



PID Controls in a PLC Environment

Robert Zaun – Rockwell Automation

ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Successfully configure PID blocks in a PLC environment

1. Select Algorithm and Control Strategy

2. The PLC Environment

3. Configure Correctly

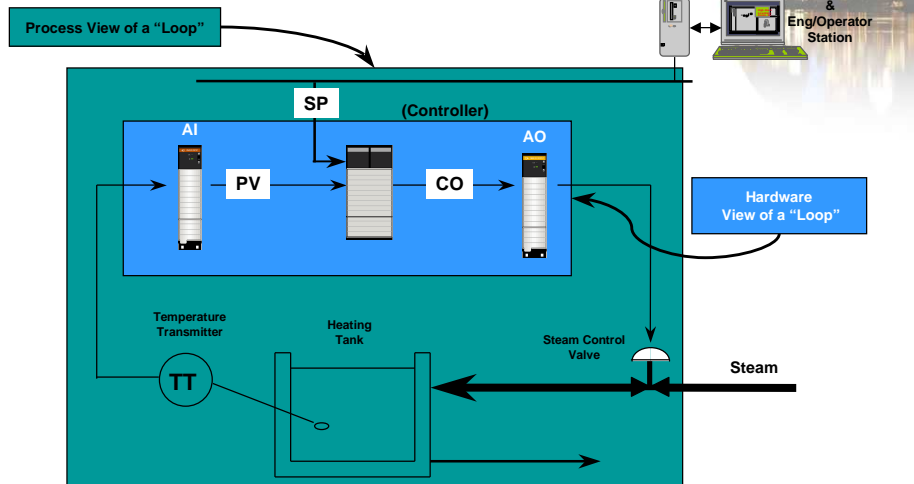
4. Timing

ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

The Process Control “Loop”



ExpertTune®

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

What Do We Know?

PV - Actual measured value which we are trying to control
Error - The difference between where we are and where we want to be

Δt - Knowledge of 'when' the PV was observed

ExpertTune®

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation



Successfully configure PID blocks in a PLC environment

1. Select Algorithm and Control Strategy

2. The PLC Environment

3. Configure Correctly

4. Timing

ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation



Loop Control Strategy

- On/Off Control
- Open Loop
- Closed Loop
- Cascade Control

ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Loop Control Algorithms in ControlLogix

- PID
- Enhanced PID (PIDE)

The tag description goes here

tag	
PIDE	
Enhanced PID	
<input type="checkbox"/> PV	<input type="checkbox"/> CVEU
<input type="checkbox"/> SPProg	<input type="checkbox"/> SP
<input type="checkbox"/> SPCascade	<input type="checkbox"/> PVHHAAlarm
<input type="checkbox"/> RatioProg	<input type="checkbox"/> PVHAlarm
<input type="checkbox"/> CVProg	<input type="checkbox"/> PVLAlarm
<input type="checkbox"/> FF	<input type="checkbox"/> PVLAlarm
<input type="checkbox"/> HandFB	<input type="checkbox"/> PVROCPoSAlarm
<input type="checkbox"/> ProgProgReq	<input type="checkbox"/> PVROCNegAlarm
<input type="checkbox"/> ProgOperReq	<input type="checkbox"/> DevHHAAlarm
<input type="checkbox"/> ProgCasRatReq	<input type="checkbox"/> DevHAlarm
<input type="checkbox"/> ProgAutoReq	<input type="checkbox"/> DevLAlarm
<input type="checkbox"/> ProgManualReq	<input type="checkbox"/> DevLAlarm
<input type="checkbox"/> ProgOverrideReq	<input type="checkbox"/> ProgOper
<input type="checkbox"/> ProgHandReq	<input type="checkbox"/> CasRat
	<input type="checkbox"/> Auto
	<input type="checkbox"/> Manual
	<input type="checkbox"/> Override
	<input type="checkbox"/> Hand

ExpertTune®

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Enhanced PID in Function Block Diagramming Editor

Velocity form algorithm

Full set of modes

Easy selection of timing modes

Easy handling of PV/CV faults

Full bumpless transfer into and out of Cascade mode

Standard PID in Ladder Editor

Positional form algorithm

Very limited set of modes

No timing modes

No handling of PV/CV faults

No bumpless transfer into and out of Cascade mode

ExpertTune®

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Velocity vs. Positional PID Algorithm

The PIDE instruction uses a “velocity” PID algorithm.

- Works on change in error.
- Same type of algorithm used by most DCS systems
- Useful when using adaptive gains

PLC-5 and Logix ladder PID instruction use a “positional” PID algorithm

- Works on error directly

ExpertTune

TiPS
INCORPORATED

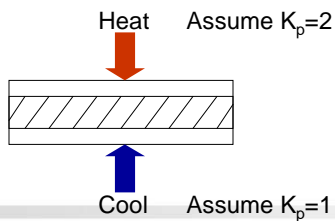
2007 User Conference
Build a Stronger Automation Foundation

Advantages of Velocity Algorithm

Simple to implement adaptive gains.

Example: Extrusion machine barrel temperature control with heating and cooling.

- Heating and cooling are different processes and require different gains

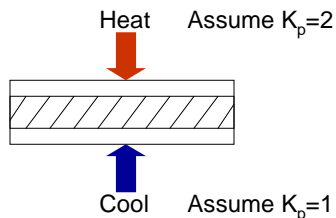


ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Adaptive Gains with Velocity Algorithm



Assume that when we switch from cool to heat the error is 5%.

$$CV = K_p E + \sum K_I E \Delta t + K_D \frac{\Delta E}{\Delta t}$$

Positional algorithm has an immediate 5% bump in output. (This is bad.)

$$CV_n = CV_{n-1} + K_p \Delta E + K_I E \Delta t + K_D \frac{E_n - 2E_{n-1} + E_{n-2}}{\Delta t}$$

Velocity algorithm has no bump in output. (This is good.)

ExpertTune

TIPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Enhanced PID

Velocity form algorithm allows for easier adaptive tuning and other advanced control schemes.

Selection of independent and dependent forms of the PID equation.

Multiple modes available:

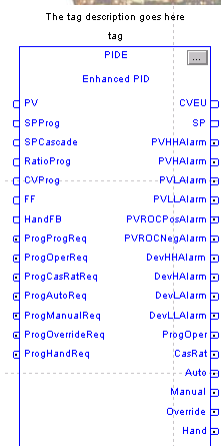
- Program/Operator control.
- Separate modes for Cascade/Ratio, Auto, Manual, Override, and Hand.

User selectable derivative smoothing.

Zero crossing deadband control.

Reset feedback Boolean inputs and outputs to prevent reset windup when used with external control algorithms.

Additive feedforward control.



ExpertTune

TIPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

PID Equation .PE

Independent (default)

$$CV = K_p \cdot E\% + \sum K_i \Delta_t E\% + K_d (d/dt)E\% + \text{Bias}$$

Dependent (ISA)

$$CV = K_p [E\% + \sum (1/T_i) \Delta_t E\% + T_d (d/dt)E\%] + \text{Bias}$$

ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Successfully configure PID blocks in a PLC environment

1. Select Algorithm and Control Strategy

2. The PLC Environment

3. Configure Correctly

4. Timing

ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Function Block Diagrams

You 'Configure' your control scheme

- The Enhanced PID algorithm (PIDE) in a ControlLogix Controller is a Function Block
- A control scheme is 'drawn'
- Blocks encapsulate enough functionality to accomplish all tasks expected in the control scenario

Values 'flow' from 'upstream' to 'downstream'

- A block functionally 'pulls' its values from preceding (upstream) blocks ... no 'MOV's or anything else that would 'push'

ExpertTune

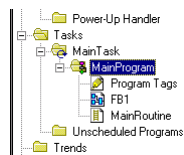
TIPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Function Block Routine

Co-exists with Ladder routines

Function Block routines follow the same organizational rules as Ladder routines with respect to Tasks and Programs



ExpertTune

TIPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Function Block Routines

Routines can have multiple sheets

Sheets help compartmentalize functionality

Routines are verified at the routine level (inclusive of all sheets in the routine)

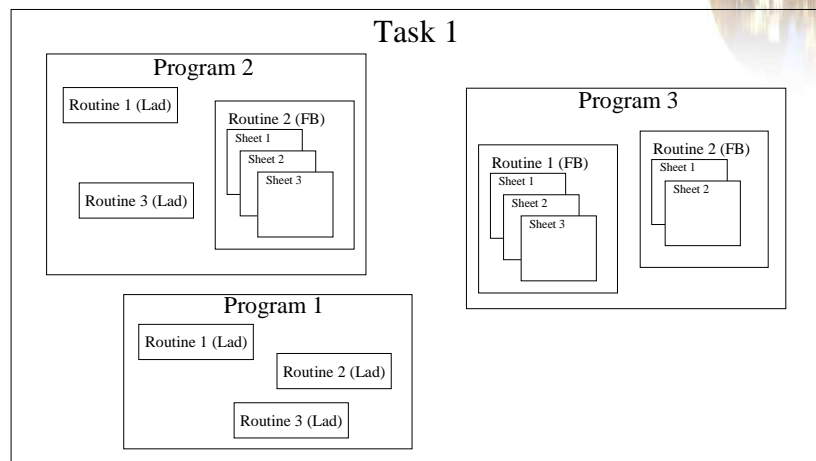
Number of sheets is limited only by memory

ExpertTune

TIPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Function Block Routines



ExpertTune

TIPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Block Order of Execution

Order of execution is generated automatically at 'compile' time

Execution ordering begins at an Input Reference and proceeds downstream

Ordering guarantees that all 'upstream' calculations occur first

ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Successfully configure PID blocks in a PLC environment

1. Select Algorithm and Control Strategy

2. The PLC Environment

3. Configure Correctly

4. Timing

ExpertTune

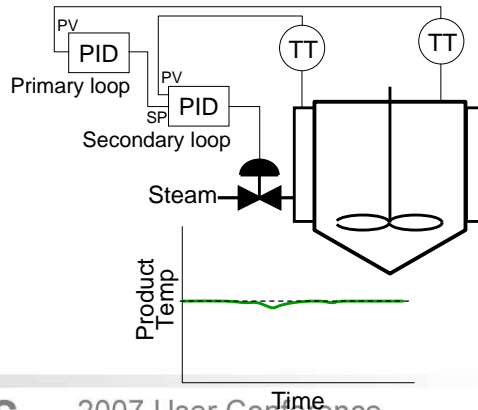
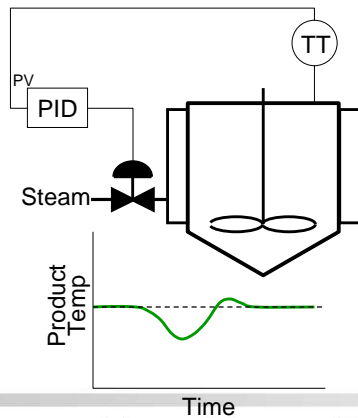
TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Cascade Control Example

Without cascade control, a drop in steam pressure causes the product temp to drop. The PID loop then responds to increase steam flow and get the temp back to setpoint.

With cascade control, a drop in steam pressure causes the *jacket* temp to drop. The secondary (inner) PID loop will then respond to increase steam flow and get the *jacket* temp back to setpoint before the *product* temp is seriously affected.



ExpertTune®

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Cascade Control Considerations

The secondary loop's response time must be less than the primary loop's.

- Otherwise, disturbances to the controlled variable cannot be compensated for by the secondary loop before they affect the primary loop.

Secondary loop is final control element of primary loop

Some provisions must exist for operator decoupling of loops

- Tuning
- Startup

ExpertTune®

TiPS
INCORPORATED

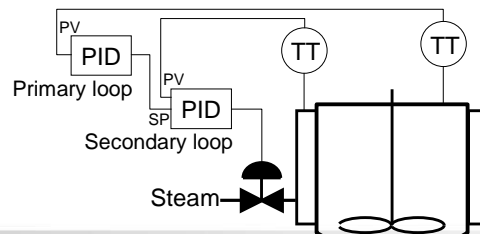
2007 User Conference
Build a Stronger Automation Foundation

Cascade Control Considerations

Must tune one loop at a time (inner first)

To tune a cascaded pair of loops:

- Place primary loop in manual
- Tune the secondary loop
- Put the secondary loop into Cascade mode.
- Tune the primary loop



ExpertTune®

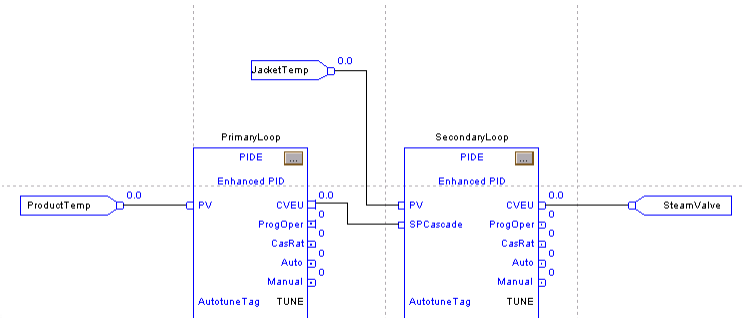
TIPS INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

PIDE Instruction and Cascade Control

The PIDE instruction has built-in cascade control capabilities.

- Cascade mode on Secondary loop is distinct from Auto mode.



ExpertTune®

TIPS INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

PIDE Configuration for Cascade Control

Set AllowCasRat true to enable Cascade/Ratio mode.

Make sure CVEUMax and CVEUMin match the engineering unit range of the secondary loop's SP.

Vis	Name	Value	Type
<input type="checkbox"/>	CVManLimiting	0	BO
<input type="checkbox"/>	CVEUMax	100.0	RE
<input type="checkbox"/>	CVEUMin	0.0	RE
<input type="checkbox"/>	CVEUMin	100.0	RE

Vis	Name	Value	Type
<input type="checkbox"/>	DevDeadband	0.0	REA
<input type="checkbox"/>	AllowCasRat	1	BOC
<input type="checkbox"/>	ManualRetrn	0	IRNF

Exp
INCORPORATED

Build a Stronger Automation Foundation

PIDE Anti-windup

PID instruction avoids reset windup by preventing the integral term from accumulating when the CV output reaches its maximum or minimum limits

- continues to calculate the proportional and derivative term.

The accumulated integral term remains frozen until the CV output drops below its maximum limit or rises above its minimum limit.

PIDE Configuration for Anti-windup

Turn on visibility of WindupHIn, and WindupLIn pins.

Turn on visibility of WindupHOut and WindupLOut pins.

PIDE Properties - PrimaryLoop

Vis	Name	Value	Type
<input type="checkbox"/>	HandFBFault	0	BOOL
<input checked="" type="checkbox"/>	WindupHIn	0	BOOL
<input checked="" type="checkbox"/>	WindupLIn	0	BOOL
<input type="checkbox"/>	ControlAction	0	BOOL

PIDE Properties - SecondaryLoop

Vis	Name	Value	Type
<input type="checkbox"/>	EPPercent	0.0	REAL
<input checked="" type="checkbox"/>	InitPrimary	1	BOOL
<input checked="" type="checkbox"/>	WindupHOut	0	BOOL
<input checked="" type="checkbox"/>	WindupLOut	0	BOOL
<input type="checkbox"/>	ProgOper	1	BOOL

ExpertTune INCORPORATED Build a Stronger Automation Foundation

PIDE Instruction and Ratio Control

The PIDE instruction has built-in ratio control capabilities.

Set AllowCasRat true to allow Cascade/Ratio mode.

Use the controlled flow as the PV.

Wire the uncontrolled flow into the SPCascade parameter.

Set UseRatio true to tell the loop to multiply SPCascade by the Ratio value. Ratio values can come from an operator display or the program.

PIDE Properties - RatioController

Vis	Name	Value	Type
<input type="checkbox"/>	DevDeadband	0.0	REAL
<input checked="" type="checkbox"/>	AllowCasRat	1	BOOL
<input type="checkbox"/>	ManualAffecting	0	BOOL

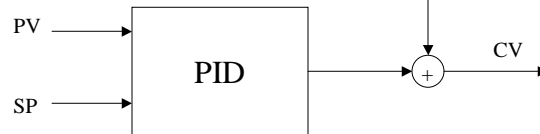
PIDE Properties - RatioController

Vis	Name	Value	Type
<input type="checkbox"/>	UseRatio	1	BOOL
<input checked="" type="checkbox"/>	RatioProg	1.0	REAL
<input type="checkbox"/>	RatioOper	0.5	REAL
<input type="checkbox"/>	RatioLimit	1.0	REAL
<input type="checkbox"/>	RatioLimit	0.0	REAL
<input type="checkbox"/>	RatioLimit	0	REAL

ExpertTune TiPS INCORPORATED 2007 User Conference Build a Stronger Automation Foundation

PID - FeedForward

Used to account for large, known disturbances in control scenario
PID used in Feedback Trim arrangement



FF Disturbance must be known and modeled explicitly

ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Feed Forward Control Considerations

Reevaluate potential disturbances

- Inlet flow and temperature

Adjust guiding principles for unknowns

- Heat losses
- Mechanical efficiencies

Adjust Feed Forward signal to compensate for lags or leads

- Principles based on steady state
- Lags in controlled variable

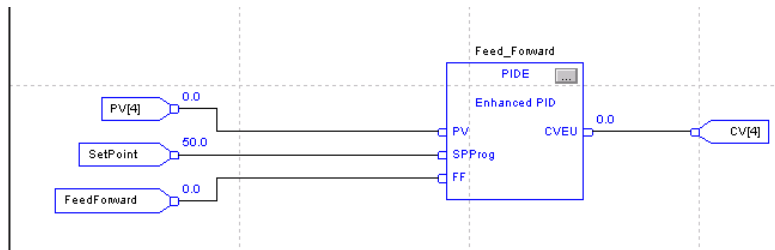
ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

PIDE Instruction

The PIDE instruction has built-in Feed Forward capabilities.



ExpertTune®

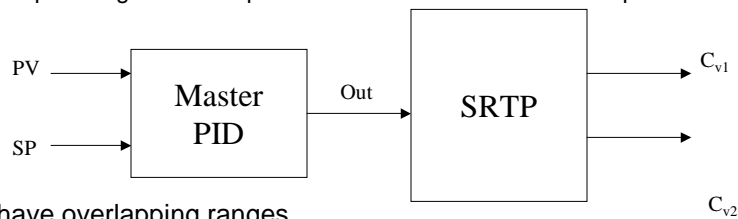
TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Split Range

Coupled Function Blocks

- Master PID: Driving Output
- Split Range Time Proportional: Determines who sees output



Can have overlapping ranges

ExpertTune®

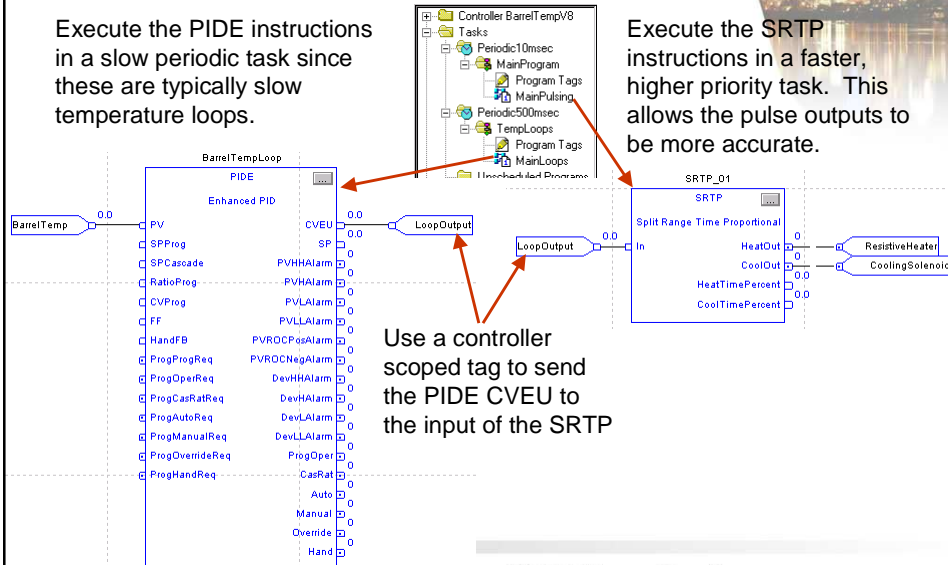
TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Heat/Cool Temperature Loops

Execute the PIDE instructions in a slow periodic task since these are typically slow temperature loops.

Execute the SRTP instructions in a faster, higher priority task. This allows the pulse outputs to be more accurate.



Use a controller scoped tag to send the PIDE CVEU to the input of the SRTP



2007 User Conference
Build a Stronger Automation Foundation

Successfully configure PID blocks in a PLC environment

1. Select Algorithm and Control Strategy
2. The PLC Environment
3. Configure Correctly
4. Timing



2007 User Conference
Build a Stronger Automation Foundation

Update Time .UPD

A floating point entry in Seconds which represents the update rate of the analog input (PV), and necessarily the execution rate of the PID instruction

$$\Delta_t$$

*Must be non-zero or a proc fault will occur

ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation


Δt

Note that the timing reference plays a major role in the calculation of both the I and D components

ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation



It is assumed that the analog input data (PV) is associated in time with this reference

Integral and Derivative component calculations use this value... if it's wrong, they're wrong!

ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Timing Modes

For the PID instruction the user must synchronize execution of the instruction with the sampling of the process variable (PV)

- Put PID instruction in a 'periodic task' to guarantee the update rate
- Insure the process variable is updated at a significantly (5 to 10x) faster rate than the periodic task

ExpertTune

TiPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

FB - Timing Modes

TimingMode = 0 Periodic (Default)

- Default mode of operation
- Will 'pick up' on scheduling type and set dT appropriately
- Preferred mode of operation (least amount of work)

TimingMode = 1 OverSample

- User defines scheduling and dT
- PLC/SLC way of doing things

TimingMode = 2 Real Time Sampling

- Uses analog time stamp for dT

ExpertTune

TIPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Periodic Timing Mode Example

Periodic mode is intended to be used by placing the block into a periodic task.

Since the timing mode is Periodic (TimingMode=0), DeltaT is automatically set to the task update time of 0.1 seconds.

The screenshot displays the ExpertTune software interface. On the left, a project tree shows a 'MainTask' containing a 'LoopControl' block. Below it, the 'Task Properties - MainTask' dialog is open, showing 'Type: Periodic' and 'Watchdog: 500.000 (ms)'. In the center, a ladder logic diagram shows a 'PIDE_01' block (Enhanced PID) with various inputs and outputs. On the right, two 'PIDE Properties - PIDE_01' dialog boxes are shown. The top one shows 'TimingMode' set to 0 and 'DeltaT' set to 0.1. The bottom one shows 'Override' and 'Hand' checked, and 'DeltaT' set to 0.1. Red arrows point from the text above to these specific settings in the dialog boxes.

ExpertTune

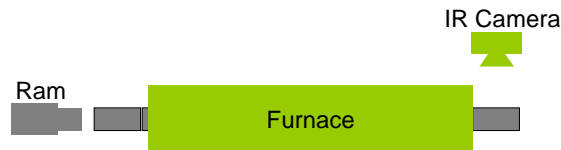
TIPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Oversample Timing Mode Example

Eddy current furnace heats steel ingots.

- New ingot is pushed into furnace approximately every 28 seconds.
- Infrared camera measures temperature every 28 seconds when heated ingot is pushed out of furnace.
- Since the temperature samples from the IR camera come at long, irregular intervals through a serial link, this loop is very difficult to implement with the periodic timing mode.



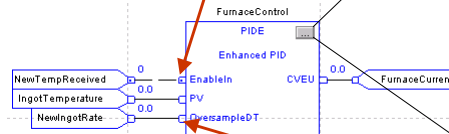
ExpertTune

TIPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Oversample Timing Mode Example

Turn on visibility for EnableIn pin and wire to the Boolean signal indicating "new temperature received" from IR camera.



PIDE Properties - FurnaceControl

Vis	Name	Value	Type
<input checked="" type="checkbox"/>	EnableIn		1 BOO

Set TimingMode to 1 for Oversample mode.

PIDE Properties - FurnaceControl

Vis	Name	Value	Type
<input type="checkbox"/>	ProgValueReset	0	BOO
<input type="checkbox"/>	TimingMode	1	DIN
<input checked="" type="checkbox"/>	OversampleDT	0.0	REA
<input type="checkbox"/>	RTSTime	1	ININ

Turn on visibility for OversampleDT pin and wire to setting for the "new ingot rate." (Optionally, you could just enter a constant value.)

ExpertTune

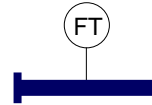
TIPS
INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Real Time Sampling Mode Example

Flow totalization using a volumetric flow meter.

- To totalize as accurately as possible, you want to execute the totalization algorithm once and only once for every analog input sample.
- Real Time Sampling lets you do this easily.

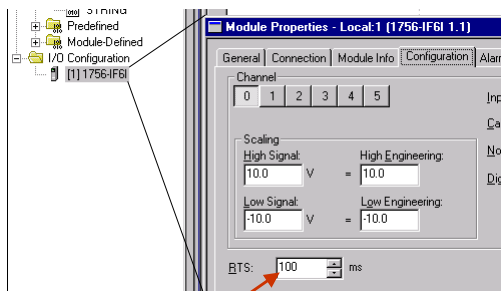


ExpertTune

TIPS INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation

Real Time Sampling Mode



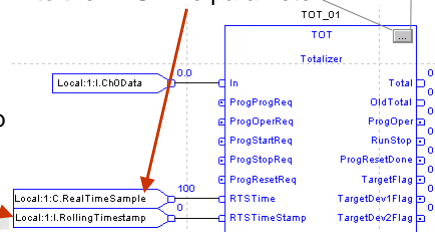
Configure the analog input module for your desired RTS rate.

Wire the RollingTimeStamp input from the analog input module to the RTSTimeStamp parameter. The totalization algorithm will now execute every time it sees the time stamp change.

Turn on visibility of RTSTime and RTSTimeStamp.

Vis	Name	Value	Type
<input type="checkbox"/>	ProgValueReset	0	BOC
<input type="checkbox"/>	TimingMode	2	DIN
<input type="checkbox"/>	OversampleDT	0.0	REA
<input checked="" type="checkbox"/>	RTSTime	1	DIN
<input checked="" type="checkbox"/>	RTSTimeStamp	0	DIN
<input type="checkbox"/>	EnableP...	0	BOC

Wire the RealTimeSample value from the analog input module configuration to the RTSTime parameter.



ExpertTune

TIPS INCORPORATED

2007 User Conference
Build a Stronger Automation Foundation