Honeywell

High-Performance Process Manager Parameter Reference Dictionary

HP09-540

TotalPlant

Honeywell

Implementation

High-Performance Process Manager - 2

High-Performance Process Manager Parameter Reference Dictionary

HP09-540
Release 530
8/97

TotalPlant

Copyright, Trademarks, and Notices

About This Publication

This publication defines the parameters for the HPM data point types implemented through **TotalPlant** Solution (TPS) system network Release 500 - 530. TPS is the evolution of TDC 3000X.

Change bars are used to indicate paragraphs, tables, or illustrations containing changes that have been made to this manual effective with Release 530. Pages revised only to correct minor typographical errors contain no change bars.

Table of Contents

Table of Contents

INTRODUCTION
Section 1

1.1 PURPOSE

This publication defines the user-visible parameters that exist in the **TotalPlant** Solution (TPS) System High-Performance Process Manager (HPM) and Network Interface Module (NIM). It also provides listings of parameters that are applicable to various HPM point types and algorithms.

For information on how the parameters are related to each other in terms of point types and algorithms, refer to the *High-Performance Process Manager Control Functions and Algorithms* manual in the *Implementation/High-Performance Process Manager - 1* binder.

1.2 USE OF THIS PUBLICATION

Use this publication during configuration and during operation when detailed information about HPM and NIM parameters is required.

For use in data point configuration, this publication provides definitions for each entry that can be made on the *High-Performance Process Manager Point Configuration Forms, HP88-500* in the *Implementation/High-Performance Process Manager - 1* binder, and in the Parameter Entry Displays at the Universal Station.

For use in process operation, this publication provides information about the parameters that appear for the process data points and HPM Box Data Point on the displays of Universal Stations that are running with the Operator personality.

1.3 PARAMETER DEFINITION FORMAT

In this dictionary, the parameter definitions are listed in alphabetical order according to the parameter name, which can be up to eight characters in length. Each parameter in this publication is defined using the format shown below for the **ALMOPT** parameter, as an example. The following paragraphs describe the entries that appear within each parameter definition.

ALMOPT (DigIn)

Type: **e($ALMOPT)** **Alarming Option—**Defines the alarming option for a digital input point whose DITYPE is Status.
Lock: **Eng/PB**
Default: **None**
PtRes: **APM**
Range: 0-**None** (No alarms are to be detected)
1-**Offnorml** (Off Normal; alarm if current PV state is not the PVNORMAL state. PVNORMAL is defined by the STATETXT(0) or STATETXT(1) descriptor, as configured by the user.)
2-**ChngofSt** (An alarm is generated when the digital input changes state in either direction).

Helpful Hint:	ALMOPT configuration requires DITYPE = Status.

For many parameters, the function of the parameter is described using the long name of the parameter (**Alarm Option**), followed by a description as shown in the above example. Some parameters in this dictionary do not have functional descriptions following the long name; this is because the long name of the parameter sufficiently describes the parameter function.

Type

This entry is the data type that defines how the parameter is viewed by the system. The following data types are used in this dictionary:

- **E:**—Enumeration; the value for the parameter is chosen from a set of predefined character strings. In the above example, the enumerations of **$ALMOPT** are **None**, **Offnorml**.
- **SD_ENM**:—Self-Defining Enumeration; the value for the parameter is chosen from the user-defined character strings.
- **Ent.Prm**—consists of a 1-16 character tag name, a period, and a 1-8 character parameter name.
- **Integer**—a 16-bit whole number that does not contain a decimal point ($\pm$ 32767).
- **Logical**—a binary type with the values of ON (True) and OFF (False), or 0 (Off) and 1 (On).
- **NaN**—although not a data type, is used to represent "Not A Number" and is stored in IEEE format.
- **Prm_Id**—1-8 character parameter name.
- **Real**—a 32-bit floating-point number in IEEE format.
- **String_L**—a character string of maximum length = L. Same as Ascii_L.
- **Time**—The time of day in one of the following formats: DDD HH:MM:SS for durations, and DDMMYY HH:MM:SS for an absolute date or time stamp.
- **Universal Ent.Prm**—Universal Entity Parameter Identifier. It is basically the same as **Ent.Prm**, but the entity name can be entered as an external 16-character tag name or as the HPM's internal hardware reference address. The hardware reference address syntax can be used to access parameters of points (within this same HPM) that are untagged or tagged.

The following are examples of hardware reference addresses*:

Type	*Hardware Reference Address*
AO Processor Output	!AO11S03.OP (Parameter OP of Slot #3 of AO processor #11)
DI Processor PV	!DI05S07.PVFL (Parameter PVFL of Slot #7 of DI processor #5)
DO Processor Status Output	!DO15S12.SO (Parameter SO of Slot #12 of DO processor #15)
DO Processor ON Pulse Command	!DO15S12.ONPULSE (Parameter ONPULSE of Slot #12 of DO processor #15)
DO Processor OFF Pulse Command	!DO15S12.OFFPULSE (Parameter OFFPULSE of Slot #12 of DO processor #15)

Lock

The access lock defines "who" or "what" can change the parameter's value or option and the access level defines "who" or "what" is requesting a parameter value or option change. For example, if a requestor with an access level of Supr tries to change a parameter that has an access lock of Engr, the request will be denied. The two charts below describe how access levels and access locks work.

Access Level	*Used By Who Or What When A Parameter Change Request Is Made*
Oper	Operator
Supr or Sup	Supervisor
Engr, Eng, or Eg	Engineer
Cont	Continuous_Control (from a Module on the LCN)
OnProc	On Process
HPMMCc	HPMM_Continuous_Control (from HPMM)
Prog	CL/HPM Sequence_Programs
PtBld or PB	Point_Builder (Data Entity Builder)

Access Lock	*Access Level of Requestors That Can Change The Parameter*						
Oper	Oper	Supr	Engr	Cont	HPMMCc	Prog	PtBld
Supr		Supr	Engr	Cont	HPMMCc	Prog	PtBld
Engr			Engr	Cont	HPMMCc	Prog	PtBld
OnProc	Oper	Supr	Engr				
Sup/Eg		Supr	Engr				
EgOnly			Engr				
Prog				Cont	HPMMCc	Prog	PtBld
Eng/PB			Engr				PtBld
PtBld							PtBld
View (Read Only)							

*The Analog Input address !AImmSss.Parameter is not supported because the Analog Input point does not have a useable default database.

Default

The default for the parameter is the default value assigned by the system. The system automatically enters the default value for a parameter when a range or a selection is not entered for a parameter during point building. The default values are also shown on the configuration forms and parameter entry displays.

PtRes

This defines where the parameter physically resides. The following residency locations are used in the parameter definitions:

PtRes	*Definition*
HPM	High-Performance Process Manager
NIM	Network Interface Module
SI	Serial Interface

Range

This defines the range of the value that can be entered for this parameter. Integers that precede HPM resident enumeration parameters are sometimes needed by advanced CL users. These integers specify the member's position within the set (that is, the ordinal). CL programs external to the UCN (such as AM/CL) will see the same enumeration strings, but in some cases, with different ordinal values.

Helpful Hint

Some parameter definitions contain a *Helpful Hint* box at the end of the definition. This box contains additional information about the parameter, such as prerequisites, etc.

1.4 PARAMETERS PER POINT TYPE AND ALGORITHM TYPE

In addition to the parameter definitions, this dictionary also contains listings of the parameters that are applicable to each HPM point type and algorithm type. Parameters-per-point-type are defined in Section 2; parameters-per-algorithm-type are defined in Section 3.

1.5 FULL POINTS AND COMPONENT POINTS

Separate functional elements of the HPM are used to implement various parts of typical control loops and control strategies. Each of these functional elements can be assigned a user-defined tag name to allow for location-independent reference to the data associated with that function. For example, point tags are assigned by the user for analog input and analog output slots. The I/O Processor data (engineering-unit range for inputs, characterization option for outputs, etc.) is configured as part of the point-build process for these points. A separate tag is configured for each regulatory control (RegCtl) slot that is linked to the assigned analog I/O tags through input/output connections.

The HPM provides a configurable parameter called PNTFORM (Point Form) that allows the user to define which points are to be used as the primary operator interface for point data. The PNTFORM parameter provides the user with two choices for point form: "Full" and "Component." Points that are configured as having "Full" point form include alarm-related parameters and sometimes, some other miscellaneous parameters. This information is needed when the point is to be used as the primary operator interface to the point's data.

Points that are configured as having "Component" point form should be used to provide inputs to the "Full" point and also for those points that handle the outputs from the "Full" points. "Component" points should be used as part of the "Full" point that has been designated a primary operator interface point.

1.6 ABBREVIATIONS

AM	Application Module
AnalgIn	Analog Input Data Point
AnalgOut	Analog Output Data Point
AO	Analog Output
HPM	High-Performance Process Manager
HPMM	High-Performance Process Manager Module
HPM Box	HPM Box Data Point
Array	Array Data Point
AutoMan	Auto Manual algorithm
Box	Box Data Point
Calcultr	Calculator algorithm
CM	Computing Module 50 or 60
DevCtl	Device Control Data Point
DI	Digital Input
DigComp	Digital Composite Data Point
DigIn	Digital Input Data Point
DigOut	Digital Output Data Point
DISOE	Digital Input Sequence of Events
DO	Digital Output
ESI	Extended Standard International Engineering Units
FBus	Field Bus
Flag	Flag Data Point
FlowComp	Flow Compensation algorithm
FTA	Field Termination Assembly
GenLin	General Linearization algorithm

HiLoAvg	High Low Average algorithm
HLAI	High Level Analog Input
IncrSum	Incremental Summer algorithm
IOL	I/O Link
IOP	I/O Processor
Logic	Logic Data Point (Slot)
LCN	Local Control Network
LLAI	Low Level Analog Input (or LLAI-8)
LLMUX	Low Level Analog Input Multiplexer (or LLAI-16/32)
MidOf3	Middle-of-3 Selector algorithm
MulDiv	Multiply Divide algorithm
NIM	Network Interface Module
ORSel	Override Selector algorithm
PI	Pulse Input
Pid	Proportional, Integral, Derivative,
PidErfb	Proportional, Integral, Derivative with External Reset Feedback algorithm
PidFf	PID with Feedforward algorithm
PidPosPr	PID With Position Proportional algorithm
PosProp	Position Proportional algorithm
ProcMod	Process Module Data Point
PSDP	Processor Status Data Point
RampSoak	Ramp Soak algorithm
RatioCtl	Ratio Control algorithm
RegCtl	Regulatory Control Data Point or algorithm
RegPV	Regulatory PV Data Point or algorithm
RHMUX	Remote Hardened Analog Input Multiplexer (or RHMUX–16/32)
SI	Serial Interface
SDI	Serial Device Interface
STI	Smart Transmitter Interface
Switch	Switch algorithm
Summer	Summer algorithm
Timer	Timer Data Point
Totalizr	Totalizer algorithm
UCN	Universal Control Network
VdtLdLag	Variable Deadtime Lead Lag algorithm

1.7 CL ACCESS

1.7.1 Parameter Not Accessible to CL

Parameter $EVNTREC is not accessible to Control Language (CL) sequences.

1.7.2 CL Restricted Parameters

The following parameters are not accessible to PM/CL sequences. They are not *directly* available to AM/CL sequences. Access to AM/CL is through a custom data segment parameters attached to AM regulatory points as described below.

BHALMFL1-BHALMFL7

NODESTS

NODETYP

UCNRECHN

These parameters are available to user schematics using the NIM reserved data point, e.g., $NMuuBnn.param, where uu = UCN number and nn = UCN node number.

AM/CL programs can access the restricted parameters as Regulatory Point General inputs (using ordinary point parameter access). They must be transferred to parameters of AM regulatory points. There are two ways to do this:

1. Boolean parameters (BHALMFLn), can be referenced as general inputs to a Switch algorithm. A CL program can access the switch parameters.

2. For Enumerations (NODEOPER, NODESTS, NODETYP, POSITION, AND UNRECHN) a custom data segment is created to allow the parameters to be referenced as general inputs and transferred to user-defined parameters (of a RegCtl Point) that can be accessed by Cl.

PARAMETERS PER POINT TYPE
Section 2

This section contains listings of parameters that are applicable to each data point type in the HPM, except for the Regulatory Control and Regulatory PV data points which can be found in Section 3. Refer to Sections $ - X for the definitions of the parameters.

2.1 Analog Input (AI)

The parameters of the Analog Input Data points are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	LRL	PVALDB (F)	PVLLPR (F)	PVTVP (F)
ALENBST (F)	LRV	PVALDBEU (F)	PVLLTP (F)	RJTEMP
ASSOCDSP	MODNUM	PVAUTO	PVLOFL	S1
AVDELTHS	NAME	PVAUTOST	PVLOPR (F)	SECVAR
AVSTS	NODENUM	PVCALC	PVLOTP (F)	SENSRTYP
BADPVFL (F)	NODETYP	PVCHAR	PVP	SERIALNO
BADPVPR (F)	NTWKNUM	PVCLAMP	PVRAW	SFSTS
C1	OTDENBLE	PVEUHI	PVRAWHI	SLOTNUM
C2	OVERVAL (F)	PVEULO	PVRAWLO	SLWSRCID
CJTACT	PIUOTDCF	PVEXEUHI	PVROCNFL	STATE
COMMAND	PNTFORM	PVEXEULO	PVROCNPR (F)	STI_EU
CONTCUT (F)	PNTMODTY	PVEXHIFL	PVROCNTP (F)	STISWVER
DAMPING	PNTNODTY	PVEXLOFL	PVROCPFL	STITAG
DECONF	PNTSTATE	PVFORMAT	PVROCPPR (F)	TCRNGOPT
EUDESC	PNTTYPE	PVHHFL	PVROCPTP (F)	TF
HIGHAL (F)	PRIMMOD (F)	PVHHPR (F)	PVSOURCE (F)	TIMEBASE
HIGHALPR (F)	PTDESC	PVHHTP (F)	PVSRCOPT (F)	UNIT
INPTDIR	PTEXECST	PVHIFL	PVSTS	URL
KEYWORD	PTINAL	PVHIPR (F)	PVTV (F)	URV
LASTPV	PV	PVHITP (F)		
LOCUTOFF		PVLLFL		

2.2 Analog Output (AO)

The parameters of the Analog Output Data point are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

ASSOCDSP	MODNUM	OPFINAL	OPOUT2	PNTTYPE
CASREQ (F)	NAME	OPIN0	OPOUT3	PRIMMOD (F)
EUDESC	NMODATTR (F)	OPIN1	OPOUT4	PTDESC
KEYWORD	NMODE (F)	OPIN2	OPOUT5	PTEXECST
LOCALMAN	NODENUM	OPIN3	OPTDIR	RCASOPT
MODATTR (F)	NODETYP	OPIN4	$OPTOL	REDTAG (F)
MODE (F)	NTWKNUM	OPIN5	PNTFORM	RINITREQ (F)
MODEAPPL (F)	OP	OPOUT0	PNTMODTY	SLOTNUM
MODEPERM (F)	OPCHAR	OPOUT1	PNTNODTY	STDBYMAN
			PNTSTATE	UNIT

2.3 Array

The parameters of the Array Data Point are listed below in alphabetical order. The Point Form parameter is set to Full.

AB_DATA1	DEVADDR	NN	PNTFORM	STR64
AB_DATA2	ERRCODE	NNDESC	PNTNODTY	STRDESC
AB_DATA3	EXTDATA	NNUMERIC	PNTTYPE	STRLEN
AB_DATA4	FL	NNSTIX	PRIMMOD	STRSTIX
ASSOCDSP	FLDESC	NODENUM	PTDESC	STSMSG
AUXDATA1	FLSTIX	NODETYP	SCANPRI	TIME
AUXDATA2	FTANUM	NSTRING	SLOTNUM	TIMEDESC
AUXDATA3	INITREQ	NTIME	SPLOCK	TIMESECS
AUXDATA4	IOPNUM	NTWKNUM	STR8	TIMESTIX
BADPVFL	KEYWORD	OVERLAP	STR16	UNIT
CNFMU	NFLAG	PERIOD	STR32	USERID
CNFPU				

2.4 Box (HPM Box)

The parameters of the High-Performance Process Manager Box Data Point are listed below in alphabetical order.

ASSOCDSP	FTA1TYPE	IOP	MNFMDDAY	NODFSTAT	RELVERS
BADPVTXT	FTA2TYPE	IOPIDAY	MNFMDINF	NOPTS	RELREV
CHPINOPR	HOUR	IOPIMON	MNFMDMTH	NORQUMAX	RJTEMP
CMFLTIME	IOLASTS	IOPIYEAR	MNFMDINF	NORSPAVG	SAFOPCMD
CMIDTXT	IOLBSTS	IOPNUM	MNFMDMTH	NORSPMAX	SCANPER
CNFPU	IOLCHAER	IOPSTR1	MNFMDSER	NOTRAAVG	SECOND
CNFPUP	IOLCHASL	IOPSTR2	MNFMDYR	NOTRAMAX	SEQPRGSZ
COMHOUR	IOLCHBER	IORECCHN	MONPER	NPARAVG	SRQUTAVG
COMMIN	IOLCHBSL	IOSSTS	MONTH	NPARMAX	SRSPTAVG
CRIOLORN	IOLCHERT	IOSTKNDR	MOVPVTXT	NPMSLOT	SRSPTMAX
CRPPXORN	IOLCMD	IOTKNSTL	MPCFWREV	NPVSLOT	STR8
CRUCNORN	IOLHWREV	LSIOLORN	MPCHWREV	NSTRING	SUMSLTSZ
CTFLTIME	IOLPERSW	LSPPXORN	NARRSLOT	NTIME	SWTCHACT
CTIDTXT	IOLREV	LSUCNORN	NCTLSLOT	NTIMER	TIERTYPE
CTLHOUR	IOLVERS	MAXCNFPU	NDCSLOT	NTRAAVG	TIME
CTLMIN	IOMCARD	MDMHWREV	NDEVSLOT	NTRAMAX	TMCMD
CTLOPT	IOMCHAER	MINUTE	NFASTCTL	PKGOPT	TMPV
CTLPATCH	IOMCHASL	MNFASIC	NFASTDC	PMMCHAER	TMRV
CYCLETIM	IOMCHBER	MNFCCDAY	NFASTDEV	PMMCHASL	TMSO
DATE	IOMCHBSL	MNFDAY	NFASTLOG	PMMCHBER	TMSP
DAY	IOMCMD	MNFCCINF	NFASTPV	PMMCHBSL	TMST
DB_VALID	IOMCOMER	MNFCCMTH	NFLAG	PMMCMD	TMTB
DIAGCMD	IOMFILE	MNFCCSER	NLOGSLOT	PMMCOMER	TRATAVG
EUNDESC	IOMOPER	MNFCCYR	NN	PMMCTLST	TRATMAX
FL	IOMREALT	MNFFPGA	NNUMERIC	PMMOPER	UTSDRIFT
FRQUTAVG	IOMRECHN	MNFIODAY	NODEASSN	PMMRECCH	UTSNODE
FRQUTMAX	IOMSEVER	MNFIOINF	NODECMD	PMMSEVER	UTSTBCRV
FRSPTAVG	IOMSTS	MNFIOMTH	NODECONF	PMMSFSTS	UTSTIME
FRSPTMAX	IOMTYPE	MNFIOSER	NODENUM	PMMSTS	UTSTIMST
	IONTOKEN	MNFIOYR	NODESTS	PNTNODTY	WEEKDAY
			NODETYP	POSITION	YEAR

Some of the parameters in the above listing are arrays and are not defined in this publication.

2.5 Box Flag

The parameters of the Box Flag Data Point are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full; an * indicates that the parameter is applicable to flag slots 1-128.

$AUXUNIT (F)*	CONTCUT (F)*	NODETYP	PRIMMOD (F)*	S1BOXCLR
ALENBST (F)*	EIPPCODE (F)*	NTWKNUM	PTDESC	SLOTNUM
ALPRIOR	EUDESC	OFFNRMPR (F)*	PV	STATE0
ASSOCDSP	HIGHAL (F)*	PNTFORM	PVFL	STATE1
BOXCLR	KEYWORD	PNTNODTY	S0BOXCLR	STATETXT
CNFMU	NAME	PNTTYPE		UNIT
CNFPU	NODENUM			

2.6 Box Numeric

The parameters of the Box Numeric Data Point are listed below in alphabetical order. The Point Form parameter is set to Full.

ASSOCDSP	EUDESC	NODETYP	PNTNODTY	PV
CNFMU	KEYWORD	NTWKNUM	PNTTYPE	PVFORMAT
CNFPU	NAME	PNTFORM	PRIMMOD	SLOTNUM
	NODENUM		PTDESC	UNIT

2.7 Box Timer

The parameters of the Box Timer Data Point are listed below in alphabetical order. The Point Form parameter is set to Full.

ASSOCDSP	NAME	PNTNODTY	PV	SP
COMMAND	NODENUM	PNTTYPE	RV	STATE
EUDESC	NODETYP	PRIMMOD	SLOTNUM	TIMEBASE
KEYWORD	NTWKNUM	PTDESC	SO	TIMOUTFL
	PERIOD			UNIT

2.8 Device Control (DevCtl)

The parameters of the Device Control Data Point are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	(#Inputs>0)	LOCALMAN	(#Outputs>0)	OVRCTIM	
ACCELTIM		LODSTN 1-2		OVRDCONF	
ALENBST (F)		LOENBL 1-2		OVRDDESC	
ASSOCDSP		LOGICSRC		OVRDI0FL	
BADPVFL (F)		LOSRC 1-2		OVRD1FL	
BADPVPR (F)		MAINDAT		OVRD2FL	
BADPVTXT		MAINTOPT		OVRDALOP	
BADSVFL	(#Inputs>0)	MANMODFL		OVRDALPR (F)	
BADSVPR (F)		MASKTIM		OVRDSIFL	
BOXCLR		MAXTIM0H		P0	(#Outputs>0)
BYPASS		MAXTIM1H		P1	(#Outputs>0)
CMDDISFL		MAXTIM2H		P2	(#Outputs>0)
CMDDISPR (F)		MAXTRAN0		PAUSETIM	
CMDFALFL		MAXTRAN1		PERIOD	
CMDFALTM		MAXTRAN2		PFDLYFL	
CNFERRFL		MODATTR	(#Outputs>0)	PGALGID 1-4	
CNFMU		MODE	(#Outputs>0)	PGDSTN 1-4	
CNFPU		MODEAPPL	(#Outputs>0)	PGPLSWTH 1-4	
CONTCUT		MODEPERM	(#Outputs>0)	PGSO 1-4	
D1		MODNUM		PIALGID 1-12	
D1_0		MOMSTATE	(#Outputs>0)	PIDEADBD1-12	
D1_1	(#Inputs>0)	MOVPVFL	(#Inputs>0)	PINN 1-12	
D2	(#Inputs=1)	MOVPVTXT		PISO 1-12	
D2D1_00	(#Inputs=1)	NAME		PISRC 1-12	
D2D1_01	(#Inputs=2)	NI0	(#Outputs>0)	PNTFORM	
D2D1_10	(#Inputs=2)	NI1	(#Outputs>0)	PNTMODTY	
D2D1_11	(#Inputs=2)	NI2	(#Outputs>0)	PNTNODTY	
D3	(#Inputs=2)	NMODATTR	(#Outputs>0)	PNTSTATE	
D4	(#Inputs=2)	NMODE	(#Outputs>0)	PNTTYPE	
D5		NN		PRGATRFL	
DB_VALID		NNINSET 1-10		PRIMMOD (F)	
DEADBAND		NODENUM		PSDLYFL	
DEADTIME		NODETYP		PTDESC	
DISRC		NODINPTS		PTEXECST	
DODSTN		NODOPTS		PTINAL	
EIPPCODE (F)	(#Inputs>0)	NOLINPTS		PULSEWTH	(#Outputs>0)
EUDESC	(#Outputs>0)	NOLOPTS		PV	(#Inputs>0)
EVTOPT (F)		NONECONF		PVAUTO	(#Inputs>0)
FBTIME (F)		NONE_OP1-3		PVFL	(#Inputs>0)
FL 1-12	(#Inputs>0)	NOPGATE		PVNORMAL (F)	
HIGHAL (F)		NORMCYCL		PVNORMFL	
HIGHALPR		NOSGATE		PVSOURCE (F)	(#Inputs>0)
HISVPEAK		NOSIOVRD		PVSRCOPT (F)	(#Inputs>0)
I0		NOSTATES		PVSTATES 0-4	(#Inputs>0)
I0CONF		NOTRANS0		PVTXTOPT	
I1	(#Outputs>0)	NOTRANS1		REDTAG (F)	(#Outputs>0)
I1CONF		NOTRANS2		RESETFL	
I2	(#Outputs>0)	NRMATRFL		S0BOXCLR	
I2CONF		NSI0	(#Outputs>0)	S1BOXCLR	
INITMAN	(#Outputs>0)	NTWKNUM		S2BOXCLR	
INITREQ		OFFNRMFL		SCHSTS	
KEYWORD		OFFNRMPR (F)		SEALOPT	
L		OP	(#Outputs>0)	SECVAR	
LIBADOPT		OPCMD	(#Outputs>0)	SGALGID 1-2	
LIDESC		OPFINAL	(#Outputs>0)	SGDSTN 1-2	
LISRC 1-12		OPRATRFL	(#Outputs>0)	SGPLSWTH 1-2	
LMREV		OROPT	(#Outputs>0)	SGSO 1-2	
LMSRC					

2.8 Device Control (DevCtl) con't

Continuation of the Device Control parameters are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

SI0
SI0ALOPT
SI0ALPR (F)
SI0CONF
SIALGID 1-12
SIDLYTIM 1-12
SIDSTN 1-12
SISO 1-12
SLOTNUM
SO 0-2
SOCMD
ST0_OP1 (#Outputs>0)
ST0_OP2 (#Outputs>=2)
ST0_OP3 (#Outputs>=3)
ST1_OP1 (#Outputs>0)
ST1_OP2 (#Outputs>=2)
ST1_OP3 (#Outputs>=3)
ST2_OP1 (#Outputs>0)
ST2_OP2 (#Outputs>=2)
ST2_OP3 (#Outputs>=3)
STATE0
STATE1
STATE2
STATTIM0
STATTIM1
STATTIM2
STATETXT0-4
STCHGOPT
STSMSG
SVALDB
SVALDBEU
SVDESC
SVEUDESC
SVEUHI
SVEULO
SVHHFL
SVHHPR (F)
SVHHTP
SVHHTPP
SVHIFL
SVHIPR (F)
SVHITP
SVHITPP
SVP
SVPEAK
SVSRC
SVTV
SVTVP
TRANTIM0
TRANTIM1
TRANTIM2
UNCMDFL
UNIT
USERID

2.9 Digital Composite (DigComp)

The parameters of the Digital Composite Data Point are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	(#Inputs>0)
ALENBST (F)	
ALPRIOR	
ASSOCDSP	
BADCTLPR	
BADPVFL (F)	(#Inputs>0)
BADPVPR (F)	
BADPVTXT	
BOXCLR	
BYPASS	
CMDDISFL	
CMDDISPR	
CMDFALFL	
CMDFALTM	
CNFERRFL	
CNFMU	
CNFPU	
CONTCUT (F)	
D1	
D1_0	
D1_1	(#Inputs>0)
D2	(#Inputs=1)
D2D1_00	(#Inputs=1)
D2D1_01	(#Inputs=2)
D2D1_10	(#Inputs=2)
D2D1_11	(#Inputs=2)
DISRC 1-2	(#Inputs=2)
DODSTN 1-3	(#Inputs=2)
EIPPCODE (F)	(#Inputs>0)
EUDESC	(#Outputs>0)
EVTOPT (F)	
FBTIME (F)	
HIGHAL (F)	(#Inputs>0)
HIGHALPR	
I0	(#Outputs>0)
I0CONF	
I0DESC	
I1	(#Outputs>0)
I1CONF	
I1DESC	
I2	(#Outputs>0)
I2CONF	
I2DESC	
INITMAN	
INITREQ	
KEYWORD	
LOCALMAN	
LOGICSRC	
MAINDAT	(#Outputs>0)
MAINTOPT	
MAXTIM0H	
MAXTIM1H	
MAXTIM2H	
MAXTRAN0	
MAXTRAN1	
MAXTRAN2	
MODATTR	(#Outputs>0)
MODE	(#Outputs>0)
MODEAPPL	(#Outputs>0)
MODEPERM	(#Outputs>0)
MODNUM	
MOMSTATE	(#Outputs>0)
MOVPVFL	(#Inputs>0)
MOVPVTXT	
NAME	
NI0	(#Outputs>0)
NI1	(#Outputs>0)
NI2	(#Outputs>0)
NMODATTR	(#Outputs>0)
NMODE	(#Outputs>0)
NODENUM	
NODETYP	
NODINPTS	
NODOPTS	
NONECONF	
NONE_OP1-3	
NORMCYCL	
NOSIOVRD	
NOSTATES	
NOTRANS1	
NOTRANS0	
NOTRANS2	
NRMATRFL	
NSI0	(#Outputs>0)
NTWKNUM	
OFFNRMFL	
OFFNRMPR (F)	
OP	(#Outputs>0)
OPCMD	(#Outputs>0)
OPFINAL	(#Outputs>0)
OPRATRFL	(#Outputs>0)
OVRDALOP	
OPSTTEXT	
OVRDALPR (F)	
OROPT	(#Outputs>0)
OVRDCONF	
OVRDDESC	
OVRDI0FL	
OVRDI1FL	
OVRDI2FL	
OVRDSIFL	
P0	(#Outputs>0)
P1	(#Outputs>0)
P2	(#Outputs>0)
PAUSETIM	
PERIOD	
PFDLYFL	
PNTFORM	
PNTMODTY	
PNTNODTY	
PNTSTATE	
PNTTYPE	
PRIMMOD (F)	
PRGATRFL	
PSDLYFL	
PTDESC	
PTEXECST	
PTINAL	
PULSEWDTH	
PV	(#Outputs>0)
PVAUTO	(#Inputs>0)
PVFL 0-2	(#Inputs>0)
PVNORMAL (F)	(#Inputs>0)
PVSOURCE (F)	
PVSRCOPT (F)	(#Inputs>0)
PVSTATES 0-4	(#Inputs>0)
PVTXTOPT	(#Inputs>0)
REDTAG (F)	
RESETFL	(#Outputs>0)
S0BOXCLR	
S1BOXCLR	
S2BOXCLR	
SCHSTS	
SEALOPT	
SI0	
SI0ALOPT	
SI0ALPR (F)	
SI0CONF	
SLOTNUM	
SO	
SOCMD	(#Outputs>0)
ST0_OP1	(#Outputs>0)
ST0_OP2	(#Outputs>0)
ST0_OP3	(#Outputs>=2)
ST1_OP1	(#Outputs>=3)
ST2_OP1	(#Outputs>0)
ST1_OP2	(#Outputs>0)
ST2_OP2	(#Outputs>=2)
ST1_OP3	(#Outputs>=2)
ST2_OP3	(#Outputs>=3)
STATE0	(#Outputs>=3)
STATE1	
STATE2	
STATETXT 0-4	
STATTIM0	
STATTIM1	
STATTIM2	
STCHGOPT	
STSMSG	
TRANTIM1	
TRANTIM2	
UNCMDFL	
UNIT	
USERID	

2.10 Digital Input (DigIn)

The parameters of the Digital Input Data point are listed below in alphabetical order. (L), (S), or (A)—parameter applies only when DITYPE = Latched, Status, or Accum. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	DLYTIME (S) (F)	OVERVAL (A) (F)	PVSOURCE (L) (F)
ALENBST (S) (F)	EIPPCODE (S) (L) (F)	PNTFORM	PVSRCOPT (S) (L) (F)
ALMOPT (S) (F)	EUDESC	PNTNODTY	RESETFL (A)
ALPRIOR	EVTOPT (L) (F)	PNTMODTY	RESETVAL (A)
ASSOCDSP	HIGHAL (S)	PNTSTATE	S0BOXCLR (S) (L)
AV (A)	INPTDIR (F)	PNTTYPE	S1BOXCLR (S) (L)
AVTV (A)	KEYWORD	PRIMMOD (F)	SLOTNUM
AVTVFL (A)	MODNUM	PTDESC	STARTFL (A)
BADPVFL (F)	NAME	PTEXECST	STATE (A)
BADPVPR	NODENUM	PV (S) (L)	STATE0 (S) (L)
BOXCLR (L)	NODETYP	PVAUTO (S) (L)	STATE1 (S) (L)
COMMAND (A)	NTWKNUM	PVCHGDLY (S) (L) (F)	STATETXT 0-2 (S) (L)
CONTCUT (S) (F)	OFFNRMFL (S)	PVFL (S) (L)	STOPFL (A)
COUNTDWN (A)	OFFNRMPR (S) (F)	PVNORMAL (S) (F)	UNIT
DEBOUNCE (S) (L)	OLDAV (A)	PVNORMFL (S)	
DITYPE (F)	OVERFLOW (A)	PVRAW	

2.11 Digital Output (DigOut)

The parameters of the Digital Output Data point are listed below in alphabetical order. (S) or (P) parameter applies only when DOTYPE = Status or Pulse Width Modulated (PWM). This point type is available only in the component form.

ASSOCDSP	NAME	OP (P)	PNTSTATE	SLOTNUM
DOTYPE	NODENUM	OPTDIR (P)	PNTTYPE	SO (S)
EUDESC	NODETYP	PERIOD (P)	PTDESC	STATE0 (S)
INITREQ	NTWKNUM	PNTFORM	PTEXECST	STATE1 (S)
KEYWORD	OFFPULSE (S)	PNTMODTY	S0BOXCLR (S)	STDBYMAN
MODNUM	ONPULSE (S)	PNTNODTY	S1BOXCLR (S)	UNIT

2.12 Reserved

2.13 IOP

The parameters of the Input/Output Processor Point are listed below in alphabetical order.

CALIBALL	IOMACTYP	IOMTYPE	IORECCHN	RJRAW
CALIBRJ	IOMFWREV	IOMSTS	LINEPERD	SLOT0SF
FAILOPT	IOMHWREV	IONTOKEN	MAXSLOTS	STDBYSTS
FTAPRES	IOMLHFST	IOPSTR1	NODETYP	SWTCHACT
FREQ6050	IOMOPER	IOPSTR2	PIUOTDCF	WARMSTRT

Some of the parameters in the above listing are arrays and are not defined in this publication.

2.14 Logic

The parameters of the Logic Data Point (otherwise referred to as the Logic Slot) are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	C2SRC	LISRC 1-12	NORMCYCL	R2 1-24
ALENBST (F)	C3SRC	LODSTN 1-12	NTWKNUM	S1 1-24
ASSOCDSP	C4SRC	LOENBL 1-12	PERIOD	S1REV 1-24
C1DESC	CNFERRFL	LOGALGID 1-24	PFDLYFL	S2 1-24
C2DESC	CNFMU	LOGMIX	PNTFORM	S2REV 1-24
C3DESC	CNFPU	LOSRC 1-12	PNTMODTY	S3 1-24
C4DESC	CONTCUT	MODNUM	PNTNODTY	S3REV 1-24
C1FL	DEADBAND 1-24	NAME	PNTSTATE	S4 1-24
C2FL	DLYTIME 1-24	NN1-8	PRIMMOD (F)	SCHSTS
C3FL	EIPPCODE (F)	NODENUM	PRMDESC 1-12 (F)	SLOTNUM
C4FL	FL1-12	NODESC (F)	PSDLYFL	SO 1-24
C1PR (F)	GENDESC (F) 1-12	NODETYP	PTDESC	STSMSG
C2PR (F)	HIGHAL	NOLINPTS	PTEXECST	UNIT
C3PR (F)	HIGHALPR	NOLOGBLK	PTINAL	USERID
C4PR (F)	L1-12	NOLOPTS	R1 1-24	
C1SRC	LIBADOPT			

2.15 Process Module (ProcMod)

The parameters of the Process Module Data Point are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

ABHEMSD	DIAGCMD	OVERSTEP	PTDESC	SSTMT 1-2
ABHHOLD	FL 1-27	OVRRUNFL	RESTART	STATMENT
ABHRSTR	IOLPSERR	OVRRUNPR	RSTROPT	STEP
ABHSHDN	IOLPSOPT	PERIOD	RUNSTATE	STR8 1-16
ACP (F)	LSTWHNER	PFDLYFL	SEQERR	STR16 1-8
ALPRIOR (F)	MAXPU	PHASE	SEQEXEC	STR32 1-4
ANAME 1-3	MSGPEND	PHASEAL	SEQMODE	STR64 1-2
ASSOCDSP	NAME (F)	PHASETIM	SEQNAME	STRLEN
ASTEP 1-3	NN 1-80	PHREMTIM	SEQOBJSZ	STSMSG
ASTMT 1-3	NODENUM	PNTFORM	SEQPR (F)	SUSPSTAT
AVGPU	NODETYP	PNTNODTY	SEQSLTSZ	SUSPTIME
BADIOLPF	NOOVRRUN	PNTTYPE	SLOTNUM	TIME 1-4
CNFMU	NTWKNUM	PROCMOD	SNAME 1-2	UNIT
CNFPU	OVERPHAS	PRIMMOD (F)	SPLOCK	USERID
CNTLLOCK	OVERSTAT	PSDLYFL	SSTEP 1-2	
CLBACK				

2.16 UCN Network

Listed below in alphabetical order are the parameters of the UCN Network Data Point (system parameter $NTWRKnn where nn = the UCN number).

CHPINHWY	HWYCTLST	NIMADDR	NIMREV	NMSGTXT
CLPZMXC	LOADSCOP	NIMDAY	NIMVERS	TIMESYNC
CLPZMXP	MSGTXT 0-15	NIMMONTH	NIMYEAR	UPGRADE

2.17 UCN Node

The parameters of the UCN Node Data Point are listed below in alphabetical order. They can be accessed as follows:

$NMuuNnn.parameter where,

uu is the UCN network number, and nn is the UCN node number.

$UCNLSB 1-50	NODESTS	NTRQUAVG*	UCNRECHN
CABLESTS	NODESTAT	NTRQUMAX*	UPGRADE
CLPZMXC	NODETYP	NTRSPAVG*	UTSDRIFT
CLPZMXP	NPRQUAVG*	NTRSPMAX*	UTSNODE
LOADSCOP	NPRQUMAX*	TIMESYNC	UTSTBCRV
MDMHWREV	NPRSPAVG*	TRATAVG	UTSTIME
MODNUM	NPRSPMAX*	TRATMAX	UTSTIMST
NMSGTXT			

*These parameters are indexed. The index is either an odd number from 1 to 63 and represents either—

- the UCN node number of a peer node for peer-to-peer statistics with that node
- 0 for the sum of all peer-to-peer statistics

Example for case 2 is: NPRQUAVG(0) = NPRQUAVG(1) + NPRQUAVG(3) + ... + NPRQUAVG(63)

PARAMETERS PER ALGORITHM TYPE
Section 3

This section contains listings of parameters that are applicable to each PV and control algorithm in the HPM. Refer to Sections $ - X for the definitions of the parameters.

3.1 Auto Manual (AutoMan)

The parameters of the Auto Manual contrrol algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)
ALENBST (F)
ARWNET
ARWOP
ASSOCDSP
AUTMODFL
B
B0
BADCTLFL
BADCTLOP
BADCTLPR (F)
BCAMODFL
CASMODFL
CASREQ
CIDSTN
CISRC
CNFMU
CNFPU
CODSTN
CONTCUT (F)
CTLEQN
CTLALGID
CTRLINIT
CV
CVEUHI
CVEULO
ESWAUTO
ESWCAS
ESWENBST
ESWMAN
EUDESC
EXTSWOPT
HIGHAL (F)
HIGHALPR (F)
INITMAN
K
KEYWORD
LOCALMAN
MANMODFL
MODATTR
MODEAPPL
MODE
MODEPERM
MODNUM
NAME
NMODATTR
NMODE
NOCINPTS
NOCOPTS
NODENUM
NODETYP
NORMCYCL
NRMATRFL
NRMMODFL
NTWKNUM
OP
OPALDB (F)
OPEU
OPHAFL (F)
OPHIFL
OPHILM
OPHIPR (F)
OPHITP (F)
OPLAFL (F)
OPLOFL
OPLOLM
OPLOPR (F)
OPLOTP (F)
OPMCHLM
OPRATRFL
OPROCLM
$OPTOL
OVERVAL (F)
PERIOD
PFDLYFL
PNTFORM
PNTMODTY
PNTNODTY
PNTSTATE
PNTTYPE
PRGATRFL
PRIMMOD (F)
PSDLYFL
PTDESC
PTEXECST
PTINAL
PTORST
RARWSTS
RATE1
RCASOPT
RCASSHED
REDTAG (F)
RINITREQ
RINITVAL
SAFEOP
SCHSTS
SHEDMODE
SHEDTIME
SHUTDOWN
SLOTNUM
STDBYMAN
STSMSG
UNIT
USERID
X1
X2
XEUHI
XEULO

3.2 Calculator (Calcultr)

The parameters of the Calculator PV algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	NAME	PFDLYFL	PVCLAMP	PVLOTP (F)
ALENBST (F)	P3	PIDSTN	PVEUHI	PVP
ASSOCDSP	NODENUM	PISRC	PVEULO	PVROCNFL
BADPVFL (F)	NODETYP	PNTFORM	PVEXEUHI	PVROCNPR (F)
BADPVPR (F)	NOPINPTS	PNTMODTY	PVEXEULO	PVROCNTP (F)
C1	NORMCYCL	PNTNODTY	PVEXHIFL	PVROCPFL
C2	NTWKNUM	PNTSTATE	PVEXLOFL	PVROCPPR (F)
C3	OVERVAL (F)	PNTTYPE	PVFORMAT	PVROCPTP (F)
C4	P1	PRIMMOD (F)	PVHHFL	PVSGCHTP (F)
CALCEXP	P1STS	PSDLYFL	PVHHPR (F)	PVSOURCE (F)
CNFMU	P2	PTDESC	PVHHTP (F)	PVSRCOPT (F)
CNFPU	P2STS	PTEXECST	PVHIFL	PVSTS
CONTCUT (F)	P3STS	PTINAL	PVHIPR (F)	PVTV (F)
EUDESC	P4	PV	PVHITP (F)	PVTVP (F)
HIGHAL (F)	P4STS	PVALDB (F)	PVINIT	SCHSTS
HIGHALPR (F)	P5	PVALDBEU (F)	PVLLFL	SLOTNUM
KEYWORD	P5STS	PVALGID	PVLLPR (F)	STSMSG
LASTPV	P6	PVAUTO	PVLLTP (F)	TF
MODNUM	P6STS	PVAUTOST	PVLOFL	UNIT
N	PERIOD	PVCALC	PVLOPR (F)	USERID

3.3 Data Acquisition (DataAcq)

The parameters of the Data Acquisition PV algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	NOPINPTS	PTEXECST	PVHHFL	PVROCNPR (F)
ALENBST (F)	NORMCYCL	PTINAL	PVHHPR (F)	PVROCNTP (F)
ASSOCDSP	NTWKNUM	PV	PVHHTP (F)	PVROCPVFL
BADPVFL (F)	OVERVAL (F)	PVALDB (F)	PVHIFL	PVROCPPR (F)
BADPVPR (F)	P1	PVALDBEU (F)	PVHIPR (F)	PVROVPTP (F)
CONCUT (F)	P1STS	PVALGID	PVHITP (F)	PVSGCHTP (F)
CNFMU	PERIOD	PVAUTO	PVINIT	PVSOURCE (F)
CNFPU	PFDLYFL	PVAUTOST	PVLLFL	PVSRCOPT (F)
EUDESC	PIDSTN	PVCALC	PVLLPR (F)	PVSTS
HIGHAL (F)	PISRC	PVCLAMP	PVLLTP (F)	PVTV (F)
HIGHALPR (F)	PNTFORM	PVEUHI	PVLOFL	PVTVP (F)
KEYWORD	PNTMODTY	PVEULO	PVLOPR (F)	SCHSTS
LASTPV	PNTNODTY	PVEXEUHI	PVLOTP (F)	SLOTNUM
MODNUM	PNTSTATE	PVEXEULO	PVP	STSMSG
NAME	PNTTYPE	PVEXHIFL	PVROCNFL	TF
NODENUM	PRIMMOD (F)	PVEXLOFL		UNIT
NODETYP	PSDLYFL	PVFORMAT		USERID
	PTDESC			

3.4 Flow Compensation (FlowComp)

The parameters of the Flow Compensation PV algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	LASTPV	PSDLYFL	PVHHFL	PVSRCOPT (F)
ALENBST (F)	MODNUM	PTDESC	PVHHPR (F)	PVSTS
ASSOCDSP	NAME	PTEXECST	PVHHTP (F)	PVTV (F)
BADPVFL (F)	NODENUM	PTINAL	PVHIFL	PVTVP (F)
BADPVPR (F)	NODETYP	PV	PVHIPR (F)	Q
C	NOPINPTS	PVALDB (F)	PVHITP	QSTS
C1	NORMCYCL	PVALDBEU (F)	PVINIT	RG
C2	NTWKNUM	PVALGID	PVLLFL	RP
CNFMU	OVERVAL (F)	PVAUTO	PVLLPR (F)	RQ
CNFPU	P	PVAUTOST	PVLLTP (F)	RT
COMPHILM	PERIOD	PVCALC	PVOFL	RX
COMPLOLM	PFDLYFL	PVCHAR	PVLOPR (F)	SCHSTS
COMPTERM	P0	PVCLAMP	PVLOTP (F)	SLOTNUM
CONTCUT (F)	PIDSTN	PVEQN	PVP	STSMSG
EUDESC	PISRC	PVEUHI	PVROCNFL	T
F	PNTFORM	PVEULO	PVROCNPR (F)	T0
FSTS	PNTMODTY	PVEXEUHI	PVROCNTP (F)	TF
G	PNTNODTY	PVEXEULO	PVROCPFL	TSTS
GSTS	PNTSTATE	PVEXHIFL	PVROCPPR (F)	UNIT
HIGHAL (F)	PNTTYPE	PVEXLOFL	PVROCPTP (F)	USERID
HIGHALPR (F)	PRIMMOD (F)	PVFORMAT	PVSGCHTP (F)	X
KEYWORD	PSTS		PVSOURCE (F)	XSTS

3.5 General Linearization (GenLin)

The parameters of the General Linearization PV algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	IN11	OUT12	PVALDBEU (F)	PVLOPR (F)
ALENBST (F)	IN12	OVERVAL (F)	PVALGID	PVLOTP (F)
ASSOCDSP	KEYWORD	P1	PVAUTOST	PVP
BADPVFL (F)	LASTPV	P1STS	PVCALC	PVROCNFL
BADVPR (F)	MODNUM	PERIOD	PVCLAMP	PVROCNPR (F)
CNFMU	NAME	PFDLYFL	PVEUHI	PVROCNTP (F)
CNFPU	NODENUM	PIDSTN	PVEULO	PVROCPFL
CONTCUT (F)	NODETYP	PISRC	PVEXEUHI	PVROCPPR (F)
EUDESC	NOPINPTS	PNTFORM	PVEXEULO	PVROCPTP (F)
HIGHAL (F)	NORMCYCL	PNTMODTY	PVEXHIFL	PVSGCHTP (F)
HIGHALPR (F)	NTWKNUM	PNTNODTY	PVEXLOFL	PVSOURCE (F)
IN0	OUT0	PNTSTATE	PVFORMAT	PVSRCOPT (F)
IN1	OUT1	PNTTYPE	PVHHFL	PVSTS
IN2	OUT2	PRIMMOD (F)	PVHHPR (F)	PVTV (F)
IN3	OUT3	PSDLYFL	PVHHTP (F)	PVTVP (F)
IN4	OUT4	PTDESC	PVHIFL	SCHSTS
IN5	OUT5	PTEXECST	PVHIPR (F)	SEGTOT
IN6	OUT6	PTINAL	PVHITP (F)	SLOTNUM
IN7	OUT7	PV	PVINIT	STSMSG
IN8	OUT8	PVAUTO	PVLLFL	TF
IN9	OUT9	PVALDB (F)	PVLLPR (F)	UNIT
IN10	OUT10		PVLLTP (F)	USERID
	OUT11		PVLOFL	

3.6 High-Low Average (HiLoAvg)

The parameters of the High-Low Average PV algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	NODETYP	PISRC	PVEUHI	PVP
ALENBST (F)	NOPINPTS	PNTFORM	PVEULO	PVROCNFL
ASSOCDSP	NORMCYCL	PNTMODTY	PVEXEUHI	PVROCNPR (F)
BADPVFL (F)	NTWKNUM	PNTNODTY	PVEXEULO	PVROCNTP (F)
BADPVPR (F)	OVERVAL (F)	PNTSTATE	PVEXHIFL	PVROCPFL
CNFMU	P1	PNTTYPE	PVEXLOFL	PVROCPPR (F)
CNFPU	P1STS	PRIMMOD (F)	PVFORMAT	PVROCPTP (F)
CONTCUT (F)	P2	PSDLYFL	PVHHFL	PVSGCHTP (F)
EUDESC	P2STS	PTDESC	PVHHPR (F)	PVSOURCE (F)
FORCE	P3	PTEXECST	PVHHTP (F)	PVSRCOPT (F)
FRCPERM	P3STS	PTINAL	PVHIFL	PVSTS
FSELIN	P4	PV	PVHIPR (F)	PVTV (F)
HIGHAL (F)	P4STS	PVALDB (F)	PVHITP (F)	PVTVP (F)
HIGHALPR (F)	P5	PVALDBEU (F)	PVINIT	SCHSTS
KEYWORD	P5STS	PVALGID	PVLLFL	SELINP
LASTPV	P6	PVAUTO	PVLLPR (F)	SLOTNUM
MODNUM	P6STS	PVAUTOST	PVLLTP (F)	STSMSG
N	PERIOD	PVCALC	PVLOFL	TF
NAME	PFDLYFL	PVCLAMP	PVLOPR (F)	UNIT
NMIN	PIDSTN	PVEQN	PVLOTP (F)	USERID
NODENUM				

3.7 Incremental Summer (IncrSum)

The parameters of the Incremental Summer control algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)
ALENBST (F)
ARWNET
ARWOP
ASSOCDSP
AUTMODFL
BADCTLFL
BADCTLPR (F)
BCAMODFL
CASMODFL
CASREQ
CIDSTN
CISRC
CNFMU
CNFPU
CODSTN
CONTCUT (F)
CTLALGID
CTRLINIT
CV
CVEUHI
CVEULO
DELCV
ESWAUTO
ESWCAS
ESWENBST
ESWMAN
EUDESC
EXTSWOPT
HIGHAL (F)
HIGHALPR (F)
INITMAN
K1
K2
K3
K4
KEYWORD
LOCALMAN
M
MANMODFL
MODATTR
MODE
MODEAPPL
MODEPERM
MODNUM
NAME
NMODATTR
NMODE
NOCINPTS
NOCOPTS
NODENUM
NODETYP
NORMCYCL
NRMATRFL
NRMMODFL
NTWKNUM
OP
OPALDB (F)
OPEU
OPHAFL (F)
OPHIFL
OPHILM
OPHIPR (F)
OPHITP (F)
OPLAFL (F)
OPLOFL
OPLOLM
OPLOPR (F)
OPLOTP (F)
OPMCHLM
OPRATRFL
OPROCLM
$OPTOL
OVERVAL (F)
PERIOD
PFDLYFL
PNTFORM
PNTMODTY
PNTNODTY
PNTSTATE
PNTTYPE
PRGATRFL
PRIMMOD (F)
PSDLYFL
PTDESC
PTEXECST
PTINAL
PTORST
RARWSTS
RCASOPT
RCASSHED
REDTAG (F)
RINITREQ
RINITVAL
SAFEOP
SCHSTS
SHEDMODE
SHEDTIME
SHUTDOWN
SLOTNUM
STDBYMAN
STSMSG
UNIT
USERID
X1
X2
X3
X4
XEUHI
XEULO

3.8 Middle-of-3 (MidOf3)

The parameters of the Middle-Of-3 PV algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)
ALENBST (F)
ASSOCDSP
BADPVFL (F)
BADPVPR (F)
CNFMU
CNFPU
CONTCUT (F)
EUDES C
HIGHAL (F)
HIGHALPR (F)
KEYWORD
LASTPV
MODNUM
NAME
NODENUM
NODETYP
NOPINPTS
NORMCYCL
NTWKNUM
OVERVAL (F)
P1
P1STS
P2
P2STS
P3
P3STS
PERIOD
PFDLYFL
PIDSTN
PISRC
PNTFORM
PNTMODTY
PNTNODTY
PNTSTATE
PNTTYPE
PRIMMOD (F)
PSDLYFL
PTDESC
PTEXECST
PTINAL
PV
PVALDB (F)
PVALDBEU (F)
PVALGID
PVAUTO
PVAUTOST
PVCALC
PVCLAMP
PVEQN
PVEUHI
PVEULO
PVEXEUHI
PVEXEULO
PVEXHIFL
PVEXLOFL
PVFORMAT
PVHHFL
PVHHPR (F)
PVHHTP (F)
PVHIFL
PVHIPR (F)
PVHITP (F)
PVINIT
PVLLFL
PVLLPR (F)
PVLLTP (F)
PVLOFL
PVLOPR (F)
PVLOTP (F)
PVP
PVROCNFL
PVROCNPR (F)
PVROCNTP (F)
PVROCPFL
PVROCPPR (F)
PVROCPTP (F)
PVSGCHTP (F)
PVSOURCE (F)
PVSRCOPT (F)
PVSTS
PVTV (F)
PVTVP (F)
SCHSTS
SELINP
SLOTNUM
STSMSG
TF
UNIT
USERID

3.9 Multiply/Divide (MulDiv)

The parameters of the Multiply/Divide control algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)
ALENBST (F)
ARWNET
ARWOP
ASSOCDSP
AUTMODFL
B
B1, B2, B3
BADCTLFL
BADCTLOP
BADCTLPR (F)
BCAMODFL
CASMODFL
CASREQ
CIDSTN
CISRC
CNFMU
CNFPU
CODSTN
CONTCUT (F)
CTLALGID
CTLEQN
CTRLINIT
CV
CVEUHI
CVEULO
ESWAUTO
ESWCAS
ESWENBST
ESWMAN
EUDESC
EXTSWOPT
HIGHAL (F)
HIGHALPR (F)
INITMAN
K
K1, K2, K3
KEYWORD
LOCALMAN
MANMODFL
MODATTR
MODE
MODEAPPL
MODEPERM
MODNUM
NAME
NMODATTR
NMODE
NOCINPTS
NODENUM
NODETYP
NORMCYCL
NRMATRFL
NRMMODFL
NTWKNUM
OP
OPALDB (F)
OPEU
OPHAFL (F)
OPHIFL
OPHILM
OPHIPR (F)
OPHITP (F)
OPLAFL (F)
OPLOFL
OPLOLM
OPLOPR (F)
OPLOTP (F)
OPMCHLM
OPRATRFL
OPROCLM
$OPTOL
OVERVAL (F)
PERIOD
PFDLYFL
PNTFORM
PNTMODTY
PNTNODTY
PNTSTATE
PNTTYPE
PRGATRFL
PRIMMOD (F)
PSDLYFL
PTDESC
PTEXECST
PTINAL
PTORST
RARWSTS
RATE1
RCASOPT
REDTAG (F)
RINITREQ
RINITVAL
SAFEOP
SCHSTS
SHEDMODE
SHEDTIME
SHUTDOWN
SLOTNUM
STDBYMAN
STSMSG
UNIT
USERID
X1, X2, X3

3.10 Override Selector (ORSel)

The parameters of the Override Selector control algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)
ALENBST (F)
ARWNET
ARWOP
ASSOCDSP
AUTMODFL
BADCTLFL
BADCTLOP
BADCTLPR (F)
BCAMODFL
BYPASS
BYPASSX1
BYPASSX2
BYPASSX3
BYPASSX4
BCAMODFL
CASMODFL
CASREQ
CIDSTN
CISRC
CNFMU
CNFPU
CODSTN
CONTCUT (F)
CTALGID
CTLEQN
CTRLINIT
CV
CVEUHI
CVEULO
ESWAUTO
ESWCAS
ESWENBST
ESWMAN
EUDESC
EXTSWOPT
HIGHAL (F)
HIGHALPR (F)
INITMAN
KEYWORD
LOCALMAN
M
MANMODFL
MODE
MODEAPPL
MODEPERM
MODNUM
NAME
NMODATTR
MODE
NOCINPTS
NOCOPTS
NODENUM
NODETYP
NORMCYCL
NRMATRFL
NRMMODFL
NTWKNUM
OP
OPALDB (F)
OPEU
OPHAFL (F)
OPHIFL
OPHILM
OPHIPR (F)
OPHITP (F)
OPLAFL (F)
OPLOFL
OPLOLM
OPLOPR (F)
OPLOTP (F)
OPMCHLM
OPRATRFL
OPROCLM
$OPTOL
OROFFSET
OROPT
OVERVAL (F)
PERIOD
PFDLYFL
PNTFORM
PNTMODTY
PNTNODTY
PNTSTATE
PNTTYPE
PRGATRFL
PRIMMOD (F)
PSDLYFL
PTDESC
PTEXECST
PTINAL
PTORST
RARWSTS
RCASOPT
RCASSHED
REDTAG (F)
RINITREQ
RINITVAL
SAFEOP
SCHSTS
SELXINP
SHEDMODE
SHEDTIME
SHUTDOWN
SLOTNUM
STDBYMAN
STSMSG
UNIT
USERID
X1
X2
X3
X4
XEUHI
XEULO

3.10 Pid

The parameters of the Pid control algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)
ADVDEVFL
ADVDEVPR (F)
ADVDEVTP (F)
ADVSP (F)
ADVSPP (F)
ALENBST (F)
ARWNET
ARWOP
ASPPROC (F)
ASSOCDSP
AUTMODFL
BADCTLFL
BADCTLOP
BADCTLPR (F)
BADPVFL (F)
BADPVPR (F)
BCAMODFL
BIAS
BSHILM
BSLOLM
CASMODFL
CASREQ
CIDSTN
CISRC
CNFMU
CNFPU
CODSTN
CONTCUT (F)
CTLACTN
CTLALGID
CTLEQN
CTRLINIT
CV
CVEUHI
CVEULO
DELCV
DEV
DEVHIFL
DEVHITP (F)
DEVLOFL
DEVLOPR (F)
DEVLOTP (F)
ESWAUTO
ESWCAS
ESWENBST
ESWMAN
EUDESC
EXTSWOPT
GAINOPT
GAPHI
GAPLO
HIGHAL (F)
HIGHALPR (F)
INITMAN
K
KEXT
KEYWORD
KGAP
KLIN
KNL
LASTPV
LMSRC
LOCALMAN
MANMODFL
MANOPCMD
MANOPTIME
MAXPULSE
MODATTR
MODE
MODEAPPL
MODEPERM
MODNUM
NAME
NLFM
NLGAIN
NMODATTR
NMODE
NOCINPTS
NOCOPTS
NODENUM
NODETYP
NORMCYCL
NRMATRFL
NRMMODFL
NTWKNUM
OP
OPCMD
OPALDB (F)
OPEU
OPHAFL (F)
OPHIFL
OPHILM
OPHIPR (F)
OPHISRC
OPHITP (F)
OPLAFL (F)
OPLOFL
OPLOLM
OPLOPR (F)
OPLOSRC
OPLOTP (F)
OPMCHLM
OPRATRFL
OPROCLM
$OPTOL
OVERVAL (F)
PERIOD
PFDLYFL
PIDFORM
PNTFORM
PNTMODTY
PNTNODTY
PNTSTATE
PNTTYPE
PRGATRFL
PSDLYFL
PTDESC
PTEXECST
PTINAL
PTORST
PV
PVALDB (F)
PVALDBEU (F)
PVAUTO
PVAUTOST
PVEUHI
PVEULO
PVFORMAT
PVHHFL
PVHHPR (F)
PVHHTP (F)
PVHIFL
PVHIPR (F)
PVHITPID
PVLLFL
PVLLPR (F)
PVLLTP (F)
PVLOFL
PVLOTP (F)
PVLOPR (F)
PVP
PVROCNFL
PVROCNPR (F)
PVROCNTP (F)
PVROCPFL
PVROCPPR (F)
PVROCPTP (F)
PVSGCHTP (F)
PVSOURCE (F)
PVSRCOPT (F)
PVSTS
PVTRACK
RAMPTIME
RARWSTS
RATIO
RBOPT
RCASOPT
RCASSHED
REDTAG (F)
RINITREQ
RINITVAL
RTHILM
RTLOLM
SAFEOP
SAFOPCMD
SCHSTS
SHEDMODE
SHEDTIME
SHUTDOWN
SLOTNUM
SP
SPEUHI
SPEULO
SPFORMAT
SPHIFL
SPHILM
SPLOFL
SPLOLM
SPOPT
SPP
$SPTOL
SPTV
SPTVP
STDBYMAN
STSMSG
T1
T2
TVPROC
UNIT
USERID

3.11 Pid with External Reset Feedback (PidErfb)

The parameters of the Pid with External Reset Feedback control algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)
ADVDEVFL
ADVDEVPR (F)
ADVDEVTP (F)
ADVSP (F)
ADVSPP (F)
ALENBST (F)
ARWNET
ARWOP
ASSOCDSP
ASPPROC (F)
AUTMODFL
BADCTFL
BADCTLOP
BADCTLPR (F)
BADPVFL (F)
BADPVPR (F)
BCAMODFL
BIAS
BSHILM
BSLOLM
CASMODFL
CASREQ
CIDSTN
CISRC
CNFMU
CNFPU
CODSTN
CONTCUT (F)
CTLACTN
CTLALGID
CTLEQN
CTRLINIT
CV
CVEUHI
CVEULO
DELCV
DEV
DEVHIFL
DEVHIPR (F)
DEVHITP (F)
DEVLOFL
DEVLOPR (F)
DEVLOTP (F)
ESWAUTO
ESWCAS
ESWENBST
ESWMAN
EUDESC
EXTSWOPT
GAINOPT
GAPHI
GAPLO
HIGHAL (F)
HIGHALPR (F)
INITMAN
K
K1
KEXT
KEYWORD
KGAP
KLIN
KNL
LASTPV
LOCALMAN
MANMODFL
MODATTR
MODE
MODEAPPL
MODEPERM
MODNUM
NAME
NLFM
NLGAIN
NMODATTR
NMODE
NOCINPTS
NOCOPTS
NODENUM
NODETYP
NORMCYCL
NRMATRFL
NRMMODFL
NTWKNUM
OP
OPALDB (F)
OPEU
OPHAFL (F)
OPHIFL
OPHILM
OPHIPR (F)
OPHITP (F)
OPLAFL (F)
OPLOFL
OPLOLM
OPLOPR (F)
OPLOTP (F)
OPMCHLM
OPRATRFL
OPROCLM
$OPTOL
OVERVAL (F)
PERIOD
PFDLYFL
PIDFORM
PNTFORM
PNTMODTY
PNTNODTY
PNTSTATE
PNTTYPE
PRGATRFL
PSDLYFL
PTDESC
PTEXECST
PTINAL
PTORST
PV
PVALDB (F)
PVALDBEU (F)
PVAUTO
PVAUTOST
PVEUHI
PVEULO
PVFORMAT
PVHHFL
PVHHTP (F)
PVHHPR (F)
PVHIFL
PVHIPR (F)
PVHITP (F)
PVLLFL
PVLLPR (F)
PVLLTP (F)
PVLOFL
PVLOTP (F)
PVLOPR (F)
PVP
PVROCNFL
PVROCNPR (F)
PVROCNTP (F)
PVROCPFL
PVROCPPR (F)
PVROCPTP (F)
PVSGCHTP (F)
PVSOURCE (F)
PVSRCOPT (F)
PVSTS
PVTRACK
RAMPTIME
RARWSTS
RATIO
RBOPT
RCASOPT
RCASSHED
REDTAG (F)
RFB
RINITREQ
RINITVAL
RTHILM
RTLOLM
S1
SAFEOP
SCHSTS
SHEDMODE
SHEDTIME
SHUTDOWN
SLOTNUM
SP
SPEUHI
SPEULO
SPFORMAT
SPHIFL
SPHILM
SPLOFL
SPLOLM
SPOPT
SPP
$SPTOL
SPTV
SPTVP
STSMSG
STDBYMAN
T1
T2
TRFB
TVPROC
UNIT
USERID

3.12 Pid with Feed Forward (PidFf)

The parameters of the Pid with Feed Forward control algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)
ADVDEVFL
ADVDEVPR (F)
ADVDEVTP
ADVSP (F)
ADVSPP (F)
ALENBST (F)
ARWNET
ARWOP
ASPPROC (F)
ASSOCDSP
AUTMODFL
BADCTLFL
BADCTLOP
BADCTLPR (F)
BADPVFL (F)
BADPVPR (F)
BCAMODFL
BFF
BIAS
BSHILM
BSLOLM
CASMODFL
CASREQ
CIDSTN
CISRC
CNFMU
CNFPU
CODSTN
CONTCUT (F)
CTLACTN
CTLALGID
CTLEQN
CTRLINIT
CV
CVEUHI
CVEULO
DELCV
DEV
DEVHIFL
DEVHIPR (F)
DEVHITP (F)
DEVLOFL
DEVLOPR (F)
DEVLOTP (F)
ESWAUTO
ESWCAS
ESWENBST
ESWMAN
EUDESC
EXTSWOPT
FF
FFOPT
GAINOPT
GAPHI
GAPLO
HIGHAL (F)
HIGHALPR (F)
INITMAN
K
KEXT
KEYWORD
KFF
KGAP
KLIN
KNL
LASTPV
LOCALMAN
MANMODFL
MODATTR
MODE
MODEAPPL
MODEPERM
MODNUM
NAME
NLFM
NLGAIN
NMODATTR
NMODE
NOCINPTS
NOCOPTS
NODENUM
NODETYP
NORMCYCL
NRMATRFL
NRMMODFL
NTWKNUM
OP
OPALDB (F)
OPEU
OPHAFL (F)
OPHIFL
OPHILM
OPHIPR (F)
OPHITP (F)
OPLAFL (F)
OPLOFL
OPLOLM
OPLOPR (F)
OPLOTP (F)
OPMCHLM
OPRATRFL
OPROCLM
$OPTOL
OVERVAL (F)
PERIOD
PFDLYFL
PIDFORM
PNTFORM
PNTMODTY
PNTNODTY
PNTSTATE
PNTTYPE
PRGATRFL
PSDLYFL
PTDESC
PTEXECST
PTINAL
PTORST
PV
PVALDB (F)
PVALDBEU (F)
PVAUTO
PVAUTOST
PVEUHI
PVEULO
PVFORMAT
PVHHFL
PVHHPR (F)
PVHHTP (F)
PVHIFL
PVHIPR (F)
PVHITP (F)
PVLLFL
PVLLPR (F)
PVLLTP (F)
PVLOFL
PVLOTP (F)
PVLOPR (F)
PVP
PVROCNFL
PVROCNPR (F)
PVROCNTP (F)
PVROCPFL
PVROCPPR (F)
PVROCPTP (F)
PVSGCHTP (F)
PVSOURCE (F)
PVSRCOPT (F)
PVSTS
PVTRACK
RAMPTIME
RARWSTS
RATIO
RBOPT
RCASOPT
RCASSHED
REDTAG (F)
RINITREQ
RINITVAL
RTHILM
RTLOLM
SAFEOP
SCHSTS
SHEDMODE
SHEDTIME
SHUTDOWN
SLOTNUM
SP
SPEUHI
SPEULO
SPFORMAT
SPHIFL
SPHILM
SPLOFL
SPLOLM
SPOPT
SPP
SPTV
$SPTOL
SPTVP
STDBYMAN
STSMSG
T1
T2
TVPROC
UNIT
USERID

3.13 PID Position Proportional (PidPosPr)

The parameters of the PID Position Proportional control algorithm are listed below in alphabetical order.

$AUXUNIT (F)
ADVDEVFL
ADVDEVPR (F)
ADVDEVTP (F)
ADVSP (F)
ADVSPP (F)
ALENBST (F)
ARWNET
ARWOP
ASPPROC (F)
ASSOCDSP
AUTMODFL
BADCTLFL
BADCTLOP
BADCTLPR (F)
BADPVFL (F)
BADPVPR (F)
BADPVR (F)
BCAMODFL
BIAS
BSHILM
BSLOLM
CASMODFL
CASREQ
CIDSTN
CISRC
CNFMU
CNFPU
CODSTN
CONTCUT (F)
CTLACTN
CTLALGID
CTLEQN
CTRLINIT
CYCLETIM
DEADBAND
DEADTIME
DELCV
DEV
DEVHIFL
DEVHIPR (F)
DEVHITP (F)
DEVLOFL
DEVLOPR (F)
DEVLOTP (F)
ESWAUTO
ESWCAS
ESWENBST
ESWMAN
EUDESC
EXTSWOPT
GAINOPT
GAPHI
GAPLO
HIGHAL (F)
HIGHALPR (F)
INITMAN
K
K1
KEXT
KEYWORD
KGAP
KLIN
KNL
LASTPV
LMSRC
LOCALMAN
LOWERTIM
LOWRDSTN
LOWRRATE
MANMODFL
MANOPCMD
MANOPTIM
MAXPULSE
MINPULSE
MODATTR
MODE
MODEAPPL
MODEPERM
MODNUM
NAME
NLFM
NLGAIN
NMODATTR
NMODE
NOCINPTS
NOCOPTS
NODENUM
NODETYP
NORMCYCL
NRMATRFL
NRMMODFL
NTWKNUM
OPCMD
OPHIFL
OPHISRC
OPLOFL
OPLOSRC
OVERVAL (F)
PERIOD
PFDLYFL
PIDFORM
PNTFORM
PNTMODTY
PNTNODTY
PNTSTATE
PNTTYPE
PRGATRFL
PRIMMOD (F)
PSDLYFL
PTDESC
PTEXECST
PTINAL
PTORST
PV
PVALDB (F)
PVALDBEU (F)
PVAUTO
PVAUTOST
PVEUHI
PVEULO
PVFORMAT
PVHHFL
PVHHPR (F)
PVHHTP (F)
PVHIFL
PVHIPR (F)
PVHITP (F)
PVLLFL
PVLLPR (F)
PVLLTP (F)
PVLOFL
PVLOPR (F)
PVLOTP (F)
PVP
PVROCNFL
PVROCNPR (F)
PVROCNTP (F)
PVROCPFL
PVROCPPR (F)
PVROCPTP (F)
PVSGCHTP (F)
PVSOURCE (F)
PVSRCOPT (F)
PVSTS
PVTRACK
RAISDSTN
RAISETIM
RAISRATE
RAMPTIME
RARWSTS
RATIO
RBOPT
RCASOPT
RCASSHED
REDTAG (F)
RINITREQ
RINITVAL
RP
RT
RTHILM
RTLOLM
SAFEOPCMD
SCHSTS
SHEDMODE
SHEDTIME
SHUTDOWN
SLOTNUM
SP
SPEUHI
SPEULO
SPFORMAT
SPHIFL
SPHILM
SPLOFL
SPLOLM
SPOPT
SPP
$SPTOL
SPTV
SPTVP
STDBYMAN
STSMSG
T1
T2
TVPROC
UNIT
USERID

3.14 Position Proportional (PosProp)

The parameters of the Position Proportional control algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)
ADVDEVFL
ADVDEVPR (F)
ADVDEVTP (F)
ADVSP (F)
ADVSPP (F)
ALENBST (F)
ARWNET
ARWOP
ASPPROC (F)
ASSOCDSP
AUTMODFL
BADCTLFL
BADCTLOP
BADCTLPR (F)
BADPVFL (F)
BADPVR (F)
BCAMODFL
CASMODFL
CASREQ
CIDSTN
CISRC
CNFMU
CNFPU
CODSTN
CONTCUT (F)
CTLALGID
CTRLINIT
CYCLETIM
DEADBAND
DEADTIME
DEV
DEVHIFL
DEVHIPR (F)
DEVHITP (F)
DEVLOFL
DEVLOPR (F)
DEVLOTP (F)
ESWAUTO
ESWCAS
ESWENBST
ESWMAN
EUDESC
EXTSWOPT
HIGHAL (F)
HIGHALPR (F)
INITMAN
KEYWORD
LASTPV
LMSRC
LOCALMAN
LOWERTIM
LOWRDSTN
LOWRRATE
LTIMHILM
MANMODFL
MANOPCMD
MANOPTIM
MAXPULSE
MINPULSE
MODATTR
MODE
MODEAPPL
MODEPERM
MODNUM
NAME
NMODATTR
NMODE
NOCINPTS
NOCOPTS
NODENUM
NODETYP
NORMCYCL
NRMATRFL
NRMMODFL
NTWKNUM
OPCMD
OPHIFL
OPHISRC
OPLOFL
OPLOSRC
OVERVAL (F)
PERIOD
PFDLYFL
PNTFORM
PNTMODTY
PNTNODTY
PNTSTATE
PNTTYPE
PRGATRFL
PSDLYFL
PRIMMOD (F)
PTDESC
PTEXECST
PTINAL
PTORST
PV
PVALDB (F)
PVALDBEU (F)
PVAUTO
PVAUTOST
PVEUHI
PVEULO
PVFORMAT
PVHHFL
PVHHPR (F)
PVHHTP (F)
PVHIFL
PVHIPR (F)
PVHITP (F)
PVLLFL
PVLLPR (F)
PVLLTP (F)
PVLOFL
PVLOPR (F)
PVLOTP (F)
PVP
PVROCNFL
PVROCNPR (F)
PVROCNTP (F)
PVROCPFL
PVROCPPR (F)
PVROCPTP (F)
PVSGCHTP (F)
PVSOURCE (F)
PVSRCOPT (F)
PVSTS
RAISDSTN
RAISETIM
RAISRATE
RAMPTIME
RARWSTS
RCASOPT
RCASSHED
REDTAG (F)
RINITREQ
RINITVAL
RP
RT
SAFOPCMD
SCHSTS
SHEDMODE
SHEDTIME
SHUTDOWN
SLOTNUM
SP
SPEUHI
SPEULO
SPFORMAT
SPHIFL
SPHILM
SPLOFL
SPLOLM
SPOPT
SPP
$SPTOL
SPTV
SPTVP
STDBYMAN
STSMSG
TVPROC
UNIT
USERID

3.15 Ramp Soak (RampSoak)

The parameters of the Ramp Soak control algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)
ADVDEVFL
ADVDEVPR (F)
ADVDEVTP (F)
ADVSP (F)
ADVSPP (F)
ALENBST (F)
ARWNET
ARWOP
ASPPROC (F)
ASSOCDSP
AUTMODFL
BADCTLFL
BADCTLPR (F)
BCAMODFL
CASMODFL
CASREQ
CIDSTN
CISRC
CNFMU
CNFPU
CODSTN
CONTCUT (F)
CTLALGID
CTRLINIT
CURSEGID
CV
CVEUHI
CVEULO
CYCLEOPT
DEV
DEVHIPR (F)
DEVHIFL
DEVHITP (F)
DEVLOFL
DEVLOPR (F)
DEVLOTP (F)
ESWAUTO
ESWCAS
ESWENBST
ESWMAN
EUDESC
EXTSWOPT
HIGHAL (F)
HIGHALPR (F)
HOLDCMD
INITMAN
KEYWORD
LASTPV
LOCALMAN
MANMODFL
MODATTR
MODE
MODEAPPL
MODEPERM
MODNUM
MXRMPDEV`
MXSOKDEV
NAME
NMODATTR
NMODE
NOCINPTS
NOCOPTS
NODENUM
NODETYP
NORSSEQ
NORMCYCL
NRMATRFL
NRMMODFL
NTWKNUM
NXTSOAKV
OP
OPEU
OPHIFL
OPHILM
OPLOFL
OPLOLM
OPMCHLM
OPRATRFL
OPROCLM
$OPTOL
OVERVAL (F)
PERIOD
PFDLYFL
PNTFORM
PNTMODTY
PNTNODTY
PNTSTATE
PNTTYPE
PRGATRFL
PRIMMOD
PSDLYFL
PTDESC
PTEXECST
PTINAL
PTORST
PV
PVEUHI
PVEULO
PVFORMAT
PVP
PVSTS
RAMPTIME
RARWSTS
RATE1
RATE2
RATE3
RATE4
RATE5
RATE6
RATE7
RATE8
RATE9
RATE10
RATE11
RATE12
REDTAG (F)
REMSOAKT
RINITREQ
RINITVAL
S1
S1BGNTIM
S1ENDTIM
S1SEGID
S2
S2BGNTIM
S2ENDTIM
S2SEGID
SAFEOP
SCHSTS
SEGTYPE
SHUTDOWN
SLOTNUM
SOAKT1
SOAKT2
SOAKT3
SOAKT4
SOAKT5
SOAKT6
SOAKT7
SOAKT8
SOAKT9
SOAKT10
SOAKT11
SOAKT12
SOAKV1
SOAKV2
SOAKV3
SOAKV4
SOAKV5
SOAKV6
SOAKV7
SOAKV8
SOAKV9
SOAKV10
SOAKV11
SOAKV12
SP
SPP
STDBYMAN
STSMSG
TVPROC
UNIT

3.16 Ratio Control (RatioCtl)

The parameters of the Ratio-Control control algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)
ADVDEVFL
ADVDEVPR (F)
ADVDEVTP (F)
ADVSP (F)
ADVSPP (F)
ALENBST (F)
ARWNET
ARWOP
ASPPROC (F)
ASSOCDSP
AUTMODFL
B1
B2
BADCTLFL
BADCTLPR (F)
BADPVFL (F)
BADPVPR (F)
BCAMODFL
CASMODFL
CASREQ
CIDSTN
CISRC
CNFMU
CNFPU
CODSTN
CONTCUT (F)
CTLALGID
CTRLINIT
CV
CVEUHI
CVEULO
DEV
DEVHIFL
DEVHIPR
DEVHITP (F)
DEVLOFL
DEVLOPR (F)
DEVLOTP (F)
ESWAUTO
ESWCAS
ESWENBST
ESWMAN
EUDESC
EXTSWOPT
HIGHAL (F)
HIGHALPR (F)
INITMAN
K1
K2
KEYWORD
LASTPV
LOCALMAN
MANMODFL
MODATTR
MODE
MODEAPPL
MODEPERM
MODNUM
NAME
NMODATTR
NMODE
NOCINPTS
NOCOPTS
NODENUM
NODETYP
NORMCYCL
NRMATRFL
NRMMODFL
NTWKNUM
OP
OPALDB (F)
OPEU
OPHAFL (F)
OPHIFL
OPHILM
OPHIPR (F)
OPHITP (F)
OPLAFL (F)
OPLOFL
OPLOLM
OPLOPR (F)
OPLOTP (F)
OPMCHLM
OPRATRFL
OPROCLM
$OPTOL
OVERVAL (F)
PERIOD
PFDLYFL
PNTFORM
PNTMODTY
PNTNODTY
PNTSTATE
PNTTYPE
PRIMMOD (F)
PSDLYFL
PTDESC
PTEXECST
PTINAL
PTORST
PRGATRFL
PV
PVALDB (F)
PVALDBEU (F)
PVAUTO
PVAUTOST
PVEUHI
PVEULO
PVFORMAT
PVHHFL
PVHHPR (F)
PVHHTP (F)
PVHIFL
PVHIPR (F)
PVHITP (F)
PVLLFL
PVLLPR (F)
PVLLTP (F)
PVLOFL
PVLOPR (F)
PVLOTP (F)
PVP
PVROCNFL
PVROCNPR (F)
PVROCNTP (F)
PVROCPFL
PVROCPPR (F)
PVROCPTP (F)
PVSGCHTP (F)
PVSOURCE (F)
PVSRCOPT (F)
PVSTS
RAMPTIME
RARWSTS
RATE1
RCASOPT
RCASSHED
REDTAG (F)
RINITREQ
RINITVAL
SAFEOP
SCHSTS
SHEDMODE
SHEDTIME
SHUTDOWN
SLOTNUM
SP
SPEUHI
SPEULO
SPFORMAT
SPHIFL
SPHILM
SPLOFL
SPLOLM
SPOPT
SPP
$SPTOL
SPTV
SPTVP
STDBYMAN
STSMSG
TVPROC
UNIT
USERID
X2
X2FILT
X2TF

3.17 Summer (RegCtl)

The parameters of the Reguatory Control Summer algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	CTRLINIT	MODEAPPL	OPLOFL	PTEXECST
ALENBST (F)	CV	MODEPERM	OPLOLM	PTINAL
ARWNET	CVEUHI	MODNUM	OPLOPR (F)	PTORST
ARWOP	CVEULO	NAME	OPLOTP (F)	RARWSTS
ASSOCDSP	ESWAUTO	NMODATTR	OPMCHLM	RATE1
AUTMODFL	ESWCAS	NMODE	OPRATRFL	RCASOPT
B	ESWENBST	NOCINPTS	OPROCLM	REDTAG (F)
BADCTLFL	ESWMAN	NODENUM	$OPTOL	RINITREQ
BADCTLOP	EUDESC	NODETYP	OVERVAL (F)	RINITVAL
BADCTLPR (F)	EXTSWOPT	NORMCYCL	PERIOD	SAFEOP
BCAMODFL	HIGHAL (F)	NRMATRFL	PFDLYFL	SCHSTS
CASMODFL	HIGHALPR (F)	NRMMODFL	PNTFORM	SHEDMODE
CASREQ	INITMAN	NTWKNUM	PNTMODTY	SHEDTIME
CIDSTN	K	OP	PNTNODTY	SHUTDOWN
CISRC	K1, K2, K3, K4	OPALDB (F)	PNTSTATE	SLOTNUM
CNFMU	KEYWORD	OPEU	PNTTYPE	STDBYMAN
CNFPU	LOCALMAN	OPHAFL (F)	PRGATRFL	STSMSG
CODSTN	M	OPHIFL	PSDLYFL	UNIT
CONTCUT (F)	MANMODFL	OPHILM	PRIMMOD (F)	USERID
CTLALGID	MODATTR	OPHIPR (F)	PTDESC	X1, X2, X3, X4
CTLEQN	MODE	OPHITP (F)		XEUHI
		OPLAFL (F)		XEULO

3.18 Summer (RegPV)

The parameters of the Summer Reguatory PV algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	N	PERIOD	PVCLAMP	PVLOPR (F)
ALENBST (F)	MODNUM	PFDLYFL	PVEQN	PVLOTP (F)
ASSOCDSP	NAME	PISRC	PVEULO	PVP
BADPVFL (F)	NODENUM	PIDSTN	PVEUHI	PVROCNFL
BADPVPR (F)	NODETYP	PNTFORM	PVEXEUHI	PVROCNPR (F)
C	NOPINPTS	PNTMODTY	PVEXEULO	PVROCNTP (F)
C1	NORMCYCL	PNTNODTY	PVEXHIFL	PVROCPFL
C2	NTWKNUM	PNTSTATE	PVEXLOFL	PVROCPPR (F)
C3	OVERVAL (F)	PNTTYPE	PVFORMAT	PVROCPTP (F)
C4	P1	PRIMMOD (F)	PVHHFL	PVSGCHTP (F)
C5	P1STS	PSDLYFL	PVHHPR (F)	PVSOURCE (F)
C6	P2	PTDESC	PVHHTP (F)	PVSRCOPT (F)
CNFMU	P2STS	PTEXECST	PVHIFL	PVSTS
CNFPU	P3	PTINAL	PVHIPR (F)	PVTV (F)
CONTCUT	P3STS	PV	PVHITP (F)	PVTVP (F)
D	P4	PVALDB (F)	PVINIT	SCHSTS
EUDESC	P4STS	PVALDBEU (F)	PVLLFL	SLOTNUM
HIGHAL (F)	P5	PVALGID	PVLLPR (F)	STSMSG
HIGHALPR (F)	P5STS	PVAUTO	PVLLTP (F)	TF
KEYWORD	P6	PVAUTOST	PVLOFL	UNIT
LASTPV	P6STS	PVCALC		USERID

3.19 Switch

The parameters of the Switch control algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	CVEUHI	NMODE	OPROCLM	S1
ALENBST (F)	CVEULO	NRMATRFL	$OPTOL	S2
ARWNET	ESWAUTO	NRMMODFL	OVERVAL (F)	S3
ARWOP	ESWCAS	NOCINPTS	PERIOD	S4
ASSOCDSP	ESWENBST	NOCOPTS	PFDLYFL	SAFEOP
AUTMODFL	ESWMAN	NODENUM	PNTFORM	SCHSTS
BADCTFL	EUDESC	NODETYP	PNTMODTY	SELXINP
BADCTLOP	EXTSWOPT	NTWKNUM	PNTNODTY	SHEDMODE
BADCTLPR (F)	HIGHAL (F)	OP	PNTSTATE	SHEDTIME
BCAMODFL	HIGHALPR (F)	OPALDB (F)	PNTTYPE	SHUTDOWN
CASMODFL	INITMAN	OPEU	PRGATRFL	SLOTNUM
CASREQ	KEYWORD	OPHAFL (F)	PRIMMOD (F)	STDBYMAN
CIDSTN	LOCALMAN	OPHIFL	PSDLYFL	STSMSG
CISRC	M	OPHILM	PTDESC	TRACKING
CNFMU	MANMODFL	OPHIPR (F)	PTEXECST	UNIT
CNFPU	MODATTR	OPHITP (F)	PTINAL	USERID
CODSTN	MODE	OPLAFL (F)	PTORST	X1
CONTCUT (F)	MODEAPPL	OPLOFL	RARWSTS	X2
CTLALGID	MODEPERM	OPLOLM	RCASOPT	X3
CTLEQN	MODNUM	OPLOPR (F)	RCASSHED	X4
CTRLINIT	NAME	OPLOTP (F)	REDTAG (F)	XEUHI
CV	NMODATTR	OPMCHLM	RINITREQ	XEULO
		OPRATRFL	RINITVAL	

3.20 Totalizer (Totalizr)

The parameters of the Totalizer PV algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	HIGHALPR (F)	PNTMODTY	PVEXEULO	PVROCPFL
ACCTYPE	KEYWORD	PNTNODTY	PVEXHIFL	PVROCPPR (F)
ALENBST (F)	LASTPV	PNTSTATE	PVEXLOFL	PVROCPTP (F)
ASSOCDSP	MODNUM	PNTTYPE	PVFORMAT	PVSGCHTP (F)
AVDEV1FL	NAME	PRIMMOD (F)	PVHHFL	PVSOURCE (F)
AVDEV1TP	NODENUM	PSDLYFL	PVHHPR (F)	PVSRCOPT (F)
AVDEV2FL	NODETYP	PTDESC	PVHHTP (F)	PVSTS
AVDEV2TP	NOPINPTS	PTEXECST	PVHIFL	PVTV (F)
AVTV	NORMCYCL	PTINAL	PVHIPR (F)	PVTVP (F)
AVTVFL	NTWKNUM	PV	PVHITP (F)	RESETFL
BADPVFL (F)	OLDAV	PVALDB (F)	PVINIT	RESETVAL
BADPVPR (F)	OVERVAL (F)	PVALDBEU (F)	PVLLFL	SCHSTS
C	P1	PVALGID	PVLLPR (F)	SLOTNUM
COMMAND	P1STS	PVAUTO	PVLLTP (F)	STARTFL
CONTCUT (F)	P2	PVAUTOST	PVLOFL	STATE
CNFMU	P2STS	PVCALC	PVLOPR (F)	STOPFL
CNFPU	PERIOD	PVCLAMP	PVLOTP (F)	STSMSG
CUTOFFLM	PFDLYFL	PVEQN	PVP	TF
EUDESC	PIDSTN	PVEUHI	PVROCNFL	TIMEBASE
HIGHAL (F)	PISRC	PVEULO	PVROCNPR (F)	UNIT
	PNTFORM	PVEXEUHI	PVROCNTP (F)	USERID

3.21 Variable Dead Time with Lead/Lag (VdtLdLag)

The parameters of the Variable Dead Time with Lead/Lag PV algorithm are listed below in alphabetical order. (F) indicates that the parameter is applicable when the PNTFORM = Full.

$AUXUNIT (F)	MODNUM	PNTSTATE	PVEXHIFL	PVROCNTP (F)
ALENBST (F)	NAME	PNTTYPE	PVEXLOFL	PVROCPFL
ASSOCDSP	NLOC	PRIMMOD (F)	PVFORMAT	PVROCPPR (F)
BADPVFL (F)	NODENUM	PSDLYFL	PVHHFL	PVROCPTP (F)
BADPVPR (F)	NODETYP	PTDESC	PVHHPR (F)	PVSGCHTP (F)
C	NOPINPTS	PTEXECST	PVHHTP (F)	PVSOURCE (F)
C1	NORMCYCL	PTINAL	PVHIFL	PVSRCOPT (F)
C2	NTWKNUM	PV	PVHIPR (F)	PVSTS
CNFMU	OVERVAL (F)	PVALDB (F)	PVHITP (F)	PVTV (F)
CNFPU	P1	PVALDBEU (F)	PVINIT	PVTVP (F)
CONTCUT (F)	P1STS	PVALGID	PVLLFL	SCHSTS
CUTOFFLM	P2	PVAUTO	PVLLPR (F)	SLOTNUM
D	P2STS	PVAUTOST	PVLLTP (F)	STSMSG
D1	PERIOD	PVCALC	PVLOFL	TD
D2	PFDLYFL	PVCLAMP	PVLOPR (F)	TF
EUDESC	PIDSTN	PVEQN	PVLOTP (F)	TLD
HIGHAL (F)	PISRC	PVEUHI	PVP	TLG1
HIGHALPR (F)	PNTFORM	PVEULO	PVROCNFL	TLG2
KEYWORD	PNTMODTY	PVEXEUHI	PVROCNPR (F)	UNIT
LASTPV	PNTNODTY	PVEXEULO		USERID

-$-

$ADD (Array)

Type: **Logical**
Lock: **PtBld**
Default: **On**
PtRes: **HPM**
Range: **Off**
On

Add Point Last Parameter Indicator—The last parameter sent to the HPM during point build of an array point.

Helpful Hint: Do not remove $ADD from an exception build file or the point will not load properly.

$AUXUNIT

Type: **E:UNIT**
Lock: **Oper**
Default: **Null**
PtRes: **NIM**

Auxiliary Unit—The Auxiliary Unit of an alarmable process point. If an Auxiliary Unit ID is specified, alarms from this point go to the Auxiliary Unit instead of the Primary Unit. If the $AUXUNIT parameter is set to null (- -), alarms go to the Primary Unit. Available in Release 520 and later software.

Range: **A-Z, 0-9 and Null** - (Any valid unit ID as configured in the NCF)

Helpful Hint: A Network configuration option sets the keylevel required to change $AUXUNIT.

$COMCFLM (HPM Box)

Type: **Real**
Lock: **Engineer**
Default: **10**
PtRes: **HPM**

Comm Processor CPU Free Low Limit in per cent —

Helpful Hint: This parameter cannot be reset.

Range: 0 - 100

$COMCUOS (HPM Box)

Type: **Real**
Lock: **View**
Default: **0.0**
PtRes: **HPM**

Comm Processor CPU Utilization (System)— Specifies the CPU Utilization (in per cent) for the Comm Processor operating system, device drivers, and interrupt handlers.

Helpful Hint:	This parameter cannot be reset.

Range: **0 - 100**

$COMCUTS(0 - 99) (HPM Box)

Type: **Real**
Lock: **View**
Default: **0.0**
PtRes: **HPM**

Comm Processor CPU Utilization (Task)— CPU Utilization (in per cent) for each Comm Processor Task.

Helpful Hint:	This parameter cannot be reset.

Range: **0 - 100**

$CTLCFLM (HPM Box)

Type: **Real**
Lock: **Engineer**
Default: **10**
PtRes: **HPM**

Control Processor CPU Free Low Limit—

Helpful Hint:	This parameter cannot be reset.

Range:

$CTLCUOS (HPM Box)

Type: **Real**
Lock: **View**
Default: **0.0**
PtRes: **HPM**

Control Processor CPU Utilization (system) — specifies the CPU Utilization (in per cent) for the Control Processor operating system, device drivers, and interrupt handlers

Helpful Hint:	This parameter cannot be reset.

Range: **0 - 100**

$CTLCUTS(0 - 99) (HPM Box)

Type:	**Real**	**Control Processor CPU Utilization (Task)** — specifies the CPU Utilization (in per cent) for each Control Processor Task
Lock:	**View**	
Default:	**0.0**	
PtRes:	**HPM**	

Helpful Hint:	This parameter cannot be reset.

Range: **0 - 100**

$DBVALID (HPM Box)

Type:	**E:$ACCSRC**	**HPM Database Valid**
Lock:	**Eng**	
Default:	**DB Invalid**	
PtRes:	**HPM**	
Range:	0-**DB_Valid** (An IOP database is valid and the IOP can be started) 1-**DBInvalid** (An IOP database is not valid and the IOP will not start)	

$IOMPADD(1)–(168)

Type:	**Integer**	**IOP Address**—Returns the physical address of the IOP with (soft address) (File-1)*16 + card + 127
Lock:	**View**	
Default:		
PtRes:	**HPM**	
Range:	**0, 129 - 255**	

$OPTOL (RegCtl, AO)

Type:	**Real (in Percent)**	**Output Tolerance Parameter Definition—**Tolerance limit for a manually entered OP. The difference between a new OP and a current OP is compared against $OPTOL. If the tolerance is violated in either a positive or negative direction from the current value of the OP, operator confirmation is required before the value is stored. A value of 0.0 disables this check. An NaN or a negative value is not allowed.
Lock:	**Engineer**	
Default:	**0.0**	
PtRes:	**HPM**	
Range:	**0.0 to 106.9**	

$SPTOL (RegCtl)

Type:	**Real (in Engineering Units)**	**Setpoint Tolerance Parameter Definition—**Tolerance limit for a manually entered SP. The difference between a new SP and a current SP is compared against $SPTOL. If the tolerance is violated in either a positive or negative direction from the current value of the SP, operator confirmation is required before the value is stored. A value of 0.0 disables this check. An NaN or a negative value is not allowed.
Lock:	**Engineer**	
Default:	**0.0**	
PtRes:	**HPM**	
Range:	**>= 0.0**	

$UCNLSB(1)–(50)

Type: **Real** **Local UCN Communications Statistics**
Lock: **View**
Default:
PtRes: **HPM**
Range:

$UCNLSB(45) (NIM)

Type: **Real** **Local Statistics Block—**The number of auto reconnects.
Lock: **View**
Default: **0**
PtRes: **HPM, NIM**
Range: ≤**0**

-A-

AB_DATA1 (SI - Array)

Type: **Real**
Lock: **Eng**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Auxiliary A-B Data 1—Specifies the Allen-Bradley PLC family type: 2.0, 3.0, or 5.0. Refer to the *APM/HPM Serial Interface Options* manual when configuring for diagnostics.

Helpful Hint:	Use of this parameter is only required to configure Serial Interface mapping to/from an Allen-Bradley programmable logic controller device. This parameter should be set to NAN if it is not being used.

AB_DATA2 (SI - Array)

Type: **Real**
Lock: **Eng**
Default: **NaN**
PtRes: **HPM**
Range: **0 - 999, NaN**

Auxiliary A-B Data 2—Specifies the Allen-Bradley PLC File Number (in decimal) from which data is read into the Array point for PLC-3 or PLC-5 controllers. Must be NaN for PLC-2.

Helpful Hint:	Use of this parameter is only required to configure Serial Interface mapping to/from an Allen-Bradley programmable logic controller device. This parameter should be set to NAN if it is not being used.

AB_DATA3 (SI - Array)

Type: **Real**
Lock: **Eng**
Default: **NaN**
PtRes: **HPM**
Range: **0 - 13**

Auxiliary A-B Data 3—Specifies the data type for Allen-Bradley PLC-2 or PLC-5 controllers or section ID for PLC-3 controllers. Refer to the *APM/HPM Serial Interface Options* manual for additional information.

Helpful Hint:	Use of this parameter is only required to configure Serial Interface mapping to/from an Allen-Bradley programmable logic controller device. This parameter should be set to NAN if it is not being used.

AB_DATA4 (SI - Array)

Type: **Real**
Lock: **Eng**
Default: **NaN**
PtRes: **HPM**
Range: **0 - 256**

Auxiliary A-B Data 4—Specifies the Allen-Bradley PLC scan frequency: 0 indicates that the point is to be scanned as fast as possible. 1–255 indicates the number of seconds for the polling period; 256 = scan once. Note that the report by exception feature can work with any scan rate selection. Refer to the *APM/HPM Serial Interface Options* manual for more information.

Helpful Hint:	Use of this parameter is only required to configure Serial Interface mapping to/from an Allen-Bradley programmable logic controller device. This parameter should be set to NAN if it is not being used.

ABHEMSD (ProcMod)

Type: **Logical** **Abnormal Handler Emergency Shutdown Enable Flag—**Indicates if the Emergency Shutdown abnormal handler sequence is currently enabled.
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **On** (Emergency Shutdown abnormal handler is enabled)
Off (Emergency Shutdown abnormal handler not enabled)

ABHHOLD (ProcMod)

Type: **Logical** **Abnormal Handler Hold Enable Flag—**Indicates if the Hold abnormal handler sequence is currently enabled.
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **On** (Hold abnormal handler is enabled)
Off (Hold abnormal handler not enabled)

ABHRSTR (ProcMod)

Type: **Logical** **Abnormal Handler Restart Enable Flag—**Indicates if the Restart abnormal handler sequence is currently enabled.
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **On** (Restart abnormal handler is enabled)
Off (Restart abnormal handler not enabled)

ABHSHDN (ProcMod)

Type: **Logical** **Abnormal Handler Shutdown Enable Flag—**Indicates if the Shutdown abnormal handler sequence is currently enabled.
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **On** (Shutdown abnormal handler is enabled)
Off (Shutdown abnormal handler not enabled)

ACCELTIM (DevCtl)

Type: **Time (Duration)** **Acceleration Time—**The amount of time the SECVAR parameter exceeded the SVHITP parameter while not in State0. This parameter resets to zero each time the state transitions to State0.
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **0** to **4000 days** (With a resolution of 1 second)

ACCTYPE (Totalizer)

Type:	**E:$ACCTYPE**	**Accumulator Operation Mode—**Specifies the type of input.
Lock:	**Eng/PB**	
Default:	**Analog**	
PtRes:	**HPM**	

Range:	0-**Pulse**	Pulse input
	1-**Analog**	Analog input

ACP (ProcMod)

Type:	**Ent_Id**	**Advanced Control Point ID—**Defines the name of the point in the CG or CM to which this process module is assigned. The NIM notifies the advanced control point when the process module sends a special sequence message.
Lock:	**PtBld**	
Default:	**Null**	
PtRes:	**NIM**	

Range: Tag name can be up to 16 characters, and the permissible character set is as follows:

Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.

ACTPRIM(1)–(40) (HPM Box)

Type:	**E:$ACTPRIM**	**Acting Primary I/O module—**Specifies the acting primary I/O module. nn = 1–40 corresponds to the 40 logical I/O modules. Applies to primary IOP only.
Lock:	**View**	
Default:		
PtRes:	**HPM**	
Range:	0-**IOM_A** (The A module is the acting primary) 1-**IOM_B** (The B module is the acting primary)	

ADVDEVFL

Type:	**Logical**	**Advisory Deviation Alarm Flag—**Indicates whether an advisory alarm has been detected.
Lock:	**View**	
Default:	**Off**	
PtRes:	**HPM**	
Range:	**Off** (Alarm has not been detected) **On** (Alarm has been detected. PV - ADVSP is greater than ADVDEVTP)	

Helpful Hint:	ADVDEVFL is never On unless SPOPT = Asp.

ADVDEVPR

Type: **E:ALPRIOR** **Advisory Deviation Alarm Priority—**Determines the priority of the advisory deviation alarm.
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

Helpful Hint:	ADVDEVPR configuration requires SPOPT = Asp.

ADVDEVTP

Type: **Real** **Advisory Deviation Alarm Trip Point—**An alarm will be generated when the difference between PV and ADVSP exceeds the value in this parameter.
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **≥ 0.0**
NaN

Helpful Hint:	1. ADVDEVTP change requires SPOPT = Asp . 2. Alarm generation requires ASPPROC = Enable and abs(PV - ADVSP) > ADVDEVTP. When abs(PV - ADVSP) < ADVDEVTP * .9 alarm returns to normal.

ADVSP

Type: **Real** **Advisory Setpoint in Engineering Units**
Lock: **Supr**
Default: **N/A**
PtRes: **HPM**
Range: **SPLOLM** to **SPHILM**

Helpful Hint:	ADVSP change requires (SPOPT = Asp) + (ASPPROC = Enable). Alarm generation requires ASPPROC = Enable and abs(PV - ADVSP) > ADVDEVTP. When abs(PV - ADVSP) < ADVDEVTP * .9 alarm returns to normal.

ADVSPP

Type: **Real** **Advisory Setpoint in Percent**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

Helpful Hint:	ADVSPP cannot be viewed unless SPOPT = Asp.

ALENBST

Type:	**E:ALENBST**	**Alarm Enable Status—**Defines the alarm reporting function that is to be used when an alarm condition is detected in this data point. Note that even when alarms are disabled, the alarm indicators still appear on the Group and Detail displays. With Release 510 and later software, the word `DIS` appears in half height text above the tag name on the Point Detail or Group Display for a point with ALENBEST set to Disable.
Lock:	**Oper**	
Default:	**Enable**	
PtRes:	**NIM**	

Range:

	Displayed	Logged	Reported to EIP
Enable	Yes	Yes	Yes
Disable	No	Yes	Yes
Inhibit	No	No	No

Helpful Hint:	ALENBST should not be set to Disable or Inhibit for points critical to safe operations. For Box Flag points, this parameter applies to only slots 1 through 128.

Reference:	See *Engineer's Reference Manual in the* Alarm Management section, for more information on what should happen under different possible alarm actions.

NOTE

The access lock for the ALENBST parameter is configurable through System-Wide Values.

ALMOPT (DigIn)

Type:	**E:$ALMOPT**	**Alarming Option—**Defines the alarming option for a digital input point whose DITYPE is Status.
Lock:	**Eng/PB**	
Default:	**None**	
PtRes:	**HPM**	

Range: 0-**None** (No alarms are to be detected.)
1-**Offnorml** (Off Normal; alarm if current PV state is not the PVNORMAL state. PVNORMAL is defined by the STATETXT(0) or STATETXT(1) descriptor, as configured by the user.)
2-**ChngofSt** (An alarm is generated when the digital input changes state in either direction. Note that IOP firmware must support Change of State Reporting.)

Helpful Hint:	ALMOPT configuration requires DITYPE = Status.

ALPRIOR (ProcMod)

Type: **E:ALPRIOR** **Alarm Priority—**Defines the alarm priority for Process Module points. Note that even when the alarm priority is Journal, the alarm indicators still appear on the Group and Detail displays.
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **Emergncy** (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
JnlPrint (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

Helpful Hint: Access to ALPRIOR is by schematic or CL. ALPRIOR is retained in R500 for compatibility with earlier software. Use SEQPR for new points.

Reference: See *Engineer's Reference Manual in the* Alarm Management section, for more information on what should happen under different possible alarm actions.

ALPRIOR (DigComp, DigIn, FL)

Type: **E:ALPRIOR** **Composite Alarm Priority—**When read, returns a value equal to the highest configured priority among all alarm parameters for the point. When written, sets all of the point's alarm priority parameters equal to the value being stored. Note that individual parameters such as BADPVPR, etc. can be stored individually. If a point's separate alarm priorities are all set to the same priority, ALPRIOR is compatible with R400 and earlier software.
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **Emergncy** (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
JnlPrint (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

Helpful Hint: Access to ALPRIOR is by schematic or CL. No value is actually read from ALPRIOR on a read and no value is actually stored to ALPRIOR on a write. Values are copied to and from the separate alarm priorities.

Reference: See *Engineer's Reference Manual in the* Alarm Management section, for more information on what should happen under different possible alarm actions.

ANAME(1)–(3) (ProcMod)

Type: **String_8** **Abnormal Sequence Name—**Indicates the name of the abnormal handler currently being executed by the process module. A value of "   " means that an abnormal handler is not executing. ANAME(1) returns the abnormal handler name, while both ANAME(2) and ANAME(3) return the names of the two abnormal subroutine levels being executed.
Lock: **View**
Default: **Spaces**
PtRes: **HPM**
Range: **N/A**

AOCALIB(1)–(168)

Type: **Logical**
Lock: **Eng/PB**
Default:
PtRes: **HPM**
Range: **Off** (No calibration in progress)
On (Calibration in progress)

AO Calibration In Progress Flag—Shows which AO modules are in the process of calibration.

ARWNET (RegCtl)

Type: **E:WINDUP**
Lock: **View**
Default: **Normal**
PtRes: **HPM**
Range: 0-**Normal** (Free to move in either direction)
1-**Hi** (Free to move in the lower direction)
2-**Lo** (Free to move in the higher direction)
3-**HiLo** (Not free to move in any direction)

Windup Status of the Input—Indicates the windup status for the SP or another initializable input.

ARWOP (RegCtl)

Type: **E:WINDUP**
Lock: **View**
Default: **Normal**
PtRes: **HPM**
Range: 0-**Normal** (Free to move in either direction)
1-**Hi** (Free to move in the lower direction)
2-**Lo** (Free to move in the higher direction)
3-**HiLo** (Not free to move in any direction)

Windup Status of the Output—Indicates the output (OP) windup status.

ASSOCDSP

Type: **String_8**
Lock: **Engr**
Default: **Blank**
PtRes: **NIM**
Range: N/A

Associated Display—Specifies a user configured schematic that is associated with this point. Available on Release 510 and later software.

Helpful Hint: The specified associated display can be called from a Point Detail Display, or from any summary display or the Group display when the point is selected.

ASPPROC (RegCtl)

Type: **E:ASPPROC**
Lock: **Supr**
Default: **Disable**
PtRes: **HPM**
Range: 0-**Disable** (Disallow advisory deviation alarming)
1-**Enable** (Allow advisory deviation alarming)

Advisory SP Processor State

Helpful Hint: ASPPROC change requires SPOPT = Asp.

ASTEP(1)–(3) (ProcMod)

Type: **String_8**
Lock: **View**
Default: **Spaces**
PtRes: **HPM**
Range: **N/A**

Abnormal Step Name—ASTEP(1) indicates the step name of the abnormal handler that is executing in this process module. A value of "   " means no abnormal handler is presently executing. Both ASTEP(2) and ASTEP(3) indicate the step names of the first and second level subroutines called from the abnormal handler.

ASTMT(1)–(3) (ProcMod)

Type: **Integer**
Lock: **View**
Default: **Blank**
PtRes: **HPM**
Range: **0** to **255**

Abnormal Statement Number—ASTMT(1) indicates the statement number of the abnormal handler that is presently executing in the process module. Both ASTMT(2) and ASTMT(3) give statement numbers for first and second level subroutines executing from an abnormal handler. A value of 0 indicates no sequence is being executed.

AUTMODFL (RegCtl)

Type: **Logical**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **Off** (Current mode is not Automatic)
On (Current mode is Automatic)

Automatic Mode Flag—Indicates whether the current mode of the point is Automatic.

AUXDATA1 (SI-Array — Generic Modbus)

Type: **Real**
Lock: **Eng**
Default: **NaN**
PtRes: **HPM**
Range: **1 - 9999, NaN**

FTA Driver Auxiliary Data 1—Keep Alive Address for Modbus devices. Specifies the address of a coil that is written to every 10 seconds (Force Single Coil On function). NaN (dashes) = Keep Alive function is inactive.

Helpful Hint:	AUXDATA1 can be configured separately for each Array point. No two Array points should write to the same coil address. This parameter should be set to NAN if it is not being used.

AUXDATA2 (SI-Array — Generic Modbus)

Type: **Real**
Lock: **Eng**
Default: **NaN**
PtRes: **HPM**
Range: **.25 - 5 Sec., NaN**

FTA Driver Auxiliary Data 2— Specifies the time interval that the FTA waits before a message retry to the Modbus is attempted. NaN (dashes) indicates a 1.5 second timeout.

Helpful Hint:	After three retries, a message timeout error is displayed on the Point Detail display. AUXDATA2 can be configured separately for each Array point. This parameter should be set to NAN if it is not being used.

AUXDATA3 (SI-Array — Generic Modbus)

Type: **Real**
Lock: **Eng**
Default: **NaN**
PtRes: **HPM**

FTA Driver Auxiliary Data 3—Signaling mode**.**Modem support (in integer/decimal format). Integer = 232 or 485.
(232 = EIA-232, 485 = EIA-485 Multidrop).
Decimal (EIA-232 only) = .0 or .1 (.0 = no modem control, .1 = modem control).
NaN (dashes) = 232.0 (EIA-232 without modem control).

Range: **232.0, 232.1, or 485.0**

> *Helpful Hint:* All array points that are loaded to the same FTA must have the same AUXDATA3 settings. This parameter should be set to NAN if it is not being used.

AUXDATA4 (SI-Array — Generic Modbus)

Type: **Real**
Lock: **Eng**
Default: **NaN**
PtRes: **HPM**

FTA Driver Auxiliary Data 4—Baud Rate**.**Parity (in integer/decimal format).
Baud Rates = 1200, 2400, 4800, 9600, or 19200.
Parity: .0 = no parity, .1 = odd parity, .2 = even parity.
NaN (dashes) = 9600.1 = (9600 baud, odd parity).

Range: integer = **1200, 2400, 4800, 9600, or 19200**
decimal **= .0, .1, or .2**

> *Helpful Hint:* All array points that are loaded to the same FTA must have the same AUXDATA4 settings. This parameter should be set to NAN if it is not being used.

AV (DigIn)

Type: **Integer**
Lock: **Oper**
Default: **0**
PtRes: **HPM**
Range: **0··32767**

Accumulated Value in Engineering Units—Indicates the current value accumulated in the accumulator.

AV (RegCtl)

Type: **Integer**
Lock: **Configurable**
Default: **0**
PtRes: **APM**
Range: **0··32767**

Accumulated Value in Engineering Units—Indicates the current value accumulated in the accumulator.

AVDELTHS (PI)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: $\geq$ **0**

The Last Half-second's AV

AVDEV1FL (Totalizr)

Type: **Logical**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **Off** (PVCALC is not > AVTV - AVDEV1TP)
On (PVCALC is > AVTV - AVDEV1TP)

Accumulated Value; 1st Deviation Flag—Indicates whether PVCALC is greater than AVTV minus AVDEV1TP. (PVCALC > AVTV - AVDEV1TP)**.** This is the first "slowdown" or "near-target" flag.

AVDEV1TP (Totalizr)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **≥ 0.0,**
NaN

Accumulated Value; 1st Deviation Trip Point (deviation from AVTV)

AVDEV2FL (Totalizr)

Type: **Logical**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **Off** (PVCALC is not > AVTV - AVDEV2TP)
On (PVCALC is > AVTV - AVDEV2TP)

Accumulated Value; 2nd Deviation Flag—Indicates whether PVCALC is greater than AVTV minus AVDEV2TP. (PVCALC > AVTV - AVDEV2TP)**.** This is the second "slowdown" or "near-target" flag.

AVDEV2TP (Totalizr)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **≥ 0.0,**
NaN

Accumulated Value; 2nd Deviation Trip Point (deviation from AVTV)

AVGPU (ProcMod)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Average PUs—Specifies the average PUs used for point processing

AVGTF (NIM, HPM Box)

Type: **Real**
Lock: **Engineer**
Default: **1.00 Minutes**
PtRes: **HPM**
Range: **0.0 - 1440.0 (0 = no filter)**

Average Statistics Single Lag Filter Time Constant—Defines the filter time in the single lag filter used to calculate average values of the performance statistics.

AVSTS (PI)

Type: **E:PVVALST** **Value Status of AV**
Lock: **View**
Default: **Bad**
PtRes: **HPM**
Range: **0-Bad**
2-Normal

AVTV (DigIn)

Type: **Integer**
Lock: **Oper**
Default: **0**
PtRes: **HPM**
Range: **0** to **32767**

Accumulator Target Value—Specifies the target value of the accumulator. AVTV appears on a group or detail display as the SP value.

Helpful Hint:	AVTV change requires DITYPE = Accum.

AVTV (Totalizr)

Type: **Real**
Lock: **Oper**
Default: **NaN**
PtRes: **HPM**
Range: **N/A,**
NaN

Accumulator Target Value—Specifies the target value of the totalizer. AVTV appears on a group or detail display as the SP value.

AVTVFL

Type: **Logical**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **Off**
On

Accumulated Value Target Reached Flag—AVTVFL is the accumulated value's "target value reached" flag. It is turned On whenever PVCALC ≥ AVTV. Parameter AVTV contains the target value last entered by the operator.

-B-

B (AutoMan)

Type: **Real**
Lock: **Oper**
Default: **0.0**
PtRes: **HPM**
Range: **N/A**

Overall Bias—Defines the overall bias which consists of BO plus BI. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

B (MulDiv, RegCtl Summer)

Type: **Real**
Lock: **Oper**
Default: **0.0**
PtRes: **HPM**
Range: **N/A**

Overall Bias—Defines the overall bias which consists of BO plus BI. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

B0 (AutoMan, MulDiv, RegCtl Summer)

Type: **Real**
Lock: **View**
Default: **0.0**
PtRes: **HPM**
Range: **N/A**

Last Operator-Entered Output Bias

B1 (RatioCtl)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: **N/A**

Output Bias Constant—If the Calcultr PV algorithm is being used in conjunction with this algorithm, the value of B1 should be the same as C3.

B2 (RatioCtl)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: **N/A**

Bias for Input X2—If the Calcultr PV algorithm is being used in conjunction with this algorithm, the value of B2 should be the same as C4.

B1, B2, B3 (MulDiv)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range:

Bias for Inputs for X1, X2, and X3—

BADCTLFL

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Bad-control alarm not present)
On (Bad-control alarm present)

Bad-Control Alarm Flag—Indicates whether a bad control alarm has been detected.

BADCTLOP

Type: **E:$BADCTLO**
Lock: **Engr**
Default: **No_Shed**
PtRes: **HPM**
Range: 0-**No_Shed** (The point holds its output and mode, resuming control after initialization upon recovery)
1-**ShedHold** (The mode sheds to manual, the mode attribute goes to operator, while the output is held and external mode switching is disabled)
2-**ShedLow** (The mode sheds to manual, the mode attribute goes to operator, while the output goes to -6.9% and external mode switching is disabled)
3-**ShedHigh** (The mode sheds to manual, the mode attribute goes to operator, while the output goes to 106.9% and external mode switching is disabled)
4-**ShedSafe** (The mode sheds to manual, the mode attribute goes to operator, while the output goes to SafeOP and external mode switching is disabled. If SafeOP is NaN, the output is held as if the Bad Control Option is ShedHold.)

Bad Control Option—Indicates if the mode sheds to manual when bad PV or CV occurs for regulatory control points. It also shows the value of the output.

BADCTLPR

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

Bad Control Alarm Priority—Defines the priority of the bad control alarm.

BADIOLPF (ProcMod)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** ( )
On (IOL prefetch value is bad)

Bad IOL Prefetch—Set to ON, by the system if any IOL prefetch value is bad.

BADOCFL (RegCtl)

Type: **Logical**
Lock: **View**
Default: **OFF**
PtRes: **HPM**

Bad Output Connection Flag (BADOC) Alarm—ON indicates that the RegCtl point cannot drive at least one Analog Output point (out of 4 possible). The alarm appears on the Alarm Summary display and in the Real Time Journal as a BADOC alarm. If it is the highest level alarm on the point, it appears on the Point Detail or Group displays as BOC.

Range: **Off** (Point is not in alarm)
On (Point is in alarm)

BADOC1FL

Type: **Logical**
Lock: **View**
Default: **OFF**
PtRes: **HPM**
Range: **Off**
On

Bad Output Connection Flag 1—ON indicates that the RegCtl point cannot drive Output 1 to an AO point (if configured).

BADOC2FL

Type: **Logical**
Lock: **View**
Default: **OFF**
PtRes: **HPM**
Range: **Off**
On

Bad Output Connection Flag 2—ON indicates that the RegCtl point cannot drive Output 2 to an AO point (if configured).

BADOC3FL

Type: **Logical**
Lock: **View**
Default: **OFF**
PtRes: **HPM**
Range: **Off**
On

Bad Output Connection Flag 3—ON indicates that the RegCtl point cannot drive Output 3 to an AO point (if configured).

BADOC4FL

Type: **Logical**
Lock: **View**
Default: **OFF**
PtRes: **HPM**
Range: **Off**
On

Bad Output Connection Flag 4—ON indicates that the RegCtl point cannot drive Output 4 to an AO point (if configured).

BADOCPR (RegCtl)

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **NoAction**
JnlPrint
Printer
Journal
Low
High
Emergncy

Bad Output Connection (BADOC) Alarm Priority—Indicates the priority of the Bad Output Connection (BADOC) alarm

Helpful Hint: The value of this parameter can be changed on the Point Detail display with Engineering keylevel access.

BADOCOPT (RegCtl)

Type: **Logical**
Lock: **Eng/Pb**
Default: **OFF**
PtRes: **HPM**
Range: **Off** (BADOC alarms are suppressed)
On (BADOC alarms are permitted)

Bad Output Connection Alarm Option (BADOC)—ON indicates that the Bad Output Connection (BADOC) alarm can be generated (or is permitted).

Helpful Hint: The value of this parameter can be changed on the Point Detail display with Engineering keylevel access.

BADPVFL

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (PV is not bad)
On (PV is bad)

Bad PV Flag—Indicates that a bad PV value has been detected at this data point. For an analog input, a bad PV is defined as a PV whose value is NaN (Not a Number).

BADPVFL (DI)

Type: **Logical**
Lock: **View**
Default: **On**
PtRes: **HPM**

Bad PV Flag—Indicates that a bad PV value has been detected at this data point. BADPVFL is shown on the detailed display only when PNTFORM = Full.

For a Digital Input, the Bad PV Flag is on when:

- The PV source is not equal to Manual and DITYPE is set to Accumulator.
- The PV source has just been switched to Substituted but the PV has not yet been updated.
- The PV source = Substitute or Auto and PTEXECST = Inactive or the module is not in the RUN state.
- PVSOURCE = Auto and there is no FTA connected or there is a Soft Fail that is preventing this chanel from working.

Range: **Off** (PV is not bad)
On (PV is bad)

BADPVFL (DevCtl, DigComp)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (PV is not bad)
On (PV is bad)

Bad PV Flag—For a Digital Composite or Device Control point, the Bad PV Flag is is set to ON when the PV is bad.

BADPVPR

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**

Bad PV Alarm Priority—Defines the priority of the bad PV alarm.

Range: **Emergncy** (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
JnlPrint (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

BADPVTXT (DevCtl, DigComp, NIM)

Type: **String_8**
Lock: **PtBld**
Default: **BAD**
PtRes: **NIM**

Bad PV State Descriptor—Defines the state descriptor that is displayed when the digital composite or device control point state is indeterminate or bad. The bad state can result when the PV input signals from the process are in an inconsistent state (e.g., for a valve, the limit switches indicating open and closed are on at the same time). This state descriptor is configured on a per point basis and is valid only when the PVTXTOPT parameter is On.

Range: The permissible character set for the up to eight character descriptor is as follows:
Alphabetics A-Z (upper case only)
Numerics 0-9,
Underscore (_)

BADSVFL (DevCtl)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Good data being read)
On (SV parameter = BAD or NaN)

Bad SV Alarm Flag—Indicates a bad secondary value alarm.

BADSVPR (DevCtl)

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

Bad SV Alarm Priority—Indicates the alarm priority for the secondary value.

BCAMODFL (RegCtl)

Type: **Logical**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **Off -** (point is not in Backup Cascade mode)
On - (point is in Backup Cascade mode)

Backup Cascade Mode Flag—Indicates if the mode of the point is Backup Cascade.

BCOMPOPT (FlowComp)

Type: **Bad Compensation Input Option—**
Lock:
Default:
PtRes: **HPM**

Range: **Set_PVCALC_Bad**
Use_Last_Goood_Comp_Term
Use_LastGood_Comp_Input

BFF (PidFf)

Type: **Real** **Feed Forward Input Bias—**Defines the bias value for multiplicative action.
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: **N/A**

BHALMFL1–BHALMFL7

Type: **String_2** **Alarm Flags**
Lock: **View**
Default:
PtRes: **NIM**
Range: **Hexadecimal characters 00 - FF**

BIAS (Pid)

Type: **Real** **Bias—**Defines the value which is added to the SP.
Lock: **Oper**
Default: **0.0**
PtRes: **HPM**
Range: **BSLOLM** to **BSHILM**

BLK_INFO

Type: **Blind Record** **Function Block Summary Information—**Provides Function Block summary information needed by the NIM for checkpointing
Lock: **View**
Default: **N/A**
PtRes: **IOP**
Range:

BNDRESET (NIM, HPM Box)

Type: **Logical** **Bounds (Minimum/Maximum) Statistics Reset Flag—**A write of ON resets the following maximum/minimum statistics to their default values: HPM CPU free percentage events, UCN transaction, UCN parameter statistics and UCN average statistics.
Lock: **Operator**
Default: **Off**
PtRes: **HPM**
Range: **Off/On**

Helpful Hint:	A read of BNDRESET always returns OFF.

BNDRSTIM (NIM, HPM Box)

Type: **Time** **Time of Last Bounds (Minimum/Maximum) Statistics Reset.**
Lock: **View**
Default: **Time of HPM Startup**
PtRes: **HPM**
Range: **N/A**

Helpful Hint: This statistic can be viewed on the Toolkit Displays.

BOXCLR(0)–(2) (DevCtl, DigComp)

Type: **E:BOXCOLOR** **Box Color for Digital Displays—**Defines the color of the upper, middle, and lower boxes that are used to display the current state of the point on the Group and Detail Displays. The lower box and its default (Red) do not apply if NOSTATES = 2 for this point.
Lock: **Eng/PB**
Default: **Green** [Upper-box default (State 1)]
Yellow [Middle-box default (State 0)]
Red [Lower-box default (State 2)]
PtRes: **NIM**
Range: **Red**
Green
White
Black
Cyan
Yellow
Blue
Magenta

Helpful Hint: BOXCLR has an access lock of View if PNTFORM = Component.

BOXCLR(0)–(1) (DigIn, Flag)

Type: **E:BOXCOLOR** **Box Color for Digital Displays—**Defines the color of the upper and lower boxes that are used to display the current state of the point on the Group and Detail Displays. Boxes are applicable when DITYPE is Latched or Status.
Lock: **Eng/PB**

Default: **Green** [Upper box default color (State 1)]
Yellow [Lower-box default color(State 0)]
PtRes: **NIM**
Range: **Red**
Green
White
Black
Cyan
Yellow
Blue
Magenta

BSHILM

Type: **Real**
Lock: **Supr**
Default: **50.0**
PtRes: **HPM**
Range: **≥ BSLOLM, NaN**

Bias High Limit—Defines the upper limit of the bias.

Helpful Hint:	Entering NaN disables the BSHILM function with NaN being stored in the database.

BSLOLM

Type: **Real**
Lock: **Supr**
Default: **-50.0**
PtRes: **HPM**
Range: **≤ BSHILM, NaN**

Bias Low Limit—Defines the lower limit of the bias.

Helpful Hint:	Entering NaN disables the BSLOLM function with NaN being stored in the database.

BYPASS (DevCtl, DigComp)

Type: **Logical**
Lock: **Oper**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Interlocks not bypassed)
On (Interlocks bypassed)

Interlock Bypass—Allows bypassing the permissive and override interlocks when ON. The Safety Override Interlock (SI0) is not affected. Only applies if Override Option (OROPT) is selected.

BYPASS (ORSel)

Type: **Logical**
Lock: **Oper**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Bypass of inputs is not allowed)
On (Bypass of inputs is allowed)

Override Input Bypass Enable—Allows the operator to select the bypass function for the X1-X4 inputs.

BYPASSX1–BYPASSX4 (ORSel)

Type: **Logical**
Lock: **Oper**
Default: **BYPASSX1 = Off**
BYPASSX2–X4 = On
PtRes: **HPM**
Range: **Off**
On

Bypass X1–X4 Input—Refer to the *HPM Control Functions and Algorithms* manual for a detailed description. BYPASSXn being On does not bypass X1–X4 unless BYPASS = On.

-C-

C (FlowComp, Summer, Totalizer, and VdtLdLg)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **Anything except NaN**

Scale Factor—Value in C is used in the calculation of PVCALC. Refer to the *HPM Control Functions and Algorithms* manual for the equation.

C1–C2 (FlowComp)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **C1 ≥ 0.1**
C2 ≥ 0.1

Correction Constants—Values in C1 and C2 are used in the calculation of PVCALC, and serve as factors in compensating for assumed design conditions. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

C1–C2 (PI)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **C1 ≥ 0**
C2 > 0

Scaling Constants—Values in C1 and C2 are used in the calculation of PVCALC. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

C1–C2 (VdtLdLg)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **≥ 0.0**

Scaling Constant For Input P1–P2—Values in C1 and C2 are used in the calculation of TD (fixed time delay) and TDNEW (calculated new delay time).

C1–C4 (Calcultr)

Type: **Real**
Lock: **Supr**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

Intermediate Results of Calculations

C1–C6 (Summer)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **N/A**

Scaling Constants 1-6—Defines the scaling constants to be used with the respective inputs P1-P6.

C1–C4DESC (Logic)

Type: **String_8**
Lock: **Engr**
Default: **Blank**
PtRes: **HPM**
Range: **8 Character String**

Custom Alarm Descriptors—Defines the state for each of the four custom alarms.

C1–C4FL (Logic)

Type: **Logical**
Lock: **Program**
Default: **Off**
PtRes: **HPM**
Range: **Off** (A custom alarm is not active)
On (A custom alarm is active)

Custom Alarm Flags—Defines the state for each of the four custom alarms. These flags can be written to if C1–C4SRC=None.

C1–C4PR (Logic)

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **NoAction**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

Custom Alarm Priorities—Defines the alarm priorities for each of the four custom alarms.

C1–C4SRC (Logic)

Type: **E:$LGALSRC**
Lock: **PtBld**
Default: **None**
PtRes: **HPM**
Range: **NONE** (No source configured for alarms)
L1..L12 (Alarm source is the configured input connection; they can be either On or Off)
SO1..SO24 (Alarm source is the status output (SOn) from another logic block)
FL1..FL12 (Alarm source is a local flag; they can be either On or Off)

Custom Alarm Source—Indicates the alarm source for each of the four custom alarms.

CABLESTS (NIM)

Type: **Integer** **Overall Cable Status for a UCN Node**
Lock: **View**
Default: **N/A**
PtRes: **NIM**
Range: 0-(Both cables are OK)
1-(Cable A has failed)
2-(Cable B has failed)
3-(Both cables have failed)

CALCEXP (Calcultr)

Type: **String_40** **Calculator Expression—**Allows the user to set up an equation that can be up to 40 characters in length, which is to be solved by the Calcultr algorithm. Input values P1-P6 and intermediate results values C1-C4 can be used in the equation.
Lock: **Eng/PB**
Default: **blank**
PtRes: **HPM**
Range: **N/A**

CALIBALL (1)–(168)

Type: **Logical** **Full Calibration Enable Flag**
Lock: **Engr**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Card calibration is disabled)
On (Card calibration is enabled)

CALIBRJ (1)–(168)

Type: **Logical** **Reference Junction Calibration Enable Flag**
Lock: **Eng/PB**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Disable Reference Junction calibration)
On (Enable Reference Junction calibration)

CASMODFL (RegCtl)

Type: **Logical** **Cascade Mode Flag—**Indicates whether the current mode of the point is Cascade.
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **On -** (point is in cascade mode)
Off - (point is not in cascade mode)

CASREQ (AnalgOut, RegCtl)

Type:	**E:CASREQ**	**Remote Cascade Request Flag—**Defines whether the remote cascade mode has been requested for the data point. The remote cascade mode exists when MODE is changed to Cas and RCASOPT is Ddc or DdcRsp. When a request to change MODE to Cas is received from a US or a program, MODE does not immediately change to Cas. Instead, CASREQ is set to Request and a -C appears to the right of the mode indicator on the Group and Detail displays. When continuous control in an AM determines that CASREQ contains Request, it requests the mode to go to Cas, and changes CASREQ to NotReq.
Lock:	**Prog**	
Default:	**NotReq**	
PtRes:	**HPM**	

Should the point shed while it is in the remote cascade mode, MODE goes to the state defined in SHEDMODE, and CASREQ goes back to Request.

Range:	0-**NotReq**	(Remote cascade mode request not made)
	1-**Request**	(Remote cascade mode request made; operator or program has requested the cascade mode)

Helpful Hint: CASREQ does not apply for an AnalgOut point if RCASOPT = None.
CASREQ does not apply for a RegCtl point unless RCASOPT = Spc, Ddc, or DdcRsp.
If Spc has been entered for the RCASOPT parameter, the AM writes to the setpoint.
Ddc is the only remote cascade option for an analog output point.

CHPINHWY (UCN)

Type: **E:CHPINDAC** **Automatic Checkpoint Inhibit**
Lock: **Supr**
Default: **Enable**
PtRes: **NIM**
Range: 0-**Enable** (Enable automatic checkpointing of data bases on this UCN)
1-**Inhibit** (Inhibit automatic checkpointing)

CHPINOPR (HPM Box)

Type:	**E:CHPINDAC**	**Automatic Checkpoint Inhibit Operation—**Defines whether automatic database saves are to be performed for the devices connected to this NIM.
Lock:	**Supr**	
Default:	**Enable**	
PtRes:	**NIM**	
Range:	**Enable** (Automatic database saves are enabled)	
	Inhibit (Automatic database saves are inhibited)	

CIDSTN(1)–CIDSTN(4) (RegCtl)

Type: **Prm_ID**
Lock: **PtBld**
Default: **Based on CTLALGID, CTLEQN, N**
PtRes: **HPM**
Range: **N/A**

Control Input Connection Destination—Defines the parameter name (PV, SP, etc.) in the RegCtl point that is to receive the value fetched using the "Tagname.Parameter" or the hardware reference address specified in parameter CISRC, Control Input Connection Source.

Helpful Hint:	1. CIDSTN must contain a legitimate parameter of one to eight characters. 2. Default to PV, SP, or some other parameter depends on parameters CTLALGID, CTLEQN, and M.

CISRC(1)–CISRC(4) (RegCtl)

Type: **Ent.Prm**
Lock: **PtBld**
Default: **null.null**
PtRes: **HPM**

Control Input Connection Source—Defines the "Tagname.Parameter" of the parameter whose value is to be obtained and then stored in one of up to four RegCtl algorithm inputs. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: Use Tagname.Parameter for tagged points where Tagname can be up to 16 characters, and the permissible character set is as follows:

Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.
An * is used to default to this point's tag name.
Parameter name can be up to eight characters and must be a legitimate parameter name.

CLBACK (ProcMod)

Type: **Integer**
Lock: **Engr**
Default: **0**
PtRes **HPM**
Range: (**0 - 240**)

Number of Backward Branches -- Specifies how many backward brances may occur when executing GOTO WHEN ERROR, & REPEAT, before preemption occurs. 0 = preempt every backward branch.

CLPZMXC (UCN)

Type: **Logical** **Overall Cable Status for UCN Cable A**
Lock: **View**
Default: **N/A**
PtRes **NIM**
Range: **Off** (Cable A status is OK)
On (Cable A status is not OK)

CLPZMXP (UCN)

Type: **Logical** **Overall Cable Status for UCN Cable B**
Lock: **View**
Default: **N/A**
PtRes **NIM**
Range: **Off** (Cable B status is OK)
On (Cable B status is not OK)

CMD (RegCtl)

Type: **Logical**
Lock: **Configurable**
Default: **Off**
PtRes: **APM**
Range: **Off** (PV moved after the output command)
On (PV did not move after the output command)

Command Fail Alarm Flag—Indicates if the PV failed to move after the output command within the allowed command fail time. Command Fail Alarm priority is determined by CMDDISPR.

CMDDISFL (DevCtl, DigComp)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (No command disagree alarm)
On (Command disagree alarm has been detected by this point)

Command Disagree Alarm Flag—Indicates whether a field device did not go to the commanded state within the allowed feedback time.

Helpful Hint:	A slow-responding field device can cause a premature alarm. If so, adjust the time in parameter FBTIME.

CMDDISPR (DevCtl, DigComp)

Type: **E:ALPRIOR** **Command Disagree Alarm Priority—**Defines the alarm priority of command disagree, command fail, and uncommanded change alarms.
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range **Emergncy** (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
JnlPrint (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

CMDFALFL (DevCtl, DigComp)

Type: **Logical** **Command Fail Alarm Flag—**Indicates if the PV failed to move after the output command within the allowed command fail time. Command Fail Alarm priority is determined by CMDDISPR.
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (PV moved after the output command)
On (PV did not move after the output command)

CMDFALTM (DevCtl, DigComp)

Type: **Integer** **Command Fail Timeout—**Sets the amount of time (in seconds) that the point should wait before generating a "command fail" alarm, if the PV has not changed after changing the output. Command Fail Alarm priority is determined by CMDDISPR.
Lock: **Supr if CMDFALTM is changed from a non-zero value to a zero value, else Eng/PB**
Default: **0**
PtRes: **HPM**
Range: **0** to **999 seconds** (0 indicates command fail alarming is disabled)

CMDHWREV

Type: **String_2** **HPMM Communications Daughter Card Revision**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Hexadecimal characters 00 - FF**

CMFLTIME (HPM BOX)

Type: **Time** **HPMM Communications Failure Time—**
Lock: **View**
Default: **N/A**
PtRes: **HPM**

> *Helpful Hint:* If a value of 0 is returned for the time from the UCN, a parameter status of Parameter Invalid is returned on the LCN.

Range:

CMIDTXT (HPM BOX)

Type: **String_16** **HPMM Communications Identification Text String**
Lock: **View**
Default: **Blank**
PtRes: **HPM**
Range: **Hexadecimal characters 00 - FF**

CMPLTIME

Type: **Time** **Compile Time—**Specifies the sequence compile time (CL object header)
Lock: **View**
Default: **0**
PtRes: **HPM**
Range:

CNFERRFL

Type: **Logical** **Configuration Error Flag—**This flag is set if any configuration requirement is violated.
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: Off
On

CNFERRPR

Type: **E: ALPRIOR** **Configuration Error Priority—**
Lock: **Eng/PB**
Default: **Low**
PtRes: **HPM**
Range: **Low**

CNFLUA(n)

Type:	**Real**	**Configured Link Units on Link A—**
Lock:	**View**	n = 1 - 64 for per cycle totals
Default:	**0.0**	n = 257 - 320 for per cycle non-SI IOP loading
PtRes:	**HPM**	n = 513 - 576 for per cycle SI array slot loading
Range:		

CNFLUB(n)

Type:	**Real**	**Configured Link Units on Link B—**
Lock:	**View**	n = 1 - 64 for per cycle totals
Default:	**0.0**	n = 257 - 320 for per cycle non-SI IOP loading
PtRes:	**HPM**	n = 513 - 576 for per cycle SI array slot loading
Range:		

CNFMU

Type:	**Integer**	**Configured Memory Units**—Configured size of slot in Memory units.
Lock:	**View**	
Default:	**N/A**	
PtRes:	**HPM**	
Range:		

CNFPU(1 - 64) (HPM Box)

Type:	**Real**	**Configured Process Units Per Cycle—**
Lock:	**View**	
Default:	**0**	
PtRes:	**HPM**	
Range:		

CNFPU

Type: **Real**
Lock: **PtBld**
Default: **2.0**
PtRes: **HPM**
Range:

Configured Process Units Per Cycle—Process Units Configured as being required to execute point processing.

Helpful Hint:	Can only be written for ProcMod points.

CNFPUP (1 -64)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range:

Configured PUs Percent—Specifies the Configured Process Units in percent

CNTLLOCK

Type: **E:ACCLVL**
Lock: **Engr**
Default: **OPERATOR**
PtRes: **HPM**

Control Lock—Attempts to write values in the following parameters are subject to the access-lock value contained in CNTLLOCK. The check is bypassed for the exceptions.

Parameter	Exceptions
PROCMOD	New value = START
SEQEXEC	None
SEQMODE	None
OVERPHAS	SEQEXEC = FAIL or ERROR
OVERSTEP	SEQEXEC = FAIL or ERROR
OVERSTAT	SEQEXEC = FAIL or ERROR

Range: 0-**OPERATOR -** Operator and higher keylock positions allow store access.
1-**SUPERVIS -** Supervisor and higher keylock positions allow store access.
2-**ENGINEER -** Engineer and higher keylock positions allow store access.
3-**PROGRAM -** Only the program has store access.

CODSTN(1)–CODSTN(4) (RegCtl)

Type: **Universal Ent.Prm**
Lock: **PtBld**
Default: **null.null**
PtRes: **HPM**

Control Output Connection Destination—Defines up to four different "Tagname.Parameter" or hardware reference address destinations to which the output value from RegCtl point is to be written. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: Use Tagname.Parameter for tagged points where Tagname can be up to 16 characters, and the permissible character set is as follows:

- Alphabetics A-Z (uppercase only)
- Numerics 0-9 (an all numeric tag name is not allowed)
- Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
- Embedded space characters are not allowed.
- An * is used to default to this point's tag name.
- Parameter name can be up to eight characters and must be a legitimate parameter name.

Use the hardware reference address !MTmmSss.Parameter for untagged or tagged points where

- MT is the IOP type, such as AO (analog output)
- mm is the IOP Card number (1-40)
- ss is the slot number on the IOP Card (refer to SLOTNUM parameter)
- Parameter name can be up to eight characters and must be a legitimate parameter name.

COMCFAVG (HPM Box)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **0 - 100**

Average HPM Communication CPU Free Percentage—The average percent of time the Communications Processor is not busy.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

COMCFMAX (HPM Box)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **0 - 100**

Maximum HPM Communication CPU Free Percentage—The maximum percent of time the Communications Processor is not busy.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

COMCFMIN (HPM Box)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **0 - 100**

Minimum HPM Communication CPU Free Percentage—The minimum percent of time the Communications Processor is not busy.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

COMDAUGH

Type: **Logical** **HPMM Communications Daughter Card Present Flag.**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Off** (No daughter card present)
On (Daughter card present)

COMDAY

Type: **Integer** **Creation Day of HPMM Communications Personality.**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **1 - 31**

COMFWREV

Type: **String_2** **HPMM Communications Firmware Revision.**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Hexadecimal characters 00 - FF**

COMDGAVG (HPM Box)

Type: **Real** **Average Diagnostic cycle time (in minutes) in the Comm CPU—**
Lock: **View**
Default: **0.0**
PtRes: **HPM**
Range:

COMDGMAX (HPM Box)

Type: **Real** **Maximum Diagnostic cycle time (in minutes) in the Comm CPU—**
Lock: **View**
Default: **0.0**
PtRes: **HPM**
Range:

COMHOUR (HPM BOX)

Type: **Integer** **HPMM Communications Personality Creation Date-Hour**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range:

COMHWREV

Type: **String_2** **HPMM Communications Hardware Revision.**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Hexadecimal characters 00 - FF**

COMLUAVG (1) - (2) (HPM Box)

Type: **Real** **Average HPM IOL Utilization (inPercent) by the Comm CPU, per I/O Link—**
Lock: **View**
Default: **0.0**
PtRes: **HPM**
Range: **0 - 100**

COMLUMAX (1) - (2) (HPM Box)

Type: **Real** **Maximum HPM IOL Utilization (inPercent) by the Comm CPU, per I/O Link—**
Lock: **View**
Default: **0.0**
PtRes: **HPM**
Range: **0 - 100**

COMMAND (DigIn)

Type: **E:COMMAND** **Accumulator Commands—**Allow the operator to control the accumulator.
Lock: **Oper**
Default: **None**
PtRes: **HPM**
Range: 0-**None** (No effect on accumulator)
1-**Start** (Start the accumulator)
2-**Stop** (Stop the accumulator)
3-**Reset** (Reset the accumulation to zero)

Helpful Hint:	COMMAND applies only when DITYPE = Accum.

COMMAND (STI)

Type: **E:COMMAND**
Lock: **Oper**
Default: **None**
PtRes: **HPM**

Command—Allows the user to do database transfers between the STI point and the smart transmitter, and to calibrate the transmitter.

> **NOTE**
>
> During an up-load operation, previously unseen data is read from the transmitter database and stored in the STI database. If this data is not desired, the data can be restored by using the checkpoint restore or load IDF functions.

Range: 0-**None** (A command has not been issued by the STI point)
1-**DnLoadDb** (Loads the transmitter parameters from the STI point data base into the transmitter)
2-**UpLoadDb** (Loads the transmitter data base from the transmitter into the STI point)
3-**Set_LRV** (Sets the Lower Range Value)
4-**Set_URV** (Sets the Upper Range Value)
5-**Cor_LRV** (Corrects the Lower Range Value)
6-**Cor_URV** (Corrects the Upper Range Value)
7-**Cor_Inpt** (Corrects the zero point for the PV value)
8-**RstCor** (Sets all input calibration parameters to their default values)

Helpful Hint:	If PV or PV_SV has been entered for the DECONF parameter, the only command supported is DnLoadDB.

COMMAND (Timer)

Type: **E:COMMAND** **Timer Commands—**Allow the operator to control the operation of the timer data point.
Lock: **Oper**
Default: **None**
PtRes: **HPM**
Range: 0-**None** (No effect on the timer)
1-**Start** (Starts the timer)
2-**Stop** (Stops the timer)
3-**Reset** (Resets the timer to zero)
4-**RestStrt** (Resets the timer, then starts the timer)

COMMAND (Totalizr)

Type: **E:COMMAND** **Totalizer Commands—**Allow the operator to control the operation of the totalizer.
Lock: **Oper**
Default: **None**
PtRes: **HPM**
Range: 0-**None** (No effect on totalizer)
1-**Start** (Starts the totalizer)
2-**Stop** (Stops the totalizer)
3-**Reset** (Resets the totalizer to RESETVAL)

COMMIN (HPM BOX)

Type: **Integer** **HPMM Communications Personality Creation Date-Minute**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range:

COMMONTH

Type: **Integer** **Creation Month of HPMM Communications Personality**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **1 - 12**

COMNAME

Type: **String_8** **HPMM Communications Personality Name**
Lock: **View**
Default:
PtRes: **HPM**
Range:

COMPHILM (FlowComp)

Type:	**Real**	**Compensation Term High Limit—**Defines the upper limit of the COMPTERM (compensation term) parameter.
Lock:	**Supr**	
Default:	**1.25**	
PtRes:	**HPM**	
Range:	**COMPLOLM** to **10.0, NaN**	

Helpful Hint:	Entering NaN disables high-limit checking by forcing its value to the extreme (10.0).

COMPLOLM (FlowComp)

Type:	**Real**	**Compensation Term Low Limit—**Defines the lower limit of the COMPTERM (compensation term) parameter.
Lock:	**Supr**	
Default:	**0.8**	
PtRes:	**HPM**	
Range:	**0.0** to **COMPHILM, NaN**	

Helpful Hint:	Entering NaN disables low-limit checking by forcing its value to the extreme (0.0).

COMPTERM (FlowComp)

Type:	**Real**	**Compensation Term—**This term differs in each of the five flow compensation equations, A through E. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.
Lock:	**View**	
Default:	**1.0**	
PtRes:	**HPM**	
Range:	**COMPLOLM** to **COMPHILM**	

COMRDRRV

Type:	**Integer**	**Rdr Revision**
Lock:	**View**	
Default:		
PtRes:	**HPM**	
Range:		

COMRDRVS

Type:	**Integer**	**Rdr Version**
Lock:	**View**	
Default:		
PtRes:	**HPM**	
Range:		

COMREV

Type: **Integer** — **HPMM Communications Software Revision**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

COMVERS

Type: **Integer** — **HPMM Communications Software Version**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

COMYEAR

Type: **Integer** — **Creation Year of HPMM Communications Personality**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **0 - 99**

CONTCUT

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**

Contact Cut Out —Defines whether alarms detected at this data point are to be cut out to prevent this data point's alarms from being reported to the operator. The alarms continue to be reported to the AM or CM through the EIPPCODE parameter.

CONTCUT can be used to cutout alarms on a point when the alarms are generated because of specific conditions at other points which themselves have alarms. As an example, the user could configure a logic point so that the logic point would monitor the nuisance alarm conditions and then store the contact cutout state of this point using an output connection. It can also be stored by the sequence program in the HPM or the AM which could monitor the process conditions to determine when the alarms have to be suppressed.

Range: **Off** (Alarms are not cut out)
On (Alarms are cut out)

Helpful Hint: Cutout alarms behave the same as inhibited alarms; that is, when a point's contact cutout state is true—

- alarms are not distributed to the US or HM
- return to normal events are not distributed to the US or HM
- EIP events triggered by the alarm condetion are not distributed

For HPM Box Flag points, CUTOUT applies to only slots 1–128.

COUNTDWN (DigIn)

Type: **Logical**
Lock: **Eng/PB**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Accumulator is to count up)
On (Accumulator is to count down)

Accumulator Count Down Flag—Determines whether the accumulator is to count down or count up.

Helpful Hint:	COUNTDWN configuration requires DITYPE = Accum.

CPMSGSEC (NIM PSDP)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **NIM**
Range: **N/A**

Number of Checkpoint Messages—Specifies the Number of Checkpoint Messages per second.

CPTIMAVG (NIM PSDP)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **NIM**
Range: **N/A**

Average Time to Complete a Checkpoint Request—Specifies the Average Time (in msec.) to Complete a Checkpoint Request.

CPTIMMAX (NIM PSDP)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **NIM**
Range: **N/A**

Maximum Time to Complete a Checkpoint Request—Specifies the Maximum Time (in msec.) to Complete a Checkpoint Request.

CRIOLORN (1) - (4) (HPM Box)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**

Current Hour IOL Fetch/Store Overrun Counter—A counter that accumulates and shows the number of I/O Link fetch/store time outs that have occurred during the current hour.

In arrays 1 through 4, the counter is indexed by the cycle.
In array 0, the counter is totaled for all cycles.

Range: **≥ 0**

CRPPXORN (0 - 8) (HPM Box)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**

Current Period Point Processing Overruns Per Cycle—A counter that accumulates and shows the number of HPMM point processing overruns that have occurred during the current hour.

In arrays 1 through 8, the counter is indexed by the cycle
In array 0, the counter is totaled for all cycles.

Range: **≥ 0**

CRUCNORN (HPM Box)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **≥ 0**

Current-Hour UCN Access Overruns —Indicates the number of UCN access overruns that have occurred in the current hour. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of overrun handling.

CTFLTIME (HPM Box)

Type: **Time** — **HPMM Control Failure Time—**defines
Lock: **View**
Default: **N/A**
PtRes: **HPM**

> *Helpful Hint:* If a value of 0 is returned for the time from the UCN, a parameter status of Parameter Invalid is returned on the LCN.

Range: **N/A**

CTIDTXT (HPM BOX)

Type: **String_16** — **HPMM Control Functionality ID Text String—**defines
Lock: **View**
Default: **Blank**
PtRes: **HPM**
Range: **N/A**

CTLACTN

Type: **E:POLARITY** — **Control Action—**Defines the direct/reverse action of this algorithm's output.
Lock: **Eng/PB**
Default: **Reverse**
PtRes: **HPM**
Range: 0-**Direct** (As PV increases, output increases)
1-**Reverse** (As PV increases, output decreases)

CTLALGID (RegCtl)

Type: **E:$PMMCTAL** — **Control Algorithm Identifier—**Defines the algorithm that is to be used for this RegCtl point.
Lock: **PtBld**
Default: **Null**
PtRes: **HPM**
Range: 0-**Null** (No algorithm selected)
1-**Pid** (Proportional, Integral, Derivative)
2-**PidFf** (PID with Feedforward)
3-**PidErfb** (PID with External Reset Feedback)
7-**RatioCtl** (Ratio Control)
8-**RampSoak** (Ramp Soak)
9-**AutoMan** (Auto Manual Station)
10-**IncrSum** (Incremental Summer)
11-**Switch** (Switch)
12-**ORSel** (Override Selector)
13-**PosProp** (Position Proportional)
14-**PIDPosProp** (PID with Position Proportional output)

CTLCFAVG (HPM Box)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **0 - 100**

Average HPM Control Processor CPU Free Percentage—The average percent of time the HPM Control Processor is not busy.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays

CTLCFMAX (HPM Box)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **0 - 100**

Maximum HPM Control Processor CPU Free Percentage—The maximum percent of time the HPM Control Processor is not busy.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

CTLCFMIN (HPM Box)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **0 - 100**

Minimum HPM Control Processor CPU Free Percentage—The minimum percent of time the HPM Control Processor is not busy.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

CTLDAY

Type: **Integer**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **1 - 31**

Creation Day of HPMM Control Personality

CTLDGAVG (1) - (2) (HPM Box)

Type: **Real**
Lock: **View**
Default: **0.0**
PtRes: **HPM**
Range: **N/A**

Average Diagnostic cycle time (in minutes) in the Control CPU—

CTLDGMAX (1) - (2) (HPM Box)

Type: **Real** — **Maximum Diagnostic cycle time (in minutes) in the Control CPU—**
Lock: **View**
Default: **0.0**
PtRes: **HPM**
Range: **N/A**

CTLEQN (AutoMan)

Type: **E:ALGOEQN** — **Control Equation Type**
Lock: **Eng/PB**
Default: **EqA**
PtRes: **HPM**
Range: 0-**EqA** (CV = X1 + B +BI)
1-**EqB** (CV = X1 + (K*X2) + BI)

CTLEQN (ORSel)

Type: **E:ALGOEQN** — **Control Equation Type—**Defines whether the highest or the lowest input is to be selected.
Lock: **PtBld**
Default: **EqA**
PtRes: **HPM**
Range: 0-**EqA** (Selects the highest input)
1-**EqB** (Selects the lowest input)

CTLEQN (Pid)

Type: **E:ALGOEQN** — **Control Equation Type—**Defines how Proportional (P) or gain, Integral (I) or reset, and Derivative (D) action is applied to a PID-type algorithm's calculated Error (PV - SP).
Lock: **PtBld**
Default: **EqA**
PtRes: **HPM**
Range: 0-**EqA** (P, I, and D act on Error)
1-**EqB** (P and I act on Error, D acts on PV)
2-**EqC** (I acts on Error, P and D act on PV)
3-**EqD** (Integral-only control)

CTLEQN (Switch)

Type: **E:ALGOEQN** — **Control Equation Type—**Defines whether the operator, the user-written program, or the logic slot controls the selection of one of the four inputs (X1-X4) as the input to this algorithm.
Lock: **Eng/PB**
Default: **EqA**
PtRes: **HPM**
Range: 0-**EqA** (Operator controls switch position)
1-**EqB** (Program or logic point controls switch position)

CTLHOUR (HPM Box)

Type: **Integer** **HPMM Control Personality Creation Date-Hour**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

CTLMIN (HPM Box)

Type: **Integer** **HPMM Control Personality Creation Date-Minute**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

CTLMONTH

Type: **Integer** **Creation Month of HPMM Control Personality**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **1 - 12**

CTLNAME

Type: **String_8** **HPMM Control Personality Name**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

CTLOPT (HPM Box)

Type: **Logical** **HPMM Control Processor Option**
Lock: **PtBld**
Default: **On**
PtRes: **HPM**
Range: **On** (All point types can be configured)
Off (DigComp, Logic, RegCtl, or RegPV points cannot be configured; only I/O points can be configured. This usually means that the control processor hardware is missing from the HPM).

CTLREDUN

Type: **Logical** **HPMM Control Redundancy Present Flag**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **Off**
On

CTLREV

Type: **Integer** **HPMM Control Personality Revision**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

CTLVERS

Type: **Integer** **HPMM Control Personality Version**
Lock: **View**
Default:
PtRes: **HPM**
Range: **N/A**

CTLYEAR

Type: **Integer** **Creation Year of HPMM Control Personality**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **0 - 99**

CTRLINIT (RegCtl)

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

Control Initialization Request Flag—A user-written program or a logic slot can cause a data point to initialize by setting the point's control initialization-request flag to On.

CURCOMFL

Type: **E:$PMMHFST** **Current HPMM Communications Board Failure**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Null** (Unknown Error)
Pwrdwn (Power Down)
Lr_Par (Parity Error)
Lr_Lram (Local Ram Error)
Lr_Ck (Local Ram Check)
Lr_Exc (Local Ram Exception)
Lr_Hrev (Local Ram Hardware Revision)
Mm_Hrev (Memory Board Hardware Revision)
Lr_Tmr (Local Ram Timer Error)
Lr_Ptrn (Local Ram Pattern Check Error)
Lr_Byte (Local Ram Byte Error)
Lr_Adcd (Local Ram Address Decode Test)
Lr_Addl (Local Ram Additional Check)
Lr_Clrr (Local Ram Scrub Incomplete)
Sr_Par (Shared Ram Parity)
Sr_Ptrn (Shared Ram Pattern Check Error)
Sr_Adcd (Shared Ram Address Decode Test)
Sr_Addl (Shared Ram Additional Checks)
Gr_Par (Global Ram Parity)
Gr_Ptrn (Global Ram Pattern Check Error)
Gr_Byte (Global Ram Byte Error)
Gr_Adcd (Global Ram Address Decode Test)
Gr_Addl (Global Ram Additional Checks)
Gr_Clrr (Global Ram Scrub Incomplete)
31_Nr (IOL Processor, No Response or Failure)
31_Aliv (IOL Processor, Transmitter Not Alive)
31_Iltn (IOL Processor, Illegal Transition)
Nmi_Unk (Unknown NMI Request)
Baducnn (UCN Address Parity or Duplicate Address)
Nr (No Response From Other Processor)
Mrft (Memory Reference Table (Pattern Build Fail)
Nomtos (No MTOS Readout)
Llc_Comm (LLC Communication Fatal Error)
Ucndrv (UCN Driver, Fatal Error)
Rd_Hrev (Redundancy Card Version/Revision Mismatch)
Sw_Error (Software Error)
Md_Hrev (Modem Card Version/Revision Mismatch)
Da_Ptrn (Daughter Card Pattern Test)
Da_Byte (Daughter Card Byte Write Test)
Da_Adcd (Daughter Card Address Decode)
Da_Addl (Daughter Card Additional Test)
Da_Clrr (Daughter Card Scrub Incomplete)
Rd_Snps (Redundancy Card 96 Kw Snapshot Error)
Rd_Bslk (Redundancy Card Bus Lock Fail)

CURCTLFL

Type: **E:$PMMHFST** **Current HPMM Control Failure**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Null** (Unknown Error)
Pwrdwn (Power Down)
Lr_Par (Parity Error)
Lr_Lram (Local Ram Error)
Lr_Ck (Local Ram Check)
Lr_Exc (Local Ram Exception)
Lr_Hrev (Local Ram Hardware Revision)
Mm_Hrev (Memory Board Hardware Revision)
Lr_Tmr (Local Ram Timer Error)
Lr_Ptrn (Local Ram Pattern Check Error)
Lr_Byte (Local Ram Byte Error)
Lr_Adcd (Local Ram Address Decode Test)
Lr_Addl (Local Ram Additional Checks)
Lr_Clrr (Local Ram Scrub Incomplete)
Sr_Par (Shared Ram Parity)
Sr_Ptrn (Shared Ram Pattern Check Error)
Sr_Adcd (Shared Ram Address Decode Test)
Sr_Addl (Shared Ram Additional Checks)
Gr_Par (Global Ram Parity)
Gr_Ptrn (Global Ram Pattern Check Error)
Gr_Byte (Global Ram Byte Error)
Gr_Adcd (Global Ram Address Decode Test)
Gr_Addl (Global Ram Additional Checks)
Gr_Clrr (Global Ram Scrub Incomplete)
31_Nr (IOL Processor, No Response or Failure)
31_Aliv (IOL Processor, Transmitter Not Alive)
31_Ilatn (IOL Processor, Illegal Transition)
Nmi_Unk (Unknown NMI Request)
Baducnn (UCN Address Parity or Duplicate Address)
Nr (No Response From Other Processor)
Mrft (Memory Reference Table - Pattern Build Fail)
Nomtos (No MTOS Readout)
Llc_Comm (LLC Communication Fatal Error)
Ucndrv (UCN Driver, Fatal Error)
Rd_Hrev (Redundancy Card Version/Revision Mismatch)
Sw_Error (Software Error)
Md_Hrev (Modem Card Version/Revision Mismatch)
Da_Ptrn (Daughter Card Pattern Test)
Da_Byte (Daughter Card Byte Write Test)
Da_Adcd (Daughter Card Address Decode)
Da_Addl (Daughter Card Additional Tests)
Da_Clrr (Daughter Card Scrub Incomplete)
Rd_Snps (Redundancy Card 96 Kw Snapshot Error)
Rd_Bslk (Redundancy Card Bus Lock Fail)

CURIOLFL

Type: **E:$IOMHF** **Current HPMM IOL Interface Failure**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Unknown** - (Unknown Error)
Powerdwn - (Power Is Off)
Invprgex (Invalid Program Execution)
Epromerr (EPROM Error)
Ramcnter (Ram Contents Error)
Ramadrer (Ram Address Error)
Dpaerror (Device Physical Address Error)
Dsaerror (Device Soft Address Error)
Rxbufofl (Receive Buffer Overflow Error)
Ioljaber (IOL Jabber Error; Module saw or talked too much on link)
Badpgjmp (Bad Program Jump)
Adcincmp (A/D Incompatible)
Adoutovf (A/D Overflow)
Adoutudf (A/D Underflow)
Adccaler (A/D Calibration Error)
Baddcltc (Bad DC LTC)
Dmt_tmot (Deadman Time Out)
Mltoutfl (Multiple Output Failure)
Datbusfl (Data Bus Failure)
Baddarng (Bad A/D Range)
Mstrtmot (Master Timeout)
Ctrcktfl (Counter Circuit Failure)

CURPINAM (n)

Type: **String_8**
Lock: **View**
Default: **N/A**
PtRes: **IOP**
Range: N/A

Current PI Filename—Defines the personality Image filename that currently resides in this IOP where n is the IOP number 1 - 40.

CURSEGID (RampSoak)

Type:	**E:CURSEGID**	**Current Segment ID—**Defines the current ramp or soak segment.
Lock:	**Oper**	
Default:	**Ramp1**	
PtRes:	**HPM**	
Range:	0-**Ramp1** 1-**Soak1**	
	2-**Ramp2** 3-**Soak2**	
	: and **:**	
	20-**Ramp11** 21-**Soak11**	
	22-**Ramp12** 23-**Soak12**	

CUTOFFLM (Totalizr)

Type:	**Real**	**Zero-Flow Cutoff Limit—**Allows the user to specify a cutoff limit such that when the value of input parameter P1 falls below the limit specified, its value is replaced by 0.0.
Lock:	**Supr**	
Default:	**0.0**	
PtRes:	**HPM**	
Range:	$\geq$ **0.0,**	
	NaN (Cutoff limit is not applicable)	

CUTOFFLM (VdtLdLag)

Type:	**Real**	**Zero-Flow/Belt-Speed Cutoff Limit—**Allows the user to specify a cutoff limit for equations C and D.
Lock:	**Supr**	
Default:	**NaN**	
PtRes:	**HPM**	
Range:	$\geq$ **0.0,**	
	NaN (Bypasses the limit check)	

CV

Type:	**Real**	**Calculated Variable—**The result (calculated value) of the calculation of the control algorithm. The value can be in percent or in engineering units depending on the control algorithm.
Lock:	**Prog**	
Default:	**NaN**	
PtRes:	**HPM**	
Range:	**N/A**	

CVEUHI

Type: **Real** — **Calculated Value's High Limit in Engineering Units**
Lock: **Engr**
Default: **100.0 (GPM, PPH, etc.)**
PtRes: **HPM**
Range: **≥ CVEULO**

Helpful Hint:	CV ranges track X-input ranges if CTLALGID = AutoMan, ORSel, IncrSum, or Switch. For CTLALGID = PidErfb and RampSoak, CV ranges are configurable. For CTLALGID = Pid, Pidff, and RatioCtl, if NOCOPTS = 0, then the CV ranges are configurable, otherwise, the CV ranges track the ranges of the secondary output connection.

CVEULO

Type: **Real** — **Calculated Value's Low Limit in Engineering Units**
Lock: **Engr**
Default: **0.0 (GPM, PPH, etc.)**
PtRes: **HPM**
Range: **≤ CVEUHI**

Helpful Hint:	Same as above for CVEUHI.

CYCLEOPT (RampSoak)

Type: **E:$CYCLOPT** — **Ramp/Soak Cycle Option—**Defines whether the ramp/soak cycle stops after a single cycle, or is continuous. For detailed information, refer to the *HPM Control Functions and Algorithms* manual.
Lock: **Oper**
Default: **Cyclic**
PtRes: **HPM**
Range: 0-**Single** (Stop after completing one complete cycle)
1-**Cyclic** (Repeat complete cycles over and over)

Helpful Hint:	If Cyclic is entered, repeats complete ramp/soak cycles after Mode is changed from Man to Auto. If Single is entered, performs one ramp/soak cycle and then stops.

CYCLETIM

Type: **Real**
Lock: **Supr**
Default: **10.0 seconds**
PtRes: **HPM**

PosProp Output Cycle Time in Seconds—Determines the rate at which raise or lower output pulses are going to be generated. PV - SP determines the width of the output pulse.

Range: **0.25** to **1000.0 seconds**

CYCOVRO (FBus)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **IOP**

Times Write Buffer Not Empty During Cycle—

Range: **N/A**

-D-

D (Summer, VdtLdLag)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: **N/A**

Overall Bias—Defines the overall bias used in calculating PVCALC.

D1 (VdtLdLag)

Type: **Real**
Lock: **Supr**
Default: **0.0 minutes**
PtRes: **HPM**
Range: **0.0** to **400.0 minutes**

Fixed Deadtime in Minutes—Bias value for the variable time delay.

D1, D2 (DigComp, DevCtl)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (No input present)
On (Input is present)

Digital Input 1 Status and Digital Input 2 Status—Separately indicates whether input 1 and input 2 are on or off.

D1_0 (DigComp, DevCtl)

Type: **E:$PVSTATS**
Lock: **View**
Default: **PVState0**
PtRes: **HPM**
Range: 0-**PVState0** (STATETXT(0) describes D1 = 0)
1-**PVState1** (STATETXT(1) describes D1 = 0)

Digital Input 1 Equal to A PV State of 0—Defines the PV state that corresponds to input D1 = Off. D1_0 = true only if INPTDIR = Direct and the field contact feeding D1 is open, or INPTDIR = Reverse and contact is cleared.

Helpful Hint:	Applies only if NODINPTS = 1. D1_0 is always the opposite state of D1_1.

D1_1 (DigComp, DevCtl)

Type: **E:$PVSTATS**
Lock: **Eng/PB**
Default: **PVState1**
PtRes: **HPM**
Range: 0-**PVState0** (STATETXT(0) describes D1 = 1)
1-**PVState1** (STATETXT(1) describes D1 = 1)

Digital Input 1 Equals A PV State of 1—D1_1 defines the PV state that corresponds to D1 (Input 1 status) = On. D1_1 = true only if INPTDIR = Direct and the field contact feeding D1 is closed, or INPDIR = Reverse and contact is open.

Helpful Hint:	D1_1, Digital Input 1 Equal To A PV State Of 1, applies only if NODINPTS = 1. D1_1 is always the opposite state of D1_0 and vice versa.

D2 (VdtLdLag)

Type: **Real** — **Bias for Input P2**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: ≥ **0.0**

D2D1_00 (DigComp, DevCtl)

Type: **E:$PVSTATS** — **D2_D1 Zero_Zero PV State—**Defines the PV state descriptor that is to be used and displayed when inputs D2 and D1 are both Off (00).
Lock: **Eng/PB**
Default: **MovPV**
PtRes: **HPM**
Range: 0-**PVState0** (STATETXT(0) descriptor)
1-**PVState1** (STATETXT(1) descriptor)
2-**BadPV** (BADPVTXT descriptor)
3-**MovPV** (MOVPVTXT descriptor)
4-**PVState2** (STATETXT(2) descriptor)

Helpful Hint:	D2D1_00 configuration requires NODINPTS = 2. Option PVState2 cannot be selected unless NOSTATES = 3. STATETXT(0–2) is configured for each DigComp or DevCtl point; BADPVTXT and MOVPVTXT are configured during Box Data Point configuration for all DigComp or DevCtl points in this box.

D2D1_01 (DigComp, DevCtl)

Type: **E:$PVSTATS** — **D2D1 Zero_One PV State—**Defines the PV state descriptor that is to be used and displayed when input D2 is Off and input D1 is On (01).
Lock: **Eng/PB**
Default: **PVState1**
PtRes: **HPM**
Range: 0-**PVState0** (STATETXT(0) descriptor)
1-**PVState1** (STATETXT(1) descriptor)
2-**BadPV** (BADPVTXT descriptor)
3-**MovPV** (MOVPVTXT descriptor)
4-**PVState2** (STATETXT(2) descriptor)

Helpful Hint:	D2D1_01 configuration requires NODINPTS = 2. Option PVState2 cannot be specified unless NOSTATES =3. STATETXT(0–2) is configured for each DigComp or DevCtl tag name; BADPVTXT and MOVPVTXT are configured during Box Data Point configuration for all DigComp or DevCtl points in the box.

D2D1_10 (DigComp, DevCtl)

Type: **E:$PVSTATS**
Lock: **Eng/PB**
Default: **PVState0**
PtRes: **HPM**
Range: 0-**PVState0** (STATETXT(0) descriptor)
1-**PVState1** (STATETXT(1) descriptor)
2-**BadPV** (BADPVTXT descriptor)
3-**MovPV** (MOVPVTXT descriptor)
4-**PVState2** (STATETXT(2) descriptor)

D2D1 One_Zero PV State—Defines the PV state descriptor that is to be used and displayed when input D2 is On and input D1 is Off (10).

Helpful Hint:	D2D1_10 configuration requires NODINPTS = 2. Option PVState2 cannot be specified unless NOSTATES = 3. STATETXT(0–2) is configured for each DigComp or DevCtl point; BADPVTXT and MOVPVTXT are configured during Box Data Point configuration for all DigComp or DevCtl points in the box.

D2D1_11 (DigComp, DevCtl)

Type: **E:$PVSTATS**
Lock: **Eng/PB**
Default: **BadPV**
PtRes: **HPM**
Range: 0-**PVState0** (STATETXT(0) descriptor)
1-**PVState1** (STATETXT(1) descriptor)
2-**BadPV** (BADPVTXT descriptor)
3-**MovPV** (MOVPVTXT descriptor)
4-**PVState2** (STATETXT(2) descriptor)

D2D1 One_One PV State—Defines the PV state descriptor that is to be used and displayed when inputs D2 and D1 are both On (11).

Helpful Hint:	D2D1_11 configuration requires NODINPTS = 2. Option PVState2 cannot be specified unless NOSTATES = 3. STATETXT(0–2) is configured for each DigComp or DevCtl point; BADPVTXT and MOVPVTXT are configured during Box Data Point configuration for all DigComp or DevCtl points in the box.

DAMPING (STI)

Type: **Real**
Lock: **Supr/View**
Default: **0.0**
PtRes: **HPM**

Damping—Defines the first-order PV filtering option for the smart transmitter. User can also implement PV filtering by using this parameter or the TF parameter; however, DAMPING is the preferred parameter. If DAMPING has been configured at the transmitter using the Universal Station, the STI IOP adjusts the entered value to one of the values in the range shown below for the appropriate transmitter type. For Multivariable transmitters with SENSRTYP = SFM the IOP will not adjust the damping value.

Any real number in the range of damping specified by the transmitter user manual can be used. It can be changed only when the STI point execution state PTEXECST is Inactive.

Range:

Transmitter Type

Spt	*Stt*	*Sfm*
0.0	**0.0**	**0.0**
0.16	**0.30**	**0.5**
0.32	**0.70**	**1.0**
0.48	**1.5**	**2.0**
1.00	**3.10**	**3.0**
2.0	**6.3**	**4.0**
4.0	**12.7**	**5.0**
8.00	**25.5**	**10.0**
16.0	**51.1**	**50.0**
32.0	**102.3**	**100**
NaN	**NaN**	**NaN**

DATE (HPM Box)

Type: **Time**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

Current Date/Time—Value of the LCN date in the HPM.

DAY (HPM Box)

Type: **Integer**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **1** to **31**

Current Day—Value of the LCN date in the HPM.

DB_VALID(1)–(40) (HPM Box)

Type: **E:$DBVALID**
Lock: **Engr**
Default: **Invalid**
PtRes: **HPM**
Range: **Valid** (Database is valid)
Invalid (Database is not valid)

Database Valid—Indicates if the database is valid. The IOP cannot be set to RUN unless the database is valid.

DEADBAND(1)–(24) (Logic)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **≥ 0.0**

Deadband Value—Defines the value of the deadband for the specified logic block within the logic slot.

Helpful Hint:	DEADBAND requires LOGALGID = EQ, NE, GT, GE, LT, or LE.

DEADBAND (PosProp, PIDPosPr)

Type: **Real**
Lock: **Supr**
Default: **5.0 %**
PtRes: **HPM**
Range: **0.0** to **100.0 %**

Deadband in Percent of Full Scale—Defines the error deadband.

DEADTIME (PosProp, PIDPosPr)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: **0.0** to **60.0** seconds

Deadtime (in seconds)—Additional pulse time required to overcome the friction in the motor when it begins to move or change direction. It is added to the calculated pulse time except when the pulse that was issued in the last cycle time was in the same direction (as the pulse this time), and the pulse width was equal to CYCLETIM.

DEBOUNCE (DigIn)

Type: **Integer**
Lock: **Engr/PB**
Default: **10 milliseconds**
PtRes: **HPM**
Range: **0** to **50 milliseconds**

Contact Debounce Time in Milliseconds—The length of time an input must remain in a new state for it to be declared as a valid event by the DISOE IOP. Refer to the Absolute Delay Across parameter located in the Digital Input Processor table of the HPM Specification and Technical Data.

DECONF (STI)

Type: **E:$DECONF**
Lock: **Eng/View**
Default: **Pv_Sv_Db**
PtRes: **HPM**

Digitally Enhanced Configuration Mode—Defines the contents of the data that will be sent by the smart transmitter to the STI point. The use of Pv_Db and Pv_Sv_Db is recommended because they offer database mismatch detection and on-process mismatch recovery.

This parameter can be changed only when the STI point execution state PTEXECST is Inactive.

Range: 0-**Analog** (Not Supported)
1-**Pv** (Transmits only the PV; 4-byte format)
2-**Pv_Sv** (Transmits the PV and the secondary variable (SV); 4-byte format)
3-**PV_Db** (Transmits the PV and the transmitter database; 6-byte format)
4-**Pv_Sv_Db** (Transmits the PV, SV, and the transmitter database; 6-byte format)

Helpful Hint:	For the PV_Db and Pv_Sv_Db selections, one byte of the transmitter database is transmitted each time the PV is transmitted to the STI IOP.

DELCV (IncrSum)

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

Delta CV in Engineering Units—Indicates the calculated change in the CV output value in engineering units**.**

DELCV (Pid)

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

Delta CV in Percent—Indicates the calculated change in the CV output value in percent**.**

DEV (RegCtl)

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

Deviation—Indicates the deviation (PV - SP) in engineering units.

DEVADDR (Array)

Type: **Real**
Lock: **PtBld**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Serial Link Device Address—Indicates the serial link address of the device containing data.

DEVHIFL (RegCtl)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (No DEVHI alarm)
On (DEVHITP has been exceeded)

Deviation High Alarm Flag—Indicates whether the DEVHITP has been exceeded.

DEVHIPR (RegCtl)

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

Deviation High Alarm Priority—Defines the priority of the deviation high alarm.

DEVHITP (RegCtl)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **≥ 0.0,**
NaN

Deviation High Alarm Trip Point—Defines the upper limit for the deviation.

Helpful Hint:	Alarm occurs when the PV is higher than SP + DEVHITP.

DEVLOFL (RegCtl)

Type: **Logical** **Deviation Low Alarm Flag—**Indicates whether the DEVLOTP has been exceeded.
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (DEVLOTP has not been exceeded)
On (DEVLOTP has been exceeded)

DEVLOPR (RegCtl)

Type: **E:ALPRIOR** **Deviation Low Alarm Priority—**Defines the priority of the deviation low alarm.
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

DEVLOTP (RegCtl)

Type: **Real** **Deviation Low Alarm Trip Point—**Defines the lower limit for the deviation.
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **≥ 0.0,**
NaN

Helpful Hint:	Alarm occurs when the PV is lower than SP - DEVLOTP.

DHTIMMAX(1) - (5) (NIM PSDP)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **NIM**
Range: **N/A**

Maximum Time to Complete a Data Handler Request—Specifies the maximum time to complete a Data Handler request in msec.

DIAGCMD (ProcMod)

Type: **E: DIAGCMD**
Lock: **Oper**
Default: **N/A**
PtRes: **HPM**
Range:

Diagnostic Command—

> *Helpful Hint:* DIAGCMD resets the ProcMod overrun statistics and AVGPU and MAXPU values.

DISP_SIM (HPM Box)

Type: **Logical**
Lock: **Prog**
Default: **On**
PtRes: **HPMM**
Range: **Off** (Simulation indicator is not required to be displayed
On (Simulation indicator is required to be displayed

Simulation Indicator Display Switch—see also SIM_TXT

DISRC(1)–(2) (DigComp, DevCtl)

Type:	**Universal Ent.Prm**	**Digital Composite and Device Control Input-Connection Source—**Specify the sources whose values are to be fetched and delivered to Digital Composite data point inputs D1 and D2. The source can be specified using the "Tagname.Parameter" format or the hardware reference address format. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.
Lock:	**PtBld**	
Default:	**null.null**	
PtRes:	**HPM**	

Range: Use Tagname.Parameter format for tagged points where Tagname can be up to 16 characters and the permissible character set is as follows:

Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.
An * is used to default to this point's tag name.
Parameter name can be up to eight characters, and must be a legitimate parameter name.

Some possible input-connection sources are

a."DigIn slot Tagname.PVFL"
b."DigOut slot Tagname.SO"
c."Logic slot Tagname.SO(nn)" where nn = 1–24
d."Logic slot Tagname.Fl(nn)" where nn = 1–12
e."ProcMod slot Tagname.Fl(nnn)" where nnn = 1–127
f."Box Flag slot Tagname.PVFL
g."!Box.FL(nnnn)" for a box flag that resides in the same box; nnnn = 1–16,384
h."$NMhhBxx.FL(nnnn)" for a box flag that resides in a different HPM box on the same UCN; hh is the NIM UCN address, xx is the HPM box number, and nnnn = 1–4095

Use the hardware reference address !MTmmSss.Parameter for untagged or tagged points where

MT is the IOP type, such as DI (Digital Input)
mm is the IOP Card number (1–40)
The letter "S" is a constant
ss is the slot number on the IOP Card (refer to SLOTNUM parameter)
Parameter name can be up-to-eight characters and must be a legitimate parameter name.

DITYPE

Type:	**E:$DITYPE**	**Digital Input Type—**Defines the type of digital input point.
Lock:	**PtBld**	
Default:	**Status**	
PtRes:	**HPM**	

Range: 0-**Status** (Point is to be used for alarming and event reporting)
1-**Latched** (Point is to be used for event reporting)
2-**Accum** (Point is to be used for accumulating pulses)

DLYTIME (DigIn)

Type:	**Integer**	**Delay Time—**For an off-normal alarm, defines the time (in seconds) that a point with a previously detected alarm condition is guaranteed to remain in alarm, even if the condition clears. If an alarm condition exists when the delay timer expires, the point is held in alarm.
Lock:	**Supr**	
Default:	**5 seconds**	For a change of state (COS) alarm, if the PV is in the same state when the delay timer expires, future state changes are immediately alarmed. If the PV is in the opposite state, a second COS alarm is produced and the delay timer is restarted.
PtRes:	**HPM**	
Range:	**0** to **60 seconds**	

DLYTIME(1)–(24) (Logic)

Type:	**Real**	**Alarm Delay in Seconds for Logic Block**
Lock:	**Supr**	
Default:	**1 second**	
PtRes:	**HPM**	
Range:	**1–8000 seconds**	

Helpful Hint:	DLYTIME requires LOGALGID = Pulse, MinPulse, MaxPulse, OnDelay, OffDelay, or Watchdog.

DODSTN(1)–(3) (DigComp, DevCtl)

Type: **Universal** **Digital Composite and Device Control Output-Connection Destination—**
Ent.Prm Specifies up to three output connection destinations that are to receive the OP
Lock: **PtBld** output from this point. The destination can be specified using the
Default: **null.null** "Tagname.Parameter" format or the hardware reference address format. Refer to
PtRes: **HPM** the *HPM Control Functions and Algorithms* manual for a detailed description.
Range: Use Tagname.Parameter for tagged points where Tagname can be up to 16 characters and the permissible character set is as follows:

Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.
An * is used to default to this point's tag name.
Parameter name can be up to eight characters, and must be a legitimate parameter name.

Some possible output-connection destinations are

a."DigOut slot tagname.ONPULSE or OFFPULSE"
b."DigOut slot Tagname.SO"
c."Logic slot Tagname.Fl(nn)" where nn = 7–12
d."ProcMod Tagname.Fl(nnn)" where nnn = 1–127
e."Flag slot Tagname.PVFL
f."!Box.FL(nnnn)" for a box flag that resides in the same HPM box; nnnn = 1–16,384.
g."$NMhhBxx.FL(nnnn)" for a box flag that resides in a different HPM box on the same UCN; hh is the NIM UCN address, xx is the HPM box number, and nnnn = 1–4095.

Use the hardware reference address !MTmmSss.Parameter for untagged or tagged points where

MT is the IOP type, such as DO (Digital Output)
mm is the IOP Card number (1–40)
The letter "S" is a constant.
ss is the slot number on the IOP Card (refer to SLOTNUM parameter)
Parameter name can be up to eight characters and must be a legitimate parameter name.

DOTYPE (DigOut)

Type: **E:$DOTYPE** **Digital Output Type—**Determines the type of digital output point.
Lock: **PtBld**
Default: **Status**
PtRes: **HPM**
Range: 0-**Status** (Status output type)
1-**Pwm** (Pulse Width Modulated output type)

DSA

Type: **Integer** **Device Soft Address—**The logical address of an IOP: 1-40 for primary IOPs
Lock: **View** and 129 - 168 for secondary IOPs.
Default: **N/A**
PtRes: **HPM**
Range: **1 - 40 for primary IOPs**
129 - 168 for secondary IOPs

-E-

EIPPCODE

Type: **Ent_Id**
Lock: **Engr**
Default: **Null**
PtRes: **NIM**

Event-Initiated Processing Point Identifier—Defines the tag name of the point in the AM or CM that is to be notified when an event is detected by this point.

Range: Tag name of the data point can be up to 16 characters and the permissible character set is as follows:
Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.

Helpful Hint:	EIPPCODE configuration requires PNTTYPE = DigIn, DigComp, Logic, Flag or DevCtl and EVTOPT = Eip or Eip_Soe. For HPM Box Flag points, this parameter applies only to slots 1 through 128.

EQUOBJNM

Type: **String**
Lock: **View**
Default: **Blank**
PtRes: **HPM**
Range: **N/A**

Equipment List Object Name—Specifies the Equipment List Object Name (CL object header)

ERRCODE (Array)

Type: **String_8**
Lock: **View**
Default: **Spaces**
PtRes: **HPM**

Serial Interface/Serial Link Communication Error Code—When the BADPVFL parameter = ON, this parameter provides additional information if initialized by the serial interface FTA driver program.

SI Array Point Error Code Values

HPM

HPM Idle—When the HPMM status is IDLE, Array point configuration may or may not be loaded to the SI IOP.
Iop Comm—When the HPMM status is RUN, Array point configuration is NOT loaded to the SI IOP.

SI IOP

No_FTA—Appears when the power adapter panel is not connected to the IOP.
FTA_Comm—Appears when the corresponding FTA is not connected to the power adapter panel, or when communication between the IOP and FTA has failed.
CFG_Load—Appears when configuration data is downloaded to the FTA.
Mod_Idle—Appears when configuration data is downloaded to the FTA and the IOP is in IDLE mode, or when the IOP operating state is switched from RUN to IDLE.

SI IOP FTA Common

Dev Addr—The device address has a configuration error
Data Type—The data type has a configuration error
Startidx—The start index has a configuration error
 # Elemnt—A number of elements configuration error has occurred
Config—An application-specific configuration error has occurred
Inv Resp—An invalid field device response has occurred
Parity, Checksum, MsgTmout, ChrTmout—A field device communication error has occurred
Ex or xx—An exception or other field device error has occurred. The “xx” error code is specific to the field device
Fac Test—A factory test is in progress
OK—No errors exist

Range: **N/A**

ESWAUTO (RegCtl)

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

External Switching Flag for Automatic Mode—When On, means that this point's operating mode has been switched from some mode other than automatic to the automatic mode by an external source.

ESWCAS (RegCtl)

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

Eternal Switching Flag for Cascade Mode—When On, means that this point's operating mode has been switched from some mode other than cascade to cascade mode, by an external source.

ESWENBST (RegCtl)

Type: **E:ENBLSTAT**
Lock: **Oper**
Default: **Disable**
PtRes: **HPM**
Range: 0-**Disable** (Does not allow external switching of point's mode)
1-**Enable** (Allows external switching of point's mode)

External Mode Switching Enable State—Defines whether external mode switching is permitted for this point.

Helpful Hint:	ESWENBST cannot be changed if parameter SHUTDOWN is On or if parameter REDTAG is On.

ESWMAN (RegCtl)

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

External Switching Flag for Manual Mode—When On, means that this point's operating mode has been switched from some mode other than the manual mode to the manual mode by an external source.

EUDESC

Type:	**String_8**	**Engineering Units Descriptor—**An eight-character descriptor that defines the name of the engineering units (EU) that are displayed on the Group and Detail Displays for this point as shown in Figure N-1 (see NAME). In this figure, LBS/SEC is the engineering unit descriptor.
Lock:	**PtBld**	
Default:	**Blank**	
PtRes:	**NIM**	

Range: Permissible character set consists of all characters on the Engineer' s Keyboard. Basically this set consists of alphabetics A-Z, numerics 0-9, and the following special characters: space ! % & ' ( ) * + - / : ; > < = ? _ , . $

EUNDESC (1)–(168)

Type:	**String_72**	**IOP Generic Descriptor—**Used as additional display text to help the operator diagnose potential problems with the IOP. It is primarily used with diagnostic displays.
Lock:	**View**	
Default:	**Blanks**	
PtRes:	**HPM**	

Range: nn = 1-40 specifies one of the 40 acting primaries.
nn = 129-168 specifies one of the 40 acting secondaries.

EVRCINPG

Type:	**Logical**	**NIM Event Recovery in Progress Flag**
Lock:	**View**	
Default:		
PtRes:	**HPM**	
Range:	**Off**	
	On	

EVTOPT (DigComp)

Type: **E:$EVTOPT**
Lock: **PtBld**
Default: **None**
PtRes: **HPM**

Event Reporting Option—If EVTOPT = Eip and the PV changes or a PV alarm is generated, the AM or CM data point named EIPPCODE is notified and a "process special" on that data point takes place.

Range: 0-**None** (Event-Initiated Processing is not allowed)
1-**Eip** (Process special is triggered in AM/CM)

Helpful Hint:	EVTOPT configuration requires NODINPTS > 0.

EVTOPT (DevCtl, DigIn)

Type: **E:$EVTOPT**
Lock: **PtBld**
Default: **None**
PtRes: **HPM**

Event Reporting Option—If EVTOPT = Eip and the PV changes, the AM or CM data point named EIPPCODE is notified and a "process special" on that data takes place. If EVTOPT = Soe and a PV change occurs, Sequence Of Events Processing is notified. If EVTOPT = EipSoe, the actions in both apply.

Range: 0-**None** (Neither Eip nor Soe is allowed)
1-**Eip** (Process special is triggered in AM/CM)
2-**Eip_Soe** (Eip and Soe are both allowed)
3-**Soe** (Point notifies Sequence of Events Processing)

Helpful Hint:	EVTOPT configuration requires DITYPE = Status or Latched. If DITYPE = Latched, EVTOPT cannot = EIPSOE or SOE.

EXTDATA (Array)

Type: **E:$EXTDATA**
Lock: **PtBld**
Default: **None**
PtRes: **HPM**

External Data Option—Indicates if either the Array point flags, numerics, or strings are mapped from a serial interface.

Range: **None** (None of the flags, numerics, or strings are mapped from a serial interface)
IO_FL (IO flags are mapped from a serial interface)
IO_NN (IO numerics are mapped from a serial interface)
IO_STR (IO strings are mapped from a serial interface)
UCN_FL (Reserved for future use)
UCN_NN (Reserved for future use)
UCN_STR (Reserved for future use)

Helpful Hint:	You can map either flags, numerics, or strings from the Serial Interface to a single Array point.

EXTSWOPT

Type: **E:EXTSWOPT**
Lock: **Eng/PB**
Default: **None**
PtRes: **HPM**

External Mode Switching Option—External mode switching is typically used to establish mode interlocks, or under certain process conditions, to restrict the use of a mode that invokes a higher level of control. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of external mode switching.

Range: 0-**None** (No external mode switching is allowed)
1-**Ems** (External source can change point's mode)
2-**Emp** (Not implemented)

-F-

F (FlowComp)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Flow Input—Indicates the value of the uncompensated flow input. This input is a square-rooted, differential pressure input.

FAILCODE

Type: **E:$IOMHF**
Lock: **View**
Default: **N/A**
PtRes: **IOP**

I/O Processor Hard Fail Status—

Range: 0-**Unknown** (Unknown status)
1-**PowerDwn**(This IOP Powered Down)
2-**InvPrgEx**(Invalid Program Execution)
3-**EpromErr**(EPROM Checksum Error)
4-**RamCntEr**(RAM Contents Error)
5-**RamAdrEr**(RAM Addressing Error)
6-**DpaError**(Device Physical Address Error)
7-**DsaError**(Device Soft Address Error)
8-**RxBufOfl**(I/O Link Receive Buffer Overflow)
9-**IOLJaber**(I/O Link Jaber Circuit Failure)
10-
11-**BadPgJmp**(Illegal Value of Case Control)
12-**AdCIncmp**(A to D Conversion Incomplete)
13-**AdOutOvf**(A to D Output Value Overflow)
14-**AdOutUdf**(A to D Output is less than Zero)
15-**AdCCalEr**(A to D Calibration is incorrect)
16-**BadDcLtc**(Bad DC LTC)
17-**Dmt_Tmot**(Dead Man Timer Timeout)
18-**MLtOutFl**(Multiple Output Failure)
20-**BadDaRng**(Bad D to A Range)
21-**MstrTmot**(Master 68K Timeout)

FAILOPT(1)–(168) (IOP)

Type: **E:FAILOPT**
Lock: **Eng/PB**
Default: **Unpower**
PtRes: **HPM**

Failure Option for Outputs—Defines the state which an AO or DO IOP goes into if the IOP itself, or the HPMM fails. If the IOP failure is due to power loss, outputs go to unpowered regardless of the FAILOPT value. When power is restored to the module, outputs are reset regardless of the FAILOPT values.

Range: 0-**Hold** (Hold output at last good value)
1-**Unpower** (Remove power from the output)

FBTIME (DevCtl, DigComp)

Type:	**Integer**	**Feedback Time—**Sets the amount of time (in seconds) that the point should wait before generating a "command disagree" alarm after the operator has issued a start/stop-type command to a field device.
Lock:	**Supr if CMDFALTM is changed from a non-zero value to a zero value, else Eng/PB**	
Default:	**0**	
PtRes:	**HPM**	
Range:	**0 to 1000 seconds** (0 indicates that command disagree alarming is disabled)	

Helpful Hint:	FBTIME can be increased to compensate for a slow-responding field device that does not respond to the operator's command in time to prevent a command-disagree alarm.

FF (PidFf)

Type:	**Real**	**Feed Forward Algorithm Input—**FF is the feedforward input signal value that is added to (FFOPT = Add) or multiplied by (FFOPT = Multiply) the PidFf algorithm's incremental output, before the full-value output is accumulated. FF is normally a parameter with a percentage value.
Lock:	**View**	
Default:	**N/A**	
PtRes:	**HPM**	
Range:	**N/A**	

FFOPT (PidFf)

Type:	**E:FFOPT**	**Feed Forward Type—**Determines whether a PidFf algorithm's feedforward input signal (FF) is added to or multiplied by the incremental output, before the full-value output is accumulated.
Lock:	**Eng/PB**	
Default:	**Multiply**	
PtRes:	**HPM**	
Range:	0-**Add** (Scaled Feedforward + Feedback) 1-**Multiply** (Feedback x Scaled, Biased Feedforward)	

FL(i) (Array)

Type:	**Logical**	**Array Point Flag Variables—**The flags are mapped from either the HPM box (defined by FLSTIX and NFLAG parameters), or from a serial interface IOP-connected device (when EXTDATA=IO_FL, mapping is defined by IOPNUM, FTANUM, DEVADDR, FLSTIX, and NFLAG parameters).
Lock:	**Determined by SPLOCK parameter**	
Default:	**N/A**	
PtRes:	**HPM**	
Range:	**1 ≤ i ≤ Array parameter NFLAG**	

FL(1)–(12) (DevCtl, Logic)

Type: **Logical**
Lock: **View; FL1-FL5 Prog; FL6 Oper; FL7-FL12**
Default: **FL2 = On, rest = Off**
PtRes: **HPM**
Range: **Off** (Flag is off)
On (Flag is set)

Logic Slot Flags—Twelve flags, FL(1) to FL(12), are provided for each logic slot. The states of flags FL(1) to FL(6) are controlled by the HPM and cannot be changed by the user. FL(7)-FL(12) are assigned by the user for controlling the path of the logic in the respective logic slot. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description

These flags are local to the logic slot and are different than the 127 flags provided with each process module, and the 1023 flags provided in each HPM box.

FL(1)–(127) (ProcMod)

Type: **Logical**
Lock: **Determined by SPLOCK parameter**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Flag is off)
On (Flag is set)

Local Flag Variables—Each process module in the HPM has 127 local flags that can be used for implementing batch operations. These flags are local to the process module and are different than the 12 logic-slot flags, and the 1023 flags provided in each HPM box.

FL(1)–(16,384) (HPM Box)

Type: **Logical**
Lock: **Oper**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Flag is off)
On (Flag is set)

Box Flag Variables—Each HPM box has a set of 16,384 local flag variables that can be used by process modules in this HPM to implement batch operations**.** The first 2047 box flags are taggable. These flags are local to the HPM box and are different than the 12 logic-slot flags, and the 127 flags provided in each process module. The LCN index limit is 4095; there is no index limit for the UCN. Array points can be used to address flags with an index greater than 4095.

Helpful Hint:	For the first 128 flags, the On state is alarmed.

FLDESC (Array)

Type: **String_64**
Lock: **PtBld**
Default: **Spaces**
PtRes: **HPM**
Range: **N/A**

FL Array Descriptor—Describes FL data for the Array point.

FLSTIX (Array)

Type: **Real**
Lock: **PtBld**
Default: **0.0**
PtRes: **HPM**
Range: **0** to **99,999** (When EXTDATA = IO_FL, 0 can be a valid device index)
0 to **16,384** (When EXTDATA ≠ IO_FL, 0 indicates that no flags are configured)

Flag Array Start Index—Defines the flag array start index in Box FL variables or serial interface-connected devices.

FORCE (HiLoAvg)

Type:	**Logical**	**Forced Input Request Flag—**Defines whether the operator, a user-written program, or an input connection has requested that an input be used as the forced input for this algorithm.
Lock:	**Oper**	
Default:	**Off**	
PtRes:	**HPM**	
Range:	**Off** (No request to force an input) **On** (Request has been made to force an input)	

Helpful Hint:	FORCE change requires FRCPERM = On.

FRCPERM (HiLoAvg)

Type:	**Logical**	**Forced Input Permissive—**Defines whether an operator or a user-written program can force-select an input. FRCPERM must be On before the operator or a program can select an input to be used as a forced input to this algorithm.
Lock:	**Eng/PB**	
Default:	**Off**	
PtRes:	**HPM**	
Range:	**Off** (Forced-selection function is disabled) **On** (Forced- selection function is enabled)	

FREQ6050(1)–(168)

Type:	**E:FRQ6050**	**Frequency 60/50Hz—**Defines the 60/50 Hz frequency configuration needed for a Low Level AI Mux or STI Temperature Transmitter. For the STI, if a mismatch occurs between this parameter and the transmitter's internal 60 Hz/50 Hz frequency parameter, a database download from the STI IOP to the transmitter will clear this condition.
Lock:	**Eng/PB**	
Default:	**60Hz**	
PtRes:	**HPM**	
Range:	0-**60 Hz** 1-**50 Hz**	

FRQUTAVG (NIM, HPM Box)

Type:	**Real**	**Average UCN Fetch Request Trip Time—**The average time in milliseconds it takes to receive a response to this node's UCN fetch requests.
Lock:	**View**	
Default:	**NaN**	
PtRes:	**HPM**	
Range:	**N/A**	

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

FRQUTMAX (NIM, HPM Box)

Type:	**Real**	**Maximum UCN Fetch Request Trip Time—**The maximum time in milliseconds it takes to receive a response to this node's UCN fetch requests.
Lock:	**View**	
Default:	**NaN**	
PtRes:	**HPM**	
Range:	**N/A**	

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

FRSPTAVG (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Average UCN Fetch Response Trip Time—The average time in milliseconds for this node to respond to fetch requests from other UCN nodes.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

FRSPTMAX (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Maximum UCN Fetch Response Trip Time—The maximum time in milliseconds for this node to respond to fetch requests from other UCN nodes.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

FSELIN (HiLoAvg)

Type: **E:PINP**
Lock: **Oper/PB**
Default: **SelectP1**
PtRes: **HPM**
Range: 1-**SelectP1** (Input P1 is the forced input)
2-**SelectP2** (Input P2 is the forced input)
3-**SelectP3** (Input P3 is the forced input)
4-**SelectP4** (Input P4 is the forced input)
5-**SelectP5** (Input P5 is the forced input)
6-**SelectP6** (Input P6 is the forced input)

Force Selected Input—Defines the one of six inputs to be used as the forced input to this algorithm.

Helpful Hint:	FSELIN change by an operator requires FRCPERM = On.

FSTS (FlowComp)

Type: **E:PVVALST**
Lock: **View**
Default: **Bad**
PtRes: **HPM**
Range: 0-**Bad** (Value is bad and replaced with NaN)
1-**Uncertn** (Status of the value is uncertain)
2-**Normal** (Value is good)

Flow Input Value Status—Indicates the current status of flow input F.

FTA1TYPE, FTA2TYPE (HPM Box)

Type: **E:$FTATYPE**
Lock: **View**
Default: **None**
PtRes: **HPM**
Range: 0-**None**
1-**TC**
2-**RTD**

Type of FTA Connected to the LLMUX IOP or RHMUX IOP—The FTA type applies to both FTA positions (1 and 2). The FTA supplies 16 points per FTA for a total of 32 points.

FTACONN(1)–(168) (HPM Box)

Type: **E:$FTACON**
Lock: **View**
Default: **CONN_A**
PtRes: **HPM**
Range: 0-**CONN_A** (Module is connected to FTA connector A)
1-**CONN_B** (Module is connected to FTA connector B)

FTA connection to I/O module file. Indicates which FTA connector is connected to this module. It is primarily used with the diagnostic displays. nn = 1–40 specifies FTA connection for one of the 40 acting primaries. nn = 129–168 specifies FTA connection for one of the 40 acting secondaries.

FTANUM (Array)

Type: **Integer**
Lock: **PtBld**
Default: **1**
PtRes: **HPM**
Range: **1** to **10**

IOP FTA Number—Indicates the FTA number of the serial interface IOP.

Helpful Hint:	Only FTA Numbers 1 and 2 are presently applicable.

FTAPRES(1)–(168)

Type: **Logical**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Off** (FTA Missing)
On (FTA Present)

IOP FTA Present Flag—For primary and secondary IOPs.

-G-

G (FlowComp)

Type: **Real**
Lock: **View**
Default: **1.0**
PtRes: **HPM**
Range: **N/A**

Specific Gravity Input—Indicates the value of the measured or calculated specific gravity or molecular weight.

GAINOPT (Pid)

Type: **E:GAINOPT**
Lock: **Eng/PB**
Default: **Lin**
PtRes: **HPM**

Gain (K) Option

Range: 0-**Lin** (Applies linear gain, with overall gain (K) = KLIN)

1-**Gap** (Reduces the sensitivity of control action when the PV is within a narrow band around the setpoint. If the PV is outside the gap, overall gain (K) = KLIN. If (SP - GAPLO) < PV < (SP + GAPHI), K = KLIN times KGAP)

2-**Nonlin** (Makes control action proportional to the error (PV - SP) squared with overall gain (K) =KLIN times KNL, where KNL = NLFM plus (NLGAIN times PV - SP)/100)

3-**Ext** (Applies external gain. Overall gain (K) = KLIN times KEXT, where KEXT is the positive external gain modifier)

GAPHI (Pid)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: $\geq$ **0.0**

Gap High Limit—Defines the upper limit of the gap in the same engineering units as the PV.

GAPLO (Pid)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: $\geq$ **0.0**

Gap Low Limit—Defines the bottom limit of the gap in the same engineering units as the PV.

GENDESC (1)–(12)

Type: **String_8**
Lock: **PtBld**
Default: **Blanks**
PtRes: **NIM**

Generic Descriptors—Define up to 12 generic descriptors that can be assigned to logic-slot parameters. As an example, six descriptors could be assigned to six logic-slot inputs, two descriptors to the logic block flags which will describe the current state of the logic slot based on the inputs, and two descriptors to the SO outputs from the logic slot. Refer to the description of the PRMDESC parameter, and to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: Permissible character set for the eight-character generic descriptors consists of all characters on the Engineer' s Keyboard. Basically this set consists of alphabetics A-Z, numerics 0-9, and the following special characters: space ! " % & ' ( ) * + - / : ; > < = ? _ , . $

Helpful Hint:	Example: GENDESC(7) is the descriptor for parameter PRMDESC(7), etc.

GENDESC(nn)

Type: **String_72**
Lock: **View**
Default: **Blanks**
PtRes: **HPM**

Generic Descriptor—Used as additional display text to help the operator diagnose potential problems with the IOP. It is primarily used with diagnostic displays.
nn = 1–40 specifies one of the 40 acting primaries.
nn = 129–168 specifies one of the 40 acting secondaries.

GISRC(1—4) (RegCtl, RegPV)

Type:
Lock:
Default:
PtRes: **HPM**
Range:

General Input Source—Specifies the Tag**.**Parameter source of General Input Connection.

GIDSTN(1—4) (RegCtl, RegPV)

Type:
Lock:
Default:
PtRes: **HPM**
Range:

Parameter Destination General Input Connection—Specifies the RegPV/RegCtl parameter destination of the General Input Connection

GIENBL(1—4) (RegCtl, RegPV)

Type:
Lock:
Default:
PtRes: **HPM**
Range:

General Input Connection Enable Flag—

GOSRC(1—4) (RegCtl, RegPV)

Type:
Lock:
Default:
PtRes: **HPM**
Range:

Parameter Source of General Output Connection—Specifies the RegPV/RegCtl parameter source of the General Output Connection

GODSTN(1—4) (RegCtl, RegPV)

Type:
Lock:
Default:
PtRes: **HPM**
Range:

Parameter Destination General Output Connection—Specifies the Tag.parameter destination of the General Output Connection

GOENBL(1—4) (RegCtl, RegPV)

Type:
Lock:
Default:
PtRes: **HPM**
Range:

General Output Connection Enable Flag—

GSTS

Type: **E:PVVALST**
Lock: **View**
Default: **Normal**
PtRes: **HPM**
Range: 0-**Bad** (Value is bad and replaced with NaN)
1-**Uncertn** (Status of the value is uncertain)
2-**Normal** (Value is good)

Gravity Input Value Status—Indicates the status of the gravity input value.

-H-

HIGHAL (AnalgIn, RegCtl, RegPV)

Type: **E:ALMTYPE**
Lock: **View**
Default: **NoAlarm**
PtRes: **NIM**

Highest Alarm Detected—Indicates the highest alarm currently detected at the data point**.** This parameter is used by the system to ensure that when two or more different types of alarms occur on a point at the same time, the most important or highest level alarm appears on the point's Group, Detail, and Alarm Summary displays. For example, if both the PV High High and PV High alarm priorities are set to Emergency, and both are in alarm, HIGHAL contains the PVHH value.

Range:
NoAlarm (No alarm exists—lowest level alarm)
AdvDev (Advisory Deviation)
DevHi (Deviation High)
DevLo (Deviation Low)
PVRocN (PV Rate Of Change Negative)
PVRocP (PV Rate Of Change Positive)
PVHi (PV High)
PVHH (PV High High)
PVLo (PV Low)
PVLL (PV Low Low)
BadCtl (Bad Control)
BadPV (Bad PV—highest level alarm)
BOC (Bad Output alarm)

HIGHAL (DevCtl, DigComp, DigIn, Flag, Logic)

Type: **E:ALMTYPE**
Lock: **View**
Default: **NoAlarm**
PtRes: **NIM**

Highest Alarm Detected—Indicates the highest alarm currently detected at the data point**.** This parameter is used by the system to ensure that when two or more different types of alarms occur on a point at the same time, the most important or highest level alarm appears on the point's Group, Detail, and Alarm Summary displays.

Range:
NoAlarm (No alarm has been detected)
OffNorm (Current PV state is not the configured PVNORMAL state. For a flag point, the off-normal state (STATE1) is the alarmed state.)
UnCEvt (Uncertain event was detected. Does not apply to a flag point.)
CmdDis (Command Disagree; field device did not respond to commanded output state. Does not apply to a flag point.)
BadPV (PV is bad)
C1 - C4ALM (1 to 4 custom logic alarms)
Chngofst (State has changed)
Cmdfail (PV failed to change after OP changed)
SVHI (SECVAR>SVHITP)
SVHH (SECVAR>SVHHTP)
BadSV (SECVAR is Bad)
OVRDI2 (Override Interlock I2)
OVRDI1 (Override Interlock I1)
OVRDI0 (Override Interlock I0)
OVRDSI0 (Safety Override Interlock)
BadCtl (Bad Control) (DevCtl and DigComp only)

HIGHALPR (AnalgIn, RegCtl, RegPV)

Type: **E:ALPRIOR**
Lock: **View**
Default: **NoAction**
PtRes: **NIM**
Range: **Emergncy** (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
JnlPrint (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

Highest Level Alarm's Priority—Defines the priority of the highest alarm currently detected at the data point. Associated with HIGHAL.

HISVPEAK (DevCtl)

Type: **Real**
Lock: **View**
Default: **0.0**
PtRes: **HPM**
Range: ≥ **0**

Highest Peak SECVAR Value—The highest peak value of the SECVAR parameter since the most recent reset of maintenance statistics.

HLCALIB(1)–(168)

Type: **Logical**
Lock: **Eng/Pb**
Default:
PtRes: **HPM**
Range: **Off -** Calibration is not in progress
On - Calibration is in progress

HLAI in Calibration Flag—Shows which HLAIs are presently in calibration

HOLDCMD (RampSoak)

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

Hold Command Flag—If On, allows users to hold the ramp or soak segment at its current position to customize the guaranteed ramp and soak function.

HOUR (HPM Box)

Type: **Integer** **Current Hour—**The value of the LCN time in the HPM.
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **0** to **23**

Only full array access is supported.

HWYCTLST (UCN)

Type: **E:$NODFSTA** **UCN Network Functional State**
Lock: **Supr**
Default: **Basic**
PtRes: **NIM**
Range: **Full** (All LCN devices can do read/write operations to this UCN)
Basic (AM and CM cannot write to this UCN)

-I-

I0–2 (DevCtl, DigComp)

Type: **Logical**
Lock: **Engr**
Default: **Off**
PtRes: **HPM**

Override Interlocks for Output States 0-2—Override interlocks force the commanded output to a specific state regardless of the condition of the permissive interlocks or the previous point state. The operator and user program cannot change the output state when any override interlock is On. An override interlock is provided for each of the three states. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: **Off** (Override interlock has no effect on the point state)
On (Override interlock sets the point to the respective state)

Helpful Hint:
1. When I0 is On, forces the output to STATE0, regardless of the permissives or any other overrides.
2. When I1 is On and I0 is Off, forces the output to a STATE1, regardless of the permissives or any other overrides.
3. When I2 is On and I0 and I1 are both Off, forces the output to STATE2 regardless of the permissives or any other overrides.
4. I0-I2 change by the engineer, requires PTEXECST = InActive or PNTSTATE = Idle for each interlock.

I0CONF (DigComp,DevCtl)

Type: **Logical**
Lock: **Oper**
Default: **Off**
PtRes: **HPM**
Range: **N/A**

Override Interlock 0 Alarm Confirmation Flag—Indicates that the Override Interlock 0 Alarm needs to be confirmed.

I0DESC-I2DESC (DigComp)

Type: **String_8**
Lock: **Engr**
Default: **Blank**
PtRes: **HPM**

I0-I2 Alarm Descriptor—The override Interlock for States 0, 1, or 2 indicating which text should be copied into the OVRDDESC parameter when an override alarm occurs. The text appears in the Alarm Display and can be configured to indicate the cause for the alarm.

Range: **8 Character String**

I1CONF (DigComp,DevCtl)

Type: **Logical**
Lock: **Oper**
Default: **Off**
PtRes: **HPM**
Range: **N/A**

Override Interlock 1 Alarm Confirmation Flag—Indicates that the Override Interlock 1 Alarm needs to be confirmed.

I2CONF (DigComp,DevCtl)

Type: **Logical**
Lock: **Oper**
Default: **Off**
PtRes: **HPM**
Range: **N/A**

Override Interlock 2 Alarm Confirmation Flag—Indicates that the Override Interlock 2 Alarm needs to be confirmed.

IN0–12 (GenLin)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **> prev. coord.**
< next coord.

Input Coordinates 0–12—Define the input value at the respective coordinate.
IN0 <IN1 <IN2 ...., <IN12

INITMAN

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**

Initialization Manual Flag—When On, indicates that this point is in Initialization Manual. The mode of the point does not change; however, INIT appears on the point's detail or group display to indicate that the point is in Initialization Manual. While the point is in Initialization Manual, an operator, supervisor, or engineer cannot change the point's output. The output is indisposable because initialization is being requested from downstream. Upon leaving Initialization Manual, the point's output is initialized from the point's secondary as determined by the point's output connection.

Range: **Off** (Mode ≠ Initialization Manual)
On (Mode = Initialization Manual)

> *Helpful Hint:* OP changes with Operator, Supervisor, or Engineer access level, requires MODE = Man and INITMAN = Off. SP changes with Operator, Supervisor, or Engineer access level, for non-PID algorithms requires MODE = Auto and INITMAN = Off, while for PID algorithms requires that MODE = Auto, and also that INITMAN = Off and PTEXECST = Active if PVTRACK = Track.

INITMAN (DigComp, DevCtl, RegCtl)

Type: **Logical**
Lock: **View**
Default: **On**
PtRes: **HPM**
Range: **Off** (Mode ≠ Initialization Manual)
On (Mode = Initialization Manual)

Initialization Manual Flag—On, indicates that an output is storing to a DO point that has its INITREQ flag set and the point is forced into initialization. When the DO point becomes available, the initialization state is cleared.

INITREQ(1)–(4) (RegCtl)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**

Initialization Request Flags (1–4)—Indicates whether an initialization request has been made. Each flag represents a request to the primary point pushing to the corresponding input to be initialized as follows:
Flag 1: SP or X1
Flag 2: RATIO or X2
Flag 3: X3
Flag 4: X4

Range: **Off** (No initialization request)
On (Initialization request)

INITREQ (Array)

Type: **Logical**
Lock: **View**

Initialization Request Flag—Indicates whether a Serial Interface-connected device can be written to, where OFF = yes, or ON = no. The flag is always OFF if EXTDATA = None.

Default: **On when EXTDATA = IO_FL, IO_NN, or IO_STR**
Off when EXTDATA = None
PtRes: **HPM**
Range: **Off** (EXTDATA=None, or Serial interface-connected device can be written to)
On (Serial interface-connected device cannot be written to)

INITREQ (AO, DO)

Type: **Logical**
Lock: **View**
Default: **On**
PtRes: **HPM**

Initialization Request Flag—When On, indicates that control strategies in the HPM cannot manipulate the output to the field. It is set to ON when:

- the PWM type output is configured
- the point is inactive
- the module is idle
- there is a soft failure such that the channel is not working
- The output is connected to standby-manual device

Range: **Off** (No initialization request)
On (Initialization request)

INITREQ(0)–(2) (DigComp, DevCtl)

Type: **Logical**
Lock: **View**
Default: **On**
PtRes: **HPM**

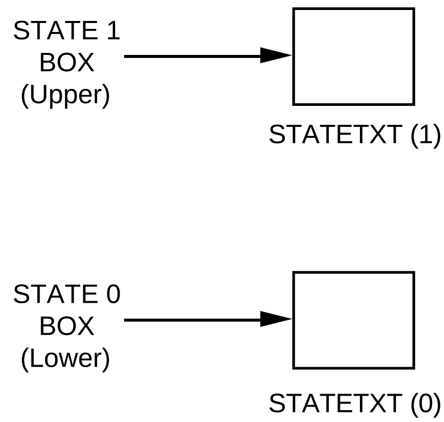
Initialization Request Flag—When On, indicates that CL programs or logic can not change the output to State(i), where i = 0, 1, or 2.

Range: **Off**
On

INITVAL

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

Initialization Value—Indicates the value to which the primary point is to be initialized.



PVRAW State	INPTDIR = Direct	INPTDIR = Reverse
ON	PV = ON. State 1 Box is lighted. State 0 Box is extinguished.	PV = OFF. State 0 Box is lighted. State 1 Box is extinguished.
OFF	PV = OFF. State 0 Box is lighted. State 1 Box is extinguished.	PV = ON. State 1 Box is lighted. State 0 Box is extinguished.

Figure I -1 — INPTDIR vs. Display Indicators 2065

INPTDIR (DigIn)

Type: **E:POLARITY**
Lock: **Eng/PB**
Default: **Direct**
PtRes: **HPM**

Digital Input Direction—Defines the contact conditions required to light the upper or lower boxes on a Group or Detail Display for a digital input point. See Figure I-1.

Range: 0-**Direct**
State 0 (lower) box lighted =>PVRAW = Off
State 1 (upper) box lighted =>PVRAW = On

1-**Reverse**
State 0 (lower) box lighted =>PVRAW = On
State 1 (upper) box lighted =>PVRAW = Off

IOLASTS (HPM Box)

Type: **Logical** **I/O Link Cable A Status**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (I/O Link cable A not in error)
On (I/O Link cable A in error)

IOLBSTS (HPM Box)

Type: **Logical** **I/O Link Cable B Status**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (I/O Link cable B not in error)
On (I/O Link cable B in error)

IOLCHAER (HPM Box)

Type: **Integer** **I/O Link Channel A Error Count**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: $\geq$ **0**

IOLCHASL (HPM Box)

Type: **Integer** **I/O Link Channel A Silence Count**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: $\geq$ **0**

IOLCHBER (HPM Box)

Type: **Integer** **I/O Link Channel B Error Count**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: $\geq$ **0**

IOLCHBSL (HPM Box)

Type: **Integer** **I/O Link Channel B Silence Count**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: $\geq$ **0**

IOLCHERT (HPM Box)

Type: **Integer**
Lock: **EngOnly**
Default: **10**
PtRes: **HPM**
Range: **≥ 0**

I/O Link Channel Error Threshold—Defines the acceptable number of I/O Link channel errors per minute before disabling the periodic I/O Link channel swap.

IOLCMD (HPM Box)

Type: **E:$IOLCMD** **I/O Link Command**
Lock: **EngOnly**
Default: **None**
PtRes: **HPM**
Range: 0-**None** (No effect)
1-**SelChnA** (Select I/O Link Channel A)
2-**SelChnB** (Select I/O Link Channel B)
3-**EnbPerSw** (Enable periodic swapping of IOL cables)
4-**DisPerSw** (Disable periodic swapping of IOL cables)
5-**RsIoLCom** (Reset IOL communication error count to 0)

IOLHWREV (HPM Box)

Type: **String_2** **HPMM I/O Link Interface Processor Card Hardware Revision—**
Lock: **View**
Default: **Blank**
PtRes: **HPM**
Range:

IOLPERSW (HPM Box)

Type: **E:ENBLSTAT** **I/O Link Periodic Cable Swap**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: 0-**Disable** (Swapping of I/O Link cables A & B is disabled)
1-**Enable** (Swapping of I/O Link cables A & B is enabled)

IOLPSERR (ProcMod)

Type: **E:Pastatus**
Lock: **View**
Default: **NoError**
PtRes: **HPM**

I/O Link Poststore Failure Option — Contains the IOL Poststore PA status failure code, or null.

Helpful Hint:	This parameter sould be used with IOLPSOPT.

Range: **NoError**

IOLPSOPT (ProcMod)

Type: **E:$IOLPSOPT** **I/O Link Poststore Failure Option —**
Lock: **Engr**
Default: **Fail**
PtRes: **HPM**

Helpful Hint: The program should check the value of IOLPSERR if this parameter is continue.

Range: **Fail** (program fails on a bad IOL store)
Continue (program continues on a bad IOL store)

IOLREV (HPM Box)

Type: **Integer** **HPMM I/O Link Software Revision—**
Lock: **View**
Default: **Blank**
PtRes: **HPM**

Range:

IOLVERS (HPM Box)

Type: **Integer** **HPMM I/O Link Software Version—**
Lock: **View**
Default: **Blank**
PtRes: **HPM**
Range:

IOMACTYP(1)–(168)

Type: **E:$PMMDTY** **IOP Actual Type —** Actual type of IOP at module address**.**
Lock: **View** This should match the configured type.
Default: **None**
PtRes: **HPM**
Range: **None** (Not Configured)
LLAI (Low Level Analog Input)
HLAI (High Level Analog Input)
DI (Digital Input)
DO (Digital Output)
AO (Analog Output
HPMM (HPM Module)
LLMUX (Low Level Analog Input Multiplexer) Also includes RHMUX (Remote Hardened Analog Multiplexer)
STIM (Smart Transmitter Interface Module)
PI (Pulse Input)

IOMCARD(1)–(168) (HPM Box)

Type: **Integer**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **1-15**

I/O module card position for the acting primary/secondary (used for diagnostic displays)**.**
nn = 1 - 40 correspond to card positions of the 40 acting primaries
nn = 129 - 168 correspond to card positions of the 40 acting secondaries

IOMCARDA(1)–(40) (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0 in IOP database; per PKGOPT on GDF**
PtRes: **HPM**
Range: **0 - 15** (0 specifies **Not Connected**)

I/O module A card position. 1–40 specifies one of the 40 logical I/O modules. The corresponding IOP must be connected to FTA connector A. Applies to the primary IOP only.

IOMCARDB(1)–(40) (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0 in IOP database; none on GDF**
PtRes: **HPM**
Range: **0 - 15 (**0 specifies **Not Connected)**

I/O module B card position. 1-40 specifies one of the 40 logical I/O modules. The corresponding IOP must be connected to FTA connector B. Applies to primary IOP only.

IOMCHAER(1)–(168)

Type: **Integer**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **0 - 255**

IOP Channel A Error Count—for a specific IOP

IOMCHASL(1)–(168)

Type: **Integer**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **0 - 255**

IOP Channel A Silence Count—for a specific IOP

IOMCHBER(1)–(168)

Type: **Integer**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **0 - 255**

IOP Channel B Error Count—for a specific IOP

IOMCHBSL(1)–(168)

Type: **Integer** **IOP Channel B Silence Count—**for a specific IOP
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **0 - 255**

IOMCMD (HPM Box)

Type: **E:$IOMCMD** **IOP Module Command**—Indicates IO module state, or whether to swap redundant pairs.
Lock: **Oper**
Default: **None**
PtRes: **HPM**
Range: **None**
Run
Idle
Swap

IOMCOMER(1)–(168)

Type: **E$:IOMCOMMIOP** **IOP Communications Error Status—**for a specific IOP
Lock: **View**
Default: **N/A**
PtRes: **HPM**

Range: **None** - No error
Invalert - Invalid alert; message bit problem
Invdest - Invalid destination
Invchcnt - Invalid character count; message corrupted
Invsourc - Invalid source
Invcmd - Invalid command
Checksum - Checksum error
No_resp - No response
Chtimout - Channel time out
Msgovrun- Message overrun
Gaperror - Gap error; message gap too long
Lpbckerr - Loopback error
Nth_0 - Next token holder equals zero
Tknrecov - Token recovery in progress
Rplbufov - Reply buffer overflow

IOMFILE(1)–(168) (HPM Box)

Type: **Integer** **I/O Module File Position for the Acting Primary/Secondary** (used for diagnostic displays)**.**
Lock: **View**
Default: **N/A**
PtRes: **HPM**

nn = 1 - 40 are file positions of the 40 acting primaries.
nn = 129 - 168 are file positions of the 40 acting secondaries.

Range: **0 - 8 (**0 specifies **Not Connected)**

IOMFILEA(1)–(40) (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0 in IOP data base; per PKGOPT on GDF**
PtRes: **HPM**
Range: **0 - 8 (**0 specifies **Not Connected)**

I/O Module A File Position—1-40 specifies one of the 40 logical I/O modules. The corresponding IOP must be connected to FTA connector A. Applies to the primary IOP only.

IOMFILEB(1)–(40) (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0 in IOP data base; none on GDF**
PtRes: **HPM**
Range: **0 - 8 (**0 specifies **Not Connected)**

I/O Module B File Position—1-40 specifies one of the 40 logical I/O modules. The corresponding IOP must be connected to FTA connector B. Applies to the primary IOP only.

IOMFWREV(1)–(168)

Type: **Ascii_2**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **X.Y** **X = Version, Y = Revision (For Release 300, X = 3)**

IOP Card Firmware Revision Status
(This is not the same as the external letter code on the card)

IOMHWREV(1)–(168)

Type: **Ascii_2**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **Hexadecimal characters 00–FF**

IOP Card Hardware Revision Status
The status of R300 boards appears as $2x, the status of R210 appears as $0x, where x is the version (0=A, 1=B, 2=C, etc.)

IOMLHFST(1)–(168)

Type: **E:$IOMHF**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: 0-**Unknown** (Unknown Status)
1-**PowerDwn** (This IOP Powered Down)
2-**InvPrgEx** (Invalid Program Execution)
3-**EpromErr** (EPROM Checksum Error)
4-**RamCntEr** (RAM Contents Error)
5-**RamAdrEr** (RAM Addressing Error)
6-**DpaError** (Device Physical Address Error)
7-**DsaError** (Device Soft Address Error)
8-**RxBufOfl** (I/O-Link Receive Buffer Overflow)
9-**IOLJaber** (I/O-Link Jabber Circuit Failure)
11-**BadPgJmp** (Illegal Value of Case Control)
12-**AdCIncmp** (A-to-D Conversion Incomplete)
13-**AdOutOvf** (A-to-D Output Value Overflow)
14-**AdOutUdf** (A-to-D Output is less than Zero)
15-**AdCCalEr** (A-to-D Calibration is incorrect)
16-**BadDcLtc** (Bad DC LTC)
17-**Dmt_Tmot** (Dead Man Timer Timeout)
18-**MLtOutFl** (Multiple Output Failures)
20-**BadDaRng** (Bad D-to-A Range)
21-**MstrTmot** (Master 68 k Timeout)

Input/Output Processor Last Hard Fail Status—Refer to the *HPM Service Manual* for a detailed description and the recommended corrective action.

IOMNUM

Type: **Integer**
Lock: **PtBld**
Default: **N/A**
PtRes: **HPM**
Range: **1** to **40**

IOP Number—IOMNUM specifies the IOP on the I/O Link that this point references for its process data.

IOMOPER(1)–(168) (HPM Box, IOP)

Type: **E:$PRIMSEC**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: 0-**Primary** (Primary IOP is operating)
1- **Secondry** (Secondary IOP is operating)

Input/Output Processor In Operation

IOMREALT (HPM Box)

Type: **E:$PMMDTY** **Actual Input/Output Processor Type**
Lock: **View**
Default: **None**
PtRes: **HPM**
Range: 0-**None** (Not Configured)
1-**LLAI**
2-**HLAI** (High-Level Analog Input)
3-**DI** (Digital Input)
4-**DO** (Digital Output)
5-**AO** (Analog Output)
7-**LLMUX** (Low-Level Analog Input Multiplexer) also includes RHMUX (Remote Hardened Analog Multiplexer)
14-**STIM** (Smart Transmitter Interface Module)
17-**PI** (Pulse Input)

IOMRECHN(1)–(168) (HPM Box)

Type: **E:$RECCHN** **IOP Receive Channel**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **ChannelA**
ChannelB

IOMSEVER(1)–(168) (HPM Box)

Type: **E:$SEVERTY** **Error Severity Based on Input/Output Processor State**
Lock: **View** **nn = 1 - 40** specifies the severity of 1 of the 40 acting primaries
Default: **nn = 129 - 168** specifies the severity of 1 of the 40 acting secondaries
PtRes: **HPM**
Range: **Ok** (I/O Processor has no errors and is OK)
Fail (I/O Processor has failed)
Inform (I/O Processor should be calibrated soon)
Warning (I/O Processor is on the verge of failing)

IOMSTS(1)–(168) (HPM Box)

Type: **E:$IOMSTS** **Input/Output Module State**
Lock: **View** **nn = 1 - 40** specifies the status of 1 of the 40 acting primaries
Default: **N/A** **nn = 129 - 168** specifies the status of 1 of the 40 acting secondaries
PtRes: **HPM**
Range: 0-**PowerOn** (Transient state when power is turned on)
1-**Idle** (In the Idle State)
2-**OK** (Running)
3-**NoResp** (No Response)
4-**IdleSF** (In the Idle State and has a Soft Failure)
5-**SoftFail** (Running and has a Soft Failure)
6-**CommErr** (Communication Error)
7-**ConfgMis** (Configuration Mismatch)
8-**NotConfg** (This IOP is Not Configured)
9-**NonExist** (This IOP does Not Exist)

UnAvail (Transient state during which status for this IOP is unavailable)

IOMTYPE(1)–(168) (IOP)

Type: **E:$PMMDTY** **Input/Output Processor Type**
Lock: **View**
Default: **None**
PtRes: **HPM**

Range: 0-**None** (Not Configured)
1-**LLAI** (Low Level Analog Input)
2-**HLAI** (High Level Analog Input, 16 slot)
3-**DI** (Digital Input, 32 slot)
4-**DO** (Digital Output, 32 slot)
5-**AO** (Analog Output, 8 slot)
7-**LLMUX** (Low Level Analog Input Multiplexer) also includes RHMUX (Remote Hardened Analog Multiplexer)
10-**SI** (Serial Interface)
14-**STIM** (Smart Transmitter Interface Module)
16**DISOE** (Digital Input, Sequence of Events)
17-**PI** (Pulse Input)
24-**AO16** (Analog Output, 16 slot)
25**DO32** (Digital Output, 32 slot)

IOMTYPE (HPM Box)

Type: **E:$PMMDTY** **Configured Input/Output Processor Type**
Lock: **PtBld**
Default: **NotConfg**
PtRes: **HPM**
Range: 0-**NotConfg** (Not Configured)
1-**LLAI** (Low-Level Analog Input)
2-**HLAI** (High-Level Analog Input)
3-**DI** (Digital Input)
4-**DO** (Digital Output)
5-**AO** (Analog Output)
7-**LLMUX** (Low Level Analog Input Multiplexer) also includes RHMUX (Remote Hardened Analog Multiplexer)
14-**STIM** (Smart Transmitter Interface Module)
17-**PI** (Pulse Input)
-**DISOE** (Digital Input, Sequence of Events)
-**SI** (Serial Interface)
-**AO-16** (Analog Output)
-**DO-32** (Digital Output)

IONTOKEN (HPM Box)

Type: **Integer** **IOP Next Token Holder**
Lock: **View**
Default: **N/A**
PtRes: **HPM**

NOTE

This parameter is available to the nodes on the LCN, but cannot be accessed on the UCN, either by HPM/CL programs or print connections.

Range: **0, or 128** to **255**

IOPDESC(1 - 40)

Type: **String_8** **IOP Description—**Provides an 8-character description of the IOP.
Lock: **View**
Default: **Spaces**
PtRes: **HPM**

Helpful Hint: An 8-character string is read from the IOP's EPROM and stored in the HPMM. The text string appears on the IOP Detail Display. Even if the IOP fails, an operator can identify the IOP/FTA for maintenance. Not all IOPs have this feature yet.

Range: 8 characters

IOPIDAY (HPM Box)

Type: **Integer** **HPMM I/O Link Personality Creation Date- Day**
Lock: **View**
Default: **0**
PtRes: **HPM**

Range:

IOPIMON (HPM Box)

Type: **Integer** **HPMM I/O Link Personality Creation Date- Month**
Lock: **View**
Default: **0**
PtRes: **HPM**

Range: **N/A**

IOPIYEAR (HPM Box)

Type: **Integer** **HPMM I/O Link Personality Creation Date- Year**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

IOPNUM (Array)

Type: **Integer**
Lock: **PtBld**
Default: **N/A**
PtRes: **HPM**
Range: **1** to **127**

Serial Interface IOP Module Number—Defines the module number of the serial interface IOP.

IOPSTR1(1)–(40) (HPM Box)

Type: **String_64**
Lock: **View**
Default: **Spaces**
PtRes: **SI**
Range: **N/A**

IOP String for FTA #1—Contains user-defined string data shown in the Box Detail display such as the FTA application name, its revision number, and date. NN = 1-40 specifies the Serial Interface IOP module number.

IOPSTR2 (1)–(40) (HPM Box)

Type: **String_64**
Lock: **View**
Default: **Spaces**
PtRes: **SI**
Range: **N/A**

IOP String for FTA #2—Contains user-defined string data shown in the Box Detail display such as the FTA application name, its revision number, and date. NN = 1-40 specifies the Serial Interface IOP module number.

IORECCHN (HPM Box)

Type: **E:$RECCHN** **I/O Link Receive Cable**—The cable the I/O module is currently listening on.
Lock: **PtBld**
Default: **None**
PtRes: **HPM**
Range: **A** (I/O module is listening on Cable A)
B (I/O module is listening on Cable B)

IOREDOPT(1)–(40) (HPM Box)

Type: **E:$REDOPT** **IOP Redundancy Option**—Indicates if an IOP is configured for redundancy
Lock: **PtBld**
Default: **NonRedun**
PtRes: **HPM**
Range: 0-**Redun**
1-**NonRedun**

IOSTKNDR

Type: **Integer** **IOP Token Drop Count**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **0 to 32767**

IOSCNCYC(1 – 40)

Type: **Integer** **Control base cycle number—**
The index to this parameter specifies the IOP number for which this information is being accessed.
Lock: **PtBld**
Default: **0.0**
PtRes: **HPM**
Range: **0 - 16**

IOSCNPER(1 – 40) (AnalgIn)

Type: **Real** **IO data Scan Period—** the I/O data scan period in seconds for IO processors that support Analong Input point types. The index to this parameter specifies the IOP number for which this information is being accessed.
Lock: **PtBld**
Default: **0.0**
PtRes: **HPM**
Range: **0.0, 0.0625, 0.125, 0.25, 0.5, 1.0**

-K-

K (AutoMan)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **N/A**

Gain Constant for X2 Input—Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

K (MulDiv, RegCtl Summer)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range:

Overall Gain—

K (Pid)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **0.0** to **240.0**

Overall Gain—Value of K depends on the chosen gain option. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

K (PosProp)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **0.0** to **10.0**

Gain Constant

K1 (PidErfb)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: **0.0** to **1.0**

External Reset Feedback Gain

K1 (PIDPosPr)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **0.0** to **10.0**

Gain Constant

K1–K2 (RatioCtl)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **N/A**

K1 = Ratio Scale Factor; K2 = Scale Factor for X2 Input—When used in conjunction with the Calcultr algorithm, K1 must be equal to C1, and K2 must be equal to C2.

K1–K3 (MulDiv)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range:

Gain Constants for X1–X3 Inputs

K1–K4 (RegCtl Summer)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range:

Gain Constants for X1–X4 Inputs

K1–K4 (IncrSum)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: ≥ **0.0**

Gain Constants for X1–X4 Inputs

KEXT(Pid)

Type: **Real**
Lock: **Prog**
Default: **1.0**
PtRes: **HPM**
Range: **0.0** to **240.0**

External Gain Modifier—Defines the external gain modification factor. It can be entered by a user-written program, or it can be an input from another data point.

KEYWORD

Type: **String_8**
Lock: **PtBld**
Default: **Blank**
PtRes: **NIM**
Range: Alphabetics A-Z (upper case only).
Numerics 0-9 (an all numeric keyword is not allowed).
Underscore (_) cannot be used as the first character or the last character in a keyword.
Consecutive underscores are not allowed. Do not use quote marks (").

Keyword Descriptor— An eight-character descriptor that is used to describe an important aspect of this particular data point. For example, in Figure N-1 (see NAME) the keyword for the data point is REFLUX.

KFF (PidFf)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: ≥ **0.0** to ≤ **1.0**

Gain for Feed Forward Input—Scale factor which is used in converting the FF input value to percent.

KGAP (Pid)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **00** to **1.0**

Gap Gain Factor—Defines the gain-modification factor.

KLIN (Pid)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **0.0** to **240.0**

Linear Gain Factor—Defines the linear gain in percent per percent.

KNL (Pid)

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

Nonlinear Gain Modifier—Indicates the calculated value of the nonlinear gain modifier.

-L-

L(1)–(12) (DevCtl, Logic)

Type: **Logical, Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **Real**

Value of the External Input—L(1)–L(12) are the 12 inputs to a logic slot fetched with input connections from other points. Each input can be a Boolean, an Integer, or a Real number. Integer input values are converted to real numbers before being stored into the database.

Helpful Hint:	L, if accessed from the LCN, must be accessed as a Logical data type.

LCNRECHN (HPM Box)

Type: **E:$RECCHN**
Lock: **View**
Default: **ChanneLA**
PtRes: **HPM**
Range: 0-**ChanneLA** (NIM is listening to LCN channel A)
1-**ChanneLB** (NIM is listening to LCN channel B)

LCN Receive Channel—Indicates the LCN channel to which the NIM is listening.

LDNGNODE (HPM Box)

Type: **Integer**
Lock: **View**
Default:
PtRes: **HPM**
Range:

UCN Node Performing Personality Image Load to This Node

LIBADOPT (DevCtl, Logic)

Type: **E:$LIBADOP**
Lock: **PtBld**
Default: **Hold**
PtRes: **HPM**
Range: 0-**On** (On state is substituted for bad input)
1-**Off** (Off state is substituted for bad input)
2-**Hold** (Last good value is substituted for bad input)

Logic Bad Input Handling Option—If a Boolean input is not successfully fetched for an input connection to the logic slot, its value is defaulted to one of the values (Off, On, Hold) selected through this parameter. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

LIBRYNUM

Type: **Integer**
Lock: **PtBld**
Default: **1**
PtRes: **NIM**
Range: **1-3**

NIM Library Number—Specifies the number of the NIM Library being configured. For Parameter Entry Display use only.

LIBRYTXT(1)–(1000)

Type: **String_8** **NIM Library Text**
Lock: **PtBld**
Default: **N/A**
PtRes: **NIM**
Range: **N/A**

LIDESC(1)–(12) (DevCtl)

Type: **String_8 in an Array (1..12)** **Input Descriptor—**External input descriptors.
Lock: **Engr**
Default: **Blank**
PtRes: **HPM**
Range: **8 Character String**

LINEPERD (1)–(168)

Type: **Real** **Line Period in Microseconds**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **15616.0** to **21759.0**

LISRC(1)–(12) (DevCtl, Logic)

Type: **Universal Ent.Prm**
Lock: **PtBld**
Default: **null.null**
PtRes: **HPM**

Logic Input Connection Source—Define the parameters whose current values are to be supplied to one or more of up-to-12 logic slot or Device Control inputs. The parameters can be specified using the "Tagname.Parameter" format or the hardware reference address format. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: Use Tagname.Parameter for tagged points where Tagname can be up to 16 characters and the permissible character set is as follows:

Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.
An * is used to default to this point's tag name.
Parameter name can be up to eight characters and must be a legitimate parameter name.

Some possible input-connection sources are

a. “AnalgIn slot Tagname.PV”
b."DigIn slot Tagname.PVFL"
c."Logic slot Tagname.SO(nn)" where nn = 1–24
d."Logic slot Tagname.Fl(nn)" where nn = 1–12
e.”Logic slot Tagname.NN(nn)” where nn = 1-8
f."ProcMod slot Tagname.Fl(nnn)" where nnn = 1–127
g.”ProcMod slot Tagname.NN(nn)” where nn = 1–80
h.”RegCtl slot Tagname.PV”
i.”RegPV slot Tagname.PV”
j."Box Flag slot Tagname.PVFL
k.”Box Numerics slot Tagname.NN” where nnnnn = 1-16,384
l."!Box.FL(nnnn)" for a box flag that resides in the same box where nnnnn = 1–16,384
m."$NMhhBxx.FL(nnnn)" for a box flag that resides in a different HPM box on the same UCN; hh is the NIM UCN address, xx is the HPM box number, and nnnn = 1–4095 (data access limit)

Use the hardware reference address !MTmmSss.Parameter for untagged or tagged points where

MT is the IOP type, such as DI (Digital Input)
mm is the IOP Card number (1–40)
The letter "S" is a constant
ss is the slot number on the IOP Card (refer to SLOTNUM parameter)
Parameter name can be up to eight characters and must be a legitimate parameter name.

LMREV (DevCtl)

Type: **E:POLARITY**
Lock: **Engr/PB**
Default: **Direct**
PtRes: **HPM**

Local Manual Polarity—Indicates whether point processing inverts the local manual input value.

Range: **Direct** (Value is not inverted)
Reverse (Value is inverted)

LMSRC (DevCtl, PosProp, PIDPosPr)

Type: **Universal Ent.Prm** — **Local Manual Source—**The input connection for the local manual input. Only inputs with logical data types are valid.
Lock: **PtBld**
Default: **Null.null**
PtRes: **HPM**
Range: Use Tagname.Parameter for tagged points where Tagname can be up to 16 characters and the permissible character set is as follows:

Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.
An * is used to default to this point's tag name.
Parameter name can be up to eight characters and must be a legitimate parameter name.

Some possible input-connection sources are

a."DigIn slot Tagname.PVFL"
b."Logic slot Tagname.SO(nn)" where nn = 1–24
c."Logic slot Tagname.Fl(nn)" where nn = 1–12
d."ProcMod slot Tagname.Fl(nnn)" where nnn = 1–127
e."Box Flag slot Tagname.PVFL
f."!Box.FL(nnnn)" for a box flag that resides in the same box where nnnn = 1–16,384
g."$NMhhBxx.FL(nnnn)" for a box flag that resides in a different HPM box on the same UCN; hh is the NIM UCN address, xx is the HPM box number, and nnnn = 1–4095 (data access limit)

Use the hardware reference address !MTmmSss.Parameter for untagged or tagged points where

MT is the IOP type, such as DI (Digital Input)
mm is the IOP Card number (1–40)
The letter "S" is a constant
ss is the slot number on the IOP Card (refer to SLOTNUM parameter)
Parameter name can be up to eight characters and must be a legitimate parameter name.

LOADFAIL

Type: **Integer** — **Node Load Failure Information**
Lock: **View**
Default:
PtRes: **HPM**
Range:

LOADFLAG

Type: **String_2** **Load Flag**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Hexadecimal characters 00 to FF**

LOADPCKT

Type: **Integer** **Current Personality Image Packet Being Loaded to This Node**
Lock: **View**
Default:
PtRes: **HPM**
Range:

LOADSCOP (NIM)

Type: **E:$LOADSCP**
Lock: **PtBld**
Default: **NIMAndPm**
PtRes: **NIM**

Load Scope—Defines the scope of the point-build procedure for NIM and HPM configuration. The point information is loaded to both the NIM and HPM or to the NIM only. A value of NIMONLY is typically used to configure points into the NIM only during installation of a new system without HPMs.

NOTE

When points are built to a NIM and the NIM is restarted with no database, the points need to be reloaded from checkpoint or the points must be reconfigured. If the database is to be reconfigured, the HPMM must be in Idle, and the point execution state must be Inactive. This allows the point build operation to override the database that already exists there.

NOTE

To delete active entities from the HPM database, the point must be put to the inactive state. An alternative is to delete the entity in the NIM only by changing the LOADSCOP parameter for the NIM to NimOnly and deleting the point. Be sure to restore LOADSCOP to NimAndPM after deleting points.

Range: **NimOnly** (Configured data is to be loaded into the NIM only)
NimAndPm (Configured data is to be loaded into the NIM and HPM)

LOADSTAT

Type: **E:LOADSTAT** **Load Status**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Notload**
Loaded
Loading
Unlding

LOCALMAN (AnalgOut, RegCtl)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Output is not being controlled by an Analog Display)
On (Output is being controlled by an Analog Display)

Local Manual Flag—Indicates whether the associated hardware output of this point is being controlled by a manually-operated analog display.

LOCALMAN (DigComp, DevCtl)

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

Local Manual Flag—When On, indicates that the output(s) is being locally controlled and not by the HPM. When this flag is on, it usually indicates that the "hand/off/auto" switch is not in the "auto" position.

LOCPRIM(1—4)

Type:
Lock: **View**
Default:
PtRes: **HPM**
Range:

Local Primary—Returns the tag name of a primary point in the same HPM that is storing to this point's parameters.

The parameter index indicates the parameter being pushed to by the primary as follows, depending on the algorithm configured in the Regulatory Control point:
1 = SP or X1
2 = Ratio or X2
3 = X3
4 = X4

> *Helpful Hint:* LOCPRIM returns a null entitty ID if the connection is not configured or the primary point is in a different node (such as, a peer-to-peer connection).

LODSTN(1)–(12) (Logic)

Type: **Blind Record in an Array (1..12)**
Lock: **PtBld**
Default: **null.null**
PtRes: **HPM**

Logic Output-Connection Destination—Specifies up to 12 destinations to which the current values of the logic slot outputs are supplied. The destinations can be specified using the "Tagname.Parameter" format or the hardware reference address format. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: Use Tagname.Parameter for tagged points where Tagname can be up to 16 characters, and the permissible character set is as follows:

Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.
An * is used to default to this point's tag name.
Parameter name can be up to eight characters, and must be a legitimate parameter name.

Some possible output-connection destinations are

a."DigOut slot tagname.ONPULSE or OFFPULSE"
b."DigOut slot Tagname.SO"
c."Logic slot Tagname.Fl(nn)" where nn = 7–12
d."ProcMod Tagname.Fl(nnn)" where nnn = 1–127
e."Flag slot Tagname.PVFL
f."!Box.FL(nnnn)" for a box flag that resides in the same HPM box where nnnn = 1-16,384.
g."$NMhhBxx.FL(nnnn)" for a box flag that resides in a different HPM box on the same UCN; hh is the UCN number, xx is the HPM box number of the destination parameter, and nnnn = 1-4095 (data access limit).

Use the hardware reference address !MTmmSss.Parameter for untagged or tagged points where

MT is the IOP type, such as DO (Digital Output)
mm is the IOP Card number (1-40)
The letter "S" is a constant
ss is the slot number on the IOP Card (refer to SLOTNUM parameter)
Parameter name can be up to eight characters and must be a legitimate parameter name.

LODSTN(1)–(2) (DevCtl)

Type: **Blind Record in an Array (1..2)**
Lock: **PtBld**
Default: **Null**
PtRes: **HPM**

Device Control Output Connection Destination—Specifies up to 2 destinations to which the current values of the Device Control slot outputs are supplied. The destinations can be specified using the "Tagname.Parameter" format or the hardware reference address format Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: Use Tagname.Parameter for tagged points where Tagname can be up to 16 characters and the permissible character set is as follows:

Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.
An * is used to default to this point's tag name.
Parameter name can be up to eight characters and must be a legitimate parameter name.

Some possible output-connection destinations are

a."DigOut slot tagname.ONPULSE or OFFPULSE"
b."DigOut slot Tagname.SO"
c."Logic slot Tagname.Fl(nn)" where nn = 7–12
d."ProcMod Tagname.Fl(nnn)" where nnn = 1–127
e."Flag slot Tagname.PVFL
f."!Box.FL(nnnn)" for a box flag that resides in the same HPM box where nnnn = 1-16,384.
g."$NMhhBxx.FL(nnnn)" for a box flag that resides in a different HPM box on the same UCN; hh is the UCN number, xx is the HPM box number of the destination parameter, and nnnn = 1-4095 (data access limit).

Use the hardware reference address !MTmmSss.Parameter for untagged or tagged points where

MT is the IOP type, such as DI (Digital Input)
mm is the IOP Card number (1–40)
The letter "S" is a constant
ss is the slot number on the IOP Card (refer to SLOTNUM parameter)
Parameter name can be up to eight characters and must be a legitimate parameter name.

LOENBL(1)–(2) (DevCtl)

Type: **E:$PMDVPRM in an Array (1..2)**
Lock: **PtBld**
Default: **FL2**
PtRes: **HPM**

Device Control Output Enable—Allows the respective output connection defined by LODSTN to write the value of the specified Device Control parameter to the destination. The logic output is allowed when the enable function, selected from the list below, is On. If the FL1 parameter is specified and the output data type is logical, output occurs only during change (normally, it is continuous).

Range: **FL1...FL12** [Local flag; either a 1 (On) or a 0 (Off)]
D1, D2 [Digital PV inputs, either a 1 (On) or a 0 (Off)]
SI0 [Safety interlocks, either a 1 (On) or a 0 (Off)]
I0, I1, I2 [Interlocks, either a 1 (On) or a 0 (Off)]
P0, P1, P2 [Permissives, either a 1 (On) or a 0 (Off)]
PISO1..PISO12 [Primary Input Gate Values (logical)]
SISO1..SISO12 [Secondary Input Gate Values (logical)]
PGSO1..PGSO4 [Primary Gate Output Values (logical)]
SGSO1, SGSO2 [Secondary Gate Output Values (logical])
L1..L12 [Logic input value to device control slot (logical)]

LOENBL(1)–(12) (Logic)

Type: **E:$PMMLGPM**
Lock: **PtBld**
Default: **FL2**
PtRes: **HPM**

Logic Output Enable—Allows the respective output connection defined by LODSTN to write the value of the specified logic-slot parameter to the destination. The logic output is allowed when the enable function, selected from the list below, is On. If the FL1 parameter is specified and the output data type is logical, output occurs only during change (normally, it is continuous).

Range: **FL1...FL12** [Local flag; either a 1 (On) or a 0 (Off)]
SO1...SO24 [Logic-block output; either a 1 (On) or a 0 (Off]
L1...L12 [Logic input value to logic slot (logical)]

LOGALGID(1)–(24) (Logic)

Type: **E:$PMMLGAL**
Lock: **PtBld**
Default: **NULL**
PtRes: **HPM**

Logic Block Algorithm Identifier—Defines the logic algorithm to be used for a particular logic block. A different logic algorithm can be specified for each logic block within a logic slot. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of each logic algorithm.

Range:

Algorithm ID	*Description*	*Input(s)*
0-**NULL**	No logic algorithm is executed	- - -
1-**AND**	AND Gate	*S1, S2, S3
2-**OR**	OR Gate	*S1, S2, S3
3-**NOT**	NOT Gate	S1
4-**NAND**	NAND Gate	*S1, S2, S3
5-**NOR**	NOR Gate	*S1, S2, S3
6-**XOR**	XOR Gate	S1, S2
7-**QOR2**	Qualified OR Gate with 2 Inputs On	S1, S2, S3, S4
8-**QOR3**	Qualified OR Gate with 3 inputs On	S1, S2, S3, S4
9-**SWITCH**	Switch	S1, S2, S3
10-**EQ**	Compare equal with deadband	R1, R2, DEADBAND
11-**NE**	Compare not equal with deadband	R1, R2, DEADBAND
12-**GT**	Compare > than with deadband	R1, R2, DEADBAND
13-**GE**	Compare > than or = with deadband	R1, R2, DEADBAND
14-**LT**	Compare < than with deadband	R1, R2, DEADBAND
15-**LE**	Compare < than or = with deadband	R1, R2, DEADBAND
16-**CheckBad**	Check for Bad	R1
17-**Pulse**	Fixed-size Pulse	S1, DLYTIME
18-**MinPulse**	Pulse with minimum time limit	S1, DLYTIME
19-**MaxPulse**	Pulse with maximum time limit	S1, DLYTIME
20-**Delay**	Either Direction	S1
21-**OnDly**	Off-On Delay	S1, DLYTIME
22-**OffDly**	On-Off Delay	S1, DLYTIME
23-**WatchDog**	Watchdog Timer	FL6
24-**FlipFlop**	Flip Flop	S1, S2, S3
25-**ChDetect**	Change Detect	S1, S2 , S3
26-**DISCREP3**	Discrepancy Gate with 3 inputs plus delay	S1, S2, S3, DLYTIME

*Inputs S1–S3 can be inverted as required

LOGICSRC (DigComp, DevCtl)

Type: **Ent_Id** **Logic Source**—Specifies the tag name of a point, usually a logic slot, that is controlling the interlock signals.
Lock: **PtBld**
Default: **Null**
PtRes: **NIM**
Range: Tag name can be up to sixteen characters and the permissible character set is as follows:
Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.

LOGMIX (Logic)

Type: **E:$LOGMIX** **Logic Mix—**Defines the number of input connections, logic blocks, and output connections this logic slot contains.
Lock: **PtBld**
Default: **12_24_4**
PtRes: **HPM**
Range:

	Input Connections LISRC(1)-LISRC(12)	*Number of Logic Blocks*	*Output Connections LOSRC(1)-LOSRC(12)*
12_24_4	12	24	4
12_16_8	12	16	8
12_8_12	12	8	12

LOSRC(1)–(2) (DevCtl)

Type: **E:$PMDVPRM in an Array (1..2)** **Device Control Output Connection Source—**Defines the Device Control parameter that is to provide its value to the output connection specified by parameter LODSTN(n), Device Control Output Connection Destination.
Lock: **PtBld**
Default: **FL1**
PtRes: **HPM**
Range: **FL1...FL12** [Local flag; either a 1 (On) or a 0 (Off)]
D1, D2 [Digital PV inputs, either a 1 (On) or a 0 (Off)]
SI0 [Safety interlocks, either a 1 (On) or a 0 (Off)]
I0, I1, I2 [Interlocks, either a 1 (On) or a 0 (Off)]
P0, P1, P2 [Permissives, either a 1 (On) or a 0 (Off)]
PISO1..PISO12 [Primary Input Gate Values (logical)]
SISO1..SISO12 [Secondary Input Gate Values (logical)]
PGSO1..PGSO4 [Primary Gate Output Values (logical)]
SGSO1, SGSO2 [Secondary Gate Output Values (logical)]
L1..L12 [Logic input value to device control slot (either logical or real)]
NN1..NN8 [Local numerics (real)]
PINN1..PINN12 [Numeric constant for arithmetic comparisons (real)]
SECVAR [Secondary variable input value (real)]

LOSRC(1)–(12) (Logic)

Type: **E:$PMMLGPM**
Lock: **Eng/PB**
Default: **FL1**
PtRes: **HPM**
Range: **FL1...FL12** [Local flag; either a 1 (On) or a 0 (Off)]
SO1...SO24 [Logic block output; logical 1 or 0)]
L1...L12 [Input to the logic slot (logical or real value)]
NN1...NN8 [Local numeric; data type of Real]

Logic Output Connection Source—Defines the logic-slot parameter that is to provide its value to the output connection specified by parameter LODSTN(n), Logic Output Connection Destination.

LOWERTIM

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

Lower Output Pulse Time (In Seconds)—Indicates the lower output pulse time in seconds. This value is clamped to MAXPULSE or CYCLETIM, whichever is lower. If LOWERTIM is smaller than RP*MINPULSE, no pulse is issued.

LOWRDSTN

Type: **Universal Ent.Prm**
Lock: **PtBld**
Default: **Null**
PtRes: **HPM**
Range: **ONPULSE**
OFFPULSE

Lower OP Pulse Destination—Defines the destination of the Lower output pulse. LOWRDSTN must point to parameter ONPULSE or parameter OFFPULSE of a DigOut point.

LOWRRATE

Type: **Real**
Lock: **Supr**
Default: **100.0%/sec.**
PtRes: **HPM**
Range: **>0.0 percent/second**

Lower OP Stroke Rate in Percent/Second

LRL (STI)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A, NAN**

Lower Range Limit—Indicates the lower range limit of the PV at the smart transmitter. This limit is fixed and cannot be changed. Refer to the description of the STI_EU parameter for the LRL engineering units.

LRV (STI)

Type: **Real**
Lock: **Supr/View**
Default: **NaN**
PtRes: **HPM**

Lower Range Value—Defines the lower end of the operating range for the PVRAW value. User entry for PVEULO is the user-entered engineering-unit value that corresponds to LRV. Refer to description of the STI_EU parameter for the LRV engineering units.

This parameter can be changed only when the STI point execution state PTEXECST is Inactive.

Range: **N/A, NaN**

LSEQNUMR

Type: **Integer**
Lock: **Eng**
Default: **0**
PtRes: **IOP**
Range: ?

Last Sequence Number—Specifies the sequence number of the last personality image file packet received by the IOP.

LSIOLORN(0) - (4) (HPM Box)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: $\geq$ **0**

Last Hour's I/O Link Fetch/Store Overruns—Indicates the number of I/O Link access overruns that have been detected in the last hour.

LSIOLORN is set equal to the contents of CRIOLORN, every hour on the hour.

LSPPXORN(0 - 8) (HPM Box)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**

Last Hour's Point Processing Overruns Per Cycle—Indicates the number of point processing overruns that have been detected in the last hour.

LSPPXORN is set equal to the contents of parameter CRPPXORN, every hour on the hour.

Range: $\geq$ **0**

LSTWHNER (ProcMod)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: $\geq$ **0**

Last When Error—The CL Fail/Error code masked by the “When Error” clause.

LSUCNORN (HPM Box)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: $\geq$ **0**

Last Hour's UCN Access Overruns—LSUCNORN is set equal to the contents of parameter CRUCNORN, the current hour's UCN Access Overruns, every hour on the hour.

-M-

M (IncrSum, ORSel, Switch)

Type: **Integer** — **Number of Inputs**
Lock: **PtBld**
Default: **2**
PtRes: **HPM**
Range: **2** to **4**

MAINDAT (DevCtl, DigComp)

Type: **Time**
Lock: **Engr**
Default: **Time of Point Build**
PtRes: **HPM**
Range: **Time Stamp** (DD MMM YY HH:MM:SS)

Maintenance Reset Statistics Date—The date and time of the reset of maintenance statistics that can also be written by the engineer. Statistics can be reset by the operator only when the device is red tagged, while programs can reset them at any time. Resetting is accomplished by setting the RESETFL to ON.

Helpful Hint:	This parameter is reset when the RESETFL parameter = ON.

MAINTOPT (DevCtl, DigComp)

Type: **Logical**
Lock: **PtBld**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Maintenance statistics are not available)
On (Maintenance statistics are available)

Maintenance Option—Indicates if the maintenance statistics option is used.

MANMODFL (RegCtl)

Type: **Logical**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **Off** (Current mode is other than Manual)
On (Current mode is Manual)

Manual Mode Flag—Indicates whether the current mode of the slot is Manual.

MANOPCMD

Type: **E:$MANOPCM**
Lock: **Oper**
Default: **None**
PtRes: **HPM**
Range: 0-**None** (No change)
1-**Raise_1** (Raise output by 1 MANOPTIM each keystroke)
2-**Lower_1** (Lower output by 1 MANOPTIM each keystroke)
3-**Raise_10** (Raise output by 10 MANOPTIMs each keystroke)
4-**Lower_10** (Lower output by 10 MANOPTIMs each keystroke)

Manual Output Pulse Command—Defines the output pulse command issued by the operator for raising and lowering the output. See also, MANOPTIM.

MANOPTIM

Type: **Real**
Lock: **Eng/PB**
Default: **1.0**
PtRes: **HPM**
Range: **0.0** to **60.0** seconds

Manual Output Pulse Time (in seconds)—Defines the width of the raise or lower output pulse that is issued by the operator.

MASKTIM (DevCtl)

Type: **Integer**
Lock: **Supr**
Default: **0**
PtRes: **HPM**
Range: **0** to **1000** seconds

Masktime—The amount of time the SECVAR parameter alarms are masked after a change in the output state.

MAXCNFPU (HPM Box)

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range:

Maximum Configurable PUs—Specifies

Helpful Hint:

MAXPU (ProcMod)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **0** to

Maximum PUs—Specifies the maximum PUs used for point processing.

Helpful Hint:

MAXPULSE

Type: **Real**
Lock: **Supr**
Default: **60**
PtRes: **HPM**
Range: **MINPULSE** to **60.0** seconds
NaN

Maximum Pulse Time Limit—Defines the maximum pulse time limit. If the calculated pulse time is greater than this value then a pulse of length MAXPULSE is issued.

MAXSLOTS

Type:	**Real**	**Maximum Available Slots—**Returns the maximum number of slots that can be configured in an IOP.
Lock:	**View**	
Default:	**0**	
PtRes:	**IOP**	
Range:	**0 - 127** slots	

Helpful Hint:	Applies to the following IOP types: AO16, DI32 and DO32.

MAXTIM0H (DevCtl, DigComp)

Type:	**Real**	**Maximum Time Allowed in State 1—**The maximum amount of time (based on the PV) in hours allowed for state 1.
Lock:	**Supr**	
Default:	**0**	
PtRes:	**HPM**	
Range:	**N/A**	

MAXTIM1H (DevCtl, DigComp)

Type:	**Real**	**Maximum Time Allowed in State 1—**The maximum amount of time (based on the PV) in hours allowed for state 2.
Lock:	**Supr**	
Default:	**0**	
PtRes:	**HPM**	
Range:	**N/A**	

MAXTIM2H (DevCtl, DigComp)

Type:	**Real**	**Maximum Time Allowed in State 2—**The maximum amount of time (based on the PV) in hours allowed for state 3.
Lock:	**Supr**	
Default:	**0**	
PtRes:	**HPM**	
Range:	**N/A**	

MAXTRAN0–2 (DevCtl, DigComp)

Type:	**Time**	**Maximum Number of Transitions into State—**This is the maximum number of transitions allowed in each state, and is the target value for maintenance statistics.
Lock:	**Supr**	
Default:	**0.0**	
PtRes:	**HPM**	
Range:	**0** (There is no limit)	

MDMHWREV (HPM Box, NIM)

Type:	**String_2**	**Modem Hardware Revision**
Lock:	**View**	
Default:		
PtRes:	**HPM, NIM**	
Range:	**Hexadecimal Characters 00 to FF**	

MEMFWREV

Type: **String_2** **Memory Firmware Revision**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Hexadecimal Characters 00 to FF**

MEMHWREV

Type: **String_2** **Memory Hardware Revision**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Hexadecimal Characters 00 to FF**

MINPULSE

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: **0.0** seconds to **MAXPULSE**
NaN

Minimum Pulse Time Limit—Defines the minimum pulse time limit for the Raise pulse. If the calculated pulse time value is smaller than this value, no pulse is issued.

MINUTE (HPM Box)

Type: **Integer**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **0** to **59**

Current Minute—Value of the LCN time in the HPM.

MNFASIC (HPM Box)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: 1 - 31

HPMM Communications Control Card ASIC Revision—

MNFCCDAY (HPM Box)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: 1 - 31

HPMM Communications Control Card Manufacturing Date-Day—

MNFCCINF (HPM Box)

Type: **String_8** **HPMM Communications Control Card Manufacturing Information—**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range:

MNFCCMTH (HPM Box)

Type: **Integer** **HPMM Communications Control Card Manufacturing Date-Month—**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: 1 - 12

MNFCCSER (HPM Box)

Type: **String_24** **HPMM Communications Control Card Serial Number—**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range:

MNFCCYR (HPM Box)

Type: **Integer** **HPMM Communications Control Card Manufacturing Date-Year—**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: 1 - 99

MNFFPGA (HPM Box)

Type: **Integer** **HPMM I/O Link Card FPGA Revision—**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: 1 - 31

MNFIODAY (HPM Box)

Type: **Integer** **HPMM I/O Link Card Manufacturing Date- Day**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **1 - 31**

MNFIOINF (HPM Box)

Type:	**String_8**	**HPMM IO Link Processor Card Manufacturing Information**
Lock:	**View**	
Default:	**Blank**	
PtRes:	**HPM**	
Range:	**0 - 99**	

MNFIOMTH (HPM Box)

Type:	**Integer**	**HPMM IO Link Card Manufacturing Date-Month**
Lock:	**View**	
Default:	**0**	
PtRes:	**HPM**	
Range:	**1 - 12**	

MNFIOSER (HPM Box)

Type:	**String_24**	**HPMM IO Link Card Serial Number**
Lock:	**View**	
Default:	**0**	
PtRes:	**HPM**	
Range:		

MNFIOYR (HPM Box)

Type:	**Integer**	**HPMM IO Link Card Manufacturing Date-Year**
Lock:	**View**	
Default:	**0**	
PtRes:	**HPM**	
Range:	**0 - 99**	

MNFMDDAY (HPM Box)

Type:	**Integer**	**HPMM UCN Interface Card Manufacturing Date-Day**
Lock:	**View**	
Default:	**0**	
PtRes:	**HPM**	
Range:	**1 - 31**	

MNFMDINF (HPM Box)

Type:	**String_8**	**HPMM UCN Interface Card Manufacturing Information**
Lock:	**View**	
Default:	**Blank**	
PtRes:	**HPM**	
Range:		

MNFMDMTH (HPM Box)

Type: **Integer** **HPMM UCN Interface Card Manufacturing Date-Month**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: 1 - 12

MNFMDSER (HPM Box)

Type: **String_24** **HPMM UCN Interface Card Serial Number**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

MNFMDYR (HPM Box)

Type: **Integer** **HPMM UCN Interface Card Manufacturing Date - Year**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: 0 - 99

MODATTR

Type: **E:MODATTR**
Lock: **Oper**
Default: **Operator**
PtRes: **HPM**

Mode Attribute—Defines whether the operator or the sequence program has the authority to change certain parameters of this data point. At the Universal Station, the mode attribute is displayed next to the mode of the data point. If the mode attribute is Program, a -P appears to the left of MODE. If the attribute is operator, blanks are displayed to the left of mode**.**

Range: 0-**Operator** (Operator can set Mode, OP, SP, Ratio, Bias)
1-**Program** (Program can set Mode, OP, SP, Ratio, Bias)
2-**Normal**
3-**None** (No mode attribute)

Helpful Hint:	MODATTR change requires SHUTDOWN = Off and REDTAG = Off. When the "normal mode" button on the Operator's keyboard is pressed, MODATTR = NMODATTR unless NMODATTR = None.

MODE (AnalgOut)

Type: **E:MODE**
Lock: **Oper**
Default: **Man**
PtRes: **HPM**

Mode—Defines the current mode of the data point. Parameter MODATTR determines whether operator or the sequence program provides the output value for this point. If PNTFORM is Component, then MODE parameter is not applicable for this data point.

Range:
1-**Man** (Operator or Program provides the point's output value (OP))
2-**Cas** (Data point receives its output value from a primary data point. If RCASOPT is DDC, data point receives its output value from an AM point.)
5-**Normal** (Parameter NMODE determines this point's mode)

> *Helpful Hint:*
> 1. MODE change by a program requires MODATTR = Program and REDTAG = Off.
> 2. MODE change by an operator requires MODATTR = Operator, MODEPERM = Permit, and REDTAG = Off.

MODE (DigComp, DevCtl)

Type: **E:MODE**
Lock: **Oper**
Default: **Man**
PtRes: **HPM**

Mode of Digital Composite and Device Control Slot—Defines the current mode of the data point. Parameter MODATTR determines whether operator or the sequence program provides the output value for this point. If PNTFORM is Component, then MODE parameter is not applicable for this data point.

Range:
1-**Man** (Operator or Program controls slot's output (OP))
5-**Normal** (Parameter NMODE contains slot's mode)

> *Helpful Hint:*
> 1. MODE change by a program requires MODATTR = Program, SHUTDOWN = Off, and REDTAG = Off.
> 2. MODE change by an operator requires MODATTR = Operator, MODEPERM = Permit, SHUTDOWN = Off, and REDTAG = Off.

MODE (RegCtl)

Type: **E:MODE**
Lock: **Oper**
Default: **Man**
PtRes: **HPM**

Mode of Regulatory Control Slot—Defines the mode of the RegCtl point.

Range:
1-**Man** (Operator or discontinuous program controls slot's output (OP), regardless of any automatic control strategy)
2-**Cas** (Upstream slot's OP is this slot's SP)
3-**Auto** (OP value is computed by the configured RegCtl algorithm, and the setpoint (SP) comes from the local setpoint (LSP) location in the RegCtl point. An operator or a discontinuous program can change the setpoint value.
4-**Bcas** (Local cascade mode where the RegCtl point receives its setpoint from the OP of a primary data point, even though the entry for the RCASOPT parameter is Spc, DdcRsp, or Rsp (where the AM provides the setpoint). In this way, should the AM or the NIM fail, the control strategy will shed to the local cascade mode.)
5-**Normal** (Parameter NMODE determines the normal mode of this slot)

> *Helpful Hint:*
> 1. MODE change by a program requires MODATTR = Program and REDTAG = Off.
> 2. MODE change by an operator requires MODATTR = Operator, MODEPERM = Permit, and REDTAG = Off.

MODEAPPL(1)–(4) (DevCtl, DigComp, RegCtl)

Type: **Logical**
Lock: **View**
Default: **Man=On**
MODEAPPL [Auto]=Off
MODEAPPL [Bcas]=Off
MODEAPPL[Cas]=Off
Static for DevCtl
and Digcomp points
PtRes: **HPM**
Range: **N/A**

Mode Applicability—Defines changes for Regulatory Control points:
MODEAPPL[1] = ON if MAN mode if valid, else it is OFF
MODEAPPL[2] = ON if AUTO mode is valid, else it is OFF
MODEAPPL[3] = ON if CAS mode is valid, else it is OFF
MODEAPPL[4] = ON if BCAS mode is valid, else it is OFF

MODEPERM (AO)

Type: **E:MODEPERM**
Lock: **Eng/PB**
Default: **Permit**
PtRes: **HPM**
Range: 0-**Permit** (Operator can change this point's mode)
1-**NotPerm** (Operator cannot change this point's mode)

Mode Permissive—Determines whether the operator can change the mode of this data point.

MODEPERM (DevCtl, DigComp, RegCtl)

Type: **E:MODEPERM**
Lock: **Eng**
Default: **Permit**
PtRes: **HPM**
Range: 0-**Permit** (Operator can change this point's mode)
1-**NotPerm** (Operator cannot change this point's mode)

Mode Permissive—Determines whether the operator can change the mode of this data point.

MODNUM

Type: **Integer**
Lock: **PtBld**
Default: **N/A**
PtRes: **NIM**
Range: **0** to **40** (0 is reserved for the HPMM)

HPMM/IOP Module Number—Defines the module number in the HPM. The HPMM is module number 0; the IOP Cards are module numbers 1–40.

MOMSTATE (DevCtl, DigComp)

Type: **E:$MOMSTAT**
Lock: **Eng/PB**
Default: **None**
PtRes: **HPM**
Range: 0-**None** (No momentary output states)
1-**Mom_1** (State 1 is momentary if NOSTATES = 2 or 3)
2-**Mom_0** (State 0 is momentary if NOSTATES = 2)
3-**Mom_2** (State 2 is momentary if NOSTATES = 3)
4-**Mom_1_2** (State 1 and State 2 are momentary; valid if NOSTATES = 3)

Momentary Output States—Defines which of the output states are momentary. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

MONPER (HPM Box)

Type: **Integer** **Monitoring Period—**Specifies the monitoring period in seconds
Lock: **Eng**
Default: **3600**
PtRes: **HPM**
Range: **4 - 3600** (must be in multiples of 4 seconds)

MONTH (HPM Box)

Type: **Integer** **Current Month—**The value of the LCN date in the HPM.
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **1** to **12** (January to December)

MOVPVFL

Type: **Logical** **Moving PV Flag—**Indicates whether the PV is moving from one state to another state.
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (PV is not moving)
On (PV is moving)

MOVPVTXT (HPM Box, DevCtl, DigComp)

Type: **String_8** **Moving PV Text Descriptor—**Defines the state descriptor that is displayed when the Digital Composite or Device Control point is changing states (moving from one state to another), or is in-between states. This descriptor, defined on the HPM box point, is displayed for all digital composite or device control points in this HPM box if PVTXTOP, defined on the Digital Composite or Device Control point, = OFF. This parameter contains the text for a configured moving PV on a per point basis if the PVTXTOPT is ON.
Lock: **PtBld**
Default: **MOVING**
PtRes: **NIM**

Range: The permissible character set for the up to eight character descriptor is as follows:
Alphabetics A-Z (uppercase only)
Numerics 0-9,
Underscore (_)

MPCFWREV (HPM Box)

Type: **String_2** **HPMM Master Processor Card Firmware Revision—**
Lock: **View**
Default: **Blank**
PtRes: **HPM**
Range:

MPCHWREV (HPM Box)

Type: **String_2** **HPMM Master Processor Card Hardware Revision—**
Lock: **View**
Default: **Blank**
PtRes: **HPM**
Range:

MSGPEND (ProcMod)

Type: **Logical**
Lock: **View**
Default: **None**
PtRes: **HPM**
Range: **N/A**

Sequence Message Pending—Indicates that a confirmable sequence message requiring confirmation has been issued to the operator.

MSGTXT(0)–(15) (NIM)

Type: **String_8 in an Array (0..15)**
Lock: **PtBld**
Default: **Blank**
PtRes: **NIM**
Range: **0** to **15**

Status Message Text—Indicates the text for the self-defined enumeration of STSMSG. MSGTXT(0) is always NONE, and cannot be configured. Refer to "Status Messages" in the Control Functions and Algorithms Manual for more information.

MXRMPDEV (RampSoak)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **≥ 0.0, NaN**

Maximum Ramp Deviation Value—If the PV falls behind the SP during a ramp segment by more than the value of MXRMPDEV, the ramping action is stopped until the PV reaches the SP.

MXSOKDEV (RampSoak)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **≥ 0.0, NaN**

Maximum Soak Deviation Value—If the PV falls behind the SP during a soak segment by more than the value of MXSOKDEV, the soak timer is stopped until the PV reaches SP.

-N-

N (Calcultr)

Type: **Integer**
Lock: **PtBld**
Default: **1**
PtRes: **HPM**
Range: **1** to **6**

Number of Inputs—Defines the number of inputs to this algorithm.

N (HiLoAvg, Summer)

Type: **Integer**
Lock: **PtBld**
Default: **2**
PtRes: **HPM**
Range: **2** to **6 inputs**

Number of Inputs—Defines the number of inputs to this algorithm.

NAME

Type: **String_16**
Lock: **PtBld**
Default: **N/A**
PtRes: **NIM**

Point Name—Identifies this point to the system and on displays, reports, and logs. Figure N-1 shows examples of the Group and Detail Displays on which the point name appears.

Digital Input, Digital Output, Analog Output, Flag, and Numeric-type data points do not have to be configured by using the point builder (DEB). All other types of data points have to be configured by using the DEB and require that a point name be specified during the point build process.

Range: Point name can be up to 16 characters, and the permissible character set is as follows:
Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric point name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.

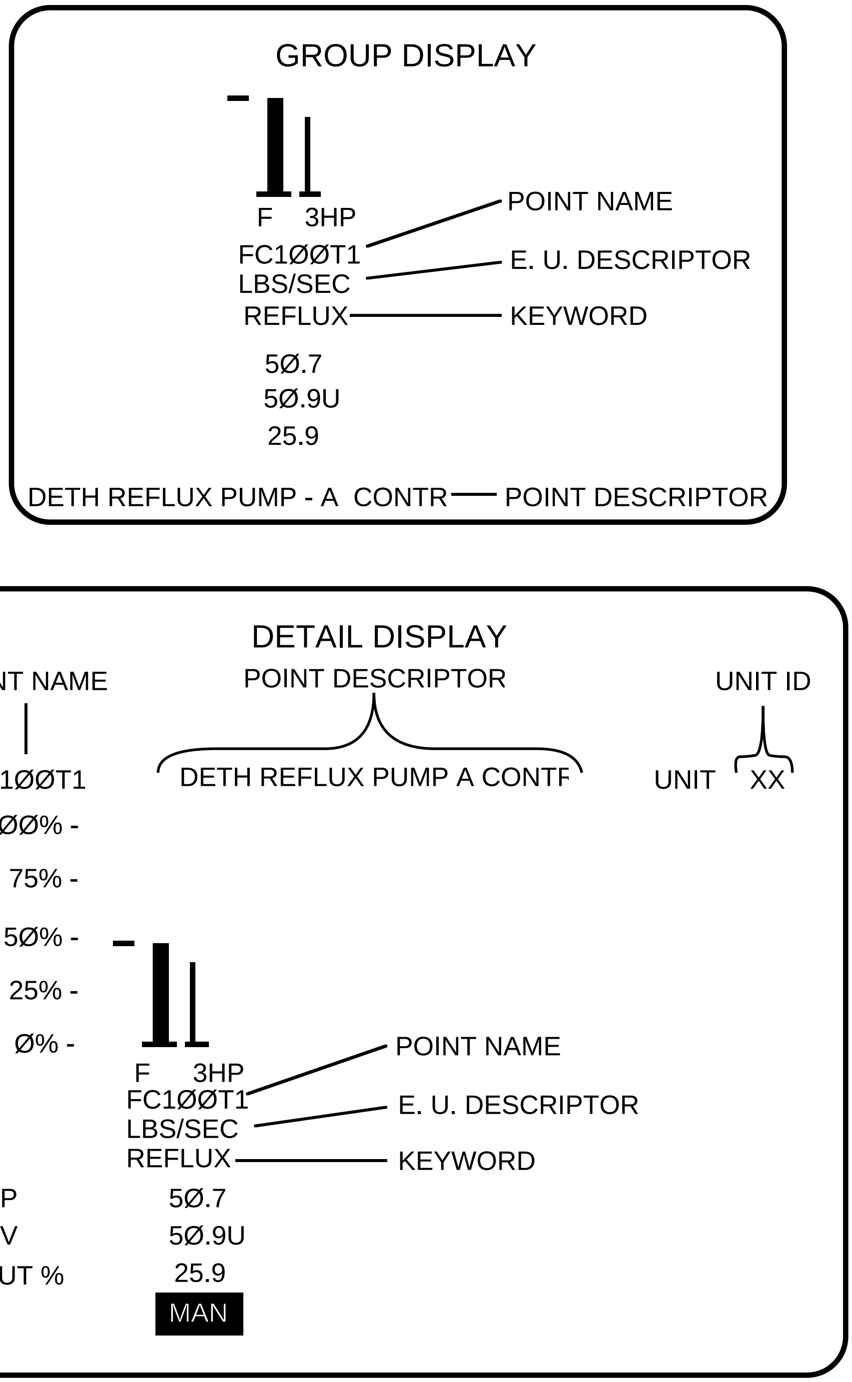


Figure N-1 — Locations of Terms on Group and Detail Displays

1850

NARRSLOT (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **500**

Number of Array Slots in an HPM

NCTLSLOT (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **250**

Number of Regulatory Control Slots in an HPM—Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of HPM processing capacity.

NDCSLOT (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **999**

Number of Digital Composite Slots in an HPM—Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of HPM processing capacity.

NDEVSLOT (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **400**

Number of Device Control Points Configured—The number of Device Control points in an HPM Box point. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of HPM processing capacity.

NEIPRQU (NIM PSDP)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **NIM**
Range: **N/A**

Number of Event Initiated Processing Requests—The number of Event Initiated Processing requests sent in the last 15 seconds.

NEVTAVG (HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Average number of Events per Second—Average number of events generated by the HPM per second.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

NEVTMAX (HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Maximum number of Events per Second—Maximum number of events generated by the HPM per second.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

NFASTCTL (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **100, cannot exceed NCTLSLOT**

Number of Fast Regulatory Control Slots—Fast slots are processed four times per second.

NFASTDC (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **999, cannot exceed NDCSLOT**

Number of Fast Digital Composite Slots—Fast slots are processed four times per second.

NFASTDEV (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **100, cannot exceed NDEVSLOT**

Number of Fast Device Control Points Configured—The number of fast device control points in an HPM Box data point. Fast slots are processed four times per second.

NFASTLOG (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **100, cannot exceed NLOGSLOT**

Number of Fast Logic Slots—Fast slots are processed four times per second.

NFASTPV (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **100, cannot exceed NPVSLOT**

Number of Fast Regulatory PV slots—Fast slots are processed four times per second.

NFLAG (HPM Box)

Type: **Integer**
Lock: **View**
Default: **16, 384**
PtRes: **HPM**
Range: **16,384**

Number of Flags in HPM Box Data Point—The HPM always provides 16, 384 box flag variables.

NFLAG (Array)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **512** (When EXTDATA = IO_FL)
0 to **1023** (When EXTDATA ≠ IO_FL)

Number of Flags in Array Point FL Array—Defines the number of mapped flags from either the HPM box (EXTDATA≠ IO_FL) or a serial interface IOP-connected device (EXTDATA=IO_FL).

NI0–2 (DevCtl, DigComp)

Type: **Logical**
Lock: **Engr**
Default: **On**
PtRes: **HPM**
Range: **On** (Interlock inactive)
Off (Interlock active)

Inverted Interlocks Value—The negative value of the corresponding interlock.

Helpful Hint:	This parameter can be changed by the engineer only if the point is inactive or if the HPM is idle.

NIMDAY (NIM)

Type: **Integer** **Day in Which the NIM Personality was Created**
Lock: **View**
Default: **1**
PtRes: **NIM**
Range: **1 to 31**

Helpful Hint: This parameter is accessed using $NTWRKuu.NIMDAY (where uu = UCN Network number).

NIMMONTH (NIM)

Type: **Integer** **Month in Which the NIM Personality was Created**
Lock: **View**
Default: **1**
PtRes: **NIM**
Range: **1 to 12**

Helpful Hint: This parameter is accessed using $NTWRKuu.NIMMONTH (where uu = UCN Network number).

NIMREV (NIM)

Type: **Integer** **Revision Number of the NIM Personality**
Lock: **View**
Default: **0**
PtRes: **NIM**
Range: **N/A**

Helpful Hint: This parameter is accessed using $NTWRKuu.NIMREV (where uu = UCN Network number).

NIMVERS (NIM)

Type: **Integer** **Version Number of the NIM Personality**
Lock: **View**
Default: **0**
PtRes: **NIM**
Range: **N/A**

Helpful Hint: This parameter is accessed using $NTWRKuu.NIMVERS (where uu = UCN Network number).

NIMYEAR (NIM)

Type: **Integer** **Year in Which the NIM Personality was Created**
Lock: **View**
Default: **0**
PtRes: **NIM**
Range: **0 - 99**

Helpful Hint: This parameter is accessed using $NTWRKuu.NIMYEAR (where uu = UCN Network number).

NLFM

Type: **Integer**
Lock: **Supr**
Default: **1**
PtRes: **HPM**
Range: **0** or **1**

Nonlinearity Form—Defines the form of the nonlinear gain.

NLGAIN (Pid)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: **0.0** to **240.0**

Nonlinear Gain—Defines the value of the nonlinear gain factor KNL.

NLOC (VdtLdLag)

Type: **Integer**
Lock: **Eng**
Default: **30**
PtRes: **HPM**
Range: **2** to **30**

Number of Locations in Delay Table

NLOGSLOT (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **400**

Number of Logic Slots in the HPM—Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of HPM processing capacity.

NMIN (HiLoAvg)

Type: **Integer**
Lock: **Supr**
Default: **1**
PtRes: **HPM**
Range: **1** to **N** (N is the number of inputs selected by N parameter)

Minimum Number of Good Inputs—Defines the minimum number of valid inputs (PV status is good or uncertain) to this algorithm.

NMODATTR (RegCtl)

Type:	**E:MODATTR**	**Normal Mode Attribute—**Defines whether an operator or a program can change certain parameters such as the mode, SP, or OP of a data point when the point is in the normal mode.
Lock:	**Engr**	
Default:	**None**	
PtRes:	**HPM**	

Range: 0-**Operator** (MODATTR can be set equal to Operator)
1-**Program** (MODATTR can be set equal to Program)
2-**Normal**
3-**None** (MODATTR is not affected by this parameter)

Helpful Hint:	If NMODATTR = Operator or Program and the "normal mode" button on the Operator's keyboard is pressed, MODATTR = NMODATTR. If NMODATTR is to be changed, the engineer must change it.

NMODE (AnalgOut)

Type:	**E:MODE**	**Normal Mode—**Allows user to define the normal mode for this data point.
Lock:	**Engr/PB**	
Default:	**None**	
PtRes:	**HPM**	

Range: 0-**None** (No configured "normal" operating mode)
1-**Man** (Manual is configured "normal" mode)
2-**Cas** (Cascade is configured "normal" mode)

Helpful Hint:	NMODE configuration for the Cas option requires RCASOPT = Ddc.

NMODE (DevCtl, DigComp)

Type:	**E:MODE**	**Normal Mode—**Allows user to define the normal mode for this data point.
Lock:	**View**	
Default:	**Man**	
PtRes:	**HPM**	

Range: 1-**Man** (Manual is the "normal" operating mode)

NMODE (RegCtl)

Type:	**E:MODE**	**Normal Mode—**Allows user to define the normal mode for this data point.
Lock:	**Engr**	
Default:	**None**	
PtRes:	**HPM**	

Range: 0-**None** (No configured "normal" operating mode)
1-**Man** (Manual is the "normal" operating mode)
2-**Cas** (Cascade is the "normal" operating mode)
3-**Auto** (Automatic is the "normal" operating mode)
4-**Bcas** (Backup Cascade is the "normal" operating mode)

Helpful Hint:	Mode. If NMODATTR = None and the "normal mode" button on the Operator's keyboard is pressed, MODE is set to the contents of NMODE.

NMODETRK (HPM Box)

Type: **E:$NMODETR**
Lock: **PtBld**
Default: **Enable**
PtRes: **HPM**
Range: **Enable**
Disable

Normal Mode Tracking Supression —Enable/disable Normal Mode and Normal Mode attribute from tracking mode and mode attribute changes.

NMSGTXT (NIM)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **NIM**
Range: **0** to **15**

Number of Message Text Items—Defines the number of message text items that you can enter. See MSGTXT.

NN(i) (Array)

Type: **Real**
Lock: **Determined by SPLOCK parameter**
Default: **N/A**
PtRes: **HPM**
Range: **1 ≤ i ≤ Array parameter NNUMERIC**

Array Point Numeric Variables—Numerics are mapped from either the HPM box (defined by NNUMERIC and NNSTIX parameters) or from a serial interface IOP-connected device (when EXTDATA=IO_NN, mapping is defined by the IOPNUM, FTANUM, DEVADDR, NNSTIX, and NNUMERIC parameters).

NN(1)–(8) (DevCtl, Logic)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Numerics 1-8—Eight numerics are provided with each device control and logic slot. The numerics can be used as reference values for the comparison logic algorithms, or they can be used as source parameters for the output connections when writing predefined analog constants to other points. The values of the numerics can be changed from the Universal Station, by other device control logic slots, or by user-written programs.

NN(1)–(80) (ProcMod)

Type: **Real**
Lock: **Determined by SPLOCK parameter**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Numeric Variables—Each process module in the HPM has 80 numerics that can be used for implementing batch operations.

NN(1)–(16,384) (HPM Box)

Type: **Real**
Lock: **Oper**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Numeric Value—This is an array of up to 16,384 numeric variables. The upper bound of this array is determined by the NNUMERIC parameter. Numerics NN(1) to NN (2047) are taggable. Numerics NN(1) to NN (4095) are accessible from the LCN by using hardware form [!Box.NN( )]. Numerics 4096 through 16,384 are accessible only through Array points.

NNDESC (Array)

Type: **String_64** **NN Array Descriptor—**Describes NN data for the Array point.
Lock: **PtBld**
Default: **Spaces**
PtRes: **HPM**
Range: **N/A**

NNINSET(1)–(10) (DevCtl)

Type: **Integer in an Array (1..10)** **Numeric 1 - 10—**A set of 10 integers that are used by the primary input gate IN_SET algorithm.
Lock: **Supr**
Default: **0**
PtRes: **HPM**
Range: **0** to **32767**

NNSTIX (Array)

Type: **Real** **Numeric Array Start Index—**Defines the start index in Box NN variables, or a serial interface-connected device.
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **99,999** (When EXTDATA=IO_NN, 0 can be valid device index)
0 to **Box parameter NNUMERIC** (When EXTDATA≠IO_NN, 0 indicates no numerics are configured)

NNUMERIC (HPM Box)

Type: **Integer** **Number of Numerics in HPM Box Data Point—**The number of box numerics is determined in intervals of 16 numerics.
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **16,384**

NNUMERIC (Array)

Type: **Integer** **Number of Numerics in Array Point NN Array—**Defines the number of numerics mapped from either the HPM box (EXTDATA≠ IO_NN), or a serial interface IOP-connected device (EXTDATA=IO_NN). For external data, the valid range depeneds on how numeric data is organized in the device.
Lock: **PtBld**
Default: **0**
PtRes: **HPM**

Range: **0 - 16 (Floats), 0 - 32 (Integers), 0 - 64 (Byte Integers)** When EXTDATA = IO_NN
0 to **240** When EXTDATA ≠ IO_NN

NOCINPTS (RegCtl)

Type: **Integer**
Lock: **PtBld**
Default: **Based on CTLALGID, CTLEQN, M**
PtRes: **HPM**
Range: **0** to **4**

Number of Control Input Connections—Defines the number of control input connections for this algorithm.

NOCOPTS (RegCtl)

Type: **Integer**
Lock: **PtBld**
Default: **1**
PtRes: **HPM**
Range: **0** to **4**

Number of Control Output Connections—Defines the number of control output connections from this RegCtl point.

Helpful Hint: Control output engineering ranges (CVEULO, CVEUHI) must be entered for CTLALGID = PidErfb and Rampsoak, and must be entered for CTLALGID = Pid, PidFf, and RatioCtl when NOCOPTS = 0. For CTLALGID = Pid, PidFf, and RatioCtl, with NOCOPTS > 0, the CV ranges are obtained from a secondary output connection.

NODEASSN (HPM Box)

Type: **E:$NODEASN**
Lock: **PtBld**
Default: **ThisNIM**
PtRes: **NIM**
Range: **ThisNIM** (The NIM on this logical UCN is responsible for parameter fetch/stores, alarming, AM control strategy and checkpointing for this UCN node.)

RemotNIM (A NIM on another logical UCN is responsible for configuring, checkpointing, and restoring the database through this UCN node.)

Node Assignment—Defines whether the NIM on this logical UCN or a NIM on another logical UCN is the primary NIM for this UCN node.

NODECMD (HPM Box)

Type: **E: $PMCMD** **Node Command—**Defines the command issued to the HPM.
Lock: **Eng**
Default: **None**
PtRes: **HPM**

Range: 0-**None** (No request made to the HPM)
1-**Run** (HPM requested to go to the Run state)
2-**Idle** (HPM requested to go to an Idle state)
3-**Warmstrt** (Warm Start requested)
4-**Coldstrt** (Cold Start requested)
5-**Pause** (HPM requested to go to the Simul_Pause state)
6-**Resume** (HPM requested to go to the Simul_Run state)

NODECONF (HPM Box)

Type: **E:$PMCONF**
Lock: **View**
Default: **Manual**
PtRes: **HPM**
Range: **Manual**
Auto

Node Configuration for the HPM—Currently not used. This parameter must always be set to manual.

NODENUM (NIM)

Type: **Integer** **Node Number—**Defines the address of the NIM on the UCN.
Lock: **PtBld**
Default: **N/A**
PtRes: **NIM**
Range: **1 to 64**

NOTE

The node number assigned to the NIM should be the lowest node number on the UCN (see Timesync).

NODENUM (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **N/A**
PtRes: **NIM**
Range: **1 to 64**

Node Number—Defines the address of the HPMs on the UCN. NODENUM assigned for any HPM must be odd whether PKGOPT equals Option 1 (nonredundant) or Option 2 (redundant). Because of this restriction and because the NIM takes up one odd address and the next even address, the maximum number of HPM's that can be on the UCN is 31. The primary HPMM is assigned an odd address, the associated secondary (redundant) HPMM is assigned the next (even) address.

NODEOPER

Type: **E:$PRIMSEC** **Node Operating Mode**
Lock: **View**
Default:
PtRes: **NIM**
Range: **Primary** (HPM/NIM is the acting primary node)
Secndry (HPM/NIM is the acting secondary)

NODESC (Logic)

Type: **Integer**
Lock: **PTBLD**
Default: **0**
PtRes: **NIM**
Range: **0** to **12**

Number of Generic Descriptors—Defines the number of user-defined generic descriptors that are to be used on this logic slot. For each descriptor, the parameter in the logic slot to which the generic descriptor is attached is defined by the PRMDESC(n) parameter, and the corresponding descriptors are defined by the GENDESC(n) parameters. This allows the user to customize the descriptors used for displaying the logic slot on the Universal Station displays.

NODESTAT (HPM Box)

Type: **E:$NODESTA** **HPM Node Status**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **OffNet** (HPM is not running on UCN)
OK (HPM is configured and running)

NODESTAT (NIM)

Type: **E:$NODESTA** **NIM's Node Status**
Lock: **View**
Default: **N/A**
PtRes: **NIM**
Range: **OffNet** (NIM is not running on UCN)
OK (NIM is configured and running)

NODESTS (NIM)

Type: **E:$DSPSTAT** **NIM Node Summary Status—**Indicates the current overall status of the NIM.
Lock: **View**
Default: **N/A**
PtRes: **NIM**
Range: **OffNet** (NIM cannot communicate with HPMM)
OK (NIM is performing normally)

NODESTS (HPM Box)

Type: **E:$DSPSTAT** **HPM Node Summary Status—**Indicates the current overall status of the HPM on the UCN.
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: 0-**OK** (HPM is performing normally)
1-**IOIDL** (At least one IOP has entered the idle state)
2-**IDLE** (HPMM has entered the idle state)
3-**PF_IOIDL** (Partial failure in one or more idle IOPs)
4-**PF_IDLE** (Partial failure in HPMM that is in idle state)
5-**PartFail** (Partial failure in HPMM that is in run state)
6-**Fail** (HPMM has sent a "failed" message to the NIM)
7-**Alive** (No event reports or point processing)
8-**Loading** (HPMM's personality or database is loading)
9-**PowerOn** (Transitional state when power applied to HPMM)
10-**OffNet** (NIM cannot communicate with HPMM)
12-**NotConf** (HPMM cannot be found on the UCN)
13-**ConfgMis** (IOP mismatch in NIM Box point)
19-**Standby**
20-**S_OK** (OK in I/O simulation mode)
21-**S_IOIDL** (IOIDL in I/O simulation mode)
22-**S_IDLE** (IDLE in I/O simulation mode)
23-**S_PFIOIL** (PF_IOIDL in I/O simulation mode)
24-**S_PFIDLE** (PF_IDLE in I/O simulation mode)
25-**S_PtFail** (PartFail in I/O simulation mode)
26-**S_Pause** (HPM is in the simulation pause state)
27-**StandbySF**
28-**Upgrade**
29-**UpgradeSF**

Helpful Hint: Loading the HPMM's operating personality requires NODESTS = Alive.

NODETYP (NIM)

Type: **E:$UCNNDTY** **UCN Node Type—**Defines the node type of this UCN node.
Lock: **PtBld**
Default: **NIM**
PtRes: **NIM**
Range: **NIM** (Network Interface Module)

NODETYP (HPM Box)

Type: **E:$UCNNDTY** **UCN Node Type—**Defines the node type of this UCN node.
Lock: **PtBld**
Default: **HPM**
PtRes: **NIM**
Range: **HPM** (High-Performance Process Manager)

NODETYP (HPM Points)

Type: **E:$UCNNDTY** **UCN Node Type—**Defines which node type supports this point.
Lock: **PtBld**
Default: **HPM**
PtRes: **NIM**
Range: **HPM** (High-Performance Process Manager)

NODFSTAT (HPM Box)

Type: **E:$NODFSTA** **Node's Functional Status—**Defines the status of the UCN node.
Lock: **Supr**
Default: **Basic**
PtRes: **NIM**
Range: **Full** (All LCN devices can read/write from/to this node)
Basic (AM and CM cannot write to this node)

NODINPTS (DevCtl, DigComp)

Type: **Integer**
Lock: **PtBld**
Default: **1**
PtRes: **HPM**
Range: **0** (No inputs)
1 (One input)
2 (Two inputs)

Number of Digital Inputs—Defines the number of digital input connections to this data point.

NODOPTS (DevCtl, DigComp)

Type: **Integer**
Lock: **PtBld**
Default: **1**
PtRes: **HPM**
Range: **0** (No outputs)
1 (One output)
2 (Two outputs)
3 (Three outputs)

Number of Digital Outputs—Defines the number of digital output connections from this data point.

NOGINPTS (RegPV, RegCtl)

Type:
Lock:
Default:
PtRes: **HPM**
Range: **0 - 4**

Number of General Input Connections—

Helpful Hint:	NOGINPTS + NOGOPTS cannot exceed four.

NOGOPTS (RegPV, RegCtl)

Type:
Lock:
Default:
PtRes: **HPM**
Range: **0 - 4**

Number of General Ouptut Connections—

Helpful Hint:	NOGINPTS + NOGOPTS cannot exceed four.

NOLINPTS (DevCtl, Logic)

Type: **Integer**
Lock: **View (Logic), PtBld (DevCtl)**
Default: **12 (Logic)**
0 (DevCtl)
PtRes: **HPM**
Range: **0** to **12**

Number of Logic Inputs—Indicates the number of logic inputs to this logic or Device Control slot.

NOLOGBLK (Logic)

Type: **Integer**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **0** to **24**

Number of Logic Blocks—Indicates the number of logic blocks that have been configured for a particular logic slot.

NOLOPTS (DevCtl, Logic)

Type: **Integer**
Lock: **View (Logic) PtBld (DevCtl)**
Default: **N/A**
PtRes: **HPM**
Range: **0** to **12** (Logic), **0** to **2** (DevCtl)

Number of Logic Output Connections—Indicates the number of output connections from this logic slot.

NOOVRRUN (ProcMOD)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range:

Number of Overruns—Indicates the number of times the point has overrun its CNFPU allocation since the last reset.

Helpful Hint:	NOOVRRUN is reset along with AVGPU and MAXPU

NOOVRRUN (ProcMod)

Type: Integer
Lock: View
Default: **0**
PtRes: **HPM**
Range: **0 - 4**

Number of Overruns—Specifies the number of times the point has overrun its CNFPU allocation since the last reset

NOPGATE (DevCtl)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **4**

Number of Primary Gates—Indicates the number of primary gates configured for a particular Device Control slot.

Helpful Hint:	All configured primary gates must have at least one input.

NOPINPTS (RegPV)

Type: **Integer**
Lock: **View**
Default: **Based on PVALGID, PVEQN, N**
PtRes: **HPM**
Range: **0** to **6**

Number of PV Input Connections—Defines the number of PV input connections to this algorithm.

NOPTS(0 - 64)

Type: **Integer** **Number of Points Per Cycle—**Defines ...
Lock: **View**
Default: **0** Index = 0 used for total count
PtRes: **HPM** Index = 1 - 64 used for per cycle count
Range:

Helpful Hint: The total count may not be equal to the sum of all cycles because most points are in more than one cycle.

NORMCYCL

Type: **Integer** **Normal Execution Cycle—**Specifies the normal execution cycle.
Lock: **PtBld**
Default:
PtRes: **HPM**
Range: **1 - 64** for points with PERIOD = 4 seconds
1 - 32 for points with PERIOD = 2 seconds
1 - 16 for points with PERIOD = 1 seconds
1 - 8 for points with PERIOD = 0.5 seconds
1 - 4 for points with PERIOD = 0.25 seconds
1 - 2 for points with PERIOD = 0.125 seconds
1 for points with PERIOD = 0.625 seconds

NORQUAVG (NIM, HPM Box)

Type: **Real** **Average number of Nodes to which UCN Requests are made—**Indicates the average number of UCN nodes per second that this node is requesting communications with.
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Helpful Hint: This statistic can be viewed on the Toolkit Displays.

NORQUMAX (NIM, HPM Box)

Type: **Real** **Maximum number of Nodes to which UCN Requests are made—**Indicates the maximum number of UCN nodes per second that this node is requesting communications with.
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Helpful Hint: This statistic can be viewed on the Toolkit Displays.

NORSPAVG (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Average number of Nodes to which UCN Responses are made—Indicates the average number of UCN nodes per second that this node is responding to.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

NORSPMAX (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Maximum number of Nodes to which UCN Responses are made—Indicates the maximum number of UCN nodes per second that this node is responding to.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

NORSSEQ (RampSoak)

Type: **Integer**
Lock: **Eng/PB**
Default: **2**
PtRes: **HPM**
Range: **2** to **12**

Number of Ramp/Soak Pairs in the Sequence

NOSGATE (DevCtl)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: 0 to 2

Number of Secondary Gates—Indicates the number of secondary gates configured for a particular Device Control slot.

Helpful Hint:	All configured secondary gates must have at least one input.

NOSIOVRD (DevCtl, DigComp)

Type: **Real**
Lock: **View**
Default: **0.0**
PtRes: **HPM**
Range: **0** (No limit)

Number of Safety Interlock Overrides—The number of safety interlock overrides that have accumulated since the most recent reset of maintenance statistics.

NOSTATES (DevCtl, DigComp)

Type: **Integer**
Lock: **PtBld**
Default: **2**
PtRes: **HPM**

Number of Digital States—Defines the number of states in this point.

- State 1 is the first active state
- State 0 is the inactive (middle) state
- State 2 is the second active state

Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: **2** (Two states can be configured: STATE 0 and STATE 1)
3 (Three states can be configured: STATE 0, STATE 1, and STATE 2)

NOTRAAVG (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Average number of Nodes to which UCN Transactions are made. This value indicates the average number of UCN nodes (per second) that this node is communicating with (both requests and responses).

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

NOTRAMAX (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Maximum number of Nodes to which UCN Transactions are made. This value indicates the maximum number of UCN nodes (per second) that this node is communicating with (both requests and responses).

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

NOTRANS0–2 (DevCtl, DigComp)

Type: **Real**
Lock: **View**
Default: **0.0**
PtRes: **HPM**

Accumulated Transitions—The number of transitions to each state of the OPFINAL parameter since the most recent reset of maintenance statistics. The MAXTRAN parameter does not limit the number of transactions unless the user writes a program to read MAXTRAN, comparing it to NOTRANS, and thereby causing it to stop.

Range: **0** (No limit)

NPARAVG (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Average number of UCN Parameter Accesses per Second—Average number of UCN parameter accesses per second between this node and all other nodes, including both requests and responses.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

NPARMAX (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Maximum number of UCN Parameter Accesses per Second—Maximum number of UCN parameter accesses per second between this node and all other nodes, including both requests and responses.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

NPMSLOT (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **250**

Number of Process Module Slots—Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of HPM processing capacity.

NPRQUAVG(0-64) (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Average number of UCN Parameter Requests—Average number of UCN parameter requests per second issued from this node to node n.

NOTE

The node address (n) is an odd number (1, 3, 5,...63). Using an even number results in a **Parameter_Invalid** error. Using n = 0 returns average total number of parameter requests to all other nodes.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

NPRQUMAX(0-64) (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Maximum number of UCN Parameter Requests—Maximum number of UCN parameter requests per second issued from this node to node n.

NOTE

The node address (n) is an odd number (1, 3, 5,...63). Using an even number results in a **Parameter_Invalid** error. Using n = 0 returns maximum total number of parameter requests to all other nodes.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

NPRSPAVG(0-64) (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Average number of UCN Parameter Responses—Average number of UCN parameter responses per second issued from this node to node n.

> **NOTE**
>
> The node address (n) is an odd number (1, 3, 5,...63). Using an even number results in a **Parameter_Invalid** error. Using n = 0 returns average total number of parameter responses to all other nodes.

Helpful Hint: This statistic can be viewed on the Toolkit Displays.

NPRSPMAX(0-64) (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Maximum number of UCN Parameter Responses—Maximum number of UCN parameter responses per second issued from this node to node n.

> **NOTE**
>
> The node address (n) is an odd number (1, 3, 5,...63). Using an even number results in a **Parameter_Invalid** error. Using n = 0 returns maximum total number of parameter responses to all other nodes.

Helpful Hint: This statistic can be viewed on the Toolkit Displays.

NPVSLOT (HPM Box)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **125**

Number of Regulatory PV Slots—Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of HPM processing capacity**.**

NRMATRFL (DigComp, DevCtl, RegCtl)

Type: **Logical** **Normal Mode Attribute Flag—**indicates if this point is in the configured Normal Mode attribute.
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: ON - (point is in the configrured Normal mode attribute)
Off **-** (point is not in the configured Normal Mode attribute or Normal Mode attribute is not configured)

Helpful Hint:	If Normal mode attribute is not configured then the value returns to OFF.

NRMMODFL (RegCtl)

Type: **Logical** **Normal Mode Flag—**indicates if the mode for this point is normal mode.
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: On - (Point is in configured normal mode)
Off - (Point is not in the configured normal mode or normal mode is not configured)

NSCANITM (HPM Box)

Type: **Integer** **Number of Scan Items in HPM Scan Table.**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **0 - 50**

NSI0 (DevCtl, DigComp)

Type: **Logical** — **Inverted Interlocks Value—**The negative value of the corresponding interlock.
Lock: **Engr**
Default: **On**
PtRes: **HPM**
Range: **On** (Safety interlock inactive)
Off (Safety interlock active)

Helpful Hint:	This parameter can be changed by the engineer only if the point is inactive, or if the HPM is idle.

NSTRING (HPM Box)

Type: **Integer** — **Number of Strings in HPM Box Data Point**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **16,384**

NSTRING (Array)

Type: **Integer** — **Number of Strings in Array Point String Array—**Defines the number of strings (length specified by the STRLEN parameter) mapped to the Array point from either the HPM box (EXTDATA≠ IO_STR), or a serial interface IOP-connected device (EXTDATA=IO_STR).
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **8** (When EXTDATA=IO_STR)
0 to **240** (When EXTDATA≠ IO_STR)

Helpful Hint:	When EXTDATA≠ IO_STR, the range for this parameter applies regardless of the value of the STRLEN parameter (up to 240 strings, either 8, 16, 32, or 64 characters in length can be mapped to the Array point from the HPM box). When EXTDATA=IO_STR, only 64 characters of string data are available (i.e., one 64-character string, two 32-character strings, four 16-character strings, or eight 8-character strings).

NTIME (HPM Box)

Type: **Integer** — **Number of Times in HPM Box Data Point**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **4,096**

NTIME (Array)

Type: **Integer**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **240**

Number of Times in Array Point Time Array—Defines the number of Times the Array point has mapped from the HPM box.

NTIMER (HPM Box)

Type: **Integer**
Lock: **View**
Default: **64**
PtRes: **HPM**
Range: **64**

Number of Timer Points in the HPM Box Data Point

NTRAAVG (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Average number of UCN Transactions —Average number of UCN transactions (requests and responses) per second between this node and all other nodes.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

NTRAMAX (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Maximum number of UCN Transactions —Maximum number of UCN transactions (requests and responses) per second between this node and all other nodes.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

NTRQUAVG(0-64) (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Average number of UCN Transaction Requests—Average number of UCN transaction requests per second issued from this node to node n.

NOTE

The node address (n) is typically an odd number (1, 3, 5,...63). Using an even number results in a **Parameter_Invalid** error. Using n = 0 returns average total number of transaction requests to all other nodes.

Helpful Hint: This statistic can be viewed on the Toolkit Displays.

NTRQUMAX(0-64) (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Maximum number of UCN Transaction Requests—Maximum number of UCN transaction requests per second issued from this node to node n.

NOTE

The node address (n) is an odd number (1, 3, 5,...63). Using an even number results in a **Parameter_Invalid** error. Using n = 0 returns maximum total number of transaction requests to all other nodes.

Helpful Hint: This statistic can be viewed on the Toolkit Displays.

NTRSPAVG(0-64) (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Average number of UCN Transaction Responses—Average number of UCN transaction responses per second issued from this node to node n.

NOTE

The node address (n) is an odd number (1, 3, 5,...63). Using an even number results in a **Parameter_Invalid** error. Using n = 0 returns average total number of transaction responses to all other nodes.

Helpful Hint: This statistic can be viewed on the Toolkit Displays.

NTRSPMAX(0-64) (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Maximum number of UCN Transaction Responses—Maximum number of UCN transaction responses per second issued from this node to node n.

NOTE

The node address (n) is an odd number (1, 3, 5,...63). Using an even number results in a **Parameter_Invalid** error. Using n = 0 returns maximum total number of transaction responses to all other nodes.

Helpful Hint: This statistic can be viewed on the Toolkit Displays.

NTWKNUM

Type: **Integer**
Lock: **PtBld**
Default: **N/A`**
PtRes: **NIM**
Range: **1** to **20**

Network Number—Defines on which UCN the NIM and HPMs reside.

NXTPINAM

Type: **String_8**
Lock: **Eng**
Default: **N/A**
PtRes: **IOP**
Range: **?**

Next Personality Image File—Defines the personality Image file that will be loaded on the next personallity load request for this IOP.

NXTSOAKV (RampSoak)

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: $\geq$ **0.0**

Next Soak Value

-O-

OFFNRMFL

Type:	**Logical**	**Off-Normal Alarm Flag—**Indicates whether an off-normal alarm has been detected at this data point.
Lock:	**View**	
Default:	**Off**	
PtRes:	**HPM**	
Range:	**Off** (No alarm) **On** (Current PV state is not the same as the configured PVNORMAL state.)	

OFFNRMPR

Type:	**E:ALPRIOR**	**Offnormal Alarm Priority—**Indicates priority of the off normal or change of state alarms.
Lock:	**ENGR**	
Default:	**Low**	
PtRes:	**NIM**	
Range:	**Emergncy** (Alarm is historized, annunciated, and reported to all alarm summary displays) **High** (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display) **Low** (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated) **JnlPrint** (Alarm is historized and reported to the printer but not annunciated) **Printer** (Alarm is reported to the printer but not historized and not annunciated) **Journal** (Alarm is historized but not reported to Universal Stations and not annunciated) **NoAction** (Alarm is not reported to the system and not annunciated)	

OFFPULSE (DigOut)

Type:	**Real**	**Off Pulse Command—**Command that sets output SO to Off for the specified number of seconds. At the end of the pulse time, SO is set to On. If 0.0 is entered for OFFPULSE, SO is immediately set to On.
Lock:	**Oper**	
Default:	**N/A**	
PtRes:	**HPM**	
Range:	**0.0** to **60.0 seconds**	

Helpful Hint:	OFFPULSE can be written to by only those entities that possess the HPMM Cont_Ctl (continuous control) access level. These are Digital Composite points, Logic points, and Regulatory Control Position Proportional points.

OLDAV (DigIn)

Type: **Integer**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **≥ 0**

Old Accumulated Value—The value of parameter AV (accumulated value) just before the accumulator was reset. This parameter makes the previous accumulated value available for those functions that need it.

OLDAV (Totalizr)

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **≥ 0.0**

Old Accumulated Value—The value of parameter PVCALC (calculated PV) just before it is reset. This parameter makes the previous total available to those functions that need it.

ONPULSE (DigOut)

Type: **Real**
Lock: **Oper**
Default: **N/A**
PtRes: **HPM**
Range: **0.0** to **60.0 seconds**

On Pulse Command—Command that sets output SO to On for the specified number of seconds. At the end of the pulse time, SO is set to Off. If 0.0 is entered for ONPULSE, SO is immediately set to Off.

Helpful Hint:	ONPULSE can be written to by only those entities that possess the HPMM Cont_Ctl (continuous control) access level. These are Digital Composite points, Logic points, and Regulatory Control Position Proportional points.

OP (AnalgOut)

Type: **Real**
Lock: **Oper**
Default: **-6.9% of full scale**
PtRes: **HPM**
Range: **-6.9** to **106.9%**

Output in Percent—Defines the output value from this point in percent.

Helpful Hint:	To manually change the output value requires MODE = Man and REDTAG = Off.

OP (DevCtl, DigComp)

Type: **E:SD-ENM:STATETXT**
Lock: **Oper**
Default: **STATETXT(0)**
PtRes: **HPM**
Range: **STATETXT(0)** Descriptor
STATETXT(1) Descriptor
STATETXT(2) Descriptor (internally set to $NULL for two-state devices)
STATETXT(3) None (Not configurable)

Digital State Output—Indicates the last commanded output state. See also OPFINAL. For Status Outputs, use SO.

Helpful Hint:	OP indicates text for the last commanded output state (i.e., On, Run, etc.). Output state change requires MODE = Man, SHUTDOWN = Off, and REDTAG = Off.

OP (DigOut)

Type: **Real**
Lock: **Prog**
Default: **0.0%**
PtRes: **HPM**
Range: **0.0** to **100.0%**

Pulsed Digital Output—OP is the percent on-time for the pulsed output. It can be written to by only the controlling slot in the HPMM, such as from the PosProp RegCtl algorithm. Output change requires DOTYPE = Pwm. OP (DigOut) has the same access-level requirement as OFFPULSE and ONPULSE; the writing entity must have an HPMM access level of Cont_Ctrl (continuous control). For Status Outputs, use SO.

For direct action, pulse-on time is calculated as follows:

$$\text{Pulse On-Time} = \frac{\text{OP\% * PERIOD}}{100}$$

For reverse action:

$$\text{Pulse On-Time} = \frac{\text{100\% - OP\% * PERIOD}}{100}$$

Refer to PERIOD parameter for length of period.

OP (RegCtl)

Type: **Real**
Lock: **Oper**
Default: **-6.9% of full scale**
PtRes: **HPM**
Range: **-6.9** to **106.9%**

Regulatory Control Output—OP is derived from CV, the variable calculated by the control algorithm. OP is checked for minimum output change, output rate-of-change, and output high and low limits. If any of the limits is exceeded, OP is adjusted or clamped as applicable. OP remains in percent of full scale if it is going to a final control element through an IOP Card. If OP is going to a secondary data point, its value is converted to the engineering units of the receiving data point's setpoint (SP).

Helpful Hint:	OP change requires MODE = Man, SHUTDOWN = Off, and REDTAG = Off. If the OP is manually set above or below the OP limits and the mode is then changed to automatic or cascade, a process bump may occur.

OPALDB (RegCtl)

Type: **Real**
Lock: **EngPB**
Default: **5 Output Units**
PtRes: **HPM**
Range: **0** to **25 Output Units**

Output Alarm Dead Band—The deadband for the Regulatory Control OP alarm. It is used to prevent excessive recurrence of alarms by adjusting the range of the output at which the alarm "returns to normal." Available on Release 510 and later software.

Helpful Hint:	The value of OPALDB must be less than or equal to (OPHITP - OPLOTP)/2.

OPCHAR

Type: **Logical**
Lock: **PtBld**
Default: **Off**
PtRes: **HPM**

Output Characterization Option—Defines whether the output characterization option is to be used for this data point. If this option is to be implemented, the user must supply the values for the input coordinates (OPIN 1–4) and output coordinates (OPOUT 1–4). Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of output characterization.

Range: **On** (Output characterization is to be used)
Off (Output characterization is not to be used)

OPCMD (DevCtl, DigComp)

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Commands the output state to State0)
On (Commands the output state to State1)

Output Command

Helpful Hint:	If state change did not occur, OPCMD has to be set to the current state, and then to the desired state.

OPCMD

Type: **E:$OPCMD**
Lock: **View**
Default: **NA**
PtRes: **HPM**
Range: 0-**Idle** (Output is not being affected by Output Command)
1-**Lower** (Output is being lowered)
2-**Raise** (Output is being raised)

Output Command—Indicates the current output command.

OPEU

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

Output Value in Engineering Units

OPFINAL (AO)

Type: **Real**
Lock: **View**
Default: **-6.9% of full scale**
PtRes: **HPM**
Range: **N/A**

Final Percent Output Sent to Control Element—Output value after direct or reverse control action and output characterization have all been applied. If output has been configured for direct action (OPTDIR), 0.0% represents 4 mA to the control element and 100% represents 20 mA. If configured for reverse action, 0.0% represents 20 mA, and 100% represents 4 mA.

OPFINAL (DevCtl, DigComp)

Type: **E:SD_ENM:STATETXT**
Lock: **Oper**
Default: **Statetxt(0)**
PtRes: **HPM**

Final Output Sent to Control Element—The output value that was last stored. This value can differ from the OP parameter if a sealin has occurred, state change is active, or the Array/SI read-back check evaluates OPFINAL to be NONE. If LOCALMAN = ON, then OP and OPFINAL follow the PV.

Range: **Statetxt(0)** (Self-defining enumeration)
Statetxt(1) (Self-defining enumeration)
Statetxt(2) (Self-defining enumeration)—internally set to $NULL for two-state devices.
Statetxt(3) NONE (not configurable)

OPHAFL (RegCtl)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**

Output High Alarm Flag—Indicates when a Regulatory Control Output High alarm has been detected at this data point. This flag is set when the output value (OP) exceeds OPHITP and is reset when OP is below OPHITP minus the deadband. Available on Release 510 and later software.

Range: **Off** (OP High alarm is off)
On (OP High alarm is on)

Helpful Hint:	Refer to the diagram with OPLAFL.

OPHIFL (RegCtl)

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On (OP value has reached its upper limit)

Output High Limit Flag—Indicates whether the OP value has reached its upper limit specified by OPHILM. If this parameter is set by a program, it will inhibit "raise" commands.

OPHILM

Type: **Real**
Lock: **Supr**
Default: **105.0%**
PtRes: **HPM**
Range: **OPLOLM** to **106.9%,**
NaN

Output High Limit in Percent

Helpful Hint:	Entering NaN disables limit checking by forcing OPHILM to its extreme value (106.9%).

OPHIPR (RegCtl)

Type: **E:Alprior**
Lock: **EngPB**
Default: **Low**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

Output High Alarm Priority—Specifies the priority of the Regulatory Control Output High alarm. Available on Release 510 and later software.

OPHISRC

Type: **Universal Ent.Prm**
Lock: **PtBld**
Default: **Null**
PtRes: **HPM**

Output High Flag Input Source—Defines the input connection that fetches the OPHIFL parameter to determine windup state.

Range: Use Tagname.Parameter for tagged points where Tagname can be up to 16 characters and the permissible character set is as follows:

Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.
An * is used to default to this point's tag name.
Parameter name can be up to eight characters and must be a legitimate parameter name.

Some possible input-connection sources are

a."DigIn slot Tagname.PVFL"
b."Logic slot Tagname.SO(nn)" where nn = 1–24
c."Logic slot Tagname.Fl(nn)" where nn = 1–12
d."ProcMod slot Tagname.Fl(nnn)" where nnn = 1–127
e."Box Flag slot Tagname.PVFL
f."!Box.FL(nnnn)" for a box flag that resides in the same box where nnnn = 1–16, 384
g."$NMhhBxx.FL(nnnn)" for a box flag that resides in a different HPM box on the same UCN; hh is the NIM UCN address, xx is the HPM box number, and nnnn = 1–4095 (Data access limit)

Use the hardware reference address !MTmmSss.Parameter for untagged or tagged points where

MT is the IOP type, such as DI (Digital Input)
mm is the IOP Card number (1–40)
The letter "S" is a constant
ss is the slot number on the IOP Card (refer to SLOTNUM parameter)
Parameter name can be up to eight characters and must be a legitimate parameter name.

OPHITP (RegCtl)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**

Output High Alarm Trip Point—The Regulatory Control Output High alarm is set when the output value (OP) exceeds the high alarm limit specified by OPHITP. The alarm is removed when OP returns to normal. A suitable dead band is provided by OPALDB. See also OPHIPR and the drawing with OPLAFL. Available on Release 510 and later software.

Range: **OPLOTP** to **OPHILM, NaN**

> *Helpful Hint:* The Regulatory Control Output High alarm is only available for points configured as full. The alarm is disabled if OPHITP is not configured.

OPIN0

Type: **Real**
Lock: **View**
Default: **-6.9%**
PtRes: **HPM**
Range: **N/A**

Input Coordinate Number 0 in Percent—Defines the OPIN0 coordinate when output characterization has been selected (OPCHAR is On). This coordinate is fixed at -6.9%.

OPIN1–4

Type: **Real**
Lock: **Supr**
Default: **N/A**
PtRes: **HPM**
Range: **≥ previous coordinate**
≤ next coordinate

Input Coordinate Number 1, 2, 3, or 4 in Percent—Define the OPIN1–OPIN4 coordinates when output characterization has been selected (OPCHAR is On).

OPIN5

Type: **Real**
Lock: **View**
Default: **106.9%**
PtRes: **HPM**
Range: **N/A**

Input Coordinate Number 5 in Percent—Defines the OPIN5 coordinate when output characterization has been selected (OPCHAR is On). This coordinate is fixed at 106.9%.

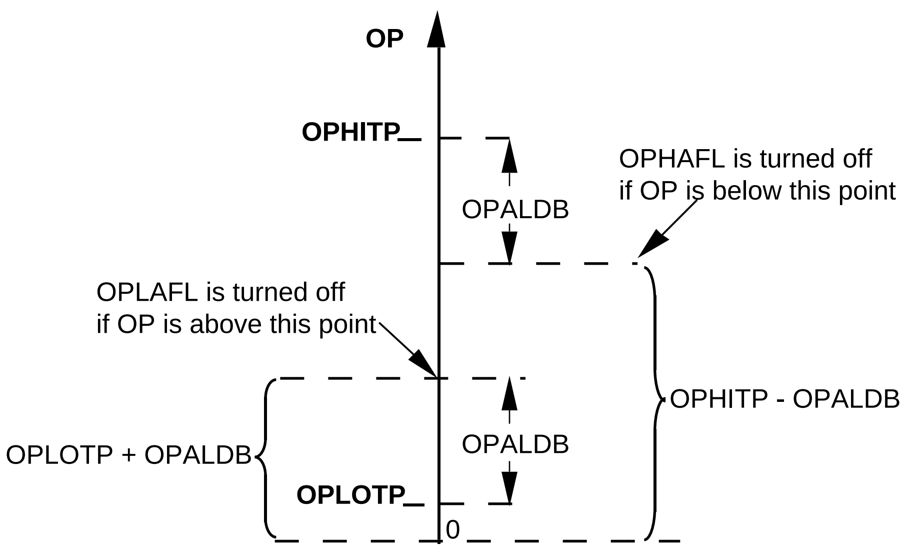
OPLAFL (RegCtl)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**

Output Low Alarm Flag—Indicates if a Regulatory Control Output Low alarm has been detected at this data point. This flag is set when the output value (OP) is less than OPLOTP and is reset when OP is above OPLOTP plus the deadband. Available on Release 510 and later software.

Range: **Off** (OP Low alarm is off).
On (OP Low alarm is on).

The drawing below illustrates the relationship of the output high/low alarm flags, the low alarm trip point OPLOTP, and the deadband OPALDB.



OPLOFL

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off** (OP is above the low limit)
On (OP has reached the low limit)

Output Low Limit Flag—Indicates whether the output value OP has reached the low limit. This parameter must be set by a program or logic point. It will inhibit "raise" commands.

OPLOLM

Type: **Real**
Lock: **Supr**
Default: **-5.0%**
PtRes: **HPM**
Range: **-6.9%** to **OPHILM,**
NaN

Output Low Limit in Percent

Helpful Hint:	Entering NaN disables limit checking by forcing OPLOLM to its extreme value (-6.9%).

OPLOPR (RegCtl)

Type: **E:Alprior**
Lock: **EngPB**
Default: **Low**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

Output Low Alarm Priority—Specifies the priority of the Regulatory Control Output Low alarm. Available on Release 510 and later software.

OPLOSRC

Type: **Universal Ent.Prm**
Lock: **HPM**
Default: **PtBld**
PtRes: **Null**

Output Low Flag Input Source—Indicates which input connection fetches the OPLOFL parameter to determine the windup state.

Range: Use Tagname.Parameter for tagged points where Tagname can be up to 16 characters and the permissible character set is as follows:

Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.
An * is used to default to this point's tag name.
Parameter name can be up to eight characters and must be a legitimate parameter name.

Some possible input-connection sources are

a."DigIn slot Tagname.PVFL"
b."Logic slot Tagname.SO(nn)" where nn = 1–24
c."Logic slot Tagname.Fl(nn)" where nn = 1–12
d."ProcMod slot Tagname.Fl(nnn)" where nnn = 1–127
e."Box Flag slot Tagname.PVFL
f."!Box.FL(nnnn)" for a box flag that resides in the same box where nnnn = 1–4095
g."$NMhhBxx.FL(nnnn)" for a box flag that resides in a different HPM box on the same UCN; hh is the NIM UCN address, xx is the HPM box number, and nnnn = 1–4095 (Data access limit)

Use the hardware reference address !MTmmSss.Parameter for untagged or tagged points where

MT is the IOP type, such as DI (Digital Input)
mm is the IOP Card number (1–40)
The letter "S" is a constant
ss is the slot number on the IOP Card (refer to SLOTNUM parameter)
Parameter name can be up to eight characters and must be a legitimate parameter name.

OPLOTP (RegCtl)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**

Output Low Alarm Trip Point—The Regulatory Control Output Low alarm is set when the output value (OP) drops below the low alarm limit specified by OPLOTP. The alarm is removed when OP returns to normal. A suitable dead band is provided by OPALDB. See also OPLOPR. Available on Release 510 and later software.

Range: **OPLOLM** to **OPHITP, NaN**

Helpful Hint: The Regulatory Control Output Low alarm is only available for points configured as full. The alarm is disabled if OPLOTP is not configured.

OPMCHLM

Type:	**Real**	**Output Minimum Change in Percent**
Lock:	**Supr**	
Default:	**0.0**	
PtRes:	**HPM**	
Range:	**≥ 0.0, NaN**	

Helpful Hint:	OP changes only if new output % - old output % is greater than the percentage in parameter OPMCHLM. Entering NaN disables limit checking by forcing OPMCHLM to its extreme value (0.0).

OPOUT0

Type:	**Real**	**Output Coordinate Number 0 in Percent—**Defines the OPOUT0 coordinate when output characterization has been selected. (OPCHAR = On) This coordinate is fixed at a value of -6.9%.
Lock:	**View**	
Default:	**-6.9%**	
PtRes:	**HPM**	
Range:	**N/A**	

OPOUT1–4

Type:	**Real**	**Output Coordinates Number 1, 2, 3, or 4 in Percent—**Define the OPOUT1–OPOUT4 coordinates when output characterization has been selected (OPCHAR is On).
Lock:	**Supr**	
Default:	**N/A**	
PtRes:	**HPM**	
Range:	**≥ previous coordinate ≤ next coordinate**	

OPOUT5

Type:	**Real**	**Output Coordinate Number 5 in Percent—**Defines the OPOUT5 coordinate when output characterization has been selected. (OPCHAR = On) This coordinate is fixed at a value of 106.9%.
Lock:	**View**	
Default:	**106.9%**	
PtRes:	**HPM**	
Range:	**N/A**	

OPRATRFL (DevCtl, DigComp, RegCtl)

Type:	**Logical**	**Operator Mode Attribute Flag—**Indicates whether the current mode attribute is Operator.
Lock:	**View**	
Default:	**N/A**	
PtRes:	**HPM**	
Range:	**Off** (Current mode attribute is Program or None) **On** (Current mode attribute is Operator)	

OPRINPUT (ProcMod)

Type: **Real**
Lock: **Oper**
Default: **0.0**
PtRes: **HPM**
Range: **N/A**

Operator Input—Defines the value entered by the operator in response to the last sequence message.

OPROCLM

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: $\geq$ **0.1**
NaN

Output Rate of Change Limit in Percent Per Minute

Helpful Hint:	Entering NaN disables limit checking.

OPTDIR (AO)

Type: **E:POLARITY**
Lock: **Eng/PB**
Default: **Direct**
PtRes: **HPM**
Range: 0-**Direct** (For final OP: 0% = 4 mA; 100% = 20 mA)
1-**Reverse** (For final OP: 0% = 20 mA; 100% = 4 mA)

Analog Output Direct/Reverse Action—Defines the output action of the OPFINAL value of the data point.

OPTDIR (DigOut)

Type: **E:POLARITY**
Lock: **Eng/PB**
Default: **Direct**
PtRes: **HPM**
Range: 0-**Direct** (OP is the % On time)
1-**Reverse** (OP is the % Off time)

Output Direction—Defines the direct/reverse action of the PWM digital output.

OROFFSET (ORSel)

Type: **Logical**
Lock: **Eng/PB**
Default: **On**
PtRes: **HPM**
Range: **Off** (No override offset is applied)
On (Offset of Gain times Error is applied)

Override Offset—When OROFFSET is On, override initialization of Pid-type algorithm points connected to this ORSel algorithm applies an override offset equal to Gain times Error (PV - SP).

OROPT (DevCtl, DigComp)

Type: **Logical**
Lock: **Eng/PB**
Default: **Off**
PtRes: **HPM**
Range: **On** (Override option enabled)
Off (Override option disabled)

Override Option—Allows the operator to bypass permissive and override interlocks by setting BYPASS On.

OROPT (ORSel)

Type: **Logical**
Lock: **Eng/PB**
Default: **Off**
PtRes: **HPM**
Range: **Off** (No override)
On (Inputs can be overridden)

Override Option—Defines whether the operator can put the point in a bypass state where any of the X1-X4 inputs can be bypassed. Also, when on, the feedback value is propagated to nonselected primaries of the override selector algorithm. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

OUT0–12 (GenLin)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **Any value but NaN**

Output Coordinates 0 -12—Define the output value at the respective coordinates.

OVERFLOW

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (No overflow)
On (Overflow)

Accumulation Overflow Flag—Indicates whether the accumulated value has overflowed.

OVERLAP (Array)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **0 through the number of Array slots** (NARRSLOT)

Overlapping Array Slot Number—Indicates the Array slot number containing the data being referenced by the slot currently being built.

OVERPHAS (ProcMod)

Type: **E:JUMPDIR**
Lock: **ONPROC and CNTLLOCK parameters**
Default: **Blank**
PtRes: **HPM**
Range: 0-**Forward** (Skip to next phase)
1-**Backward** (Go back to previous phase)

Override Current Phase—Allows the operator to override the current phase of the sequence by skipping forward to the next phase, or backward to the previous phase. A phase can be overridden in this manner only when the sequence execution state is PAUSE, FAIL, or ERROR.

OVERSTAT (ProcMod)

Type: **E:JUMPDIR**
Lock: **ONPROC and CNTLLOCK parameters**
Default: **Blank**
PtRes: **HPM**
Range: 0-**Forward** (Skip to next statement)
1-**Backward** (Go back to previous statement)

Override Current Statement—Allows the operator to override the current statement of the sequence by skipping forward to the next statement, or backward to the previous statement. A statement can be overridden in this manner only when the sequence execution state is PAUSE, FAIL, or ERROR.

OVERSTEP (ProcMod)

Type: **E:JUMPDIR**
Lock: **ONPROC**
Default: **Blank**
PtRes: **HPM**
Range: 0-**Forward** (Skip to next step)
1-**Backward** (Go back to previous step)

Override Current Step—Allows the operator to override the current step of the sequence by skipping forward to the next step, or backward to the previous step. A step can be overridden in this manner only when the sequence execution state is PAUSE, FAIL, or ERROR.

OVERVAL

Type:	**Integer**	**Overview Value in Percent—**Defines the amount of deviation (PV - SP, in percent) that causes the PV to reach the overview limit. For digital points, the display shows the current state of the point.
Lock:	**Eng/PB**	As shown in Figure O-1, The baseline shows the normal operating value for this PV.
Default:	**25**	
PtRes:	**NIM**	
Range:	**0** to **100** (Entering a 0 suppresses the value; value is not shown on the display)	

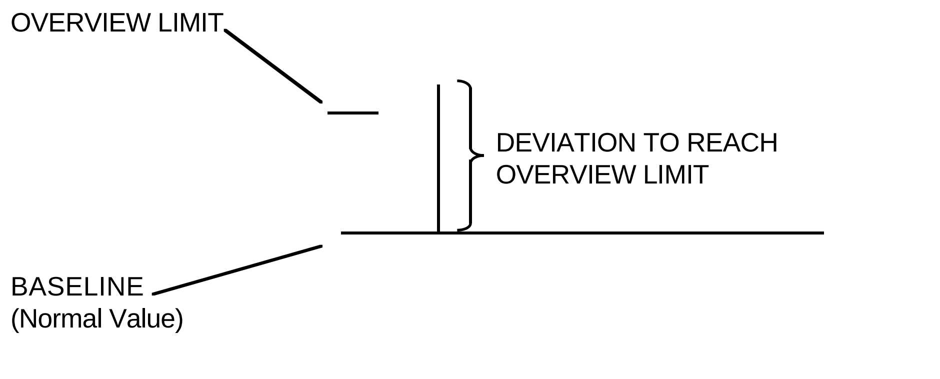


Figure O-1 — Overview Limit

OVRCTIM (DevCtl)

Type:	**Time**	**Time Over High Trip Limit—**The amount of time the SECVAR parameter is continuously greater than the SECVAR high trip limit.
Lock:	**View**	
Default:	**0**	
PtRes:	**HPM**	
Range:	**Duration** (0 to 9999 days, with a resolution to 1 second)	

OVRDALOP (DevCtl, DigComp)

Type:	**E:$OVRALOP**	**Override Alarm Option—**The override alarm option for I0, I1, and I2 parameters.
Lock:	**Eng/PB**	
Default:	**None**	
PtRes:	**HPM**	
Range:	0-**None** (No override alarming) 1-**Auto_Rtn** (Return to normal when override is cleared) **2-Cnfm_Rqd** (Confirm to clear, after interlock is cleared)	

OVRDALPR (DevCtl, DigComp)

Type: **E:ALPRIOR** **Override Alarm Priority—**Defines the priority of an override alarm.
Lock: **Engr**
Default: **Low**
PtRes: **HPM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

OVRDCONF (DevCtl, DigComp)

Type: **Logical**
Lock: **Oper**
Default: **Off**
PtRes: **HPM**

Override Confirmation Flag—Indicates one of the four override alarms SI0CONF or I0CONF-I2CONF has not yet been confirmed. This flag is also used to confirm the alarm.

Range: **Off** (An alarm is not waiting for confirmation)
On (An alarm is waiting for confirmation)

OVRDDESC (DevCtl, DigComp)

Type: **String_8**
Lock: **View**
Default: **Blank**
PtRes: **HPM**

Override Alarm Descriptor—Input connections and logic gating are examined in order to determine which input was the source of change for the interlock. For the Device Control point, descriptor text for this parameter is taken from LIDESC (1-12). Descriptive text for the Digital Composite point is taken from SI0 DESC or I0-I2 DESC parameters.

Range: **SI0 Desc** (Current interlock or input descriptor)
I0 - I2 Desc (Current interlock or input descriptor)
L1 - L12 Desc (Current interlock or input descriptor, Device Control only)

OVRDI0FL, OVRDI1FL, OVRDI2FL, OVRDSIFL (DevCtl, DigComp)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**

Override and Safety Override Alarm Flag—Indicates that an override is active, or that a confirmable override was cleared, but not yet confirmed.

Range: **Off** (Override flag is not active)
On (Override flag is active)

-P-

P (FlowComp)

Type: **Real**
Lock: **View**
Default: **1.0**
PtRes: **HPM**
Range: **≥ 0.0**

Pressure Input—Indicates the measured actual gage pressure.

P0 (FlowComp)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: **N/A**

Zero Reference for Pressure—P0 is the zero reference pressure input and is in the same engineering units as the P input. P0 is typically 14.696 if P is in psig or 101.325 if P is in kiloPascals. Enter the absolute value of the number.

P0–P2 (DevCtl, DigComp)

Type: **Logical**
Lock: **Engr**
Default: **On**
PtRes: **HPM**

Permissive Interlocks for Output States 0, 1, & 2—Permissive interlocks are controlled by logic slot outputs, and each interlock determines whether the operator and user program are allowed to use the respective state, or are locked out from that state. A permissive interlock (P0-P2) is provided for each state (STATE0-STATE2). The permissive interlocks themselves never cause the outputs to change. P0-P2 can be changed by a logic block or a program when the point is active and the mode attribute is Program. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: **Off** (Respective state is locked out)
On (Respective state is permitted to be used)

Helpful Hint:	P0–P2 configuration requires PTEXECST = InActive or PNTSTATE = Idle.

P1–P6 (RegPV)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A, NaN**

PV Inputs 1-6—Indicates the current values at the inputs to the RegPV algorithm. For Totalizers, P2 is the floating point input of AV.

P1STS–P6STS

Type: **E:PVVALST** **P1–P6 Status—**Indicate the status of the up to six inputs at the RegPV algorithm.
Lock: **View**
Default: **Bad**
PtRes: **HPM**
Range: 0-**Bad** (Value is bad and replaced with NaN)
1-**Uncertn** (Status of the value is uncertain)
2-**Normal** (Value is good)

PAUSETIM (DevCtl, DigComp)

Type: **Integer** **State 0 Pause Time—**The amount of time to pause in State 0 on an OP state change, if the STCHGOPT parameter equals STATE0.
Lock: **Supr**
Default: **0**
PtRes: **HPM**
Range: **0** to **1000 seconds**

PERIOD (ProcMod, Timer)

Type: **Real** **Period—**Defines the processing period in seconds.
Lock: **View**
Default: **1 seconds**
PtRes: **HPM**
Range: **1 seconds**

PERIOD (DevCtl, DigComp, Logic, RegCtl, RegPV)

Type: **Real** **Period—**Defines the processing period in seconds.
Lock: **View**
Default: **1 second**
PtRes: **HPM**
Range: **0.25, 0.5, or 1.0 seconds**

PERIOD

Type: **Real** **Period—**specifies the scan period in seconds.
Lock: **PtBld**
Default: **.25 sec for Logic, DigComp, & DevCtl**
.50 sec for RegPv and RegCtl
1.00 sec for ProcMod
PtRes: **HPM**
Range: **0.0625, 0.125, 0.25, 0.5, 1.0, 2.0, and 4.0 seconds**

PERIOD (DigOut)

Type: **Real** **Period—**Defines length of period for an SO output from DigOut point that has been configured for a PWM output.
Lock: **Eng/PB**
Default: **10.0 seconds**
PtRes: **HPM**
Range: **1.0** to **120.0 seconds**

PFDLYFL (RegCtl, RegPV, DevCtl, DigComp, Logic, ProcMod)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (prefetch data is available for slot execution.)
On (prefetch data is not available for slot execution.)

PreFetch Delayed Flag—Set when prefetch data is not available for slot execution.

PGALGID(1)–(4) (DevCtl)

Type: **E:$GTALGID in an Array (1..4)**
Lock: **PtBld**
Default: **Null**
PtRes: **HPM**
Range: **NULL** (No algorithm)
AND (And Gate algorithm)
OR (Or Gate algorithm)
NAND (Nand Gate algorithm)
NOR (Nor Gate algorithm)
XOR (Exclusive Or Gate algorithm)
PAND (Pulse Nand Gate algorithm)
POR (Pulse Or Gate algorithm)
PNAND (Pulse Nand Gate algorithm)
PNOR (Pulse Nor Gate algorithm)
PXOR (Pulse Exclusive-Or Gate algorithm)

Primary Gate Algorithm ID—Defines the algorithm IDs for primary gates. The Boolean logic gates beginning with “P” have a user-defined pulse size.

PGDSTN(1)–(4) (DevCtl)

Type: **E:$GATDSTN in an Array (1..4)**
Lock: **PtBld**
Default: **None**
PtRes: **HPM**
Range: **None** (No destination)
SI0 (Output goes to Safety Interlock)
I0, I1, I2 (Output goes to Interlock)
P0, P1, P2 (Output goes to Permissives)
SOCMD0, SOCMD1, SOCMD2 (Output is commanded to go to SOCMD0, 1 or 2)
OPCMD (Output is commanded to go to OPCMD parameter)
SG1, SG2 (Output goes to Secondary gates 1 or 2)

Primary Gate Destination—Defines the output destination of the primary gate.

PGPLSWTH(1)–(4) (DevCtl)

Type: **Integer in an Array (1..4)**
Lock: **Supr**
Default: **0**
PtRes: **HPM**
Range: **0** to **8000** seconds

Primary Gate Pulse Width—Indicates the pulse width for primary gates whose algorithm starts with a “P”.

PGSO(1)–(4) (DevCtl)

Type: **Logical in an Array (1..4)**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

Primary Gate Status Output—Indicates the output value of the primary gate.

PHASE (ProcMod)

Type: **String_8**
Lock: **View**
Default: **Spaces**
PtRes: **HPM**
Range: **N/A**

Phase Name—Indicates the current phase of the sequence executing in the process module.

PHASEAL (ProcMod)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **On** (Phase has not been completed in the specified time)
Off (No phase alarm)

Phase Alarm—Indicates whether the current phase of the sequence has been completed within the specified time.

PHASETIM (ProcMod)

Type: **Integer**
Lock: **View**
Default: **0 seconds**
PtRes: **HPM**
Range: **0** to **9999** minutes

Phase Time —Indicates the time remaining in minutes in the current phase before a phase alarm is generated. The maximum time allowed for the phase is defined by the phase statement.

PHREMTIM (ProcMod)

Type: **Time**
Lock: **View**
Default: **0 seconds**
PtRes: **HPM**
Range: **N/A**

Phase Time Remaining—Indicates the time remaining in time duration before a phase alarm is generated. This value is displayed in the HPM Detail display.

PIALGID (1)–(12) (DevCtl)

Type: **E:$I1ALGID in an Array (1..12)**
Lock: **PtBld**
Default: **Null**
PtRes: **HPM**
Range: **NULL** (No algorithm)
INVERT (Invert Logical algorithm)
GT (Greater Than algorithm)
GE (Greater Than or Equal To algorithm)
LT (Less Than algorithm)
LE (Less Than or Equal To algorithm)
EQ (Equal To algorithm)
NE (Not Equal To algorithm)
GT2 (Greater Than algorithm)
GE2 (Greater Than or Equal To algorithm)
LT2 (Less Than algorithm)
LE2 (Less Than or Equal To algorithm)
EQ2 (Equal To algorithm)
NE2 (Not Equal To algorithm)
IN_SET (Compares the input to values in the INSET array)

Primary Input Gate Algorithm ID—The XX algorithms compare this input to the PINN (1)-(12) parameter, and the IN_SET algorithm compares it to the range of 10 values in the NNINSET (1)-(10) parameter. The XX2 algorithms compare this input defined by the PISRC(1)-(12) parameter.

PIDEADBD(1)–(12) (DevCtl)

Type: **Real in an Array (1..12)**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **>0**

Primary Input Gate Deadband—The deadband for primary input gates that is configured for an arithmetic algorithm.

PIDFORM

Type: **E:PIDFORM**
Lock: **Eng/PB**
Default: **Interact**
PtRes: **HPM**
Range: 0-**Interact** [(Proportional + Integral) x Derivative]
1-**Ideal** (Proportional + Integral + Derivative)

PID Controller Form—Defines the type of Pid controller form. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

PIDSTN(1)-(6)

Type: **Ent.Prm**
Lock: **View**
Default: **Based on PVALGID, PVEQN, & N**
PtRes: **HPM**
Range: **1–8 character valid parameter name**

PV Input Connection Destination—Defines the parameter in the data point that is to receive the value provided by the respective input connection. There can be up to six input-connection destinations for a RegPV point.

PINN (1)–(12) (DevCtl)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: **<> NaN**

Primary Input Constants Numeric—The numeric constant for arithmetic comparisons of primary input gates using XX algorithms.

PISO (1)–(12) (DevCtl)

Type: **Logical in an Array (1..12)**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

Primary Input Gate Output Value—Indicates the output value of the primary input gate.

PISRC(1)–(12) (DevCtl)

Type: **E:$GATESRC in an Array (1..12)**
Lock: **PtBld**
Default: **Null**
PtRes: **HPM**

Primary Input Source—The source for the second input of primary input gates for arithmetic comparison algorithms that use a second external input (i.e., XX2 algorithms).

Range: **NULL** (No source for input)
L1..L12 (These values correspond with the LISRC(1)—(12) parameter)

PISRC(1)–PISRC(6)

Type: **Ent.Prm**
Lock: **PtBld**
Default: **null.null**
PtRes: **HPM**

PV Input Connection Source—Define the parameters whose current values are to be fetched and then written to the up to six RegPV algorithm inputs. The source parameter name can be specified using the "Tagname.Parameter" format. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: Use Tagname.Parameter for tagged points where Tag name can be up to 16 characters and the permissible character set is as follows:
- Alphabetics A-Z (uppercase only)
- Numerics 0-9 (an all numeric tag name is not allowed)
- Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
- Embedded space characters are not allowed.
- An * is used to default to this point's tag name.
- Parameter name can be up to eight characters, and must be a legitimate parameter name.

PIUOTDCF (STI, LLMUX, RHMUX)

Type: **Logical**
Lock: **Eng/View**
Default: **On**
PtRes: **HPM**

Open Thermocouple Detection Enable—Defines whether the point is to detect an open thermocouple condition. This parameter is configurable for each STI point that is connected to a smart temperature transmitter and for each LLMUX point. NOTE: There is no special point type for RHMUX. The LLMUX point type also applies to the RHMUX IOP.

This parameter is a view-only parameter when the point execution state PTEXECST is Active.

Range: **On** (Detect an open thermocouple condition)
Off (Do not detect an open thermocouple)

PIUOTDCF(1)–(168) (LLAI)

Type: **Logical**
Lock: **Supr**
Default: **On**
PtRes: **HPM**

LLAI Open Sensor Detection Enable—Defines whether an open-sensor condition is to be detected for all eight LLAI points.

Range: **On** (Detect open-sensor conditions)
Off (Do not detect open-sensor conditions)

PKGOPT (HPM Box)

Type: **E:$PKGOPT**
Lock: **PtBld**
Default: **REDUN_2F**
PtRes: **HPM**

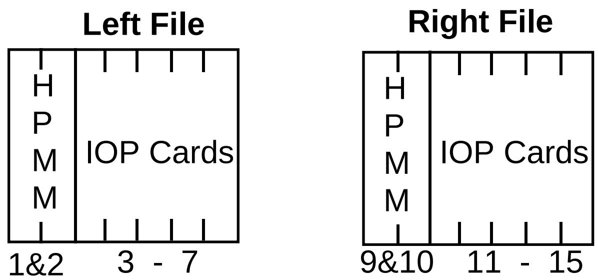
HPMM Hardware Packaging Option. The tables below show the default hardware location for each choice.

Range: 1-**REDUN** (HPMMs in two 7-slot files/can have up to 40 IOPs)
2-**REDUN_2F** (HPMMs in two 15-slot files/can have up to 40 IOPs)
3-**REDUN_IO** (HPMMs/two separate 15-slot files/can have full redundant IOPs)
4-**NODEFALT** (Sets file/card positions of IOPs to 0. Used to bypass all defaults for IOP File/Card positions).

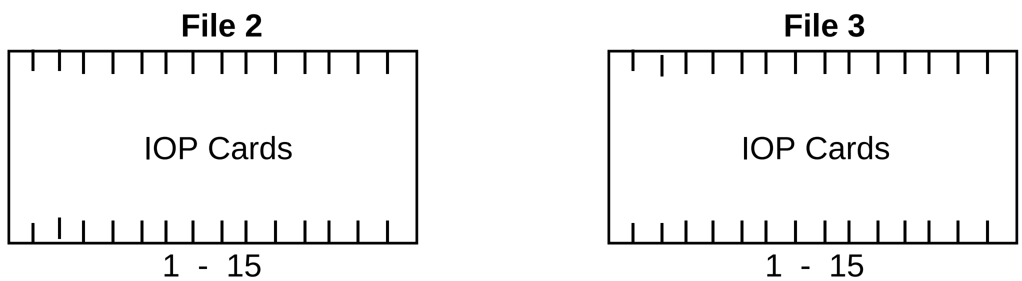
> *Helpful Hints:* Parameter NODENUM must be equal to an odd number no matter which option is selected for PKGOPT.
>
> During Node Specific configuration, if you choose Redun_IO, you must type in file and card numbers for the IOP cards. Refer to the tables below or the HPM Node Specific Configuration Form if necessary.
>
> An HPMM can be operated as a non-redundant node independent of the PKGOPT selected.

The options are illustrated or discussed further in the following pages:

REDUN



The Primary and Secondary HPMM Cards must be in Left File Card slots 1 & 2 and Right Card File, Card slots 9 & 10.



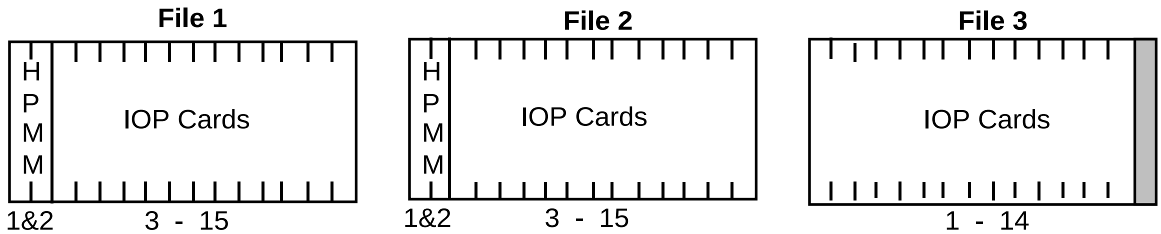
The File and Card position of the IOPs are defaulted as follows:

Hardware	File	Card Slot		Hardware	File	Card Slot
IOM-A 1-5	1	3 - 7		IOM-A 11- 25	2	1 - 15
IOM-A 6-10	1	11 - 15		IOM-A 26-40	3	1 - 15

NOTE

To operate the HPMM as non-redundant, configure PKGOPT = REDUN as above but install only one of the HPMM card sets above. The backplane slot positions vacated by the second HPMM cards can be used to house IOP cards if necessary.

REDUN_2F



The Primary and Secondary HPMMs must be in File 1, Card slots 1 & 2 and in File 2, Card Slots 1 & 2.

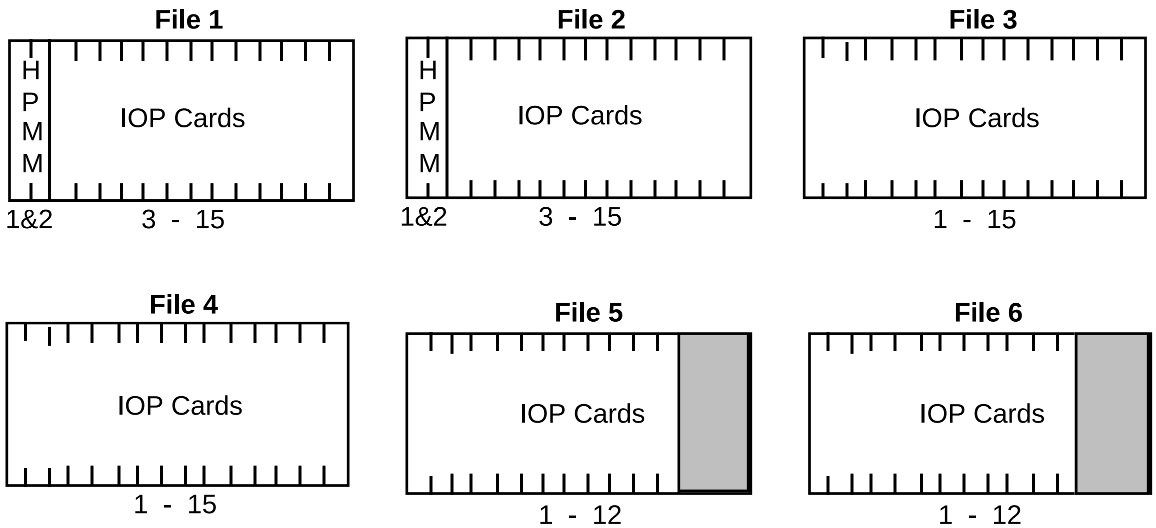
File and Card positions of the IOPs are defaulted as follows:

Hardware	File	Card Slot
IOM-A 1 - 13	1	3 - 15
IOM-A 14 - 26	2	14 - 26
IOM-A 27 - 40	3	1 - 14

NOTE

To operate the HPMM as non-redundant, configure PKGOPT = REDUN_2F as above but install only one of the HPMM card sets above. The backplane slot positions vacated by the second HPMM cards can be used to house IOP cards if necessary.

REDUN_IO



The Primary and Secondary HPMMs must be in File 1, Card Slots 1 & 2 and in File 2, Card Slots 1 & 2.

The File and Card position of the IOPs are defaulted as follows:

Hardware	File	Card Slot		Hardware	File	Card Slot
IOM-A 1-13	1	3 - 15		IOM-B 1 - 13	2	3 - 15
IOM-A 14-28	3	1 - 15		IOM-B 14 - 28	4	1 - 15
IOM-A 29-40	5	1 - 12		IOM-B 29 - 40	6	1 - 12

Note that on download of this configuration to the HPM, the PKGOPT is changed back to REDUN_2F.

NODEFALT

The HPMM File(s) may be like any of the previous three configurations and up to 40 IOPs are allowed. The IOP file/Card positions must be configured by the user. Note that on download to the HPM, PKGOPT changes to REDUN, or REDUN_2F based on the actual hardware.

I/O Simulator Option

The optional I/O Simulator can be used to build points for this (the host) HPM or another HPM. When using the I/O Simulator personality you may choose a packaging option (PKGOPT) that is different from the physical backplane/hardware configuration of the host HPMM. The intent is to let you choose a PKGOPT based on either the host's hardware configuration or that of another HPMM. This allows you to create databases for other HPMs using a single HPM I/O Simulator independent of its actual physical configuration. The following rules apply:

Host HPM Configuration	Other HPM Configuration	PKGOPT Selection
Any	7 - Slot	REDUN or NODEFALT
Any	15 - Slot	REDUN_2F, REDUN_IO, or NODEFALT

PMEVOVFL

Type: **Logical** **HPMM Event Overflow Flag**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Off** (No overflow)
On (Overflow has occurred)

PMMCHAER (HPM Box)

Type: **Integer** **HPMM I/O Link Channel A Error Count**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: $\geq$ **0**

PMMCHASL (HPM Box)

Type: **Integer** **HPMM I/O Link Channel A Silence Count**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: $\geq$ **0**

PMMCHBER (HPM Box)

Type: **Integer** **HPMM I/O Link Channel B Error Count**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: $\geq$ **0**

PMMCHBSL (HPM Box)

Type: **Integer** **HPMM I/O Link Channel B Silence Count**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: $\geq$ **0**

PMMCMD (HPM Box)

Type: **E:$PMMCMD** **HPMM Command**
Lock: **OnProc**
Default: **None**
PtRes: **HPM**

> **NOTE**
>
> When points are built to a NIM and the NIM is restarted with no database, the points need to be reloaded from checkpoint or the points must be reconfigured. If the database is to be reconfigured, the HPMM must be in Idle, and the point execution state must be Inactive. This allows the point build operation to override the database that already exists there.

Range: 0-**None** (No command request has been issued)
1-**Run** (To "Run" state for processing points)
2-**Idle** (To "Idle" state for reloading the database)
3-**RsIoLCom** (Reset I/O Link communication error count = 0)
4-**ShutDown** (To "Alive" state for reloading personality)
5-**RsUcnLsb** (Reset the Local Statistics Block to zeroes)
6-**SelChnA** (Select Input/Output Link Channel A)
7-**SelChnB** (Select Input/Output Link Channel B)
8-**Warmstrt** (Warm Start)
9-**Coldstrt** (Cold Start)
10-**SwapPri** (Switchover to the redundant HPMM)

PMMCOMER (HPM Box)

Type: **E:$IOMCOMM** **HPMM I/O Link Communication Error Status**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: 0-**None** (No communication errors)
1-**InvAlert** (Invalid Alert—message bit problem)
2-**InvDest** (Invalid Destination)
3-**InvChCnt** (Invalid Character Count)
4-**InvSourc** (Invalid Source)
5-**InvCmd** (Invalid Command)
6-**Checksum** (Data record Checksum Error)
7-**No_Resp** (No Response)
8-**ChTimOut** (Channel Time Out)
9-**MsgOvRun** (Message Overrun)
10-**GapError** (Message gap is too long)
11-**LpBckErr** (Loop Back Error)
12-**NTH_0** (Next Token Holder equals zero)
13-**TknRecov** (Token Recovery in progress)
14-**RplBufOv** (Reply Buffer Overflow)

PMMCTLST (HPM Box)

Type: **Logical** **HPMM Control Processor Status**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Control processor has not failed)
On (Control processor has failed)

PMMIOLST (HPM Box)

Type: **E:$IOMSTS**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Notconfg** (IOP not configured)
Confgmis (Configuration mismatch detected)
OK (Module is running with no soft fail errors)
Idle (Module is idle with no soft fail errors)
Softfail (Module is running with soft fail error(s) present)
Idlesf (Module is idle with soft failure error(s) present)
Nonexist (Module does not exist at this address)
Noresp (No response from module)
Poweron (Module state is Power On)
Commerr (Communication error to IOP)
Unavail (Module is unavailable for communication)

PMMOPER (HPM Box)

Type: **E:$OPERATE** **Process Manager Module Operation—**Indicates the type of HPMM.
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: 0-**NonRedun** (This HPMM has no redundant HPMM to back it up)
1-**Primary** (This HPMM is the primary point processor)
2-**Secondry** (This HPMM is the secondary HPMM that backs up the primary HPMM)

PMMRECCH (HPM Box)

Type: **E:$RECCHN** **HPMM I/O Link Current Receive Channel**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: 0-**ChannelA** (Channel A is channel currently receiving)
1-**ChannelB** (Channel B is channel currently receiving)

PMMRECHN

Type: **E:$RECCHN** **HPMM Active Receive Channel**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **ChannelA**
ChannelB

PMMSFST(1)–(96)

Type: **Logical** **HPMM Soft Failure**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Off**
On

PMMSTS (HPM Box)

Type: **E:$NODESTA** **HPMM Primary Status**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: 0-**OffNet** (NIM cannot communicate with the HPMM)
1-**ConfgMs** (Configuration mismatch detected)
2-**Idle** (Event reports but no point processing)
3-**IdleSF** (Soft failure occurred in Idle state)
4-**OK** (HPMM is operating normally)
5-**SoftFail** (Soft failure while HPMM is running)
6-**Fail** (HPMM can be accessed but CPU is halted; box hard failure has been detected)
10-**Alive** (No event reports or point processing)
11-**AliveSF** (Soft failure during Idle state)
12-**Test** (HPMM is in the test mode)
13-**TestSF** (Soft failure has been detected while the HPMM is in the test mode)
15-**Loading** (Personality or data base is loading)
16 **S_Idle** (Idle in Simulation Mode)
17 **S_IdleSf** (IdleSF in Simulation Mode)
18 **S_OK** (OK in Simulation Mode)
19 **S_SFFail** (SoftFail in Simulation Mode)
20 **S_Pause** (HPM is in the Simulation Pause state)
Standby
StandbySF
Upgrade
UpgradeSF

Helpful Hint:	Loading the HPMM's operating personality software requires PMMSTS = Alive. Loading the HPMM's database requires PMMSTS = Idle. Use parameter PMMCMD's "Shutdown" and "Idle" command requests, respectively.

PNAMIOPA

Type: **String_16**
Lock: **View**
Default: **Parameter_Invalid**
PtRes: **IOP**
Range:

Physical Node Name Assigned to IOP A— Returns the Fieldbus physical node name assigned to IOPA.

PNAMIOPB

Type: **String_16**
Lock: **View**
Default: **Parameter_Invalid**
PtRes: **IOP**
Range:

Physical Node Name Assigned to IOP B— Returns the Fieldbus physical node name assigned to IOPB.

PNTFORM

Type: **E:$PNTFORM**
Lock: **View/PB**
Default: **Full**
PtRes: **HPM**
Range: 0-**Full** (Point is fully displayed and alarmed)
1-**Componnt** (Point is partially displayed but not alarmed)

Point Form—Defines the form of the data point that is implemented. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of this function.

Helpful Hint: This parameter is not applicable to DigOut points.

PNTMODTY

Type: **E:$PMMDTY**
Lock: **View**
Default: **N/A**
PTRes: **NIM**
Range:
AO (Analog Output)
AO_16 (Analog Output/high density)
DI (Digital Input)
DO (Digital Output)
DO_32 (Digital Output/high density)
HLAI (High-Level Analog Input)
LLAI (Low-Level Analog Input)
STI8M(Smart Transmitter Interface)
NotConfg (Not Configured)
PI (Pulse Input)
HPMM (High-Performance Process Manager Module)
LLMUX (Low-Level Analog Input Multiplexer) All references to LLMUX also apply to RHMUX, except that RHMUX does not support SENSRTYP of RTD.
DISOE (Sequence of Events)
SI (Serial Interface)
AO_16 (Analog Output 16)
DO_32 (Digital Output 32)

Point's Module Type—Defines where the data point resides. Control points such as DigComp, RegPV, RegCtl, Logic, Process Module, Array, Box Flags, Box Numerics, and Box Timers reside in the HPMM.

PNTNODTY

Type: **E:$UCNNDTY** **Point's Node Type—**Defines the type of node on the UCN
Lock: **View**
Default: **N/A**
PtRes: **NIM**
Range: **NIM** (Network Interface Module)
HPM (High-Performance Process Manager)
NotConfg (Node not configured)

PNTSTATE

Type: **E:PNTSTATE** **Point's Overall State—**Defines the state of the data point, which is based on the state of the HPMM and the IOP Card in which it resides.
Lock: **View**
Default: **N/A**
PtRes: **NIM**
Range: **Failed** (NIM cannot communicate with point's HPMM/IOP)
Idle (Point's HPMM or IOP is in the Idle State)
OK (Point's HPMM or IOP is the Run State and is OK)
UNCERTN (Point's HPMM or IOP state is uncertain)

PNTTYPE

Type: **E:PNTTYPE** **Point Type—**Defines the type of point in the HPM.
Lock: **PtBld**
Default: **Null**
PtRes: **HPM**
Range: 0-**Null** (Not configured)
1-**AnalgIn** (Analog Input)
2-**AnalgOut** (Analog Output)
4-**DigIn** (Digital Input)
5-**DigOut** (Digital Output)
6-**DigCom** (Digital Composite)
8-**RegPV** (Regulatory PV)
9-**RegCtl** (Regulatory Control)
10-**Logic** (Logic)
11-**Array** (Array)
12-**Flag** (Flag)
13-**Numeric** (Numeric)
14-**ProcModl** (Process Module)
22-**Timer (Timer)**
28-**DevCtl** (Device Control)

Helpful Hint: PNTTYPE of DigOut has a restriction that PNTFORM cannot be = Full.

POSITION (HPM Box)

Type: **E:$POSITIN** **HPMM File Position**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: 0-**Right** (HPMM cards are in card file slots 6–10)
1-**Left** (HPMM cards are in card file slots 1–5)
2-**File_1** (HPMM cards are in card file 1)
3-**File_2** (HPMM cards are in card file 2)
4-**Pref**
5-**Non_Pref**
6-**None**
7-**Unknown** (Not able to determine file position from hardware)

PR2PREFF (HPM Box)

Type: **Real**
Lock: **View**
Default: **100**
PtRes: **HPM**
Range: 0 - 100

Peer-to-Peer Communication Efficiency (in percent)—Indicates the rate of successful and on time UCN transactions from this node.

Helpful Hint:	This statistic is displayed on the Control Configuration page of the HPM Diagnostic Display.

PRGATRFL (DigComp, DevCtl, RegCtl)

Type: **Logical**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **On -** (point is in Program mode attribute)
Off - (point is not in Program mode attribute)

Program Mode Attribute Flag —indicates if the point is in Program Mode attribute.

PRIMMOD

Type: **Ent_Id**
Lock: **Engr**
Default: **Null**
PtRes: **NIM**

Primary Module Point Identifier—Typically used in Batch Processing, this parameter contains the tag name of an HPM point to which this data point is assigned. Other points that belong to the Batch equipment unit should have their PRIMMOD set to this same point. Primmod is used to collect alarms and events from this point along with others related to the specified Primary Module point. Information is collected into a common file, accessible from the Event History Menu.

Range: Tag name of the process module point can be up to 16 characters, and the permissible character set is as follows:
Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.

> *Helpful Hint:* For Box Flag points, this parameter applies to only slots 1 through 128.
> LCN entities that can be stored to PRIMMOD in NIM points are restricted to local NIM points.

PRMDESC(1)–(12) (Logic)

Type: **E:$PMMLGPM**
Lock: **Eng/PB**
Default: **N/A**
PtRes: **NIM**

Parameter Descriptor Assignment—Defines up to 12 logic-slot parameters to which custom generic descriptors entered through parameters GENDESC(1-12) are to be assigned.

Range: **L1...L12** (Logic-slot inputs)
FL1...FL12 (Logic-slot flags)
NN1...NN8 (Logic-slot numerics)
SO1...SO24 (Logic-slot outputs)

PROCMOD (ProcMod)

Type: **E:PROCMOD**
Lock: **Determined by CNTLLOCK parameter**
Default: **Off**
PtRes: **HPM**

Process Module Operating State—Represents the operational condition of a process module. Refer to the *HPM Control Functions and Algorithms* manual for a state diagram.

Range: 0-**Off** (Off)
2-**Norm** (Normal)
4-**Hold** (Hold)
5-**Shdn** (Shutdown)
6-**Emsd** (Emergency Shutdown)
7-**Strt** (Start)
8-**Stop** (Stop)

PRPMMSTS

Type: **E:$NODESTA** **Previous HPMM Status**
Lock: **View**
Default:
PtRes: **HPM**

Range: **Offnet** (HPMM is offnet with no communications possible)
Confgmis (HPMM is in configuration mismatch)
Idle (HPMM is idle)
Idlesf (HPMM is idle with soft failure(s))
OK (HPMM is running with no errors)
Softfail (HPMM is running with soft failure(s))
Fail (HPMM has failed)
Poweron (HPMM is in Power On state-startup condition from power loss)
Alive (HPMM has passed self diagnostics and is ready to accept personality)
Alivesf (HPMM diagnostics have soft failure)
Loading (HPMM is loading personality)
Notconfg (HPMM is not configured on network)
Unavail (HPMM is unavailable on network for communications)
Test (HPMM is in Test mode)
Testsf (HPMM is in Test mode with a soft failure(s))
Standby
StandbySF
Upgrade
UpgradeSF

PRVCOMFL

Type: **E:$PMMHFST** **Previous HPMM Communications Board Failure**
Lock: **View**
Default:
PtRes: **HPM**
Range: **NULL** (Unknown Error)
PWRDWN (Power is Off)
LR_PAR (Local Ram Parity Error)
LR_LRAM (Local Ram Error)
LR_CK (Local Ram Check)
LR_EXC (Local Ram Exception)
LR_HREV (Local Ram Hardware Revision)
MM_HREV (Memory Board Hardware Revision)
LR_TMR (Local Ram Timer Error)
LR_PTRN (Local Ram Pattern Check Error)
LR_BYTE (Local Ram Byte Error)
LR_ADCD (Local Ram Address Decode Test)
LR_ADDL (Local Ram Additional Checks)
LR_CLRR (Local Ram Scrub Incomplete)
SR_PAR (Shared Ram Parity)
SR_PTRN (Shared Ram Pattern)
SR_ADCD (Shared Ram Address Decode Test)
SR_ADDL (Shared Ram Additional Checks)
GR_PAR (Global Ram Parity)
GR_PTRN (Global Ram Pattern Check Error)
GR_BYTE (Global Ram Byte Error)
GR_ADCD (Global Ram Address Decode Test)
GR_ADDL (Global Ram Additional Checks)
GR_CLRR (Global Ram Scrub Incomplete)
31_NR (IOL Processor, No Response or Failure)
31_ALIV (IOL Processor, Transmitter Not Alive)
31_ILTN (IOL Processor, Illegal Transition)
NMI_UNK (Unknown NMI Request)
BADUCNN (UCN Address Parity or Duplicate Address)
NR (No Response From Other Processor)
MRFT (Memory Reference Table - Pattern Build Fail)
NOMTOS (No MTOS Readout)
LLC_COMM (LLC Communication Fatal Error)
UCNDRV (UCN Driver, Fatal Error)
RD_HREV (Redundancy Card Version/Revision Mismatch)
SW_ERROR (Software Error)
MD_HREV (Modem Card Version/Revision Mismatch)
DA_PTRN (Daughter Card Pattern Test)
DA_BYTE (Daughter Card Byte Write Test)
DA_ADCD (Daughter Card Address Decode)
DA_ADDL (Daughter Card Additional Tests)
DA_CLRR (Daughter Card Scrub Incomplete)
RD_SNPS (Redundancy Card 96 Kw Snapshot Error)
RD_BSLK (Redundancy Card Bus Lock Fail)

PRVCTLFL

Type: **E:$PMMHFST** **Previous HPMM Control Failure**
Lock: **View**
Default:
PtRes: **HPM**

Range: **NULL** (Unknown Error)
PWRDWN (Power is Off)
LR_PAR (Local Ram Parity Error)
LR_LRAM (Local Ram Error)
LR_CK (Local Ram Check)
LR_EXC (Local Ram Exception)
LR_HREV (Local Ram Hardware Revision)
MM_HREV (Memory Board Hardware Revision)
LR_TMR (Local Ram Timer Error)
LR_PTRN (Local Ram Pattern Check Error)
LR_BYTE (Local Ram Byte Error)
LR_ADCD (Local Ram Address Decode Test)
LR_ADDL (Local Ram Additional Checks)
LR_CLRR (Local Ram Scrub Incomplete)
SR_PAR (Shared Ram Parity)
SR_PTRN (Shared Ram Pattern)
SR_ADCD (Shared Ram Address Decode Test)
SR_ADDL (Shared Ram Additional Checks)
GR_PAR (Global Ram Parity)
GR_PTRN (Global Ram Pattern Check Error)
GR_BYTE (Global Ram Byte Error)
GR_ADCD (Global Ram Address Decode Test)
GR_ADDL (Global Ram Additional Checks)
GR_CLRR (Global Ram Scrub Incomplete)
31_NR (IOL Processor, No Response or Failure)
31_ALIV (IOL Processor, Transmitter Not Alive)
31_ILTN (IOL Processor, Illegal Transition)
NMI_UNK (Unknown NMI Request)
BADUCNN (UCN Address Parity or Duplicate Address)
NR (No Response From Other Processor)
MRFT (Memory Reference Table - Pattern Build Fail)
NOMTOS (No MTOS Readout)
LLC_COMM (LLC Communication Fatal Error)
UCNDRV (UCN Driver, Fatal Error)
RD_HREV (Redundancy Card Version/Revision Mismatch)
SW_ERROR (Software Error)
MD_HREV (Modem Card Version/Revision Mismatch)
DA_PTRN (Daughter Card Pattern Test)
DA_BYTE (Daughter Card Byte Write Test)
DA_ADCD (Daughter Card Address Decode)
DA_ADDL (Daughter Card Additional Tests)
DA_CLRR (Daughter Card Scrub Incomplete)
RD_SNPS (Redundancy Card 96 Kw Snapshot Error)
RD_BSLK (Redundancy Card Bus Lock Fail)

PRVIOLFL

Type: **E:$IOMHF** **Previous IOL Failure**
Lock: **View**
Default:
PtRes: **HPM**
Range: **UNKNOWN** (Unknown Error)
POWERDWN (Power is Off)
INVPRGEX (Invalid Program Execution)
EPROMERR (EPROM Error)
RAMCNTER (Ram Contents Error)
RAMADRER (Ram Address Error)
DPAERROR (Physical Address Error)
DSAERROR (Soft Address Error)
RXBUFOFL (Receive Buffer Overflow)
IOLJABER (IOL Jabber Circuit - saw too much traffic)
BADPGJMP (Bad Program Jump)
ADCINCMP (A/D Incomplete)
ADOUTOVF (A/D Output Overflow)
ADOUTUDF (A/D Output Underflow)
ADCCALER (A/D Calibration Error)
BADDCLTC (Bad DC LTC)
DMT_TMOT (Dead Man Time Out)
MLTOUTFL (Multiple Output Failure)
DATBUSFL (Data Bus Failure)
BADDARNG (Bad D/A Range)
MSTRTMOT (Master Time Out 68 K)
CTRCKTFL (Counter Circuit Failure)

PSDLYFL

Type: **Logical** **Poststore Delayed Flag—**Set when poststore data is older than 1 second.
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On (poststore data is older than 1 second.)

PSTS (FlowComp)

Type: **E:PVVALST** **Pressure Input Value Status—**Status of the P input value.
Lock: **View**
Default: **Normal**
PtRes: **HPM**
Range: 0-**Bad** (Value is bad and replaced with NaN)
1-**Uncertn** (Status of the value is uncertain)
2-**Normal** (Value is good)

PTDESC

Type: **String_24**
Lock: **PtBld**
Default: **Blank**
PtRes: **NIM**

Point Descriptor—A 24-character descriptor which is used to describe the point and appears on the Group and Detail Displays for the point. Refer to Figure N-1.

Range: Permissible character set consists of all characters on the Engineer' s Keyboard. Basically this set consists of alphabetics A-Z, numerics 0-9, and the following special characters: space ! % & ' ( ) * + - / : ; > < = ? _ , . $

PTEXECST

Type: **E:PTEXECST**
Lock: **Supr**
Default: **Inactive**
PtRes: **HPM**

Point Execution State—Defines the current execution state of the point.

Range: 0-**Inactive** (Point is not scanned or processed)
1-**Active** (Point is scanned and processed)

PTINAL (RegCtl, RegPV)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**

Point in Alarm Indicator—Indicates when an alarm condition has been detected at this point.

Range: **Off** (Point is not in alarm)
On (Point is in alarm)

PTORST (RegCtl)

Type: **E:ORSTATUS**
Lock: **Prog**
Default: **NotCon**
PtRes: **HPM**

Point Override Status—Indicates the override status of the point.

Range: 0-**NotCon** (Not connected to ORSel algorithm. Also indicates that point has been returned from inactive to active status, or it is undergoing a cold restart, or it is being initialized.)
1-**Sel** (Selected as a part of ORSel strategy)
2-**NotSel** (Not selected as a part of ORSel strategy)

PTSTSIOL

Type: **E:$NODESTA Redundant Partner Status as Seen From the IOL**
Lock: **View**
Default:
PtRes: **HPM**
Range: **OffNet** (NIM cannot communicate with the HPMM)
ConfgMis (Configuration mismatch detected)
Idle (Event reports but no point processing)
Idlesf (Soft failure occurred in Idle state)
Ok (HPMM is operating normally)
SoftFail (Soft failure while HPMM is running)
Fail (HPMM can be accessed but CPU is halted; box hard failure has been detected)
Powron (Power is on)
Alive (No event reports or point processing)
Alivesf (Soft failure during Idle state)
Test (HPMM is in the Test mode)
TestSF (Soft failure has been detected while the HPMM is in the Test mode)
Loading (Personality or database is loading)
Notconfg
Nosynch
Unavail
Standby
StandbySF
Upgrade
UpgradeSF

PTSTSUCN

Type: **E:$NODESTA Redundant Partner Status as Seen From the UCN**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Offnet** (NIM cannot communicate with the HPMM)
Confgms (Configuration mismatch detected)
Idle (Event reports but no point processing)
Idlesf (Soft failure occurred in Idle state)
Ok (HPMM is operating normally)
Softfail (Soft failure while HPMM is running)
Fail (HPMM can be accessed but CPU is halted; box hard failure has been detected)
Powron (Power is on)
Alive (No event reports or point processing)
Alivesf (Soft failure during Idle state)
Test (HPMM is in the Test mode)
Testsf (Soft failure has been detected while the HPMM is in the Test mode)
Loading (Personality or database is loading)
Notconfg
Nosynch
Unavail
Standby
StandbySF
Upgrade
UpgradeSF

PULSEWTH (DevCtl, DigComp)

Type: **Real** — **Pulse Width for Point Outputs**
Lock: **Supr**
Default: **1.0 second**
PtRes: **HPM**
Range: **0.0** to **60.0 seconds**

Helpful Hint:	PULSEWTH change requires DODSTN= "Tagname.ONPULSE" or OFFPULSE. When On is to be written to the DigOut module, a pulse of the specified width is generated. When Off is to be written, no pulse is generated.

PV (AnalgIn, PI)

Type: **Real**
Lock: **Oper**
Default: **NaN**
PtRes: **HPM**
Range: **PVEXEUHI** to **PVEXEULO, NaN**

Process Variable—PV is the PV's current value after the PV is selected from one of the following possible sources: a field device, an operator, or a program. See PVSRCOPT and PVSOURCE.

Helpful Hint:	PV change by a program requires PVSRCOPT = All and PVSOURCE = Sub. PV change by an operator requires PVSRCOPT = All and PVSOURCE = Man.

PV (DevCtl, DigComp)

Type: **E:SD_ENM:PVSTATES**
Lock: **Oper**
Default: **BADPVTXT**
PtRes: **HPM**
Range:
0-**PVSTATES (0)** (Defined by STATETXT (0))
1-**PVSTATES (1)** (Defined by STATETXT (1))
2-**PVSTATES (2)** (Defined by BADPVTXT)
3-**PVSTATES (3)** (Defined by MOVPVTXT)
4-**PVSTATES (4)** (Defined by STATETXT(2))

Current State—PV is the PV's current state after the PV is selected from one of the following possible sources: a field device, an operator, a program, or the output of the PV's data point. See PVSRCOPT and PVSOURCE.

Helpful Hint:	PV change by a program requires PVSRCOPT = All and PVSOURCE = Sub. PV change by an operator requires PVSRCOPT = All and PVSOURCE = Man.

PV (DigIn)

Type: **E:SD_ENM:STATETXT**
Lock: **Oper**
Default: **Off**
PtRes: **HPM**
Range: **STATETXT(0) or STATETXT(1)**

Current State—Indicates the current state of the PV input to this data point. Not appropriate if DITYPE = ACCUM.

Helpful Hint:	PV is derived from the open or closed state of field contacts and from the configured direct or reverse input direction (INPTDIR). PV change by a program requires DITYPE = Latched or status, PVSRCOPT = All, and PVSOURCE = Sub. PV change by an operator requires DITYPE = Latched or status, PVSRCOPT = All, and PVSOURCE = Man.

PV (Flag)

Type: **E:SD_ENM:STATETXT**
Lock: **Oper**
Default: **Blank**
PtRes: **HPM**
Range: **STATETXT(0) or STATETXT(1)**

Current State—Indicates the current state of the flag data point, and it is derived from PVFL. STATETXT(1) is the alarmed state.

PV (Numeric)

Type: **Real**
Lock: **Oper**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Process Variable—Indicates the value of the numeric. This value maps into parameter NN(n) in the HPM box where n = SLOTNUM.

PV (RegCtl, RegPV)

Type: **Real**
Lock: **View for RegCtl, Oper for RegPV**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Process Variable—Indicates the current value of the PV after the PV is selected from one of the following possible sources: a field device, an operator, or a program. See PVSRCOPT and PVSOURCE.

Helpful Hint:	PV change by a program requires PVSRCOPT = All and PVSOURCE = Sub. PV change by an operator requires PVSRCOPT = All and PVSOURCE = Man.

PV (Timer)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **0** to **32000**

Current Value—Indicates the current time in seconds or minutes. The timer starts at 0 and is incremented towards the preset time established by the SP parameter.

PVALDB (RegCtl, RegPV)

Type: **E:PVALDB**
Lock: **Eng/PB**
Default: **One**
PtRes: **HPM**
Range: 0-**Half** (1/2 of 1% of Engineering Unit range)
1-**One** (1% of Engineering Unit range)
2-**Two** (2% of Engineering Unit range)
3-**Three** (3% of Engineering Unit range)
4-**Four** (4% of Engineering Unit range)
5-**Five** (5% of Engineering Unit range)
6-**EU** (Specify deadband in Engineering Units)

PV Alarm Deadband—Alarm deadband is used to prevent excessive recurrence of alarms by adjusting the percent of Engineering Unit range at which the alarm "returns to normal."

PVALDBEU (RegCtl, RegPV)

Type: **Real** — **EU value of alarm deadband**
Lock: **Eng/PB**
Default: **NaN**
PtRes: **HPM**
Range: ≥ **0.0**

PVALGID

Type: **E:$PMMPVAG** — **PV Algorithm Identifier—**Defines which PV algorithm is to be used for a RegPV point.
Lock: **PtBld**
Default: **Null**
PtRes: **HPM**
Range: 0-**Null** (No algorithm configured)
1-**DataAcq** (Data Acquisition)
2-**FlowComp** (Flow Compensation)
3-**MidOf3** (Middle-Of-3 Selector)
4-**HiLoAvg** (High Low Average Selector)
5-**Summer** (Summer)
6-**VdtLdLag** (Variable Dead Time with Lead Lag)
7-**TotaLizr** (Totalizer)
8-**GenLin** (General Linearization)
9-**Calcultr** (Calculator)

PVAUTO (AnalgIn, PI)

Type: **Real** — **PV Auto Value—**Value of the PV after PVCALC is range checked, filtered, and clamped.
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

PVAUTO (DevCtl, DigComp)

Type: **E:PVSTATES** — **Current PV State—**Indicates the current PV state, based on the states of the inputs to the point.
Lock: **View**
Default: **BADPVTXT**
PtRes: **HPM**
Range: 0-**STATETXT(0)**
1-**STATETXT(1)**
2-**BADPVTXT**
3-**MOVPVTXT**
4-**STATETXT(2)** (only if NOSTATES is 3)

PVAUTO (DigIn)

Type: **E:STATETXT** — **Current PV State—**Indicates the current PV state that corresponds to the field contact input after direct/reverse correction.
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **STATETXT(0) or STATETXT(1)**

PVAUTO (RegCtl)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

PV Auto Value Fetched Using Control Input Connection—Indicates the current value of the PV when the RegCtl point is in the Auto mode.

PVAUTO (RegCtl, RegPV)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

PV Auto Value—Indicates the current value of the PV after the algorithm calculation is performed, the range is checked, and the PV is filtered and clamped.

PVAUTOST (RegCtl, RegPV)

Type: **E:PVVALST**
Lock: **View**
Default: **Bad**
PtRes: **HPM**
Range: 0-**Bad** (All inputs, or result in PVCALC is bad)
1-**Uncertn** (Final result in PVCALC is an uncertain value)
2-**Normal** (Final result in PVCALC is a normal value)

PV Auto Value Status—Indicates the current status of the PVAUTO value.

PVCALC (AnalgIn, PI)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **PVEXEUHI** to **PVEXEULO,**
NaN

Calculated PV—PVCALC is the PV value in Engineering Units after the raw PV (PVRAW) input to this data point has been characterized. The value of PVRAW is the PV value provided by the Field Termination Assembly (FTA).

PVCALC (RegPV)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A,**
NaN

Calculated PV—Indicates the value of the PV after the PV has been calculated by the PV algorithm.

PVCHAR

Type: **E:VALCHAR**
Lock: **PtBld**
Default: **Linear**
PtRes: **HPM**

PV Characterization Option—Defines the display characterization to be used for characterizing the input PV value. Characterization is based on the field sensor type.

HLAI, LLMUX, RHMUX, & LLAI — PV Characterization

Range	PNT-MODTY (NOTE 1)	*X = Allowable Sensor Type (SENSRTYP)* pt4-2V	0-5V	1-5V	0-100 mV	TC	Slide wire	RTD	*Valid normal range (PVEULO–PVEUHI) in Degrees C (when TCRNGOPT = Normal for SENSRTYP = Thermcpl)*	*Valid extended range (PVEXEULO–PVEXEUHI) in Degrees C (when TCRNGOPT = Extended for SENSRTYP = Thermcpl)*
0-**Jtherm**	HLAI	X	X	X					-200 to 1200	N/A
	LLAI					X			-100 to 750	-200 to 1200
	LLMUX					X			-100 to 750	-200 to 1200
1-**Ktherm**	HLAI	X	X	X					0 to 1100	-200 to 1370
	LLAI					X			0 to 1100	-200 to 1370
	LLMUX					X			0 to 1100	-200 to 1370
2-**Etherm**	HLAI	X	X	X					-200 to 1000	N/A
	LLAI					X			-150 to 500	-200 to 1000
	LLMUX					X			-150 to 500	-200 to 1000
3-**Ttherm**	HLAI	X	X	X					-230 to 400	N/A
	LLAI					X			-200 to 300	-230 to 400
	LLMUX					X			-200 to 300	-230 to 400
4-**Btherm**	HLAI	X	X	X					100 to 1820	N/A
	LLAI					X			600 to 1650	100 to 1820
	LLMUX					X			600 to 1650	100 to 1820
5-**Stherm**	HLAI	X	X	X					0 to 1700	N/A
	LLAI					X			550 to 1500	0 to 1700
	LLMUX					X			550 to 1500	0 to 1700
6-**Rtherm**	HLAI	X	X	X					0 to 1700	N/A
	LLAI					X			550 to 1500	0 to 1700
	LLMUX					X			550 to 1500	0 to 1700
7-**RPtherm**	HLAI	X	X	X					0 to 1700	N/A
	LLAI					X			550 to 1500	0 to 1700
	LLMUX					X			550 to 1500	0 to 1700
8-**DinRtd**	HLAI	X	X	X					-180 to 800	N/A
	LLAI							X	-200 to 850	N/A
	LLMUX							X	-200 to 850	N/A
9-**JisRtd**	HLAI	X	X	X					-180 to 650	N/A
	LLAI							X	-200 to 650	N/A
	LLMUX							X	-200 to 650	N/A
10-**NicklRtd**	HLAI	X	X	X					-45 to 315	N/A
	LLAI							X	-45 to 315	N/A
	LLMUX							X	-45 to 315	N/A
11-**CopprRtd**	HLAI	X	X	X					-20 to 250	N/A
	LLAI							X	-20 to 250	N/A
	LLMUX							X	-20 to 250	N/A
12-**Linear**	HLAI	X	X	X			X		N/A	N/A
	LLAI		X	X	X				N/A	N/A
	LLMUX				X				N/A	N/A
13-**Sqrroot**	HLAI	X	X	X					N/A	N/A
	LLAI		X	X					N/A	N/A

N/A = Not Applicable

NOTE 1: PNTMODTY LLMUX includes RHMUX for all thermocouple types except RTD; RTD type is not supported by RHMUX IOP.

STI — PV Characterization (Pressure and Magnetic Flow Transmitters)

Range	*Spt_Dp*	*Spt_Gp*	*Spt_Ap*	*Sfm*
Linear	X	X	X	X
Sqrroot	X	--	--	--

X = Allowable Sensor Type

STI — PV Characterization (Temperature Transmitters)

Range	*Normal Range (PVEULO to PVEUHI) in Degrees C (except where noted)*	*Maximum Range (PVEXEULO to PVEXEUHI) in Degrees C (except where noted)*
Linear	-50 to 220 mV	-1000 to 1000 mV
Thermocouples		
Btherm	400 to 1820	200 to 1820
Etherm	-100 to 1000	-200 to 1000
Jtherm	-180 to 1200	-200 to 1200
Ktherm	-170 to 1250	-200 to 1370
NiNiMoTC	600 to 1300	600 to 1300
Ntherm	-100 to 1300	-200 to 1300
Rtherm	0 to 1760	-50 to 1760
Stherm	0 to 1760	-50 to 1760
Ttherm	-120 to 400	-250 to 400
W3W25TC	0 to 2300	0 to 2300
W5W26TC	0 to 2300	0 to 2300
RTDs		
Cu10RTD	-20 to 250	-20 to 250
Cu25RTD	-20 to 250	-20 to 250
Pt100 DinRtd	-200 to 450	-200 to 850
Pt100 JisRtd	-200 to 450	-200 to 640
Pt200 RTD	-200 to 450	-200 to 850
Pt500 RTD	-200 to 450	-200 to 850
RH Rad	420 to 1800	700 to 1800
RTD Ohms	0 to 4KΩ	0 to 4KΩ

PVCHAR (FlowComp)

Type: **E:VALCHAR**
Lock: **PtBld**
Default: **SqRoot**
PtRes: **HPM**
Range: **12-Linear** (The COMPTERM compensation is not square rooted)
13 SqRoot (The COMPTERM compensation is square rooted)

PV Characterization Option—Defines the display characterization to be used for characterizing the input PV value of Regulatory PV point configuring with Flow Compensation.

PVCHGDLY

Type: **Integer**
Lock: **Supr**
Default: **0 seconds**
PtRes: **HPM**
Range: **0** to **60 seconds**

PV Change Delay time in Seconds—Defines the time(in seconds) that a point with a previously detected PV change event is guaranteed to remain at the new value even if the PV returns to its original value. If the point remains at its new value when the delay timer expires, the point is held at the new value.

Helpful Hint: PVCHGDLY requires that EVTOPT = EIP, SOE, or EIPSOE.

PVCLAMP

Type: **E:PVCLAMP**
Lock: **Eng/PB**
Default: **NoClamp**
PtRes: **HPM**

PV Clamping Option—Defines whether PV clamping is to be used for this data point. If PVCLAMP = Clamp and the PV extended range is exceeded, PV value status PVSTS is marked Uncertain and the PV is set equal to the extended limit that was violated.

Range: 0-**NoClamp** (No clamping of the PV value)
1-**Clamp** (Clamp PV value at range extension limit)

PVEQN (FlowComp)

Type: **E:ALGOEQN**
Lock: **PtBld**
Default: **EqA**
PtRes: **HPM**

PV Equation Type—Defines the equation type (EqA-EqE) to be used for this PV algorithm. Refer to the *HPM Control Functions and Algorithms* manual for more information.

Range:

	Comp. Inputs	*Type of Compensation*
0-**EqA**	G	Mass/Volumetric flow of liquid
1-**EqB**	P and T	Mass flow of gases and vapors
2-**EqC**	G, P, and T	Mass flow of gases and vapors w/specific gravity
3-**EqD**	G, P, and T	Volumetric flow of gases and vapors
4-**EqE**	P, T, X, and Q	Mass flow of steam

G = measured or calculated specific gravity or molecular weight, P = measured actual gage pressure, T = measured actual temperature, X = measured actual steam compressibility and Q = measured actual steam quality.

PVEQN (HiLoAvg)

Type: **E:ALGOEQN**
Lock: **Eng/PB**
Default: **EqA**
PtRes: **HPM**
Range: 0-**EqA** (Select and identify highest of up to six inputs)
1-**EqB** (Select and identify lowest of up to six inputs)
2-**EqC** (Calculate the average of up to six inputs)

PV Equation Type—Defines the equation type (EqA-EqC) to be used for this PV algorithm. Refer to the *HPM Control Functions and Algorithms* manual for more information.

PVEQN (MidOf3)

Type:	**E:ALGOEQN**	**PV Equation Type—**Defines the equation type (EqA-EqC) to be used for this PV algorithm. Refer to the *HPM Control Functions and Algorithms* manual for more information.
Lock:	**Eng/PB**	
Default:	**EqA**	
PtRes:	**HPM**	
Range:	0-**EqA** (Highest good input when one or two are bad) 1-**EqB** (Lowest good input when one or two are bad) 2-**EqC** (Average of all good inputs)	

PVEQN (Summer)

Type:	**E:ALGOEQN**	**PV Equation Type—**Defines the equation type (EqA or EqB) to be used for this PV algorithm. Refer to the *HPM Control Functions and Algorithms* manual for more information.
Lock:	**PtBld**	
Default:	**EqA**	
PtRes:	**HPM**	
Range:	0-**EqA** (P1 input is scaled and biased) 1-**EqB** (Up to six inputs are scaled and summed with an overall bias applied)	

PVEQN (Totalizr)

Type:	**E:ALGOEQN**	**PV Equation Type—**Defines the equation type (EqA-EqF) to be used for this PV algorithm. Refer to the *HPM Control Functions and Algorithms* manual for more information.
Lock:	**Eng/PB**	
Default:	**EqA**	
PtRes:	**HPM**	

Range:

Option	*Warm Restart Action*	*Bad Input Handling*
0-**EqA**	Continue	Use zero
1-**EqB**	Continue	Use last good value
2-**EqC**	Continue	Set Bad and stop
3-**EqD**	Set Bad, and stop	Use zero
4-**EqE**	Set Bad, and stop	Use last good value
5-**EqF**	Set Bad, and stop	Set Bad and stop

PVEQN (VdtLdLag)

Type:	**E:ALGOEQN**	**PV Equation Type—**Defines the equation type (EqA-EqD) to be used for this PV algorithm. Refer to the *HPM Control Functions and Algorithms* manual for more information.
Lock:	**PtBld**	
Default:	**EqA**	
PtRes:	**HPM**	
Range:	0-**EqA** (Lead-Lag) 1-**EqB** (Fixed dead time) 2-**EqC** (Variable dead time) 3-**EqD** (Variable dead time with two lags)	

Helpful Hint:	For Equations C and D, the dead time is changed in steps of NLOC*NRATE*TS where NLOC is configurable from 2 to 30, for better resolution of dead time.

PVEUHI

Type:	**Real**	**PV High Range in Engineering Units—**Note that PVEUHI cannot be written with NaN. NaN is the default value only.
Lock:	**Eng/PB**	
Default:	**NaN**	
PtRes:	**HPM**	
Range:	**PVEULO** to **PVEXEUHI, NaN**	

Helpful Hint: For Smartline transmitters, refer to Table A-3 in the *PM/APM Smartline Transmitter Integration Manual*, PM12-410.

PVEULO

Type:	**Real**	**PV Low Range in Engineering Units—**Note that PVEULO cannot be written with NaN. NaN is the default value only.
Lock:	**Eng/PB**	
Default:	**NaN**	
PtRes:	**HPM**	
Range:	**PVEXEULO** to **PVEUHI, NaN**	

Helpful Hint: For Smartline transmitters, refer to Table A-3 in the *PM/APM Smartline Transmitter Integration Manual*, PM12-410.

PVEXEUHI

Type:	**Real**	**PV Extended Engineering Unit Range High—**Both PVEXEUHI and PVEXEULO are used to clamp or detect a bad PV value. Refer to parameter PVEXEULO. Note that PVEXEUHI cannot be written with NaN. NaN is the default value only.
Lock:	**Engr**	
Default:	**NaN**	
PtRes:	**HPM**	
Range:	**≥ PVEUHI, NaN**	

PVEXEULO

Type:	**Real**	**PV Extended Engineering Unit Low Range—**For the LLAI IOP with Thermocouple and RTD sensor types, extended PV range parameters are VIEW ONLY. Their values are defaulted based on the sensor types, the thermocouple range option, and temperature scale. The tables below show the default values in degrees C. For other engineering units, these values are appropriately converted. Note that PVEXEULO cannot be written with NaN. NaN is the default value only.
Lock:	**Engr**	
Default:	**NaN**	
PtRes:	**HPM**	
Range	**≤ PVEULO, NaN**	

Defaults for Extended Range PV Parameters When SENSRTYP = THERMCPL, PVTEMP = Degrees C

	TCRNGOPT = NORMAL		TCRNGOPT = EXTENDED	
PVCHAR	PVEXEULO	PVEXEUHI	PVEXEULO	PVEXEUHI
Btherm	600	1650	100	1820
Etherm	-150	500	-200	1000
Jtherm	-100	750	-200	1200
Ktherm	0	1100	-100	1370
Rtherm	550	1500	0	1700
RPtherm	550	1500	0	1770
Stherm	550	1500	0	1700
Ttherm	-200	300	-230	400

PVEXEULO (continued)

Defaults for Extended Range PV Parameters When SENSRTYP = RTD, PVTEMP = Degrees C

PVCHAR	PVEXEULO	PVEXEUHI
PtDinRTD	-180	800
PtJisRTD	-180	650
NicklRTD	-45	315
CopprRTD	-20	250

PVEXHIFL

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Extended high range not exceeded)
On (Extended high range exceeded)

PV Extended High Range Violation—Indicates that the PV has exceeded the extended-high range alarm trip point.

PVEXLOFL

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Extended low range not exceeded)
On (Extended low range exceeded)

PV Extended Low Range Violation—Indicates that the PV has exceeded the extended-low range alarm trip point.

PVFL(0)–(2) (DevCtl, DigComp)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (PV is not in the respective state)
On (PV is in the respective state)

PV Flag—Indicates the current PV state as three separate Boolean parameters. PVFL(n) is On when the PV is in state "n" where n is 0, 1, or 2.

PVFL (DigIn, Flag)

Type: **Logical**
Lock: **Oper**
Default: **Off**
PtRes: **HPM**
Range: **Off** [PV = STATETXT(0)]
On [PV = STATETXT(1)]

PV Flag—Represents the current PV state as a Boolean value.

PVFORMAT (RegCtl, RegPV)

Type: **E:VALFORMT**
Lock: **Engr (/PB)**
Default: **D1**
PtRes: **HPM**
Range: 0-**D0** (-XXXXXX**.**)
1-**D1** (-XXXXX**.**X)
2-**D2** (-XXXX**.**XX)
3-**D3** (-XXX**.**XXX)

PV Decimal Point Format—Defines the decimal format that is to be used to display the PV and SP values. It contains up to eight characters including the minus sign and decimal point.

PVHHFL (RegCtl, RegPV)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (High High limit not exceeded)
On (High High limit exceeded)

PV High High Alarm Flag—Indicates whether the PV has exceeded the alarm trip point established by the PVHHTP parameter.

PVHHPR (RegCtl, RegPV)

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

PV High High Alarm Priority—Defines the priority of the PV high high alarm.

Helpful Hint:	PVHHPR configuration requires PVHHTP ≠ NaN.

PVHHTP (AnalgIn, RegPV)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **PVHITP** to **PVEXEUHI, NaN**

PV High High Alarm Trip Point—Defines the PV high high alarm trip point for this point.

Helpful Hint:	PVHHTP configuration requires PVHITP ≠ NaN.

PVHHTP (RegCtl, RegPV)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **PVHITP** to **PVEUHI, NaN**

PV High High Alarm Trip Point—Defines the PV high high alarm trip point for this point.

Helpful Hint:	PVHHTP configuration requires PVHITP≠ NaN.

PVHIFL (RegCtl, RegPV)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (No PV High alarm)
On (High PV alarm)

PV High Alarm Flag—Indicates that the PV has exceeded the alarm trip point established by parameter PVHITP.

PVHIPR (RegCtl, RegPV)

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

PV High Alarm Priority—Defines the priority of the PV high alarm for this point.

Helpful Hint:	PVHIPR configuration requires PVHITP ≠ NaN.

PVHITP

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **PVLOTP** to **PVHHTP, NaN**

PV High Alarm Trip Point—Defines the trip point for the PV high alarm for this point.

PVINIT

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off** (No 1-shot initialization)
On (Initializes the PV filter and the algorithm for a 1-shot single sample time).

PV Initialization Request Flag—Indicates that an initialization request has been made for this point.

PVLLFL (RegCtl, RegPV)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (PV ≥ Low Low alarm trip point)
On (PV ≤ Low Low alarm trip point)

PV Low Low Alarm Flag—Indicates that the PV has exceeded the alarm trip point established by the PVLLTP parameter.

PVLLPR (RegCtl, RegPV)

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

PV Low Low Alarm Priority—Determines the priority of the PV low low alarm for this data point.

Helpful Hint:	PVLLPR configuration requires PVLLTP ≠ NaN.

PVLLTP (AnalgIn, RegPV, PI)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **PVEXEULO** to **PVLOTP, NaN**

PV Low Low Alarm Trip Point—Defines the trip point for the PV low low alarm for this point.

Helpful Hint:	PVLLTP configuration requires PVLOTP ≠ NaN.

PVLLTP (RegCtl)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **PVEULO** to **PVLOTP, NaN**

PV Low Low Alarm Trip Point—Defines the trip point for the PV low low alarm for this point.

Helpful Hint:	PVLLTP configuration requires PVLOTP ≠ NaN.

PVLOFL (RegCtl, RegPV)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (PV ≥ Low alarm trip point)
On (PV ≤ Low alarm trip point)

PV Low Alarm Flag—Indicates that the PV has exceeded the alarm trip point established by parameter PVLOTP.

PVLOPR (RegCtl, RegPV)

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

PV Low Alarm Priority—Defines the priority of the PV low alarm for this point**.**

Helpful Hint:	PVLOPR configuration requires PVLOTP ≠ NaN.

PVLOTP (RegCtl, RegPV)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **PVLLTP** to **PVHITP,**
NaN

PV Low Alarm Trip Point—Defines the trip point for the PV low alarm for this point.

PVNORMAL (DevCtl, DigIn, DigComp)

Type: **E:STATETXT**
Lock: **Supr (Engr to change to/from NONE)**
Default: **N/A**
PtRes: **HPM**
Range: **STATETXT(0) descriptor** (Defaulted to Off for PV State 0)
STATETXT(1) descriptor (Defaulted to On for PV State 1)
STATETXT(2) descriptor (Defaulted to State2 for PV State 2; internally set to $NULL for two-state devices; does not apply to DigIn point)
NONE (No off normal checking)

PV Normal State—Defines the normal state of the PV using the appropriate STATETXT descriptor.

Helpful Hint:	PV normal state text descriptor describes the normal (desired) state, such as Run, Stop, Open, Closed.

PVNORMFL (DevCtl, DigIn, DigComp)

Type: **Logical** **PV Normal State Flag—**Indicates whether the normal state of the PV is active.
Lock: **Supr**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Point is in a state other than the normal state)
On (Normal state is active)

Helpful Hint:	PVNORMFL change requires ALMOPT = Offnorm for Digital Input points, or that PVNORMAL ≠ None for Digital Composite or Device Control points. If set to On, causes text in STATETXT (1) to be used to describe the normal state of the PV, otherwise text in STATETXT (0) is used.

PVP (RegCtl, RegPV)

Type: **Real** **PV in Percent—**Defines the PV as a percentage.
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

PVRAW (AnalgIn)

Type: **Real** **PV Raw Value—**Indicates the raw input value of the PV from the Field Termination Assembly (FTA) before PV characterization is performed. The units of value for the PV are determined by the field sensor type as described below.
Lock: **Operator**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Helpful Hint:	If sensor type is 0.4–2 V, 1- 5 V, 0 - 5 V, PVRAW is in percent; if sensor type is T/C, PVRAW is in microvolts; if sensor type is in RTD, PVRAW is in milliohms; if sensor type is slidewire, PVRAW is in ratio; if sensor type is 0–100 mV, PVRAW is in millivolts.

PVRAW (DigIn)

Type: **Logical** **Raw State of Field Contacts—**Indicates the current state of the field contacts.
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range:

Off Open contacts
On Closed contacts

PVRAW (PI)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

PV Raw Value—Indicates the raw input value of the PV in pulses per second.

PVRAW (STI)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

PV Raw Value—Indicates the raw input value of the PV in % of span based on the transmitter PV after PV characterization (PVCHAR) and DAMPING have been performed. The span of the PV is determined by using LRV as a 0%-point and URV as a 100%-point.

PVRAWHI

Type: **Real**
Lock: **Eng/PB**
Default: **NaN**
PtRes: **HPM**
Range: **PVRAWLO** to **100** for a 0_100_mv input (microvolts)
PVRAWLO to **1** for a slidewire input (ratio)

PV Raw High Range—Defines the high end of the normal operating range for the raw PV value (PVRAW). For a slidewire input, the units are ratio, for a 0_100_MV input, the units are in millivolts.

PVRAWLO

Type: **Real**
Lock: **Eng/PB**
Default: **NaN**
PtRes: **HPM**
Range: **0–PVRAWHI** (0 to 100 microvolts for a 0_100 mv input, or 0 to 1 ratio for a slidewire input)

PV Raw Low Range—Defines the low end of the normal operating range for the raw PV value (PVRAW).

PVROCNFL (RegCtl, RegPV)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (No PV negative rate-of-change alarm)
On (PV negative rate-of-change alarm)

PV Negative Rate-of-Change Alarm Flag—Indicates that the PV negative rate-of-change has exceeded the value established by the PVROCNTP parameter.

PVROCNPR (RegCtl, RegPV)

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**

PV Negative Rate-of-Change Alarm Priority—Defines the priority of the PV negative rate-of-change alarm for this point.

Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

Helpful Hint:	PVROCNPR configuration requires PVROCNTP ≠ NaN.

PVROCNTP (RegCtl, RegPV)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**

PV Negative Rate-of-Change Trip Point—Defines the trip point for the PV negative rate-of-change alarm for this point. Operation is the same as for PVROCPTP except for the direction of change.

The maximum rate of change value must be less than the absolute value of:
$(PVEUHI\text{-}PVEULO) * \frac{60}{8}$.

Range: $\geq$ **0.0,**
NaN

Helpful Hint:	For RegPV points, RegCtl points, and points in a HLAI or LLAI, the maximum rate of change is one step away from PVEXEUHI to PVEXEULO in eight seconds; therefore, the maximum rate of change is (PVEXEUHI - PVEXEULO) * 7.5 units/minute. For SENSRTYP = RTD, the maximum is [800 - (-180)] * 7.5, which is 7350.

PVROCPFL (RegCtl, RegPV)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**

PV Positive Rate-of-Change Alarm Flag—Indicates that the positive rate-of-change of the PV has exceeded the value established by the PVROCPTP parameter.

Range: **Off** (No PV positive rate-of-change alarm)
On (PV positive rate-of-change alarm)

PVROCPPR (RegCtl, RegPV)

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**

PV Positive Rate-of-Change Alarm Priority—Defines the priority of the positive rate-of-change PV alarm for this point.

Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

Helpful Hint:	PVROCPPR configuration requires PVROCPTP ≠ NaN.

PVROCPTP (RegCtl, RegPV)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**

PV Positive Rate-of-Change Trip Point—Defines the positive PV rate-of-change limit in engineering units/minute for this point; for example 25 degrees per minute. The PV value is checked every four seconds. The rate of change alarm trips if the PV rate-of-change value is exceeded for two successive scans. The alarm is reset if the PV rate-of-change falls below the rate of change value for two successive scans.

The maximum rate of change value must be less than the absolute value of:
(PVEUHI-PVEULO) * $\frac{60}{8}$.

Range: **≥ 0.0,**
NaN

Helpful Hint:	For RegPV points, RegCtl points, and points in HLAI or LLAI, the maximum rate of change is one step away from PVEXEUHI to PVEXEULO in eight seconds; therefore, the maximum rate of change is: (PVEXEUHI - PVEXEULO) * 7.5 units/minute. Example: for SENSRTYP = RTD, the maximum is: [800 - (-180)] *7.5, which = 7350.

PVSGCHTP (RegCtl, RegPV)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **≥ 0.0,**
NaN

PV Significant Change Alarm Trip Point—Defines the alarm trip point for an increment of change that occurs between configured PVHITP and PVHHTP or PVLOTP and PVLLTP alarms. For example, consider a temperature point with limits of PVHITP = 800 degrees, PVHHTP = 850, and PVSGCHTP = 10. When the temperature rises to 800 degrees, the PVHITP alarm is annunciated, and should the temperature continue to increase, the alarm is annunciated again when the temperature reaches 810 degrees, 820 degrees, and so on. This allows the alarm to be reannunciated to remind the operator of the existence of an alarm condition.

PVSOURCE (RegCtl, RegPV)

Type: **E:PVSOURCE**
Lock: **Oper**
Default: **Auto**
PtRes: **HPM**
Range: 0-**Sub** (Value is provided by a sequence program)
1-**Man** (PV is supplied by operator or program)
2-**Auto** (Field wiring or memory fetch supplies PV)
3-**Track** (PV tracks OP (DigComp points only))

PV Source—Defines the source of the PV input to this data point. The PV goes to bad when PVSOURCE is switched from Man to Sub.

Helpful Hint:	PVSOURCE change by an operator requires PVSRCOPT = All and DITYPE = Latched if PNTTYPE = DigIn.

PVSRCOPT (RegCtl, RegPV)

Type: **E:PVSRCOPT**
Lock: **Eng/PB**
Default: **OnlyAuto**
PtRes: **HPM**
Range: 0-**OnlyAuto** (PV source selection is not available and field wiring or memory fetch supplies PV)
1-**All** (PV is provided by an operator, by a sequence program, or by field wiring)

PV Source Option—Defines the PV source options available in this data point.

Helpful Hint:	PVSRCOPT change by an operator requires DITYPE = Latched if PNTTYPE= DigIn.

PVSTATES(0)–(4) (DevCtl, DigComp)

Type: **String_8**
Lock: **View**
Default: **N/A**
PtRes: **NIM**
Range: **N/A**

PV State Descriptors—The PV state descriptors contain the text that describes the five (0-4) possible states of a DigComp or DevCtl PV. The descriptors are set equal to whatever is configured in BADPVTXT and MOVPVTXT during HPM Box configuration and in STATETXT(0), STATETXT(1), and STATETXT(2) during point configuration (if PVTXOPT = ON).

Helpful Hint: PVSTATES, if accessed by Control Language programs, obey the following rules:

a. PVSTATES (0) = STATETXT (0)
b. PVSTATES (1) = STATETXT (1)
c. PVSTATES (2) = BADPVTXT
d. PVSTATES (3) = MOVPVTXT
e. PVSTATES (4) = STATETXT (2); does not apply unless NOSTATES = 3

PVSTS (RegCtl, RegPV)

Type: **E:PVVALST** **Status Of PV Input Value—**Defines the current status of the PV value.
Lock: **View**
Default: **Bad**
PtRes: **HPM**
Range: 0-**Bad** (Value is bad and replaced with NaN. For an STI point, value can be set to Bad based on transmitter gross status.)
1-**Uncertn** (Status of the value is uncertain)
2-**Normal** (Value is good)

PVTEMP

Type: **E:TEMPTURE** **PV Temperature Scale—**Defines the temperature scale to be used in characterizing the PV input.
Lock: **PtBld**
Default: **Degrees C**
PtRes: **HPM**
Range: 0-**Degrees C** (Celsius)
1-**Degrees F** (Fahrenheit)
2-**Degrees R** (Rankin)
3-**Degrees K** (Kelvin)

Helpful Hint: PVTEMP is to be configured when PVCHAR = TC or RTD.

PVTRACK (Pid)

Type: **E:TRACKING** **PV Tracking Option—**Defines whether SP is to be set equal to PV.
Lock: **Eng/PB**
Default: **NoTrack**
PtRes: **HPM**
Range: 0-**NoTrack** (SP is never set equal to PV)
1-**Track** (Man mode or initialization causes SP to track PV)

Helpful Hint: SP is set equal to PV if PVTRACK = Track and the point is:
a. in manual mode
b. being initialized from a secondary
c. undergoing 1-shot initialization during the first sample time after becoming active.

PVTV

Type:	**Real**	**PV Target Value in Engineering Units—**Defines the target value of the PV in engineering units.
Lock:	**Configurable**	
Default:	**NaN**	
PtRes:	**HPM**	
Range:	**PVEXEULO** to **PVEXEUHI,**	

PVTVP

Type:	**Real**	**PV Target Value in Percent—**Indicates the target value of the PV in percent.
Lock:	**View**	
Default:	**NaN**	
PtRes:	**HPM**	
Range:	**≥ 0.0%, NaN**	

PVTXTOPT (DevCtl, DigComp)

Type:	**Logical**	**PV Text Option—**Indicates whether the BADPVTXT and MOVPVTXT parameters are configured for this point, or if the default from the box data point should be used.
Lock:	**PtBld**	
Default:	**Off**	
PtRes:	**NIM**	
Range:	**Off** (The parameters are not configured for this point) **On** (The parameters are configured for this point)	

-Q-

Q

Type: **Real**
Lock: **View**
Default: **1.0**
PtRes: **HPM**
Range: $\geq$ **0.0,**
NaN

Steam Quality Factor Input—Indicates the measured actual steam quality factor.

QSTS (FlowComp)

Type: **E:PVVALST**
Lock: **View**
Default: **Normal**
PtRes: **HPM**
Range: 0-**Bad** (Value is bad and replaced by NaN)
1-**Uncertn** (Status of the value is uncertain)
2-**Normal** (Value is good)

Q Input Value Status—Indicates the status of the steam quality-factor input.

-R-

R1(1)–(24), R2(1)–(24) (Logic)

Type: **E:$PMMLGPM**
Lock: **PtBld**
Default: **L1**
PtRes: **HPM**
Range: 32··47-**L1...L12** (Values from Input Connections)
48··51-**NN1...NN8** (Local Numerics)

Real Inputs 1 & 2—Defines the sources that provide the input values for the R1 and R2 inputs of certain logic blocks.

Helpful Hint:	R1 configuration requires LOCALGID = EQ, NE, GT, GE, LT, LE, or CHECKBAD.

RAISDSTN

Type: **Universal Ent.Prm**
Lock: **PtBld**
Default: **Null**
PtRes: **HPM**
Range: **ONPULSE**
OFFPULSE

Raise Output Pulse Destination—Defines the destination of the Raise output pulse. RAISDSTN must point to parameters ONPULSE or OFFPULSE of a DigOut data point.

RAISETIM

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A,**
NaN

Raise Output Pulse Time—Indicates the Raise output pulse time in seconds. It is clamped to MAXPULSE or CYCLETIM, whichever is lower. If the value of RAISETIM is smaller than MINPULSE, no pulse is issued.

RAISRATE

Type: **Real**
Lock: **Supr**
Default: **100.0% per sec.**
PtRes: **HPM**
Range: **> 0.0% per second**

Raise OP Stroke Rate in Percent/Second

RAMPTIME

Type: **Real**
Lock: **Oper**
Default: **0.0**
PtRes: **HPM**
Range: **≥ 0.0**

Ramp Time in Minutes

Helpful Hint:	RAMPTIME change by an operator requires SPOPT = Tv. The minimum value is clamped to TS, the point sample time in minutes, while the maximum value is clamped to 32767*TS. To change the RAMPTIME from a logic point requires Node=Auto and Mode Attribute=Program.

RARWSTS (RegCtl)

Type: **E:WINDUP** **Remote Anti-Reset Windup Status**
Lock: **View**
Default: **Normal**
PtRes: **HPM**
Range: 0-**Normal** (Free to move in either direction)
1-**Hi** (Free to move lower)
2-**Lo** (Free to move higher)
3-**HiLo** (Not free to move in any direction)

Helpful Hint:	RARWSTS applies only if RCASOPT = Spc, Ddc, or DdcRsp.

RATE1 (AutoMan, MulDiv, RegCtl Summer)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**

Rate at Which Bias Ramps Down—Rate at which the bias (B) ramps down from the initialization value to the last value entered by the operator. If a positive value is entered, Rate1 determines the ramp down rate of the internal bias value (BI). If 0 is entered, the ramp down of B is disabled. If NaN is entered, the internal bias does not decay, but instantaneously changes to 0 and will cause a bump in the output.

Range: **≥ 0.0 EU's per minute, NaN**

RATE1 (RatioCtl)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **≥ 0.0, NaN**

Internal Bias Ramps Down Rate—Rate in EUs per minute at which the internal bias (BI) ramps down from the initialization value to the last value entered by the operator. If BIAS = NaN, initialization for the primary is determined through back calculation. If 0 is entered, the ramp down of B is disabled.

RATE1–12 (RampSoak)

Type: **Real** **Ramp Rate for Ramp Soak Segments 1–12**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: Negative values are accepted to allow ramping down.

RATIO (Pid)

Type: **Real** **Ratio—**Defines the ratio value by which the SP is multiplied.
Lock: **Oper**
Default: **1.0**
PtRes: **HPM**
Range: **RTLOLM** to **RTHILM**

RBOPT (Pid)

Type: **E:RBOPT**
Lock: **Eng/PB**
Default: **NoRatBi**
PtRes: **HPM**

Ratio/Bias Option—Defines the type of ratio/bias option to be used for this algorithm. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of bias and ratio options.

Range: 0-**NoRatBi** (No ratio/bias is used to calculate the SP)
1-**FixRatBi** (Fixed ratio (R) and fixed bias (B) are used)
2-**AutoRat** (R is back-calculated during initialization)
3-**AutoBi** (B is back-calculated during initialization)

Helpful Hint:	RBOPT applies to only PID-type RegCtl algorithms.

RCASOPT (AnalgOut)

Type: **E:$RCASOPT**
Lock: **Eng/PB**
Default: **None**
PtRes: **HPM**

Remote Cascade Option—Defines whether the AM is to provide the output value for this data point. This option is available only when the point has been configured as a Full point.

Range: 0-**None** (No cascade mode of any type is allowed)
2-**Ddc** (Direct Digital Control; in cascade mode, AM point controls this point's OP)

RCASOPT (AutoMan, IncrSum, ORSel, Switch)

Type: **E:$RCASOPT**
Lock: **Eng/PB**
Default: **None**
PtRes: **HPM**

Remote Cascade Option—Defines the type of remote cascade mode to be used. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: 0-**None** (Only local cascade mode is allowed)
2-**Ddc** (In cascade mode, AM point provides the output OP for this data point)

RCASOPT (Pid)

Type: **E:$RCASOPT**
Lock: **Eng/PB**
Default: **None**
PtRes: **HPM**

Remote Cascade Option—Defines the type of remote cascade mode to be used. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: 0-**None** (Local cascade is the only valid cascade)
1-**Spc** (AM writes to SP within SP limits)
2-**Ddc** (AM writes to OP. No OP limits)
3-**Rsp** (AM writes to SP when this point is in Auto and is being initialized by its secondary. SP limits are applied)
4-**DdcRsp** (AM does Ddc and Rsp functions)

Helpful Hint:	MODE for a point cannot be changed to Cascade by the operator or program if a. parameter RCASOPT is configured to Ddc, Spc, or DdcRsp b. the AM strategy has not yet stored to MODE, SP, or OP. Even if PVTRACK = Track, PV tracking is not performed in auto mode with INITMAN = On if RCASOPT = Rsp.

RCASOPT (RegCtl)

Type: **E:$RCASOPT**
Lock: **Eng/PB**
Default: **None**
PtRes: **HPM**

Remote Cascade Option—Defines the type of remote cascade mode to be used. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: 0-**None** (Only local cascade mode is allowed)
1-**Spc** (In cascade mode, AM point provides the SP for this point)
2-**DDC** (In cascade mode, AM point provides the OP for this point)
3-**Rsp** (In Auto mode with INITMAN = ON, the AM point provides the SP for this point)
4-**DDCRsp** (In cascade mode, AM point provides the OP for this point)

RCASSHED (RegCtl)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**

Remote Cascade Shed—Indicates whether the mode has shed from the Cas mode to the preconfigured backup mode.

Range: **Off** (No mode shed)
On (Mode has shed to the preconfigured backup mode)

Helpful Hint:	RCASSHED applies only if RCASOPT = Spc, Ddc, or DdcRsp.

RDNHWREV

Type: **String_2**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Hexadecimal Characters 00 to FF**

HPMM Control Daughter Card Revision

REDTAG (RegCtl)

Type: **E:REDTAG**
Lock: **Sup/Eng**
Default: **Off**
PtRes: **HPM**

Red Tag State—Allows the user to set the point as being "out of service," indicating that this point or the associated control loop needs repair or is being repaired. Once this point is put in the red tag condition, the output OP is frozen at the last value or state.

Range: 0-**Off** (Data point is in service. Point's OP is not frozen)
1-**On** (Data point is out of service = point's OP is frozen)

Helpful Hint:	REDTAG change requires MODE = Man and MODATTR = Oper. Once a point is red tagged, parameters MODE, MODATTR, and OP (output) cannot be changed. In addition, for a RegCtl point, ESWENBST cannot be changed.

RELREV (HPM)

Type: **String_1**
Lock: **View**
Default: **00** (Hex.)
PtRes: **HPM**

Overall Software Release Revision Code —

Range: N/A

RELVERS (HPM)

Type: **String_1**
Lock: **View**
Default: **00** (Hex.)
PtRes: **HPM**

Overall Software Release Version Code —

Range: N/A

REMSOAKT (RampSoak)

Type: **Real**
Lock: **Oper**
Default: **0.0**
PtRes: **HPM**
Range: **0.0** to **120.0 minutes**

Remaining Soak Time—Indicates the amount of time remaining in the current soak segment.

RESETFL (DevCtl, DigComp)

Type: **Logical**
Lock: **Oper**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On (Storing to this parameter resets maintenance statistics)

Reset Maintenance Statistics Flag—Used to reset maintenance statistics.

Helpful Hint:	This parameter can be reset by the operator only while it is red tagged. A program may reset at anytime.

RESETFL (DigIn)

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off** (No Reset command)
On (Reset command is issued to the accumulator)

Reset Flag—Resets the accumulator to zero when the command flag transitions from the Off to the On state.

Helpful Hint:	RESETFL change requires DITYPE = Accum

RESETFL (Timer)

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

Reset Timer Command Flag—Resets the total when this flag changes from Off to On.

RESETFL (Totalizr)

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

Reset Totalizer Command Flag—Resets the total RESETVAL when this flag changes from Off to On.

RESETVAL (DigIn)

Type: **Integer**
Lock: **Oper**
Default: **0**
PtRes: **HPM**
Range: **0–32767**

Accumulator Reset Value—Value that is preset in the accumulator. Value can then be incremented or decremented depending on the COUNTDWN parameter.

RESETVAL (Totalizr)

Type: **Real**
Lock: **Oper**
Default: **0.0**
PtRes: **HPM**
Range: **N/A**

Reset Value—Value used for presetting the value to be totaled.

RESTART (ProcMod)

Type: **E:RESTART**
Lock: **View**
Default: **None**
PtRes: **HPM**

Process Module Restart State—Indicates the type of restart last performed by the process module. This value will be set to a value other than “None” until the first preemption point.

Helpful Hint:	RESTART can be used to determine if the startup was caused by failover. Following failover, RESTART takes on the value “Failover.” This value remains until the first preemption point after which it returns to “None.”

Range: 0-**None** (Has not been restarted)
1-**Failover** (Running for the first time after a failover)
3-**Warm** (Running for the first cycle after a warm start)
4-**Cold** (Running for the first cycle after a cold start, or a power up to Run)
5-**PTACTVN** (Running for the first cycle following the transition from Off state to Run state)

RFB (PidErfb)

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **$\geq$ 0.0**

Reset Feedback Input in Percent—Indicates the PV value of another data point that is receiving its setpoint from this data point.

RG (FlowComp)

Type: **Real**
Lock: **Supr**
Default: **1.0(molecular weight)**
PtRes: **HPM**
Range: **N/A**

Reference Specific Gravity—Defines the reference specific gravity or reference molecular weight, in the same engineering units as G (measured or calculated specific gravity or molecular weight).

RINITREQ (RegCtl)

Type: **Logical**
Lock: **View**
Default: **On for AnalgOut, Off for RegCtl**
PtRes: **HPM**
Range: **Off** (No request)
On (Request has been made)

Remote Initialization Request—Indicates whether an initialization request has been made.

Helpful Hint:	RINITREQ does not apply if RCASOPT = None.

RINITVAL (RegCtl)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A, NaN**

Remote Initialization Value

Helpful Hint:	RINITVAL does not apply if RCASOPT = None.

RJRAW(1)–(168)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Reference Junction's Raw Count in µVolts

RJTEMP (LLAI)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Reference Junction Temperature—Displays the current temperatures (°C) of the reference junction sensor. This value is updated only if the thermocouple input is configured.

RNGCODE3 (ProcMod)

Type: **Blind Record** **RNGCODE3—** .
Lock: **Oper**
Default: **N/A**
PtRes: **HPM**
Range:

RP (FlowComp)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **N/A**

Reference Pressure—RP is the reference pressure input and it is in the same Engineering Unit as the P (measured or actual gage pressure) input.

RP

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **0.01** to **100.0**

Minimum Pulse Time Ratio

RQ (FlowComp)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **N/A**

Reference Steam Quality Factor—Defines the reference steam quality factor which is in the same units as the Q (measured actual steam quality) input.

RSPBGP$$ (RampSoak)

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: $\geq$ **0.0**

Ramp/Soak Percent Bar Graph Parameter—If in a ramp sequence, the value of the next soak percent is displayed. If in a soak sequence, the value of parameter REMSOAKT as a percent of total soak time is displayed.

RSTROPT (ProcMod)

Type: **E:$RSTROPT**
Lock: **Eng**
Default: **Off**
PtRes: **HPM**
Range: 0-**Off** (Sequence is waiting for the operator command to start)
1-**Restart** (Sequence is to be restarted from the beginning)
2-**Stop** (Sequence positions to beginning of the last preemption following a Warm, Cold, or Power Up Restart and waits for the operator to start

Restart Option—Defines how the sequence program is to be started following an Idle to Run, or power up to Run transition or a warm restart.

RT (FlowComp)

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **N/A**

Reference Temperature—RT is the reference temperature input and is in the same Engineering Unit as the T (measured actual temperature) input.

RT

Type: **Real**
Lock: **Supr**
Default: **1.0**
PtRes: **HPM**
Range: **0.01** to **100.0**

Deadtime Ratio

RTHILM (Pid)

Type: **Real**
Lock: **Supr**
Default: **100.0**
PtRes: **HPM**
Range: **RTLOLM** to **100.0,**
NaN

Ratio High Limit

Helpful Hint:	Entering NaN forces RTHILM to its extreme value (100.0%).

RTLOLM (Pid)

Type: **Real**
Lock: **Supr**
Default: **0.01**
PtRes: **HPM**
Range: **0.01** to **RTHILM,**
NaN

Ratio Low Limit

Helpful Hint:	Entering NaN forces RTLOLM to its extreme value (0.01).

RUNSTATE (ProcMod)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Process Module point is not in the RUN state)
On (Process Module point is in the RUN state)

Run State—Indicates that the point is in the RUN sequence execution state.

RV (Timer)

Type:	**Integer**	**Remaining Time—**Indicates the amount of time remaining (in seconds or minutes) that the timer is to run.
Lock:	**View**	
Default:	**0**	
PtRes:	**HPM**	
Range:	**>0**	

Helpful Hint:	RV represents remaining time computed as SP - PV. If SP = 0, RV is always 0.

RX (FlowComp)

Type:	**Real**	**Reference Steam Compressibility—**Defines the reference steam compressibility, and is in the same engineering units as the X (measured actual steam compressibility).
Lock:	**Supr**	
Default:	**1.0**	
PtRes:	**HPM**	
Range:	**Anything except NaN**	

-S-

S0BOXCLR, S1BOXCLR, S2BOXCLR

Type: **E:BOXCOLOR**
Lock: **View**
Default: **N/A**
PtRes: **NIM**
Range: **Red**
Green
White
Black
Cyan
Yellow
Blue
Magenta

State Box Color—Used only for US displays; corresponds to the box colors configured using S0BOXCLR - BOXCLR(0), S1BOXCLR - BOXCLR(1), S2BOXCLR - BOXCLR(2).

S1 (PidErfb)

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off** (CV value is not replaced)
On (CV value is replaced by the tracking value)

Tracking Switch—Determines whether the CV value of this data point is replaced by the tracking value**.**

S1 (STI)

Type: **String_127**
Lock: **View**
Default: **Blank**
PtRes: **HPM**
Range: **N/A, Blank**

Transmitter Status—Indicates the current status of the smart transmitter associated with this STI point. Transmitter status consists of
- Transmitter scratch pads 1, 2, 3 & 4
- Detailed transmitter status
- List of parameters whose values are not the same in both the STI IOP database and the transmitter's database. (Parameters are mismatched.)

S1, S2 (RampSoak)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

Mark 1 and Mark 2 Flags—These flags are used to indicate to other data points that a specified time has elapsed from the beginning of a specified ramp or soak segment. Refer to the *HPM Control Functions and Algorithms* manual for detailed information.

S1(1)–(24) through S4(1)–(24) (Logic)

Type: **E:$PMMLGPM**
Lock: **PtBld**
Default: **L1**
PtRes: **HPM**
Range: **FL1...FL12** (Input source is a local flag; they can be either On or Off)
SO1...SO24 (Input source is the status output (SOn) from another logic block.)
L1...L12 (Input source is the configured input connection; they can be either On or Off)

Status Inputs 1-4—Defines the input source for each of the S1-S4 inputs to the logic block.

S1–S4 (Switch)

Type: **Logical**
Lock: **Oper**
Default: **On**
PtRes: **HPM**
Range: **Off**
On (Respective input has been selected)

Select X1-X4 Request Flag—Indicate whether the respective input (X1-X4) has been selected as the input to this algorithm.

S1BGNTIM, S2BGNTIM (RampSoak)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: $\geq$ **0.0 to 120 minutes**

Mark 1 and Mark 2 Begin Times—Times at which Mark Function Flag S1 or S2 is turned on. Refer to the *HPM Control Functions and Algorithms* manual for detailed information.

S1ENDTIM, S2ENDTIM (RampSoak)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: $\geq$ **0.0 to 120 minutes**

Mark 1 and Mark 2 End Time—Times at which Mark Function Flags S1 and S2 are turned Off. Refer to the *HPM Control Functions and Algorithms* manual for detailed information.

S1REV(1)–(24) through S3REV(1)–(24) (Logic)

Type: **Logical**
Lock: **PtBld**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Input is direct)
On (Input is reversed)

S1, S2, S3 Inputs Reversed—Allows the user to selectively reverse (invert) any of the inputs to a logic block.

Helpful Hint:	Reversed (inverted) inputs apply only to LOGALGID = And, Or, Nand, and Nor.

S1SEGID, S2SEGID (RampSoak)

Type: **E:CURSEGID**
Lock: **Supr**
Default: **Ramp1**
PtRes: **HPM**
Range: 0-R**amp1** 1-**Soak1**
2-R**amp2** 3-**Soak2**
: and :
20-**Ramp11** 21-**Soak11**
22-**Ramp12** 23-**Soak12**

Mark 1 and Mark 2 Segment Identifiers—Refer to the *HPM Control Functions and Algorithms* manual for detailed information.

SAFEOP

Type: **Real**
Lock: **Engr**
Default: **N/A**
PtRes: **HPM**
Range: **-6.9** to **106.9%, NaN**

Safe Operation For Safety Shutdown—Defines the safe output value (OP) for a point when the SHUTDOWN parameter is set to On.

SAFOPCMD (PosProp, PIDPosPr)

Type: **E:$SFOPCMD**
Lock: **Engr**
Default: **Idle**
PtRes: **HPM**
Range: 0-**IDLE** (Output does not change)
1-**RAISE** (Output is raised)
2-**LOWER** (Output is lowered)

Safe OP Command—Defines the Safe OP state for position proportional and PID position proportional.

SCANPER (HPM Box)

Type: **Real**
Lock: **PtBld**
Default: **1.0**
PtRes: **HPM**
Range: **.25** seconds
.5 seconds
1.0 seconds

SI Data Scan Period—Defines the period that the HPMM Control Processor scans serial interface data that is mapped to the Array point.

SCANPRI (Array)

Type: **E:$SCANPRI**
Lock: **PtBld**
Default: **Low**
PtRes: **HPM**
Range: **Low** (Scan at low priority)
High (Scan at high priority)

SI Data FTA Scan Priority—Indicates which scan priority the serial interface FTA is using when reading data from the serial link.

SCANRATE (HPM Box)

Type: **E:$PMMSNRT**
Lock: **PtBld**
Default: **Reg1Log1**
PtRes: **HPM**

Scan Rate—Defines the number of times that all slots of a particular type are scanned and processed. Refer to the *HPM Control Functions and Algorithms* manual for information on how to determine the processing capacity of the HPM. During the load of the HPM Box Data point, the point mix (number of points and box variables) and the scan rate are written to the HPMM by the store of the SCANRATE parameter.

Range:

	RegCtl & RegPV Scan Freq	*Logic, DigComp & DevCtl Scan Freq*	*ProcMod Scan Freq*
0-N**ull**	- - - -	- - - -	- - - -
1-R**eg1Log1**	1 second	1 second	1 second
2-R**eg1Log2**	1 second	1/2 second	1 second
3-R**eg1Log4**	1 second	1/4 second	1 second
5-R**eg2Log2**	1/2 second	1/2 second	1 second
6-R**eg2Log4**	1/2 second	1/4 second	1 second
8-R**eg4Log4**	1/4 second	1/4 second	1 second

CAUTION

If a new point mix or a new SCANRATE is loaded from the DEB, the following items should be noted:

a. Before making changes to the point mix or SCANRATE, any configured points being removed due to a reduction in the point mix should first be deleted from the system.

b. Any other currently configured points are preserved in the new point mix (the point database is not defaulted).

c. If the SCANRATE or any part of the point mix is rejected by the HPMM then the HPMM database remains unchanged; the SCANRATE and the point mix also remain unchanged.

d. If the SCANRATE and the point mix are equivalent to the previous values, then the HPMM database remains unchanged.

SCHSTS

Type: **Logical**
Lock: **View**
Default: **OK**
PtRes: **HPM**

Schedule Status—Indicates the status of the schedule configuration option processing (for example, before/after relationship).

Range: **OK (**the point is correctly assigned to the desired scan cycle or before/after another point with the same status)
Incomplete (the point did not complete loading to the point where the proper scan cycle or before/after point could be determined.)
error (the point could not be placed on the desired scan cycle or before/after the desired point)
Alarm (the schedule configuration of the point was violated after the configuration of the point was complete and its status was Ok.)

Helpful Hint: The point cannot be made active if SCHSTS = Incomplete or Error.

SEALOPT (DevCtl, DigComp)

Type: **E:$SEALOPT** **Seal-in Circuit Option—**Configures the seal-in circuit option.
Lock: **Eng/PB**
Default: **None**
PtRes: **HPM**
Range: 0-**None** (Sealin is not configured)
1-**Sealin** (Sealin is configured)

SECOND (HPM Box)

Type: **Integer** **Current Second**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **0** to **59**

SECSYNC

Type: **E:$SECSYNC** **Secondary Synchronization Status**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Synched** (Modules are synchronized)
NoSynch (Modules are out of synchronization)

SECVAR (DevCtl)

Type: **Real** **Secondary Variable—**The analog feedback, normally the motor current or flow.
Lock: **View**
Default: **0.0**
PtRes: **HPM**
Range: **Real Numbers including NaN**

SECVAR (STI)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A, NaN**

Secondary Variable—Displays the value of the secondary variable of the smart transmitter as follows:
Pressure transmitter—temperature of the transmitter
Temperature transmitter—cold junction temperature
Flow transmitter—totalized value.

SEGTOT (GenLin)

Type: **Integer** **Total Number of Segments—**Defines the total number of segments in the curve.
Lock: **Supr**
Default: **1**
PtRes: **HPM**
Range: **1** to **12**

SEGTYPE (RampSoak)

Type: **E:SEGTYPE**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: 0-**Ramp** (Ramp segment)
1-**Soak** (Soak segment)

Segment Type—Indicates the current segment being executed by the RegCtl point.

SELINP (HiLoAvg, MidOf3)

Type: **E:PINP**
Lock: **View**
Default: **SelectP1**
PtRes: **HPM**
Range: 1-**SelectP1** (HiLoAvg and MidOf3 algorithms)
2-**SelectP2** (HiLoAvg and MidOf3 algorithms)
3-**SelectP3** (HiLoAvg and MidOf3 algorithms)
4-**SelectP4** (Only HiLoAvg algorithm)
5-**SelectP5** (Only HiLoAvg algorithm)
6-**SelectP6** (Only HiLoAvg algorithm)

Selected Input—Indicates the selected input for the algorithm.

SELXINP (ORSel, Switch)

Type: **E:XINP**
Lock: **View (for ORSel)** **Oper (for Switch)**
Default: **SelectX1**
PtRes: **HPM**
Range: 1-**SelectX1**
2-**SelectX2**
3-**SelectX3**
4-**SelectX4**

Selected X Input—For the ORSel algorithm, this parameter indicates the inputs to the algorithm that have not been bypassed by the BYPASS1-BYPASS4 parameters. For the Switch algorithm, this parameter allows the operator to specify the input (X1-X4) to the algorithm. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

SENSRTYP (HLAI & LLAI)

Type: **E:$SENSRTY**
Lock: **PtBld**
Default: **1_5_V**
PtRes: **HPM**
Range: 0-**1_5_V** (1 to 5 volts)
1-**0_5_V** (0 to 5 volts)
2-**0_100_mV** (0 to 100 millivolts)
3-**Thermcpl** (Thermocouple)
4-**RTD** (Resistance Temperature Device)
5-**P4_2_V** (0.4 to 2 volts)
6-**Slidwire** (Slidewire Resistance Device)

Sensor Type—Defines the type of field sensor connected to the Field Termination Assembly (FTA). 0_100_mV, Thermocouple, and RTD sensor types do not apply for HLAI. P4_2_V and slidewire sensor types do not apply for LLAI. Refer to PVCHAR for more information.

SENSRTYP (LLMUX, RHMUX)

Type: **E:$SENSRTY**
Lock: **PtBld**
Default: **0 - 100 mV**
PtRes: **HPM**
Range: 2-**0_100_mV** (0 to 100 millivolts)
3-**Thermcpl** (Thermocouple)
4-**RTD** (Resistance Temperature Device) NOTE: RTD is not supported by RHMUX.

Sensor Type—Defines the type of field sensor connected to the Field Termination Assembly (FTA). Refer to PVCHAR for more information.

SENSRTYP (STI)

Type: **E:$SENSRTY**
Lock: **PtBld**
Default: **Spt_Dp**
PtRes: **HPM**
Range: 8-S**PT_DP** (ST3000—differential pressure)
9-S**PT_GP** (ST3000—gauge pressure)
10-**SPT_AP** (ST3000—absolute pressure)
11-**STT** (STT3000—temperature)
12-**SFM** (MagneW 3000—magnetic flow and most Multivariable transmitters)

Sensor Type—Defines the Smart Transmitter type. Refer to PVCHAR for more information. Note that SENSRTYP must match the FTA. The point status is set to SOFTFAIL if a mismatch occurs.

Helpful Hint: For multivariable transmitters, refer to the transmitter manual for the default value of the specific device.

SEQERR (ProcMod)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **0** (No error)
1-100 (Not used)

Sequence Error—Indicates that a sequence error or failure was detected. A code is displayed to indicate the nature of the error or failure. When an error is detected, the sequence execution state is changed to ERROR; when a failure is detected, the execution state is changed to FAIL.

Error Codes

101 (Not used)
102 (Array index error)
103 (Illegal IMD code)
104 (Illegal variable/operator code)
105 (Interpreter stack overflow)
106 (GOTO destination error)
107 (Key level error)
108 (Configuration mismatch error)
109 (I/O Link prefetch overflow)
110 (Subroutine nesting level error)
111 (Illegal value error)
112 (Fail statement)
113 (IOL-Prefetch buffer full)
114 (IOL-Poststore buffer full)
115 (UCN-Prefetch buffer full)
116 ((UCN-Postore buffer full)
117-164 (Not used)

Failure Codes

165 (Sequence has been halted by the operator)
166 (Sequence jumped to an abnormal condition handler which was not enabled)
167 (Not used)
168 (Timeout condition occurred on WAIT statement)
169 (An attempt was made to start a sequence that has not been loaded)
170 (Communication error in READ/WRITE statement)
171 (Communication error detected during I/O Link access. This error is also generated for all post-store problems)
172 (Range Error)
173 (An attempt was made to write to a point that was not in the proper mode)
174 (Interlock error)
175-255 (Not used)

SEQEXEC (ProcMod)

Type: **E:SEQEXEC**
Lock: **View**
Default: **NL**
PtRes: **HPM**
Range: 0-**NL** (Not Loaded)
1-**DLL** (Down-line loading is in progress)
2-**Loaded** (Sequence has been loaded into the process module)
3-**End** (Sequence has stopped because it has run to completion)
4-**Pause** (Sequence has stopped because of a PAUSE statement, or after each step is executed while in the SnglStep sequence execution mode.)
5-**Fail** (Sequence has stopped because a sequence failure was detected)
6-**Error** (Sequence has stopped because a sequence error was detected)
7-**Run** (Sequence is running in the process module)

Sequence Execution State—Indicates the current execution state of the sequence program that is executing in the process module. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of the execution states.

SEQMODE (ProcMod)

Type: **E:SEQMODE**
Lock: **Determined by CNTLLOCK parameter**
Default: **Auto**
PtRes: **HPM**

Sequence Execution Mode—Defines the manner in which the sequence is executed.

Range: 0-**Auto** (Normal mode of sequence operation. Sequence runs from beginning to end without operator intervention.)
1-**SemiAuto** (Sequence stops at all PAUSE statements in the sequence. Operator action is required to restart the sequence.)
2-**SnglStep** (Sequence is executed one step at a time, and operator action is required to resume execution. This mode is normally used for debugging.)

SEQNAME (ProcMod)

Type: **String_8**
Lock: **View**
Default: **Blank**
PtRes: **HPM**
Range: **N/A**

Sequence Name—Defines the name of the CL sequence that currently resides in the process module.

SEQOBJSZ (ProcMod)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **>0**

Sequence Program Size—Indicates the number of memory blocks used by the sequence currently loaded in this process module.

SEQPR (ProcMod)

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**

Sequence Priority—Defines the alarm priority for Process Module points. Note that even when the Sequence priority is Journal, the alarm indicators still appear on the Group and Detail displays.

Range: **Emergncy** (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
JnlPrint (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

SEQPRGSZ (HPM Box)

Type: **Integer**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

Sequence Program Size—Gives total HPM memory available for sequence program, in blocks.

SEQPROC (HPM Box)

Type:	**E:$SEQPROC**	**Sequence Processing Rate—**Specifies the number of processing units per Process Module data points that can be processed each quarter second cycle.
Lock:	**PtBld**	
Default:	**1_PU**	
PtRes:	**HPM**	

Range: **1_PU** One PU per Process Module point is allocated per scan (200 points can be processed per scan)
2_PU Two PUs per Process Module point are allocated per scan (100 points can be processed per scan)

SEQSLTSZ (ProcMod)

Type:	**Integer**	**Sequence Slot Size—**Defines the size of the program memory allocated for this process module in terms of blocks. Each block is 32 words long. The actual size is limited by the available memory.
Lock:	**PtBld**	
Default:	**0**	
PtRes:	**HPM**	
Range:	$\geq$**0**	

SERIALNO (STI)

Type:	**String_8**	**Serial Number/PROM Number of the Smart Transmitter**
Lock:	**View**	
Default:	**Blank**	
PtRes:	**HPM**	
Range:	**N/A**	

SGALGID(1)–(2) (DevCtl)

Type:	**E:$GTALGID in an Array (1..2)**	**Secondary Gate ID—**Defines the algorithm IDs for secondary gates.
Lock:	**PtBld**	
Default:	**None**	
PtRes:	**HPM**	

Range:
NULL (No algorithm)
AND (And Gate algorithm)
OR (Or Gate algorithm)
NAND (Nand Gate algorithm)
NOR (Nor Gate algorithm)
XOR (Exclusive Or Gate algorithm)
PAND (Pulse Nand Gate algorithm)
POR (Pulse or Gate algorithm)
PNAND (Pulse Nand Gate algorithm)
PNOR (Pulse Nor Gate algorithm)
PXOR (Pulse Exclusive or Gate algorithm)

SGDSTN(1)–(2) (DevCtl)

Type:	**E:$GATDSTN in an Array (1..2)**	**Secondary Gate Destination**—Defines the output destination for the secondary gates.
Lock:	**PtBld**	
Default:	**None**	
PtRes:	**HPM**	
Range:	**None** (No destination) **SI0** (Output goes to Safety Interlock) **I0, I1, I2** (Output goes to Interlocks) **P0, P1, P2** (Output goes to Permissives) **SOCMD0, SOCMD1, SOCMD2** (Output is commanded to go to SOCMD0, 1 or 2) **OPCMD** (Output is commanded to go to OPCMD parameter)	

SGPLSWTH(1)–(2) (DevCtl)

Type:	**Integer in an Array (1..2)**	**Pulse Width for Secondary Gate—**Indicates the pulse width for gates whose algorithms begin with “P”.
Lock:	**Supr**	
Default:	**0**	
PtRes:	**HPM**	
Range:	**0** to **8000**	

SGSO(1)–(2) (DevCtl)

Type:	**Logical in an Array (1..2)**	**Status Output for Secondary Gates**
Lock:	**View**	
Default:	**Off**	
PtRes:	**HPM**	
Range:	**Off** **On**	

SHEDMODE (RegCtl)

Type: **E:MODE**
Lock: **Engr**
Default: **Man**
PtRes: **HPM**
Range: 1-**Man** (Manual)
3-**Auto** (Automatic; applies to only Pid, PosProp, and RatioCtl algorithms)
4-**Bcas** (Backup Cascade)

Shedmode—Defines the mode to which this point sheds when it sheds from the Cas mode.

Helpful Hint:	1. SHEDMODE configuration requires RCASOPT = Spc or Ddc for Pid algorithm. 2. SHEDMODE configuration requires RCASOPT = Ddc for the following algorithms: AutoMan IncrSum ORSel Switch 3. SHEDMODE configuration requires RCASOPT = Spc for the following algorithms: PosProp RatioCtl

SHEDTIME (RegCtl)

Type: **Integer**
Lock: **Eng/PB**
Default: **0**
PtRes: **HPM**
Range: **0** to **1000 seconds**

Remote Cascade Shed Time—Defines the amount of time between successive updates of the SP or OP value from the AM. If the update is not received within the specified time, the AM or the NIM is assumed to have failed, and the backup control strategy is substituted by means of changing the mode to a preconfigured backup mode.

Helpful Hint:	SHEDTIME configuration requires RCASOPT = Spc, Ddc, or DdcRsp, which indicates that the SP or OP value is provided by the AM. To disable mode shed, use the default value of 0 for this parameter.

SHUTDOWN (RegCtl)

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

Shutdown Command Flag—This optional parameter allows the user to implement safety interlocks that can effectively shutdown a single control loop. When the SHUTDOWN flag is set to On by a user-written program or logic block, the mode and mode attribute are changed to Man and Oper, respectively, and the OP output is set equal to a predefined safe output value (SAFEOP). As long as the SHUTDOWN flag is On, the MODE, MODATTR, ESWENBST, and OP parameter values cannot be changed. When the SHUTDOWN flag is set to Off, the control loop must be manually restarted.

Helpful Hint:	Before a program sets this flag to the On state, it should write into parameter SAFEOP a safe shutdown value of 0%, 100%, or NaN (which causes the last good OP value to be used). A Logic Point or CL program must be used to reset the Shutdown Flag.

SI0 (DevCtl, DigComp)

Type: **Logical** **Safety Override Interlock Flag**
Lock: **Engr**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Override is not active)
On (Override is active)

Helpful Hint:	This parameter can be changed by the engineer only when the point is inactive or when the HPM is idle.

SI0ALOPT (DevCtl, DigComp)

Type: **E:$OVRALOP** **SI0 Safety Override Alarm—**Indicates the required action to be performed when a safety interlock occurs.
Lock: **Eng/PB**
Default: **None**
PtRes: **HPM**
Range: **None** (No override alarming)
Auto_Rtn (Return to normal when override is cleared)
Cnfm_Rqd (Confirm to clear, after interlock is cleared)

SI0ALPR (DevCtl, DigComp)

Type: **E:ALPIROR** **Override SI0 Alarm Priority—**Indicates the alarm priority for the safety override.
Lock: **Engr**
Default: **NoAction**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

SIALGID(1)–(12) (DevCtl)

Type: **E:$I2ALGID in an Array (1..12)** **Secondary Input Gate Algorithm ID—**Indicates the algorithm IDs for secondary input gates.
Lock: **PtBld**
Default: **Null**
PtRes: **HPM**
Range: **NULL** (No algorithm)
DLY (Input is Delayed algorithm)
ONDLY (On Delay algorithm, transition to ON is delayed)
OFFDLY (Off Delay algorithm, transition to OFF is delayed)
PULSE (Input is Pulsed algorithm)
MAXPULSE (Maximum Pulse Width algorithm)
MINPULSE (Minimum Pulse Width algorithm)

SIDLYTIM(1)–(12) (DevCtl)

Type: **Integer in an Array (1..12)**
Lock: **Supr**
Default: **0**
PtRes: **HPM**
Range: **0** to **8000 seconds**

Secondary Input Gate Delay/Pulse Width—Indicates the delay or pulse width for secondary input gates.

SIDSTN(1)–(12) (DevCtl)

Type: **E:$GATDSTN in an Array (1..12)**
Lock: **PtBld**
Default: **None**
PtRes: **HPM**
Range: **None** (No destination)
SI0 (Output goes to Safety Interlock)
I0, I1, I2 (Output goes to Interlocks)
P0, P1, P2 (Output goes to Permissives)
SOCMD0, SOCMD1, SOCMD2 (Output is commanded to go to SOCMD0, 1 or 2)
OPCMD (Output is commanded to go to OPCMD)
SG1, SG2 (Output goes to Secondary gates 1 or 2)
PG1, PG2, PG3, PG4 (Output goes to Primary gates 1, 2, 3 or 4)

Destination for Secondary Input Gates—Defines the output destination of the secondary gate.

SIM_TXT (NIM)

Type: **String_8**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

Simulation Indicator—see also DISP_SIM

SI0CONF (DigComp,DevCtl)

Type: **Logical**
Lock: **Oper**
Default: **Off**
PtRes: **HPM**
Range: **N/A**

Safety Override Interlock Alarm Confirmation Flag—Indicates that the safety override interlock alarm needs to be confirmed.

SISO(1)–(12) (DevCtl)

Type:	**Logical in an Array (1..12)**	**Status Output for Secondary Input Gates—**Indicates the output value of the secondary input gate.
Lock:	**View**	
Default:	**Off**	
PtRes:	**HPM**	
Range:	**Off** **On**	

SLOTNUM

Type:	**Integer**	**Slot Number—**Defines the slot number where this point resides. For IOP point types (AnalgIn, AnalgOut, DigIn, DigOut), it defines the hardware subslot on the module (IOP card) in which the point resides; refer to description of MODNUM parameter. For control points (DigComp, DevCtl, Array, Logic, RegPV, RegCtl, Flag, Numeric, Timer and ProcMod) it defines the software slot in the PMM. The processing capacity of the HPM depends on the number and mix of configured control points. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of how to determine the processing capacity of the HPM, based on the number of control points being used. For multivariable Smartline transmitters, note that although the transmitter is connected to only one slot, you must allocate adjacent slots for the other PVs.
Lock:	**PtBld**	
Default:	**N/A**	
PtRes:	**NIM**	

Range:
AnalgIn (1-16 for HLAI and STI; 1–32 for LLMUX and RHMUX)
AnalgIn (1-8 for LLAI)
Array (1-500, but ≤ the value of NARRSLOT)
DevCtl (1-400, but ≤ the value of NDEVSLOT)
DigComp (1-999, but ≤ the value of NDCSLOT)
DigIn (1-32)
DigOut (1-16 or 1-32)
Flag (HPM Box Flag 1-2047)
LLMUX (1-32)
Logic (1-400, but ≤ the value of NLOGSLOT)
Numeric (HPM Box Numeric 1-2047)
ProcMod (1-250, but ≤ the value of NPMSLOT)
PI (1-8)
RegCtl (1-250, but ≤ the value of NCTLSLOT)
RegPV (1-125, ≤ the value of NPVSLOT)
Timer (HPM box Timer 1-64)
AnalgOut (1-8 or 1-16)

Helpful Hint:	SLOTNUM configuration for Digcomp, DevCtl, Array, Flag, Numeric, Timer, ProcMod,Logic, RegCtl, and RegPV points require CTLOPT = On.

SLOT0SF(1)–(168)

Type: **String_96**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Hexadecimal Characters 00 to FF**

Slot 0 Soft Failures—Returns blind record of box soft failures present at a module address.

SLWSRCID (AnalgIn)

Type: **Integer**
Lock: **Eng/PB**
Default: **1**
PtRes: **HPM**
Range: **1–16**

Slidewire Voltage Source Identifier—Defines the slot number of the voltage source for the slidewire.

Helpful Hint:	SLWSRCID configuration requires SENSRTYP = Slidewire.

SNAME(1)–(2) (ProcMod)

Type: **String_8**
Lock: **View**
Default: **Spaces**
PtRes: **HPM**
Range: **N/A**

Subroutine Name—Indicates the name of the subroutine currently used by the process module. A value of " " means that no subroutine is executing. SNAME(1) and SNAME(2) display the name of the first and second level subroutines called from the main sequence.

SO (DigOut)

Type: **Logical**
Lock: **Oper**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Field contact is to be de-energized.)
On (Field contact is to be energized.)

Status Output—The output from a DigOut point.

Helpful Hint:	Only the HPMM Cont_Ctl access level can write to this parameter.

SO (Timer)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **On** (PV has reached the SP)
Off (PV has not reached the SP)

Status Output of Timer—Indicates whether the PV (elapsed time) has reached the SP (preset time).

SO(1)–(24) (Logic)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Output is false)
On (Output is true)

Logic Block Status Output—Indicates the output state of the logic block.

SO(0)–(2) (DevCtl, DigComp)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

Status output array—Indicates the current output state of the Digital Composite slot.
s = (0) for state 0
s = (1) for state 1
s = (2) for state 2

SOAKT1–12

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: **0.0** to **120.0 minutes**

Soak Time for Soak Segments 1–12—Define the soak time in minutes for each soak segment.

SOAKV1–12

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Soak Value for R/S Segments 1–12— Defines the soak values in engineering units for each soak segment.

SOCMD(0)–(2) (DevCtl, DigComp)

Type: **Logical**
Lock: **Prog**
Default: **OFF**
PtRes: **HPM**
Range: **On** (The OP is commanded to the state corresponding to 'i'. 0=State 0, 1=State 1, 2=State 2 if SOCMD (i) was previously OFF)
Off (No action)

Output Status Command—When commanding an OFF to ON write, the OP is commanded to the state corresponding to the array element written on an off-to-on transition.

SP (RegCtl)

Type: **Real**
Lock: **Oper**
Default: **0.0**
PtRes: **HPM**
Range: **SPLOLM** to **SPHILM**

Setpoint of the PV in Engineering Units

Helpful Hint:	SP usually does not require a control input connection. If a cascade connection to SP is required, it is typically configured by specifying a control output connection on the primary point.

SP (Timer)

Type: **Integer**
Lock: **Oper**
Default: **0**
PtRes: **HPM**
Range: **0** to **32000**

Preset Time—Defines the amount of time in seconds or minutes that the timer is to run.

SPEUHI (RegCtl)

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **≥ SPEULO**

Setpoint Engineering Unit High Range

SPEULO (RegCtl)

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **≤ SPEUHI**

Setpoint Engineering Unit Low Range

SPFORMAT (RegCtl)

Type: **E:VALFORMT**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: 0-**D0** (XXXX.)
1-**D1** (XXX.X)
2-**D2** (XX.XX)
3-**D3** (X.XXX)

Setpoint Decimal Point Format—Indicates the format of the SP value. SPFORMAT tracks with the selected PVFORMAT.

SPHIFL (RegCtl)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (High limit not exceeded)
On (High limit exceeded)

Setpoint High Limit Violation Flag—Indicates the SP has exceeded the upper limit established by SPHILM.

SPHILM (RegCtl)

Type: **Real** — **Setpoint High Limit—**Defines the upper limit for the SP.
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **SPLOLM** to **SPEUHI, NaN**

Helpful Hint:	1. Entering NaN disables limit checking by forcing SPHILM to its extreme value (SPEUHI). 2. SPHILM does not apply for the RampSoak algorithm.

SPLOCK (ProcMod)

Type: **E:ACCLVL** — **Setpoint Lock—**Stores to the process module point's own flags, numeric, strings, and time parameters, and are checked against the access lock specified by SPLOCK.
Lock: **Engr**
Default: **Operator**
PtRes: **HPM**
Range:
0-**Operator (**Operator and higher keylock positions allow store access.)
1-**Supervis (**Supervisor and higher keylock positions allow store access.)
2-**Engineer (**Engineer and higher keylock positions allow store access.)
3-**Program(**Only the program has store access.)

SPLOCK (Array)

Type: **E:ACCLVL** — **Setpoint Lock—**Indicates the access lock for array point parameters FL, NN, STRn and TIME.
Lock: **Engr**
Default: **Operator**
PtRes: **HPM**
Range:
0-**Operator** (Operator and higher keylock positions allow store access.)
1-**Supervis (**Supervisor and higher keylock positions allow store access.)
2-**Engineer (**Engineer and higher keylock positions allow store access.)
3-**Program(**Only the program has store access.)

SPLOCK (Totalizer)

Type: — **Setpoint Lock—**Stores to AVTV/PVTV parameters, are checked against the access lock specified by SPLOCK.
Lock: **Engr**
Default:
PtRes: **HPM**
Range:
0-**Operator (**Operator and higher keylock positions allow store access.)
1-**Supervis (**Supervisor and higher keylock positions allow store access.)
2-**Engineer (**Engineer and higher keylock positions allow store access.)
3-**Program(**Only the program has store access.)

SPLOFL (RegCtl)

Type: **Logical** **Setpoint Low Limit Violation Flag**—Indicates that the SP has exceeded the lower limit established by SPLOLM.
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Low limit is not exceeded)
On (Low limit is exceeded)

SPLOLM (RegCtl)

Type: **Real** **Setpoint Low Limit**—Defines the lower limit for the SP.
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **SPEULO** to **SPHILM,**
NaN

Helpful Hint:	1. Entering NaN disableslimit checking by forcing SPLOLM to its extreme value (SPEULO). 2. SPLOLM does not apply for the RampSoak algorithm.

SPOPT (RegCtl)

Type: **E:SPOPT** **Setpoint Option**
Lock: **Eng/PB**
Default: **None**
PtRes: **HPM**
Range: 0-**None** (No specialized options are available)
1-**TV** (Target Value processing; provides a smooth transition from an existing setpoint to a desired setpoint)
2-**Asp** (Advisory setpoint processing for Advisory Deviation Alarming)

Helpful Hint:	1. If component has been entered for the PNTFORM parameter, the Asp option cannot be configured. 2. SPOPT does not apply for the RampSoak algorithm.

SPP (RegCtl)

Type: **Real** **Setpoint in Percent**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range: **N/A**

SPTV (RegCtl)

Type: **Real** **Setpoint Target Value in Engineering Units**
Lock: **Oper**
Default: **N/A**
PtRes: **HPM**
Range: **SPLOLM to SPHILM,**
NaN

Helpful Hint:	SPTV change requires SPOPT = TV.

SPTVP (RegCtl)

Type:	**Real**	**Setpoint Target Value in Percent**
Lock:	**View**	
Default:	**N/A**	
PtRes:	**HPM**	
Range:	**≥ 0.0**	

Helpful Hint:	SPTVP change requires SPOPT = TV.

SRQUTAVG (NIM, HPM Box)

Type:	**Real**	**Average UCN Store Request Trip Time—**The average time in milliseconds that it takes to receive a response to this node's UCN store request.
Lock:	**View**	
Default:	**NaN**	
PtRes:	**HPM**	
Range:	**N/A**	

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

SRQUTMAX (NIM, HPM Box)

Type:	**Real**	**Maximum UCN Store Request Trip Time—**The maximum time in milliseconds that it takes to receive a response to this node's UCN store request.
Lock:	**View**	
Default:	**NaN**	
PtRes:	**HPM**	
Range:	**N/A**	

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

SRSPTAVG (NIM, HPM Box)

Type:	**Real**	**Average UCN Store Response Trip Time—**The average time in milliseconds that it takes this node to respond to UCN store requests from other nodes.
Lock:	**View**	
Default:	**NaN**	
PtRes:	**HPM**	
Range:	**N/A**	

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

SRSPTMAX (NIM, HPM Box)

Type:	**Real**	**Maximum UCN Store Response Trip Time—**The maximum time in milliseconds that it takes this node to respond to UCN store requests from other nodes.
Lock:	**View**	
Default:	**NaN**	
PtRes:	**HPM**	
Range:	**N/A**	

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

SSTEP(1)–(2) (ProcMod)

Type: **String_8**
Lock: **View**
Default: **Spaces**
PtRes: **HPM**
Range: **N/A**

Subroutine Step Name—Indicates the current step of the subroutine executing in this Process Module. A value of " " means that no subroutine is executing. SSTEP(1) and SSTEP(2) display the step name of the first and second level subroutines called from the main sequence.

SSTMT(1)–(2) (ProcMod)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **0** to **255**

Subroutine Statement Number—This parameter points to the statement number (in the NIM sequence library) of the current subroutine. A value of 0 indicates that no subroutine is executing. The array index indicates nesting level.

ST0_OP1–3 (DevCtl, Digcomp)

Type: **Logical**
Lock: **Eng/PB**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

State 0, Outputs 1 through 3—Defines the value (On or Off) that is to be written to output number 1, 2, and 3 when the OP is in State 0.

ST1_OP1–3 (DevCtl, Digcomp)

Type: **Logical**
Lock: **Eng/PB**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

State 1, Outputs 1, 2, and 3—Defines the value (On or Off) that is to be written to output number 1, 2, and 3 when the OP is in State 1.

ST2_OP1–3 (DevCtl, Digcomp)

Type: **Logical**
Lock: **Eng/PB**
Default: **Off**
PtRes: **HPM**
Range: **Off**
On

State 2–Outputs 1, 2, and 3—Defines the value (On or Off) that is to be written to output number 1, 2, and 3 when the OP is in State 2.

STARTFL

Type: **Logical**
Lock: **Prog**
Default: **Off**
PtRes: **HPM**
Range: **Off** (No effect on accumulator/totalizer)
On (Allows the accumulator/totalizer to begin counting up/down)

Start Command Flag—Starts DigIn accumulator, RegPV totalizer, or Box Timer when flag transitions from Off to On.

STATE (STI)

Type: **E:STATE**
Lock: **View**
Default: **N/A**
PtRes: **HPM**
Range:
2-**Loading** (Indicates that database loading between the STI point and the transmitter is occurring.)
3-**Loadcomp** (Indicates that the database transfer between the STI point and the transmitter has been successfully completed)
4-**Loadfail** (Indicates that the parameter transfer between the STI point and the transmitter has not been successfully completed)
5-**Calib** (Indicates that certain parameters are being calibrated at the transmitter by the STI point)
6-**Calcomp** (Indicates that the calibration has been successfully completed)
7-**Calfail** (Indicates that the calibration has not been successfully completed)
8-**OK** (Normal state; indicates that the STI point and the transmitter are OK. Transmitter is updating the PV value at the STI point. STATE remains OK when the point is made inactive.)
9-**DBChange** (Indicates that a database mismatch between the STI point and the transmitter has been detected. Transmitter is not updating the PV value at the STI point. STATE remains DBChange when the point is made inactive.)

Current State—Indicates the current state of the STI point.

STATE (Timer, DigIn)

Type: **E:STATE**
Lock: **View**
Default: **Stopped**
PtRes: **HPM**
Range: 0-**Stopped**
1-**Running**

Timer State—Indicates the current state of the timer data point.

STATE (Totalizr)

Type:	**E:STATE**	**Accumulator State**—Indicates the current state of the totalizer.
Lock:	**View**	
Default:	**Stopped**	
PtRes:	**HPM**	
Range:	0-**Stopped** (Stopped) 1-**Running** (Accumulating)	

STATE0–2

Type:	**String_8**	**Current State**—These parameters represent the state text (STATETXT) descriptors as follows: STATE0 = STATETXT(0) STATE1 = STATETXT(1) STATE2 = STATETXT(2); Digital Composite and DevCtl points. Refer to the *HPM Control Functions and Algorithms* manual for a detailed description of the Digital Composite point states.
Lock:	**View**	
Default:	**1 = On** **0 = Off** **2 = State 2**	
PtRes:	**NIM**	
Range:	**N/A**	

STATETXT(0)–(3) (DevCtl, DigIn, DigComp, Flag)

Type:	**String_8**	**State Descriptor Text**—Define the states of the point using descriptors which can have up to eight characters.
Lock:	**PtBld**	
Default:	**1 = On** **0 = Off** **2 = State 2** **3 = None (State 3)**	
PtRes:	**NIM**	

STATETXT (1) corresponds to the first ACTIVE state, or the state corresponding to PVFL = On (direct acting) or PVFL = Off (reverse acting). On the Group or Detail Display, it is in the upper box.

STATETXT (0) corresponds to the INACTIVE state, or the state corresponding to PVFL = Off (direct acting) or On (reverse acting). On the Group or Detail Display, it is the middle box for a Digital Composite or DevCtl point. For a Digital Input point, it is the lower box.

STATETXT (2) corresponds to the second ACTIVE state. On the Group or Detail Display for a Digital Composite or DevCtl point, it is in the lower box. STATETXT(2) does not apply to Digital Input and Flag points. When a two- state device is configured, STATETXT(2) is internally set to $NULL.

STATETXT (3) “NONE” (not configurable).
STATETXT(3) does not apply to Digital Input and Flag points.

Range: N/A

Helpful Hint:	STATETXT has an access lock of View if PNTFORM = Componnt. STATETXT (2) for State 2 applies only if NOSTATES = 3 for digital composite or device control points.

STATMENT (ProcMod)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: 1 to **255**

Statement—Indicates the current statement of the sequence executing in this process module. A statement number of 0 indicates that no statement is being executed.

STATTIM0–2 (DevCtl, DigComp)

Type: **Time (Duration)**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **0 - 4000 Days** (1 second resolution)

State Time—The amount of time based on the PV that has accumulated for States 0, 1, and 2 since the most recent reset of maintenance statistics.

STDBYMAN (AnalgOut, DigOut, RegCtl)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Output is not connected to standby-manual device)
On (Output is connected to standby-manual device)

Standby Manual Flag—Indicates whether the associated hardware output is connected to a standby manual device. It is not an indication of whether or not the output is isolated from the process.

STDBYSTS(1)–(168)

Type: **Logical**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Off** (No Standby Manual present)
On (Standby Manual is activated)

Standby Status Flag—Returns blind record of STDBYMAN status

STCHGOPT (DevCtl, DigComp)

Type: **E:$STCHGOP**
Lock: **Engr/PB**
Default: 0-**None**
PtRes: 1-**HPM**

State Change Option—State0 passed through before entering a new state. If command disagree alarming is not configured, the point will wait for the number of seconds designated in the PAUSETIM parameter after State0 is commanded. If alarming is configured, the system will wait for the PV to go to State0 (or when the feedback timer expires) before starting the pause timer.

Range: **None** (State change option is not configured)
State0 (State change option is configured)

STEP (ProcMod)

Type: **String_8**
Lock: **View**
Default: **Spaces**
PtRes: **HPM**
Range: **N/A**

Step Name—Indicates the step name of the sequence executing in this process module.

STI_EU (STI)

Type: **E:STI_EU**
Lock: **Eng**
Default: **InH20**
PtRes: **HPM**

Smart Transmitter Engineering Units—Specifies the units of measurement for parameters LRL, LRV, URL, and URV. These units are used for display only.

For multivariable Smart Transmitters with a SENSRTYP of SFM, choose BLANK. For multivariable slots with a SENSRTYP of SPT_DP, SPT_AP, SPT_GP, or STT, choose the preferred STI_EU (engineering units). When BLANK is selected the limit values URL, LRL, URV and LRV are displayed in the base engineering units specified in the transmitter user manual.

> *Helpful Hint:* Loading an invalid STI_EU type causes an error. An attempt to correct it from the Detail Display is rejected as `CONFIG MISMATCH`. To recover, load the correct STI_EU parameter from the PED or perform an UPLOAD from the point Detail Display.

Range:

0-**InH20** (Pressure transmitter—Inches of water)
1-**MMHG** (Pressure transmitter—Millimeters of mercury)
2-**PSI** (Pressure transmitter—Pounds per square inch)
3-**KPA** (Pressure flow transmitter—Kilopascals)
4-**MPA** (Pressure transmitter—Millipascals)
5-**MBar** (Pressure transmitter—Millibars)
6-**Bar** (Pressure transmitter—Bars)
7-**G_SQCM** (Pressure transmitter—Grams per square centimeter)
8-**KG_SQCM** (Pressure flow transmitter—Kilograms per square centimeter)
9-**MMH20** (Pressure transmitter—Millimeters of water)
10-**INHG** (Pressure transmitter—Inches of mercury)
11-**Deg_C** (Temperature transmitter—Degrees Centigrade)
12-**Deg_F** (Temperature transmitter—Degrees Fahrenheit)
13-**Deg_K** (Temperature transmitter—Degrees Kelvin)
14-**Deg_R** (Temperature transmitter—Degrees Rankine)
15-**MV** (Temperature transmitter—Millivolts)
16-**V** (Temperature transmitter—Volts)
17-**Ohms** (Temperature transmitter—RTD Ohms)
18-**CM_HR** (Magnetic flow transmitter (volume)—Cubic Meters per hour)
19-**Gal_HR** (Magnetic flow transmitter (volume)—Gallons per hour)
20-**LIT_HR** (Magnetic flow transmitter (volume)—Liters per hour)
21-**CC_HR** (Magnetic flow transmitter (volume)—Cubic Centimeters per hour)
22-**CM_Min** (Magnetic flow transmitter (volume)—Cubic Meters per hour)
23-**Gal_Min** (Magnetic flow transmitter (volume)—Gallons per minute)
24-**Lit_Min** (Magnetic flow transmitter (volume)—Liters per minute)
25-**CC_Min** (Magnetic flow transmitter (volume)—Cubic centimeters per minute)
26-**CM_Day** (Magnetic flow transmitter (volume)—Cubic meters per day)
27-**Gal_Day** (Magnetic flow transmitter (volume)—Gallons per day)
28-**KGal_Day** (Magnetic flow transmitter (volume)—Thousands of gallons per day)
29-**BRL_Day** (Magnetic flow transmitter (volume)—Barrels per day)
30-**CM_Sec** (Magnetic flow transmitter (volume)—Centimeters per second)
31-**KG_HR*** (Magnetic flow transmitter (mass)—Kilograms per hour)
32-**LBS_HR*** (Magnetic flow transmitter (mass)—Pounds per hour)
33-**Ft_Sec** (Magnetic flow transmitter (velocity)—Feet per second)
34-**M_Sec** (Magnetic flow transmitter (velocity)—Meters per second)
35-**KG_Min*** (Magnetic flow transmitter (mass)—Kilograms per minute)
36-**KG_Sec*** (Magnetic flow transmitter (mass)—Kilograms per second)
37-**LBS_Min*** (Magnetic flow transmitter (mass)—Pounds per minute)
38-**LBS_Sec*** (Magnetic flow transmitter (mass)—Pounds per second)
39-**PRCNT** (Percent)

*Not implemented

STI_EU (STI) (continued)

40-**BLANK** (Blank) - Multivariable transmitter with SFM SENSRTYP
41-**LBS** (Pounds)
42-**KG** (Kilograms)
43-**TONS** (Tons)
44-**GRAMS** (Grams)
45-**OZ** (Ounces)
46-**GAL** (Gallons)
47-**BRL** (Barrels)
48-**CUB_M** (Cubic Meters)
49-**LITERS** (Liters)
50-**MLITRES** (Milliliters)
51-**FL_OZ** (Fluid ounce)
52-**FEET** (Feet)
53-**METERS** (Meters)
54-**MM** (Millimeters)
55-**INCHES** (Inches)
56-**KG_CUM** (Kilograms per cubic meter)
57-**G_CUM** (Grams per cubic Meter)
58-**LBS_CUFT** (Pounds per cubic foot)
59-**LBS_CUIN** (Pounds per cubic inch)

STISWVER (STI)

Type:	**String_8**	**Software Revision Level of the Smart Transmitter**
Lock:	**View**	
Default:	**Blank**	
PtRes:	**HPM**	
Range:	**N/A**	

STITAG (STI)

Type:	**String_8**	**Transmitter Tag Name—**Identifies the name of the Smart Transmitter to the system and on displays, reports, and logs.
Lock:	**Eng/PB**	
Default:	**All Spaces**	
PtRes:	**HPM**	

> *Helpful Hint:* For multivariable transmitters, an identical STITAG must be entered for each active slot related to that transmitter. The IOP uses the number of identical STITAG names to calculate the number of PVs associated with with a given transmitter.

Range: Tag name can be up to 8 characters, and the permissible character set is as follows:
Alphabetics A-Z (uppercase or lowercase)
Numerics 0-9
Embedded space characters are allowed.

STOPFL

Type:	**Logical**	**Stop Command Flag—**Stops the DigIn accumulator, RegPV totalizer, or Box Timer when flag transitions from Off to On.
Lock:	**Prog**	
Default:	**Off**	
PtRes:	**HPM**	
Range:	**Off** (No effect on the accumulator/totalizer) **On** (Stops the accumulator/totalizer from counting up/down)	

STR8(1)–(16,384) (HPM Box)

Type:	**String_8**	**Box String Variables—**The upper bound limit of this array is determined by the NSTRING Box parameter. The LCN index limit is 4095, while no limit exists for the UCN. Array points may be used to address strings with an index greater than 4095.
Lock:	**Oper**	
Default:	**Spaces**	
PtRes:	**HPM**	
Range:	**N/A**	

STR8(1)–(16) (ProcMod)

Type:	**String_8**	**Local String Variables—**Each Process Module point has 16 local STR8 variables that are different from the HPM Box STR8 variables.
Lock:	**Determined by SPLOCK parameter**	
Default:	**Spaces**	
PtRes:	**HPM**	
Range:	**N/A**	

STR16(1)–(8) (ProcMod)

Type:	**String_16**	**Local String Variables—**Each Process Module point has 8 local STR16 variables that overlay the local STR8 variables [for example, STR16(1)=STR8(1) concatenated with STR8(2)].
Lock:	**Determined by SPLOCK parameter**	
Default:	**Spaces**	
PtRes:	**HPM**	
Range:	**N/A**	

STR32(1)–(4) (ProcMod)

Type:	**String_32**	**Local String Variables—**Each Process Module point has four local STR32 variables that overlay the local STR8 variables.
Lock:	**Determined by SPLOCK parameter**	
Default:	**Spaces**	
PtRes:	**HPM**	
Range:	**N/A**	

STR64(1)–(2) (ProcMod)

Type:	**String_64**	**Local String Variables—**Each Process Module point has two local STR64 variables that overlay the local STR8 variables [for example, STR64(1)=STR8(1-8)].
Lock:	**Determined by SPLOCK parameter**	
Default:	**Spaces**	
PtRes:	**HPM**	
Range:	**N/A**	

STR8(i) (Array)

Type:	**String_8**	**Array Point String Variables—**8-character string variables that are mapped to the Array point. The number of variables is dependent on the NSTRING and STRLEN variables.
Lock:	**Determined by SPLOCK parameter**	
Default:	**N/A**	
PtRes:	**HPM**	
Range:	**1 ≤ i ≤ (Array parameter NSTRING)/(8/STRLEN)**	

STR16(i) (Array)

Type:	**String_16**	**Array Point String Array Variables—**16-character string variables that are mapped to the Array point. These variables overlay the STR8 variables.
Lock:	**Determined by SPLOCK parameter**	
Default:	**N/A**	
PtRes:	**HPM**	
Range:	**1 ≤ i ≤ (Array parameter NSTRING)/(16/STRLEN)**	

STR32(i) (Array)

Type:	**String_32**	**Array Point String Variables—**32-character string variables mapped to the Array point that overlay the STR8 variables.
Lock:	**Determined by SPLOCK parameter**	
Default:	**N/A**	
PtRes:	**HPM**	
Range:	**1 ≤ i ≤ (Array parameter NSTRING)/(32/STRLEN)**	

STR64(i) (Array)

Type:	**String_64**	**Array Point String Variables—**64-character string variables mapped to the Array point that overlay the STR8 variables.
Lock:	**Determined by SPLOCK parameter**	
Default:	**N/A**	
PtRes:	**HPM**	
Range:	**1 ≤ i ≤ (Array parameter NSTRING)/(64/STRLEN)**	

STRDESC (Array)

Type: **String_64**
Lock: **PtBld**
Default: **Spaces**
PtRes: **HPM**
Range: **N/A**

String Array Descriptor—64-character string describing the Array point string data.

STRLEN (Array)

Type: **Integer**
Lock: **PtBld**
Default: **8**
PtRes: **HPM**
Range: **8, 16, 32, 64**

Array Point String Length—Indicates the length of the configured string (with the STRSTIX and NSTRING parameters) displayed on the Array Point Detail display. Strings can be accessed by STR8, STR16, STR32 or STR64 regardless of this value.

STRLEN (ProcMod)

Type: **Integer**
Lock: **PtBld**
Default: **8**
PtRes: **HPM**
Range: **8, 16, 32, 64**

Process Module String Length—Indicates the displayed string length on the Process Module Detail display. Strings can still be accessed by STR8, STR16, STR32, or STR64 regardless of this value.

STRSTIX (Array)

Type: **Real**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **Box parameter NSTRING** (When EXTDATA≠IO_STR, 0 indicates no strings are configured)
0 to **99,999** (When EXTDATA=IO_STR, 0 can be a valid device index)

String Array Start Index—Defines the string array start index in Box STR8 variables, or the Serial Interface-connected device.

STRTFAIL(1)–(6)

Type: **String_2**
Lock: **View**
Default:
PtRes: **HPM**
Range: **Hexadecimal characters 00 to FF**

Startup/Failover Information

STSMSG

Type: **E:MSGTXT**
Lock: **Oper**
Default: **MSGTXT(0)**
PtRes: **HPM**
Range: **MSGTXT(0)** to **MSGTXT(15)**

Status Message—A self-defining enumeration of the MSGTXT parameter that provides additional descriptive information regarding the red tag, batch state, or device state.

SUMSLTSZ (HPM Box)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **>0**

Total Configured Memory for Sequence Programs—This parameter equals the sum of all Process Module SEQSLTSZ parameters, and is shown on the HPMM Control Configuration display.

SUSPSTAT (ProcMod)

Type: **E:$SUSPST**
Lock: **View**
Default: **None**
PtRes: **HPM**
Range: 0-**None**
2-**Feedback**
3-**Wait**
4-**ConfMsg**
5-**InputMsg**

Suspend State

SUSPTIME (ProcMod)

Type: **Integer**
Lock: **View**
Default: 0
PtRes: **HPM**
Range: **0** to **32, 767** minutes

Suspension Timer—The remaining time (in minutes) before the sequence program resumes execution. This timer is started when a sequence program executes a wait statement.

SVALDB (DevCtl)

Type: **E:PVALDB**
Lock: **Engr/PB**
Default: **One**
PtRes: **HPM**
Range: 0-**Half** (1/2 of 1% of Engineering Unit range)
1-**One** (1% of Engineering Unit range)
2-**Two** (2% of Engineering Unit range)
3-**Three** (3% of Engineering Unit range)
4-**Four** (4% of Engineering Unit range)
5-**Five** (5% of Engineering Unit range)
6-**EU** (Value is defined by SVALDBEU parameter)

SECVAR Alarm Deadband—The deadband for the SECVAR alarm.

SVALDBEU (DevCtl)

Type: **Real**
Lock: **Engr/PB**
Default: **0.0**
PtRes: **HPM**
Range: **Allowable Engineering Units**

SECVAR Alarm Deadband in Engineering Units—Indicates the alarm deadband in engineering units when the SVALDB parameter = EU.

SVDESC (DevCtl)

Type: **String_8**
Lock: **PtBld**
Default: **Blank**
PtRes: **HPM**
Range: **8 Character String**

SECVAR Descriptor—Defines the SECVAR parameter or secondary variable descriptor.

SVEUDESC (DevCtl)

Type: **String_8**
Lock: **PtBld**
Default: **Blank**
PtRes: **HPM**
Range: **8 Character String**

SECVAR Engineering Unit Descriptor—Defines the engineering unit descriptor for the SECVAR parameter or secondary variable descriptor.

SVEUHI (DevCtl)

Type: **Real**
Lock: **Engr/PB**
Default: **NaN**
PtRes: **HPM**
Range: **<> NaN**

SECVAR Range High—Defines the high engineering unit range for the SECVAR parameter.

SVEULO (DevCtl)

Type: **Real**
Lock: **Engr/PB**
Default: **NaN**
PtRes: **HPM**
Range: **<> NaN**

SECVAR Range Low—Defines the low engineering unit range for the SECVAR parameter.

SVHHFL (DevCtl)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (SECVAR parameter is below the SVHHTP parameter minus the deadband)
On (SECVAR parameter has exceeded the SVHHTP parameter)

SECVAR High-High Alarm Flag

SVHHPR (DevCtl)

Type: **E:ALPIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized, reported to printer, but not annunciated)
Printer (Reported to printer only)
Emergncy (Reported to all alarm summary displays)
High (Reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Reported to Unit Alarm Summary Display)
Journal (Logged but not reported to Universal Stations)
NoAction (Alarm is not reported to the system)

SECVAR High-High Alarm Priority

SVHHTP (DevCtl)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **≥ SVHITP or NaN**

SECVAR High-High Alarm Trip Point—No alarms are generated when this parameter is set to NaN.

SVHHTPP (DevCtl)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **0** to **100**

SECVAR High-High Trip Point Percent—The SECVAR High-High Trip Point in terms of engineering units in percent.

SVHIFL (DevCtl)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (SECVAR parameter is below SVHIFL minus the deadband)
On (SECVAR parameter has exceeded SVHIFL)

SECVAR High Alarm Flag—This flag is set when the SECVAR exceeds SVHITP and is reset when SECVAR is below SVHIFL minus deadband.

SVHIPR (DevCtl)

Type: **E:ALPRIOR**
Lock: **Engr**
Default: **Low**
PtRes: **NIM**
Range: **JnlPrint** (Alarm is historized and reported to the printer but not annunciated)
Printer (Alarm is reported to the printer but not historized and not annunciated)
Emergncy (Alarm is historized, annunciated, and reported to all alarm summary displays)
High (Alarm is historized, reported to Area Alarm Summary Display and Unit Alarm Summary Display)
Low (Alarm is historized, reported to the Unit Alarm Summary Display, and annunciated)
Journal (Alarm is historized but not reported to Universal Stations and not annunciated)
NoAction (Alarm is not reported to the system and not annunciated)

SECVAR High Alarm Priority

SVHITP (DevCtl)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **≥ SVEULO or NaN**

SECVAR High Alarm Trip Point—When this parameter is set to NaN, no alarms are generated.

SVHITPP (DevCtl)

Type: **Real**
Lock: **Supr**
Default: **NaN**
PtRes: **HPM**
Range: **0** to **100**

SECVAR High Alarm Trip Point Percent—The SECVAR High Trip Point in terms of engineering units percent.

SVP (DevCtl)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **0** to **100**

SECVAR in Percent of Engineering Range—The percentage for this parameter is calculated from the SECVAR parameter, using both the SVEVHI and SVELVO parameters.

SVPEAK (DevCtl)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **Real or NaN**

Peak Value of SECVAR—Indicates the highest value of the SECVAR parameter since the device changed from state 0.

SVSRC (DevCtl)

Type: **Universal Ent.Prm**
Lock: **PtBld**
Default: **Null**
PtRes: **HPM**

SECVAR Input Connection—Specifies input connection to which the current values of Device Control inputs are supplied. The connection can be specified using the "Tagname.Parameter" format or the hardware reference address format Refer to the *HPM Control Functions and Algorithms* manual for a detailed description.

Range: Use Tagname.Parameter for tagged points where Tagname can be up to 16 characters and the permissible character set is as follows:

Alphabetics A-Z (uppercase only)
Numerics 0-9 (an all numeric tag name is not allowed)
Underscore (_) cannot be used as the first character or the last character, and consecutive underscores are not allowed.
Embedded space characters are not allowed.
An * is used to default to this point's tag name.
Parameter name can be up to eight characters and must be a legitimate parameter name.

Some possible input-connection sources are
a."Logic slot Tagname.NN(nn)" where nn = 1–8
b."ProcMod slot Tagname.NN(nnn)" where nnn = 1–80
c."Box Numerics slot Tagname.NN" where nnnnn = 1-16,384
d."!Box.FL(nnnn)" for a box flag that resides in the same box where nnnn = 1–16,384

Use the hardware reference address !MTmmSss.Parameter for untagged or tagged points where
MT is the IOP type, such as AI (Analog Input)
mm is the IOP Card number (1–40)
The letter "S" is a constant
ss is the slot number on the IOP Card (refer to SLOTNUM parameter)
Parameter name can be up to eight characters and must be a legitimate parameter name.

SVTV (DevCtl)

Type: **Real**
Lock: **Oper**
Default: **NaN**
PtRes: **HPM**
Range: **SVEULO** to **SVEUHI**

Secondary Variable Target Value—Indicates the normal or operating setpoint for the SECVAR parameter.

SVTVP (DevCtl)

Type: **Real**
Lock: **Oper**
Default: **NaN**
PtRes: **HPM**
Range: **0** to **100**

Secondary Variable Target Value in Percent—Indicates the normal or operating setpoint for the SECVAR parameter as a percent.

SWTCHACT (1)–(40) (HPM Box)

Type: **Logical**
Lock: **View**
Default: **On**
PtRes: **HPM**
Range: On-**Active**
Off-**Inactive**

IOP Synchronization—When this parameter is Off (inactive), the backup request line from the IOP's partner is not asserted. When this parameter is On (active), the backup request line from the IOP's partner is asserted.

SYNCHSTS(1)–(40) (HPM Box)

Type: **E:$SYNCHST**
Lock: **View**
Default: **None**
PtRes: **HPM**

Primary/secondary Database Synchronization Status. It is set to WARNING when a database inconsistency is first detected, but the secondary appears capable of providing backup. It changes to FAIL when the HPMM has tried to resync but does not succeed or a secondary failure prevents synchronization. nn = 1–40 corresponds to the 40 logical I/O modules.
Applies to primary IOP only.

Range: 0-**OK -** (the secondary is synchronized with the primary and can provide backup if necessary.)
1-**WARNING - (**database inconsistency detected but secondary can probably provide backup).
2-**FAIL - (**HPMM has tried to resynchronize but has not succeeded or, secondary has also failed).

-T-

T (FlowComp)

Type: **Real**
Lock: **View**
Default: **1.0**
PtRes: **HPM**
Range: **≥ 0.0**

Temperature Input—Indicates the measured actual temperature.

T0 (FlowComp)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: **N/A**

Zero Reference for Temperature—T0 is the zero reference temperature input and it is in the same engineering units as the T (measured actual temperature) input. T0 is typically -459.69 degrees F or -273.15 degrees C. Enter the absolute value of the temperature.

T1

Type: **Real**
Lock: **Supr**
Default: **0.0 minutes**
PtRes: **HPM**
Range: **0.0** to **1440.0 minutes**

Integral Time in Minutes—Defines the integral time constant in minutes-per-repeat.

Helpful Hint:	Integral action can be disabled by setting T1 equal to 0.0.

T2

Type: **Real**
Lock: **Supr**
Default: **0.0 minutes**
PtRes: **HPM**
Range: **0.0** to **1440.0 minutes**

Derivative Time in Minutes—Defines the derivative time constant.

Helpful Hint:	Derivative action can be disabled by setting T2 equal to 0.0

TCRNGOPT

Type: **E:$TCRNGOP**
Lock: **Eng/PB**
Default: **Normal**
PtRes: **HPM**
Range: 0-**Normal** (Use PVEXEULO's normal range table)
1-**Extended** (Use PVEXEULO's extended range table)

Thermocouple Range Option—Defines the thermocouple range option. Applies only if a thermocouple (TC) sensor type is used for this data point.

Helpful Hint:	TCRNGOPT configuration requires IOTYPE = LLAI, LLMUX, or RHMUX and SENSRTYP = Thermcpl. Refer to parameter PVEXEULO.

TD (VdtLdLag)

Type: **Real**
Lock: **Supr**
Default: **0.0**
PtRes: **HPM**
Range: **≥ 0.0 minutes**

Total Dead Time in Minutes—Defines the fixed delay time in minutes for equation B, and the actual variable delay time in minutes for equations C and D.

TF

Type: **Real**
Lock: **Supr**
Default: **0.0 minutes**
PtRes: **HPM**
Range: **0.0** to **60.0 minutes**

PV Filter Lag Time in Minutes—Defines the filtering time lag to be used after the PV range has been checked. A value of 0.0 specifies that the PV is not delayed.

TIERTYPE (HPM Box)

Type: **E:$TIERTYP**
Lock: **View**
Default:
PtRes: **HPM**
Range:

HPMM Tier Type

TIME(1)–(4095) (HPM Box)

Type: **Time**
Lock: **Oper**
Default: **0 seconds**
PtRes: **HPM**
Range: **N/A**

Box Time Variables—The upper limit of this array is determined by the NTIME parameter. The LCN index limit is 4,095, while the limit on the UCN is 4096. Array points may be used to address Times with an index greater than 4095.

TIME(i) (Array)

Type: **Time**
Lock: **Determined by SPLOCK parameter**
Default: **N/A**
PtRes: **HPM**
Range: **1 ≤ i ≤ Array parameter NTIME**

Array Point Time Variables—Times are mapped from the HPM Box defined by the TIMESTIX and NTIME parameters.

TIME(1)–(4) (ProcMod)

Type: **Time**
Lock: **Determined by SPLOCK parameter**
Default: **0 seconds**
PtRes: **HPM**
Range: **N/A**

Local Time Variables—Four local Time variables are available in each Process Module point. These variables are different than the HPM Box Time variables.

TIMEBASE (Timer)

Type: **E:TIMEBASE**
Lock: **Engr**
Default: **Seconds**
PtRes: **HPM**
Range: 0-**Seconds**
1-**Minutes**

Time Base—Defines the time base to be used for the Timer data point.

TIMEBASE (Totalizr, PI)

Type: **E:TIMEBASE**
Lock: **Eng/PB**
Default: **Minutes**
PtRes: **HPM**
Range: 0-**Seconds** (PV and Setpoint engineering units (gallons, etc.) per second)
1-**Minutes** (PV and Setpoint engineering units (gallons, etc.) per minute)
2-**Hours** (PV and Setpoint engineering units (gallons, etc.) per hour)

Totalizer Time Base—Defines whether time base is in seconds, minutes, or hours.

TIMEDESC (Array)

Type: **String_64**
Lock: **PtBld**
Default: **Spaces**
PtRes: **HPM**
Range: **N/A**

Time Array Descriptor—Sixty four-character string describing Time data.

TIMESECS(1)–(240) (Array)

Type: **Time**
Lock: **Determined by SPLOCK parameter**
Default: **N/A**
PtRes: **HPM**
Range: **1 ≤ i ≤ NTIME Array parameter**

Array Point Time Variables—Times mapped from the HPM box defined by TIMESTIX and NTIME parameters.

TIMESTIX (Array)

Type: **Real**
Lock: **PtBld**
Default: **0**
PtRes: **HPM**
Range: **0** to **Box parameter NTIME** (0 indicates there are no Times configured for this point)

Time Array Start Index—Defines the Time data start index in the Box Time variables.

TIMESYNC (UCN)

Type: **E:ENBLSTAT**
Lock: **PtBld**
Default: **Disable**
PtRes: **NIM**
Range: **Enable** (This NIM or NIM pair is able to perform SOE time synchronization)
Disable (This NIM or NIM pair does not perform SOE time synchronization, but can receive and report SOE events)

Timesynch Control—Defines whether SOE timesynch can be performed by this NIM or NIM pair. Normally, the NIM with the lowest address is configured for this function.

TLD (VdtLdLag)

Type: **Real**
Lock: **Supr**
Default: **0.0 minutes**
PtRes: **HPM**
Range: **-1440.0** to **1440.0 minutes**

Lead Time Constant in Minutes—Defines the lead-compensation time constant in minutes. A 0 (zero) entry specifies no lead compensation.

TLG1, TLG2 (VdtLdLag)

Type: **Real**
Lock: **Supr**
Default: **0.0 minutes**
PtRes: **HPM**
Range: **0.0** to **1440.0** minutes (**0** specifies no lag compensation)

Lag Time Constant

TMCMD(1)–(64) (HPM Box)

Type: **E:COMMAND**
Lock: **Oper**
Default: **N/A**
PtRes: **HPM**
Range: 0-**None** (A command has not been issued to the timer)
1-**Start** (Starts the timer)
2-**Stop** (Stops the timer)
3-**Reset** (Resets the timer to 0)
4-**RestStrt** (Resets the timer to 0, and starts the timer)

Timer Command—An array of commands issued to the 64 Timer data points.

TMPV(1)–(64) (Timer)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **>0**

Timer PV—Indicates the current (elapsed) time of the Timer data point in seconds or minutes.

TMRV(1)–(64) (Timer)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **>0**

Timer RV—Indicates the remaining time (TMSP minus TMPV) for the Timer data point.

TMSO(1)–(64) (Timer)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes: **HPM**
Range: **Off** (TMPV ≠ TMSP; elapsed time has not reached the preset time)
On (TMPV = TMSP; elapsed time has reached the preset time)

Timer Status Output—Indicates the current state of the timer output.

TMSP(1)–(64) (Timer)

Type: **Integer**
Lock: **Oper**
Default: **0**
PtRes: **HPM**
Range: **0** to **32000**

Timer Setpoint—Defines the preset time of the Timer data point, in seconds or minutes.

TMST(1)–(64) (Timer)

Type: **E:STATE**
Lock: **View**
Default: **Stopped**
PtRes: **HPM**
Range: 0-**Stopped** (Timer is currently stopped)
1-**Running** (Timer is currently running)

Timer State—Indicates the current state of the Timer data point.

TMTB(1)–(64) (Timer)

Type: **E:TIMEBASE**
Lock: **Engr**
Default: **Seconds**
PtRes: **HPM**
Range: 0-**Seconds**
1-**Minutes**

Timer Time Base—Defines the time base of the timer.

TOTLUAVG (1) - (2) (HPM Box)

Type: **Real**
Lock: **View**
Default: **0.0**
PtRes: **HPM**
Range: **0 - 100**

Average IOL Utilization (in per cent) by the HPM, per I/O Link— (total utilization by the Comm and Control CPUs)

TOTLUMAX (1) - (2) (HPM Box)

Type: **Real**
Lock: **View**
Default: **0.0**
PtRes: **HPM**
Range: **0 - 100**

Maximum IOL Utilization (in per cent) by the HPM, per I/O Link— (total utilization by the Comm and Control CPUs)

TRACKING

Type: **Logical**
Lock: **Eng/PB**
Default: **Off**
PtRes: **HPM**
Range: **Off** (Tracking disabled)
On (Tracking is to be used)

Selected Input Tracking—Allows the selected input to be changed without bumping the output.

Helpful Hint:	If On, causes nonselected inputs to track the selected input.

TRANTIM0–2 (DevCtl, DigComp)

Type: **Time**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **Time Stamp**

Transition Time—The date and time of the most recent transition to each state based on the PV.

TRATAVG (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Average UCN Transaction Trip Time—The average UCN transaction trip time in milliseconds for both fetch and store responses from this node to other UCN nodes.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays

TRATMAX (NIM, HPM Box)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Maximum UCN Transaction Trip Time—The maximum UCN transaction trip time in milliseconds for both fetch and store responses from this node to other UCN nodes.

Helpful Hint:	This statistic can be viewed on the Toolkit Displays.

TRFB (PidErfb)

Type: **Real**
Lock: **View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A**

Tracking Feedback Input in Engineering Units—Indicates the value of the PV or SP of another data point that is receiving its setpoint from this data point.

TSCOMP

Type: **Time**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Time Stamp, CL Source Compatibility —Specifies the CL Source compatibility time stamp (CL object header)

TSSRC

Type: **Time**
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

Time Stamp, CL Source—Specifies the CL Source time stamp (CL object header)

TSTS (FlowComp)

Type: **E:PVVALST** **Temperature Input Value Status—**Status of the T input value.
Lock: **View**
Default: **Normal**
PtRes: **HPM**
Range: 0-**Bad** (Value is bad and replaced with NaN)
1-**Uncertn** (Status of the value is uncertain)
2-**Normal** (Value is good)

TSUNICHG

Type: **Time** **Time Stamp, Unit Change—**Specifies the CL Unit Change time stamp (CL object header)
Lock: **View**
Default: **0**
PtRes: **HPM**
Range: **N/A**

TVPROC (RegCtl)

Type: **E:TVPROC** **Target Value Processor State**
Lock: **Oper**
Default: **Off**
PtRes: **HPM**
Range: 0-**Off** (No target value processing)
1-**Preset** (Set up setpoint target value and ramp time)
2-**Run** (Perform ramping function)

Helpful Hint:	TVPROC applies only if SPOPT = TV.

-U-

UCNRECHN (HPM Box, NIM)

Type: **E:$RECCHN** **UCN Receive Channel—**Indicates the channel to which the node is listening.
Lock: **View**
Default: **ChannelA**
PtRes **HPM, NIM**
Range: 0-**ChannelA**
1-**ChannelB**

UCNSCANT (HPM Box)

Type: **Real** **Peer-to-Peer Scan Period in seconds**
Lock: **Eng/PB**
Default: **0.5**
PtRes: **HPM**
Range: **0.5, 1.0**

UCNSFREV

Type: **Integer** **UCN Software Revision**
Lock: **View**
Default: **N/A**
PtRes **HPM**

UCNSFVER

Type: **Integer** **UCN Software Version**
Lock: **View**
Default: **N/A**
PtRes **HPM**

UCNWRTLK (HPM Box)

Type: **E: UCNWRTL**
Lock: **Eng**
Default: **WrtLkOff**
PtRes: **HPM**

HPM Write Lockout—When HPM Write Lockout is set to On, all Writes to the HPM (except writes to UCNWRTLK and some IOL parameters) are locked out including peer-to-peer writes. All parameter reads are allowed as well as cable swaps, HPMM swaps and IOP swaps.

Range: 0-WrtLkOff - (Write Lock Off, UCN node is read/write)
1-WrtLkOn - (Write Lock On, UCN node is read only)

> *Helpful Hint:* Write Lockout must be set to Off before any changes are made to configuration, modes, or setpoints. The state of UCNWRTLK can only be changed when the HPM is either in RUN or RUNSOFTFAIL state.
>
> UCNWRTL can be changed (under Engineer Key Level) from the HPM Write Lock Control display. Refer to the *HPM Implementation Guidelines* for more information.

UNCMDFL (DevCtl, DigComp)

Type: **Logical**
Lock: **View**
Default: **Off**
PtRes **HPM**
Range: **Off** (No uncommanded change alarm)
On (Uncommanded change alarm has been detected by this point)

Uncommanded Change Alarm Flag—Indicates whether an uncommanded change has been detected in the field device. (Field device has changed its state without a command.)

UNIT

Type: **String_2**
Lock: **PtBld**
Default: **N/A**
PtRes **NIM**

Unit Identifier—Defines the process unit to which this point is assigned. The unit identifier is originally assigned during network configuration, and it appears in displays and listings throughout the system.

Restriction: Two characters are required; blanks are not allowed. For example, unit 3 must be entered as 03.

CL and Picture Editor — An integer is returned. This number is equivalent to the unit position in the Unit Names configuration list.

Range: **A-Z, 0-9** (up to 100 unit IDs can be configured)

UPGRADE (UCN)

Type: **E:UPGRADE** **NIM Upgrade Status**
Lock: **Oper**
Default: **OK**
PtRes **NIM**
Range: **OK** (NIM has not been upgraded and is OK)
Upgrade (NIM is upgraded and is questionable)

URL (STI)

Type: **Real**
Lock: **Eng/View**
Default: **NaN**
PtRes: **HPM**

Upper Range Limit—Indicates the upper range limit of the PV at the Smart Transmitter. This limit is a fixed limit and cannot be changed. Refer to the description of the STI_EU parameter for the URL engineering units. During configuration, the value entered for this parameter must agree with the URL value of the transmitter. Although any value can be entered during configuration, a database mismatch will occur when the point is put on-process because the transmitter's URL value and the STI IOP's URL value are not the same. If the values are not the same, the STATE parameter value becomes DBChange and PVSTS becomes Bad. Refer to URL in the *PM/APM Smartline Transmitter Integration Manual* for more information.

The corresponding LRL parameter is not a configurable parameter at the Universal Station.

The upper range limits for the Smart Transmitters are as follows:

For the ST3000 Smart Pressure Transmitters (Spt_Dp, Spt_Gp and Spt_Ap):

Xmtr Range	*URL (In H_2O)*
400 inH_2O	400.0
600 inH_2O	600.0
780 mmHga	400.0
100 PSI	2768.0
200 PSI	5536.13
500 PSI/A	13840.34
1500 PSI	41521.0
2000 PSI	55361.35
3000 PSI	83042.02
6000 PSI	166084.0
10000 PSI	276806.7

For the STT3000 Smart Temperature Transmitter (STT):

Sensor Type (PVCHAR)	*URL (in Degrees C except where noted)*
Linear (mV)	1000 mV
Thermocouples:	
B	1820
E	1000
J	1200
K	1370
NiNiMoTC	1300
N	1300
R	1760
S	1760
T	400
W5W26TC	2300
W3W25TC	2300

URL (STI) (continued)

RTDs:

Cu10RTD	250
Cu25RTD	250
DINRTD	850
JISRTD	640
NicklRTD	150
Pt200	850
Pt500	850
RHRad	1800
RTD (ohms)	4000 Ω

For the MagneW 3000 Magnetic Flowmeter (Sfm):

$$\text{URL (in meters}^3\text{/hour)} = \frac{\pi D^2}{4x10^6} \text{ x } 3600 \text{ x } (N + 1)$$

where: D = the detector diameter in millimeters as follows: 2.5, 5, 10, 15, 25, 40, 50, 80, 100, 150, 200, 300, 350, 400, 500, 600, or 700
N = the number of dummy submerged detectors, from 0 to 9

Range: **N/A, NaN**

URV (STI)

Type: **Real**
Lock: **Supr/View**
Default: **NaN**
PtRes: **HPM**
Range: **N/A, NaN**

Upper Range Value—Defines the upper end of the operating range for the PVRAW value. Refer to the description of the STI_EU parameter for the URV engineering units.

Although the following maximum values can be entered, values greater than the URL are not recommended and accuracy is not guranteed in such cases.

For a pressure transmitter (Spt): URV_{max} = 2.0 x URL
For a temperature transmitter (Stt): URV_{max} = 2.0 x URL
For a magnetic flow transmitter (Sfm): URV_{max} = 12.0 x URL

This parameter is a view-only parameter when the STI point execution state PTEXECST is Active (indicating that changes cannot be made in this parameter value from the Universal Station).

USERID (Array, DevCtl, DigComp, ProcMod, RegCtl, RegPV)

Type: **String_16**
Lock: **Oper**
Default: **Dashes**
PtRes **HPM**
Range: **16 Character String**

User ID Reservation—The user ID that currently has reserved this point. The User ID can be changed by either a point, program, or operator. The operator can overwrite the USERID parameter at anytime. A program can store a nonblank string in this parameter only if it is blank. If the USERID string starts with three or more dashes (- - -), only the operator can overwrite the ID.

UTSDRIFT (HPM Box, NIM)

Type: **Integer**
Lock: **View**
Default: **N/A**
PtRes **HPM, NIM**
Range:

UCN Node Clock Drift—Indicates the current HPMM clock drift rate, calculated by averaging the LCN clock interval and SYNCH CLOCK interval over multiple synchs. Averaging does not occur until UCN time synchronization is in a steady state.

UTSNODE (HPM Box, NIM)

Type: **Integer**
Lock: **View**
Default: **0**
PtRes **HPM, NIM**

Last UCN Syncher Node—Describes which node is the synch master or syncher. Normally, this is the primary NIM, even though the secondary NIM can also be the syncher. The syncher function performs periodic time synchronization on the UCN.

Range: **0, 1** to **64** (0 specifies **No Syncher Node**)

UTSTBCRV (HPM Box, NIM)

Type: **String_2**
Lock: **View**
Default: **N/A**
PtRes **HPM, NIM**
Range: **5** to **15**

TBC Revision—The token bus controller revision number in hexidecimal format**.**

UTSTIME (HPM Box, NIM)

Type: **Time**
Lock: **View**
Default: **N/A**
PtRes **HPM, NIM**
Range: **N/A**

Current Time in LCN Node—Identifies the current time of day for this LCN node, and is useful if there are multiple LCNs or UCNs.

UTSTIMST (HPM Box, NIM)

Type: **E:$UCNTMST**
Lock: **View**
Default: **0**
PtRes **HPM, NIM**

Timesynch State of the UCN Node—The state of time synchronization for each UCN node.

Range: 0-**Initial** (Waiting for the first complete synch operation)
1-**Failed** (The maximum amount of time has elapsed and no synch operation has occurred, or the NIM does not have a functioning EPNI board)
2-**Degraded** (In nonsyncher NIMs and HPMMs, an excessive amount of time has elapsed without a complete synch operation. In HPMMs, the drift limit between the LCN and HPMM clock has been exceeded)
3-**LCN_Bad** (Synch operations are taking place on a regular basis, but the NIM's clock is not synched with the LCN)
4-**LCN_OK** (Synch operations are occurring regularly and the NIM's clock is properly synched with the LCN)
5-**OK** (Synch operations are working in an optimal manner)

-W-

WARMSTRT(1)–(168)

Type: **Logical** **Warm Start Flag**
Lock: **View**
Default: **N/A**
PtRes **IOP**
Range: **On** (Warm start executed)
Off (Cold start executed)

WEEKDAY (HPPM Box)

Type: **Integer** **Current Weekday—**The current weekday based on LCN wall clock time.
Lock: **View**
Default: **N/A**
PtRes **HPM**
Range: **1** to **7** (Sunday to Saturday)

WITHBIAS(1)–(40) (HPPM Box)

Type: **Logical**
Lock: **View**
Default: **ON**
PtRes **HPM**

I/O Module Physical Bias State—ON Indicates that the preferred primary is really the acting primary; OFF indicates that it is not. For IOPs that do not have hardware bias (e.g., HLAI, DI, etc.), the status of ON is always returned. Applies to primary IOP only.

Range: **On** (The preferred primary is the acting primary)
Off (The preferred primary is not the acting primary)

-X-

X (FlowComp)

Type: **Real**
Lock: **View**
Default: **1.0**
PtRes **HPM**
Range: **≥ 0.0**

Steam Compressibility Input—Indicates the measured actual steam compressibility.

X1 (AutoMan)

Type: **Real**
Lock: **Prog**
Default: **NaN**
PtRes **HPM**
Range: **NaN**

X1 Input Value to be Biased

X1–3 (MulDiv)

Type: **Real**
Lock: **Prog**
Default: **NaN**
PtRes **HPM**
Range: **≥ 0.0, NaN**

Inputs 1-3—Current values of the inputs to this algorithm.

X1–4 (IncrSum, ORSel, RegCtl Summer)

Type: **Real**
Lock: **Prog**
Default: **NaN**
PtRes **HPM**
Range: **≥ 0.0, NaN**

Inputs 1-4—Current values of the inputs to this algorithm.

X2 (AutoMan)

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes **HPM**
Range: **N/A**

Bias Adjustment Input

X2 (RatioCtl)

Type: **Real**
Lock: **View**
Default: **N/A**
PtRes **HPM**
Range: **N/A**

Input Number 2—Indicates the value of the uncontrolled process variable. Source should be the same as for P2 of the Calcultr algorithm, if it is being used in conjunction with the Calcultr algorithm.

X2FILT (RatioCtl)

Type:	**Real**	**Filtered value of the X2 input**
Lock:	**View**	
Default:	**N/A**	
PtRes	**HPM**	
Range:	**N/A**	

> *Helpful Hint:* Filter time is determined by X2TF. This filter is only active if the point is in the AUTO or CASC mode.

X2TF (RatioCtl)

Type:	**Real**	**X2 input filter lag time in minutes**
Lock:	**Supr**	
Default:	**0.0**	
PtRes	**HPM**	
Range:	**0 - 60 minutes**	

XEUHI (AutoMan, IncrSum, ORSel)

Type:	**Real**	**X Input Engineering Unit High Range—**Defines the upper limit of the value of the X input or inputs.
Lock:	**Engr**	
Default:	**100.0 (equivalent to 100%)**	
PtRes	**HPM**	
Range:	**> XEULO**	

XEULO (AutoMan, IncrSum, ORSel)

Type:	**Real**	**X Input Engineering Unit Low Range—**Defines the lower limit of the value of the X input or inputs.
Lock:	**Engr**	
Default:	**0.0 (equivalent to 0%)**	
PtRes	**HPM**	
Range:	**< XEUHI**	

XSTS (FlowComp)

Type:	**E:PVVALST**	**X Input Value Status—**Status of the steam compressibility input.
Lock:	**View**	
Default:	**Normal**	
PtRes	**HPM**	
Range:	0-**Bad** (Value is bad and replaced with NaN) 1-**Uncertn** (Status of the value is uncertain) 2-**Normal** (Value is good)	

-Y-

YEAR (HPM Box)

Type: **Integer**
Lock: **View**
Default: **N/A**
PtRes **HPM**
Range: **1979** to **2115**

Current Year—The value of the LCN date in the HPM.

FAX Transmittal — FAX No.: (602) 313-4842

TO: **Andrea Simmons** ______________________

Total FAX pages: _____
(including this page)

Reader Comments

Title of Document: **HPM Parameter Reference Dictionary**

Document Number: **HP09-540** Issue Date: **8/97**

Comments: ____

____

____

____

____

____

Recommendations: ____

____

____

____

____

____

FROM:

Name: ______________________________ **Date:** ______________

Title: ____

Company: ____

Address: ____

City: ______________ **State:** ______________ **ZIP:** ______________

Telephone: ______________ **FAX:** ______________

You may also call 800-822-7673 (available in the 48 contiguous states except Arizona; in Arizona dial 602-313-5558, or write to:

Honeywell Inc.
Industrial Automation and Control
Automation College
2820 West Kelton Lane
Phoenix, AZ 85023-3028

Honeywell

Industrial Automation and Control

Automation College

2820 W. Kelton Lane

Phoenix, AZ 85023-3028

Helping You Control Your World