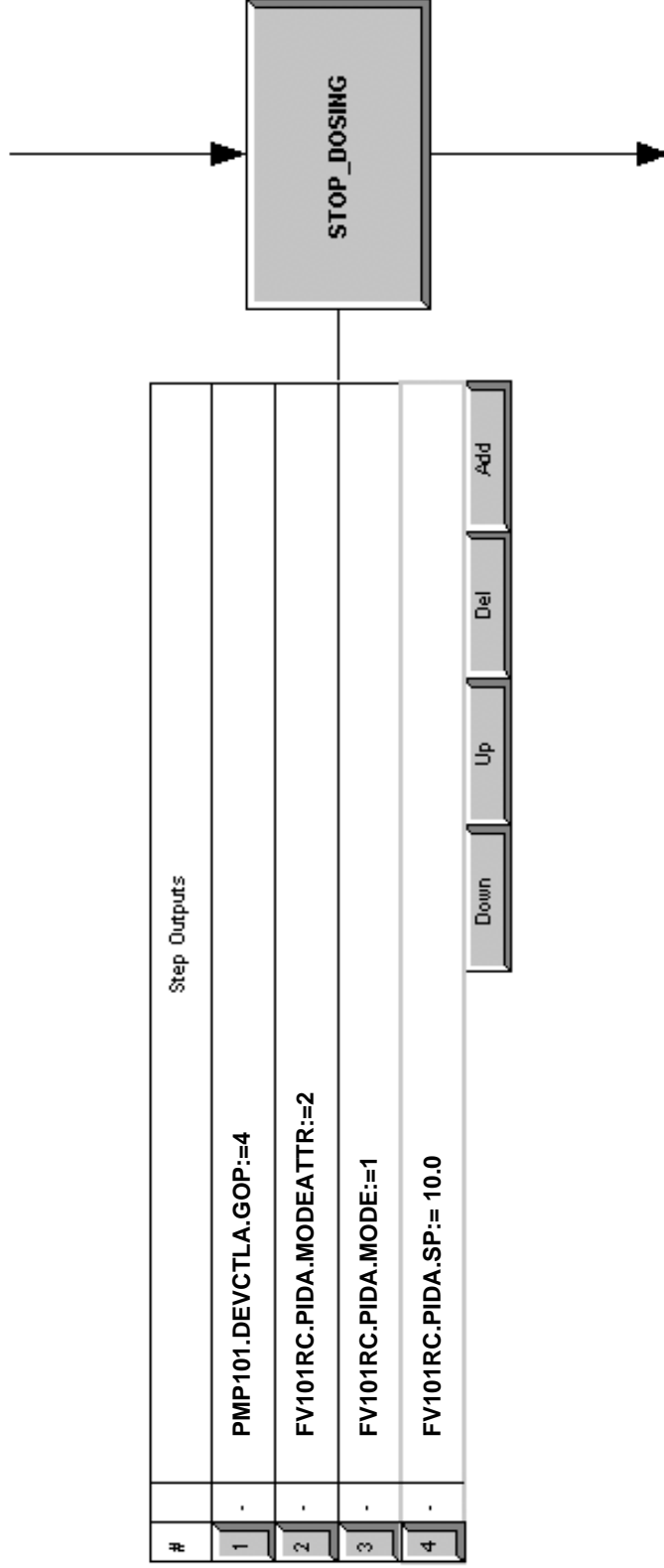
SCM27.STEP8.MINTIME := 12

SCM27.STEP8.MAXTIME := INT (SCM27.RECTARGET[6])

SCM Output Descriptions



SCM Output Code



Rules for Defining an SCM

Rule 1. Every SCM begins with an Invoke Transition.

Rule 2. SCMs can have multiple endings.

Rule 3. Steps and transitions must alternate in an SCM.

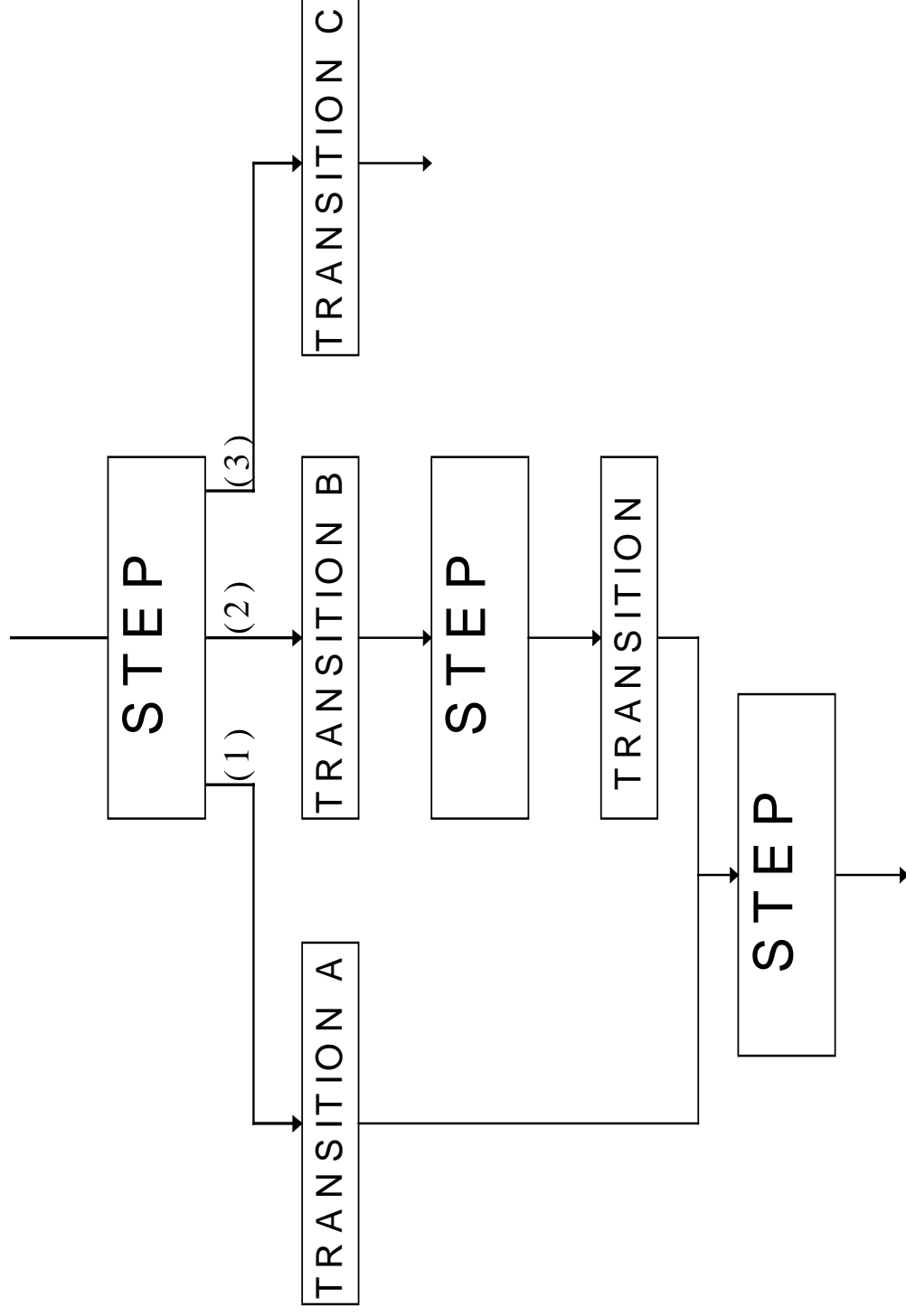
SCM Modes

- Three execution modes are available
 - Automatic, Semi-automatic, and Single step
 - Automatic : Run to Complete , no normal interruptions
 - Semi-automatic : Pause after each step, wait for Resume command
 - Single step: Pause after each step, wait for Resume command and target step (ignores transition conditions)
- Manual Mode: stops SCM execution (normally used with TPB to link SCM execution with Recipe execution.
- Normal Mode: used with Operator or GUS keyboard to go to configured execution mode

SCM Branching

- Operation branches at Step, not Transition
- Up to 10 branches per step
- Outputs evaluated in order
- Execution goes to first transition that is true
- Branching, NOT parallel execution

Example of SCM Branching



Exception Handlers

- transitions and steps that run only after specified conditions have been met
 - SCMs can have multiple handlers
- handler types, in priority order:
 - Abort
 - Stop
 - Hold
 - Restart
 - Interrupt
 - Check
 - Main

SCM Recipe Data

- maximum of 50 recipe values per SCM
- data
 - descriptors
 - SP, PV access locks
 - scaling options
 - high and low limits
 - default values
 - display and print options

SCM History Data

- maximum of 50 history parameters per SCM
- store process data collected during SCM operation
- parameters
 - description
 - variable type
 - value

Additional SCM Code Examples

- **SCM235.HISTVALUE[1] := (CM151.PIDA.PV + CM251.PIDA.PV) / 2**
- **MESSAGES.XFERB.SENDFL[3] := 1**
- **ENBHANDLER[8] := 2**
- **CM235.PIDA.SP:=AVG (MIN(CM1.REGCALC.X[1], CM1.REGCALC.X[2], CM1.REGCALC.X[3]), MAX(CM1.REGCALCX[1], CM1.REGCALC.X[2], CM1.REGCALC.X[3]))**
- **CM456.PIDA.SP := (CM456.SPPEC1FLAG.PVFL) ? SCM457.RECTARGET[1] : SCM457.RECTARGET[2]**