

***Design
Specifications***

4410S

Copyright, Notices, and Trademarks

© Copyright 1993, 1997, 1999 by Honeywell Inc.

Revision 04 – September 28, 1999

Honeywell IAC courseware is subject to change without notice.

FLEXTRAINING™ courseware is copyrighted and all rights are reserved by Honeywell Inc. These materials are intended for use solely in conjunction with Honeywell products. The materials comprising the courseware may not, in whole or in part, be copied, photocopied, reproduced, translated or reduced to any electronic medium or machine readable form without the prior, express written consent of Honeywell Inc.

This module supports **TotalPlant** Solution (TPS) system network.

FLEXTRAINING and **TotalPlant** are US registered trademarks of Honeywell Inc.

TPS is the evolution of TDC 3000^X.

Other brand and product names are trademarks of their respective owners.

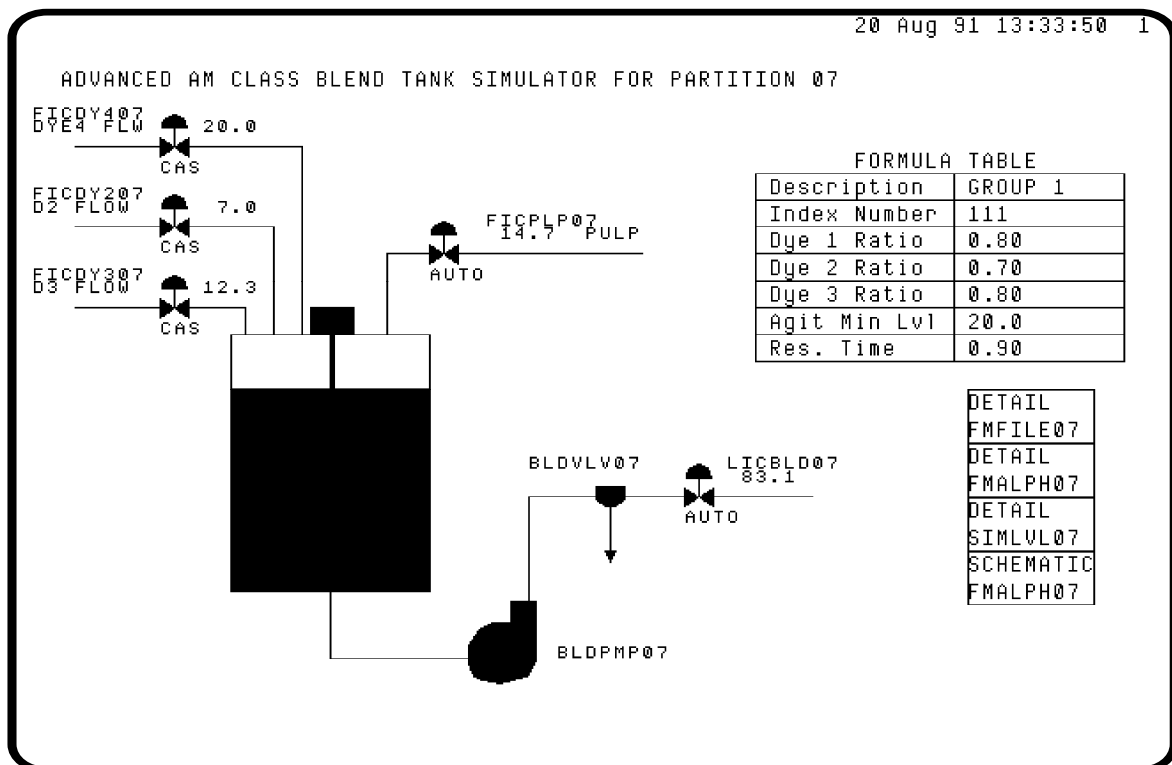
Honeywell
Industrial Automation and Control
Automation College
2820 West Kelton Lane
Phoenix, AZ 85053
1-800-852-3211

BLEND TANK SIMULATOR

1. OVERVIEW

The process being simulated is a continuous operation. Base pulp and three dyes are metered into a blend tank. The feed rate of the dyes are determined by throttling control valves whose setpoints are ratioed to the base pulp setpoint using data supplied by an operator-loaded formula. In the normal running mode, the mix is pumped out of the tank through a control valve at a rate proportional to the "residence time" loaded with the formula. In the drain mode, a three-way valve opens and allows the mix to be pumped out of the tank to the drain at a fixed rate. The tank agitator turns on automatically when the level reaches a minimum value set by the formula.

PROCESS SCHEMATIC



BLEND TANK SIMULATOR

2. FUNCTIONAL DESCRIPTION

The flow control valves are simulated by AM Regulatory points with PID algorithms, configured so that their PV source is their own output. The dye flow controllers' setpoints are ratioed to the setpoint of the pulp flow controller. The ratio values are downloaded from the Formula Manager application.

The level in the tank is simulated by an AM Regulatory point, whose PV is calculated in an AM Custom point. The calculation for normal-running operation is as follows:

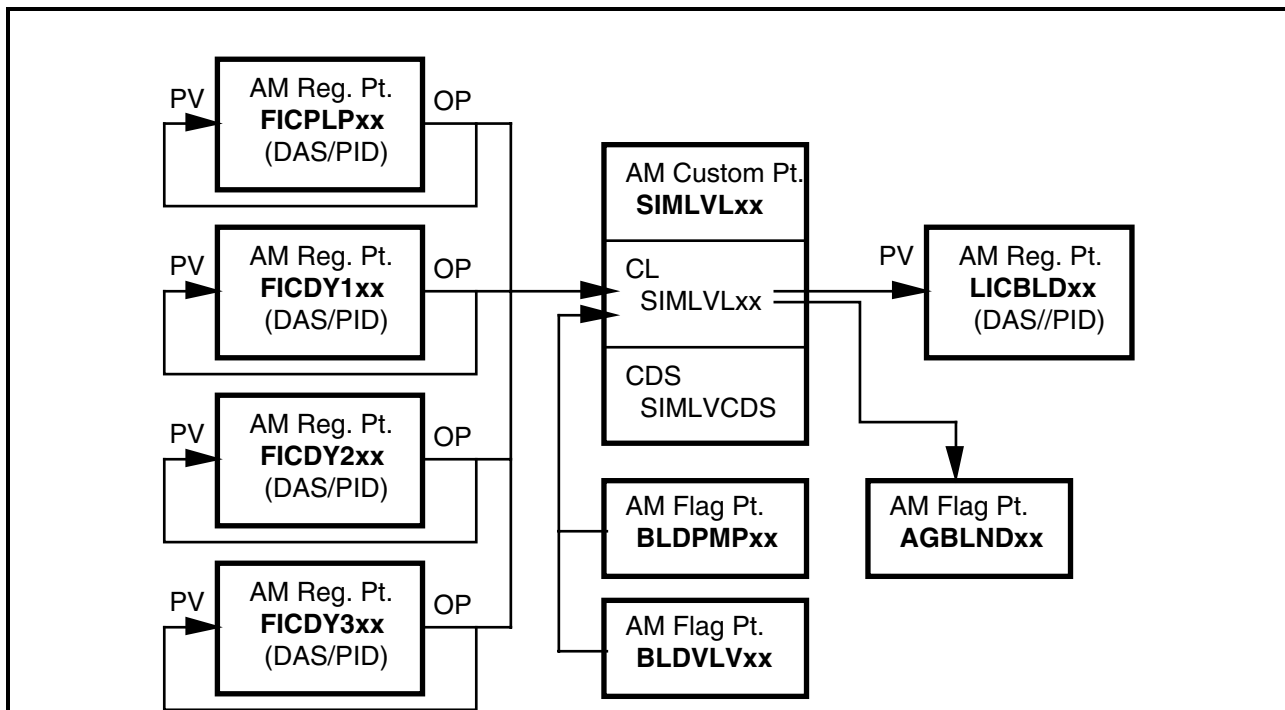
$$\text{Tank_level.PV} = \text{Tank_level.PV} + C * (\text{Pulp.OP} + \text{Dye1.OP} + \text{Dye2.OP} + \text{Dye3.OP} - \text{Tank_level.OP})$$

The calculation for draining is as follows:

$$\text{Tank_level.PV} = \text{Tank_level.PV} - 10.0$$

The tank agitator, pump 3-way valve and the pump are simulated by AM flag points.

FUNCTIONAL BLOCK DIAGRAM



Note: replace xx with your partition number

BLEND TANK SIMULATOR

3. FLAG POINT SPECIFICATIONS

The Flag points listed below reside in the AM.

TAGNAME	DESCRIPTOR	KEYWORD	UPPER BOX	LOWER BOX
BLDPMPxx	BLND TK PUMP STATUS	PUMP STS	RUN	STOP
BLDVLVxx	PUMP 3WAY VALVE POS	VLV POS	RUN	DRAIN
AGBLNDxx	BLEND TANK AGIT STS	AGIT STS	ON	OFF

4. REGULATORY POINT SPECIFICATIONS

The following regulatory points reside in the AM. They are processed on a 5-second schedule. They have Data Acquisition (DAS) PV algorithms and PID control algorithms. The PV inputs are configured as shown in the table below. All will have REVERSE control action, except the level controller, which is DIRECT action. The setpoints of the dye controllers are all calculated from the setpoint of FICPLPxx by a CL Block, so their normal mode will be CAS. The normal mode for the rest of the controllers is AUTO. For the smoothest operation, LICBLDxx is processed before the flow controllers.

TAGNAME	DESCRIPTOR	EUDESC	KEYWORD	EU RANGE	PV INPUT
FICPLPxx	PULP FLOW CONTR	GPM	PLP FLOW	0-100	FICPLPxx.OP
FICDY1xx	DYE 1 FLOW CONTR	GPM	D1 FLOW	0-100	FICDY1xx.OP
FICDY2xx	DYE 2 FLOW CONTR	GPM	D2 FLOW	0-100	FICDY2xx.OP
FICDY3xx	DYE 3 FLOW CONTR	GPM	D3 FLOW	0-100	FICDY3xx.OP
LICBLDxx	BLND TK LVL CONT	% LEVEL	TANK LVL	0-100	SIMLVLxx.TREAL(7)

BLEND TANK SIMULATOR

5. POINT SIMLVLxx

This AM Custom point is processed on a 5-second schedule, after the level and flow calculations. It has the pre-built CDS Package SIMLVCDS attached to it, and a generic CL Block, SIMLVLxx, which is linked to the point and runs continuously.

CDS Pseudo-code

```
CUSTOM  -- file name SLEVLCDL

TSTRING(1..10) -- String array, used to store the formula name
TREAL(1..10)   -- Number array, used to store formula data and the
                -- results of the level calculation
RGPOINTS(1..5) -- Entity array, stores tagnames of regulatory
                -- points used for flow and level controllers
DCPOINTS(1..5) -- Entity array, stores tagnames of digital
                -- composite points
END CUSTOM
```

CL/AM Pseudo-code

BLOCK SIMLVLxx

If the pump valve is set to "drain" and the pump is running,
decrement the calculated tank level by 10%

If the pump is on and the pump valve is set to "run" then
use the following formula to calculate the level:

$$\text{Level.PV} = \text{Level.PV} + \text{res_time} * (\text{PULP.OP} + \text{DYE1.OP} + \text{DYE2.OP} + \text{DYE3.OP} - \text{LEVEL.OP})$$

Keep the calculated level in the 0-100% range

Set tank agitator on only if the level > agitator_min_level

Calculate and store new dye controller setpoints by multiplying
the pulp controller setpoint by the formulated dye ratios

END SIMLVLxx

BLEND TANK SIMULATOR

Parameter value assignments

TSTRING(1) FORMULA NAME - downloaded formula data
TREAL(1) FORMULA CODE - downloaded formula data
TREAL(2) DYE 1 RATIO - downloaded formula data
TREAL(3) DYE 2 RATIO - downloaded formula data
TREAL(4) DYE 3 RATIO - downloaded formula data
TREAL(5) AGITATOR MIN LEVEL - downloaded formula data
TREAL(6) RESIDENCE TIME - downloaded formula data
TREAL(7) CALCULATED LEVEL - input to LICBLDxx PV algorithm
RGPOINTS(1) ... Tagname of pulp flow controller
RGPOINTS(2) ... Tagname of dye 1 flow controller
RGPOINTS(3) ... Tagname of dye 2 flow controller
RGPOINTS(4) ... Tagname of dye 3 flow controller
RGPOINTS(5) ... Tagname of blend tank level controller
DCPOINTS(1) ... Tagname of blend tank pump
DCPOINTS(2) ... Tagname of pump 3-way valve
DCPOINTS(3) ... Tagname of blend tank agitator

FORMULA MANAGER

1. OVERVIEW

This application consists of AM points and parameters that hold formula data and a schematic display. When a particular formula is selected at the schematic, it can be downloaded to the AM Blend Tank Simulator point SIMLVLxx. The application also has the ability to accept the entry of new formulas into the file and to sort formulas alphabetically by name to make it easier for the operator to use.

2. FUNCTIONAL DESCRIPTION

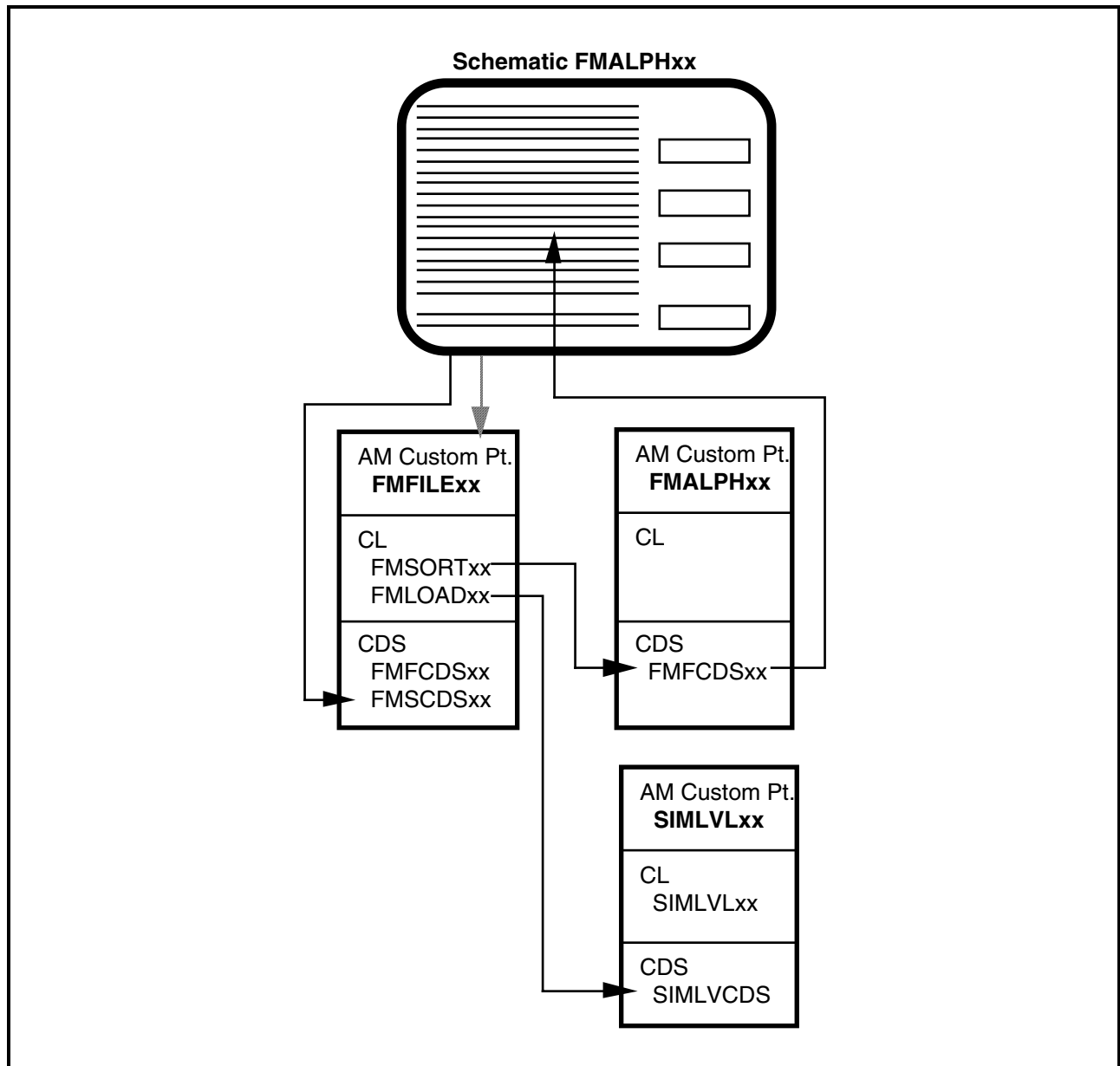
The AM Custom point FMFILExx has the CDS package FMFCDSxx attached to it., which contains parameters that will hold data for up to 20 formulas. A formula consists of a 3-digit index number, a 10-character name, three dye ratio values, an agitator minimum level value in percent and a residence time constant. The CDS package FMSCDSxx, also attached to the point, holds the currently-selected formula.

The CL Block FMSORTxx inserted at point FMFILExx adds newly-entered formulas to the first empty space in the file and then sorts all the formula entries alphabetically. Alpha-sorted data is then written to an identical CDS FMFCDSxx on Custom point FMALPHxx.

The CL Block FMLOADxx inserted at point FMFILExx writes the data for a selected formula to parameters on the point SIMLVLxx. FMLOADxx also causes a pre-built batch ticket to be printed out when new formula data is written. Pre-built schematic FMALPHxx displays the alpha-sorted formulas. It provides an interface for selecting, changing and adding formulas to the file (FMFILExx), as well as downloading a selected formula to the Blend Tank Simulator.

FORMULA MANAGER

FUNCTIONAL BLOCK DIAGRAM



FORMULA MANAGER

3. POINT FMFILExx

This AM Custom point is unscheduled. It is demand-processed from the Formula Manager schematic. It has CDS packages FMFCDSxx and FMSCDSxx attached to it. Two specific CL Blocks, FMSORTxx and FMLOADxx are inserted at the point. This point stores formula data as it is added from the schematic and performs the sort and load formula functions.

CL Pseudo-code

```
BLOCK FMSORTxx (specific, executes when TBOOL(1) is ON)

-- Check for sort only. This occurs when a formula read from ----
-- the History Module is done. This is indicated by a bad value ---
-- for the formula code in TREAL(1). This supports the Product --
-- Group Application in day 4 -----

IF TREAL(1) = bad value THEN goto begining of sort function

-- Store new entry in file -----

-- Store changed data if formula index number already exists

DO I = 1 to 20
  IF FMCODE(I) = bad value THEN repeat loop
  ELSE IF FMCODES(I) = TREAL(1) THEN
    FMNAMES(I) = TSTRING(1)
    FMDYE1R(I) = TREAL(2) -- allow for bad values
    FMDYE2R(I) = TREAL(3) -- allow for bad values
    FMDYE3R(I) = TREAL(4) -- allow for bad values
    FMMINLV(I) = TREAL(5) -- allow for bad values
    FMRESTM(I) = TREAL(6) -- allow for bad values
    goto NEXT1
  repeat

-- Else store new formula at first empty location

DO I = 1 to 20
  IF FMCODES(I) = bad value THEN
    FMCODES(I) = TREAL(1)
    FMNAMES(I) = TSTRING(1)
    FMDYE1R(I) = TREAL(2) -- allow for bad values
    FMDYE2R(I) = TREAL(3) -- allow for bad values
    FMDYE3R(I) = TREAL(4) -- allow for bad values
    FMMINLV(I) = TREAL(5) -- allow for bad values
    FMRESTM(I) = TREAL(6) -- allow for bad values
  goto NEXT1
repeat
```

FORMULA MANAGER

```
-- Name Sort Function (bubble sort) -----
-- Start of the Sort Function -----

-- Get names from CDS array into local array variable

NEXT1: DO I = 1 to 20
    NAMES(I) = FMNAMES(I)
    repeat

-- Init the sort, must use lower case z

DO I = 1 to 20
    SCODES(I) = COUNTER
    IF NAMES(I) = "" THEN NAMES(I) = "zzzzzzzzzz"
    repeat

-- Sort on formula names

S1: change_flag = OFF

DO I = 2 to 20
    IF NAMES(SCODES(I-1)) > NAMES(SCODES(I)) THEN
        change_flag = ON
        temp = SCODES(I)
        SCODES(I) = SCODES(I-1)
        SCODES(I-1) = temp
    repeat

IF change_flag is ON then goto S1

-- Store name-sorted data in storage point

DO I = 1 to 20
    FMALPHxx.FMCODES(I) = FMCODES(SCODES(I)) -- allow bad values
    FMALPHxx.FMNAMES(I) = FMNAMES(SCODES(I))
    FMALPHxx.FMDYE1R(I) = FMDYE1R(SCODES(I)) -- allow bad values
    FMALPHxx.FMDYE2R(I) = FMDYE2R(SCODES(I)) -- allow bad values
    FMALPHxx.FMDYE3R(I) = FMDYE3R(SCODES(I)) -- allow bad values
    FMALPHxx.FMMINLV(I) = FMMINLV(SCODES(I)) -- allow bad values
    FMALPHxx.FMRESTM(I) = FMRESTM(SCODES(I)) -- allow bad values
repeat

-- Reset sort flag

TBOOL(1) = OFF

END FMSORTxx
```

FORMULA MANAGER

```
BLOCK FMLOADxx (specific, executes when TBOOL(2) is ON)

-- Save last formula name

SIMLVLxx.TSTRING(2) = SIMLVLxx.TSTRING(1)

-- Load new formula name, formula code and ratio values

SIMLVLxx.TSTRING(1) = TSTRING(1)
SIMLVLxx.TREAL(1) = TREAL(1)
SIMLVLxx.TREAL(2) = TREAL(2)
SIMLVLxx.TREAL(3) = TREAL(3)
SIMLVLxx.TREAL(4) = TREAL(4)
SIMLVLxx.TREAL(5) = TREAL(5)
SIMLVLxx.TREAL(6) = TREAL(6)

-- Turn off load flag

SET TBOOL(2) = OFF

-- Generate Down Load ticket via a Free Format Log
SEND:"$OUT_RPT TICKETxx"

END FMLOADxx
```

Parameters

Local:

CODES(20)	Number array to store FMCODE values locally
SCODES(20)	Number array to store sorted FMCODE values
NAMES(20)	String array to store FMNAME values locally

CDS Package: FMFCDSxx

Parameters:

PRODUCT	Number, Product Group Number
DYETAGS(3)	Entity array (\$REG_CTL) for dye controller tags
FMCODES(20)	Number array for formula index numbers
FMNAMES(20)	String array for formula names
FMDYE1R(20)	Number array for Dye 1 ratio values
FMDYE2R(20)	Number array for Dye 2 ratio values
FMDYE3R(20)	Number array for Dye 3 ratio values
FMMINLV(20)	Number array for minimum level values
FMRESTM(20)	Number array for residence time values

CDS Package: FMSCDSxx

Parameters:

TSTRING(10)	String array for selected formula names
TREAL(10)	Number array for selected formula values
TBOOL(10)	Logical array for sort and load run flags

FORMULA MANAGER

```
Parameter value assignments:
TSTRING(1) ..... Selected formula name
TREAL(1) ..... Selected formula code
TREAL(2) ..... Selected formula dye 1 ratio
TREAL(3) ..... Selected formula dye 2 ratio
TREAL(4) ..... Selected formula dye 3 ratio
TREAL(5) ..... Selected formula minimum level value
TREAL(6) ..... Selected formula residence time
TBOOL(1) ..... Sort function ON/OFF
TBOOL(2) ..... Load function ON/OFF
```

4. POINT FMALPH_{xx}

This AM Custom point is unscheduled. It has the CDS package FMFCDSxx attached to it, and no CL Blocks. This point holds alpha-sorted formula data stored to it by the CL Block FMSORTxx on point FMFILExx.

SCHEMATIC FMALPH_{xx}

```
03 Sep 91 09:58:16   1
```


ADVANCED AM CLASS FORMULA MANAGER DISPLAY FOR PARTITION 07
CURRENT PRODUCT GROUP : 4

CODE	DESCRIPTOR	DYE1	DYE2	DYE3	MLEV	REST
123	ASDK	5.00	4.00	3.00	55.0	0.80
258	BLUE STOCK	0.56	0.87	1.23	45.0	0.81
189	BROWN STOK	2.50	1.50	0.87	55.0	0.77
259	LT BLU STK	0.55	0.94	0.33	46.0	0.84
000	PINK STOCK	2.50	1.50	0.87	55.0	0.77
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----
---		----	----	----	----	----

DYE CONTROLLERS TAGS
DYE1<CUR>::FICDDY407
 <NEW>::FICDDY407
DYE2<CUR>::FICDDY207
 <NEW>::FICDDY207
DYE3<CUR>::FICDDY307
 <NEW>::FICDDY307

TO CHANGE CURRENT DYE TAG
SELECT AND CHANGE NEW TAG
THEN WRITE PROD GROUP AND

SELECT NEW TASK

ADD NEW FORMULA

MODIFY FORMULA

LOAD FORMULA

READ PRODUCT GROUP

WRITE PRODUCT GROUP

DETAIL FAFICE07 DETAIL FMALPHA07 DETAIL SIMLVL07

SCHEMATIC SCDINR07C

PRODUCT GROUP SELECTOR DESIGN

1. OVERVIEW

This application is an additional function that is added to the Formula Manager Application. Its purpose is to allow the user to select from 12 product group formula files on the HM. The user can read or write the formulas from the files. Additionally, three dye flow controllers are selected for each overall product group. The three controllers are selected from a total of six available dye flow controllers.

2. FUNCTIONAL DESCRIPTION

A new CL block, RWPRODxx, is created which reads and writes formula files from the HM. This block is linked to the BACKGRND insertion point for the AM point FMFILExx. This block is activated whenever TBOOL(3) or TBOOL(4) are set ON. When set ON, they indicate whether a read or write is to be performed, respectively.

On a read the new product group number is specified in TREAL(8). On a write the current product group is specified in PRODUCT. The program converts the product group number to a string and creates a file pathname of the form "NET>S0xx>P##.XX" where ## is the product group number. This file is then written to or read from using the CDS name "FMFCDSxx". After a successful read operation, the sort block on the point must be executed to sort and move the data to FMALPHxx. This is accomplished by setting TBOOL(1) ON and PPSing FMFILExx.

Three additional dye controllers are created, for a total of six controllers. These new controllers are FICDY4xx, FICDY5xx and FICDY6xx. Each product group is assigned three dye controllers to manipulate. The point IDs are stored in the entity array parameter DYETAGS. After a read operation the new dye controller tags are moved from DYETAGS into the parameter array RGPOINTS on the simulator point SIMLVLxx.

PRODUCT GROUP SELECTOR DESIGN

3. CL BLOCK RWPRODxx

The following is a listing of the CL block RWPRODxx.

CL Pseudo-code

BLOCK RWPRODxx (specific,at backgrnd, executes when TBOOL(3) is ON
or TBOOL(4) is ON)

IF read convert the product group in TREAL(8) to a string
IF write convert the product group in PRODUCT to a string

Create the pathname using the converted product group

IF the read function is selected THEN use CDS_READ
to read the file using the created pathname into
CDS FMFCDSxx

IF the write function is selected THEN use CDS_WRITE
to WRITE the file using the created pathname from
CDS FMFCDSxx

Save the file status words in TREAL(9) and TREAL(10)

IF a succesfull read THEN
MOVE DYETAGS(1) to SIMLVLxx.DYETAGSC(1)
MOVE DYETAGS(2) to SIMLVLxx.DYETAGSC(2)
MOVE DYETAGS(3) to SIMLVLxx.DYETAGSC(3)
Store bad value in TREAL(1)
Set TBOOL(1) ON to trigger sort and move function
Set PPS ON to re-trigger this point

IF unsuccessful file operation THEN
SEND error message reporting error codes

Turn program run flags TBOOL(3) and TBOOL(4) OFF

END RWPRODxx

PRODUCT GROUP SELECTOR DESIGN

4. PARAMETER AND LOCAL VARIABLE DEFINITIONS

Parameters

TBOOL(1)	Set ON to trigger sort and move of data by FMSORTxx
TBOOL(3)	Set ON to trigger a read operation *
TBOOL(4)	Set ON to trigger a write operation *
PRODUCT	Current/new product group number
TREAL(8)	Selected product group to read
TREAL(9)	File Manager status of file operation
TREAL(10)	Other status of file operation
TSTRING(2)	Pathname skeleton for file operation
TSTRING(3)	CDS name of structure for file operation
DYETAGS	Three element entity array. Holds the three dye controllers for the current product group.

* Note: TBOOL(3) and TBOOL(4) are mutually exclusive.
Only one should be set ON at a time.

Local Variable

FORMAT:STRING	Format string for number to string conversion
PGASSTR:STRING	Product group number as a string
NTSSTAT:\$CONVRS	Number to string conversion status
MSSTAT:\$MODSTR	Modify string status
RWSTAT:\$CLFSTAT	CDS read/write status
MOVESTAT:CLERRSTS	Move parameter status
FMSTAT:NUMBER	File manager status
PGROUP:NUMBER	Product group as a number

PRODUCT GROUP SELECTOR DESIGN

