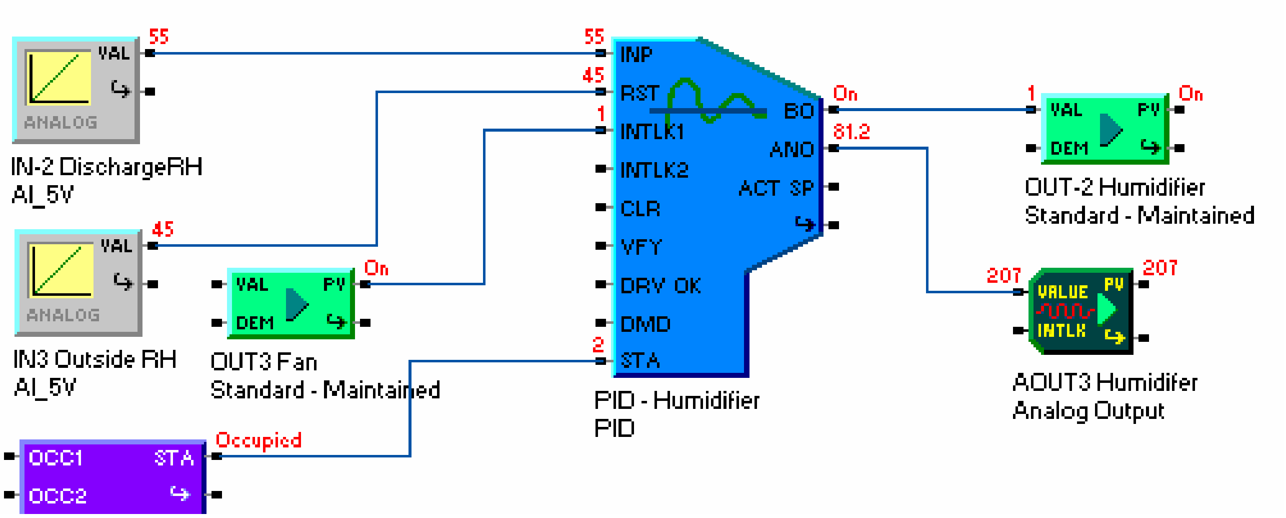
Object 18 - PID

PID Summary



Overview: The PID (Proportional-Integral-Derivative) object typically takes a measured variable such as static pressure or temperature as its control input and uses this value in a PID algorithm to control an output so as to maintain the control input at a user-chosen setpoint. The PID object has input handles for a Control Input, a Reset Input, 4 interlocks, an Output Verify, and Drive OK input. The PID has a binary and an analog output which are used to control physical outputs or to reset other PID loops.

Application Examples: If used to control duct static pressure, the PID object retrieves the pressure as its control input, calculates the output value, and places it in memory. The output object in charge of variable fan drive speed may now retrieve this value and set the voltage output to the appropriate level. If used for Discharge Air Temperature control, the PID algorithm retrieves the hot or cold duct temperature as its control input, calculates the output value, and places it in memory. The output object in charge of heating or cooling coil valve position retrieves the output value and properly sets the voltage output.

PID Algorithm: The absolute PID algorithm is much preferred, set Absolute PID Enable to Yes. The ASIC/2-7540, ASIC/2-7040 and ASIC/2-8040 have an option for either absolute or incremental PID algorithm. The incremental PID algorithm is included for backward compatibility. The PID error is the difference between the Control Input and the Active Setpoint. The PID algorithm is tunable. The proportional, integral, and derivative portions of the algorithm are calculated separately and the user may weight each portion accordingly. The calculation time between successive PID calculations and minimum and maximum limits on the output are user-adjustable. Additionally, acceleration and deceleration times may be used to control the maximum rate at which an output may change up or down.

PID Analog Output: The PID calculation of the analog output is performed at a user-adjustable interval. A smoothed control input is always in the control algorithm. When turning off an output, the PID object may be configured to step an output to minimum rather than abruptly dropping it to minimum.

PID Binary Output: A binary output is provided to turn on a fan , a pump, etc. ,as determined by the current control state and Interlock Status. The user-specified interlocks and output verify must be present for the binary output to remain on.

Initial Output Value: If the Initial Value Enable is set to yes, the PID Analog Output starts at an initial value for an Interstage Time. During this time the Error Sum is zero. At the end of the interstage time the PID algorithm takes over. If the PID loop is off then the analog output is at the Off Output Value.

State Sensitive Setpoints: Separate control setpoints may be used for each state (Occupied, Unoccupied, Morning Ready, Night Setback). The PID object checks the State object and sets its control setpoint according the current state in effect. Optionally, state may be ignored and the control setpoint set to a single value, the Occupied Control Setpoint.

Reset Input: The reset input to the PID object is used to reset the active control setpoint by an amount determined by the reset schedule. The reset schedule designates appropriate reset values proportional to the reset input.

If the reset input rises above a maximum input value, the control setpoint is adjusted by a maximum reset value. If the reset input falls below a minimum value, then the active control setpoint is reset by a minimum reset value. For example, if the outdoor air temperature is the reset input and rises above 80 F, exceeding the maximum reset value, the cold duct temperature setpoint could be reset downward by the maximum reset value of -4 F. (from 62 F to 58 F).

Demand Limit: A demand level calculated by the Demand Limit object may be monitored by the PID object. Up to 6 demand levels have a corresponding user-adjustable reset value which is added to obtain the active control setpoint.

High/Low Alarm: The user may set-up high and low alarm setpoints which causes a high or low alarm to be noted when the control input rises above or below the control setpoint by the alarm setpoint value.

Interlocks: Only 2 Interlocks Handles are available in the ASIC/2-7040 FW740A... and ASIC/2-8040 FW840A.. Interlock 3 Handle is now used to Clear Verify Alarm. Interlock 4 Handle is now used for the Initial Output Value and for the Off Output Value parameters.

Four interlock inputs are available in the ASIC/2-7000 FW700A..I with user-changeable handles. For any interlock which the user has enabled, the PID object will hold its analog output value to 0 and its binary output to off for long as the interlock is not present.

Output Verification: The PID object may be configured to check an input for a signal verifying that the PID's output has in fact turned on the piece of equipment in question. If the verification is not present after a short wait, the PID object turns off its binary outputs. After a short delay it tries to turn it on again. This is repeated up to a maximum number of attempts, at which point an alarm is set. Once this verify alarm has been set, the user must clear it for the output to be operational. The verify alarm may be cleared by communication, or by turning Clear Verify Alarm Input On and Off.

Drive OK Interlock: If the PID object sends its output to a driven device, then the PID object may be configured to check a given input to determine if the mechanical drive is operative. If the drive is not operative, then the output is forced to the Off Output Value immediately until the drive becomes operative.

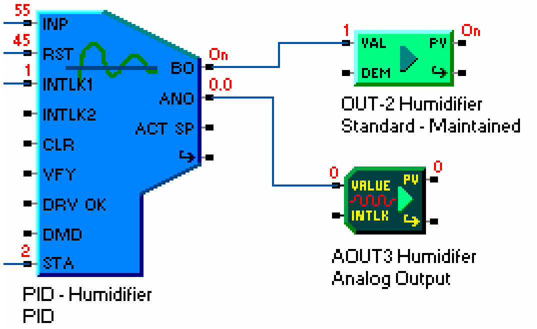
Operational Monitoring: The PID object keeps track of the starts today, which is the number of times during the current day that the output has turned on.

PID Pause Enable: A new feature causes Interlock 2 to be used to cause the PID Loop to pause in the same way as the override Action. (FW740E2.0, 840E2.0)

PID Loop Operation

Binary Output

A binary outputAttr-0 LO Bit 0 is provided to turn on a fan or a pump or the like as determined by the current control state and Interlock Status. The user-specified interlocks and output verify must be present for the binary output to remain on.



PID Analog Output

The PID calculation of the analog output is performed at a user-adjustable interval. A smoothed control input is always in the control algorithm. When turning off an output, the PID object may be configured to step an output to minimum rather than abruptly dropping it to minimum.

Initial Output Value

A new Initial Output Value feature has been added to the ASIC/2-7040 FW740A.. When the PID algorithm starts up the initial value of the Analog Output is typically zero. If Initial Value Enable is set, the PID algorithm will hold to an Initial Output Value for a period of time give by the Interstage Time. During this time the integral term will be kept to zero. Note: In the ASIC/2-7000 the Initial Output Value is Minimum Output Limit.

At the end of the interstage time the output will then go to the value calculated by the PID algorithm. The initial value feature will apply to both the absolute and the incremental form of the algorithm.

Off Output Value

An Off Output Value has been added to the ASIC/2-7040 FW740A.. If the Output Value is used for the PID Analog Output whenever the PID Binary is Off.

Interlocks

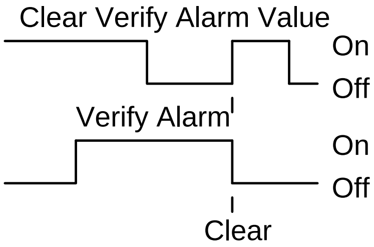
Interlocks: Two interlock inputs are available with user-changeable handles.(FW740A, 840A). For any interlock which the user has enabled, the PID object will hold its analog output value to 0 and its binary output to off for long as the interlock is not present. With FW700I four interlocks were available.

PID Pause : If Polling Pause Enable is yes, then Interlock 2 is used to cause the PID Loop to pause in the same way as the override Action. (FW740E2.0, 840E2.0) When Interlock goes true it add at least 1 second interstage to recalculate integral term so that it continues smoothly (FW7/854a1.4)

Output Verification

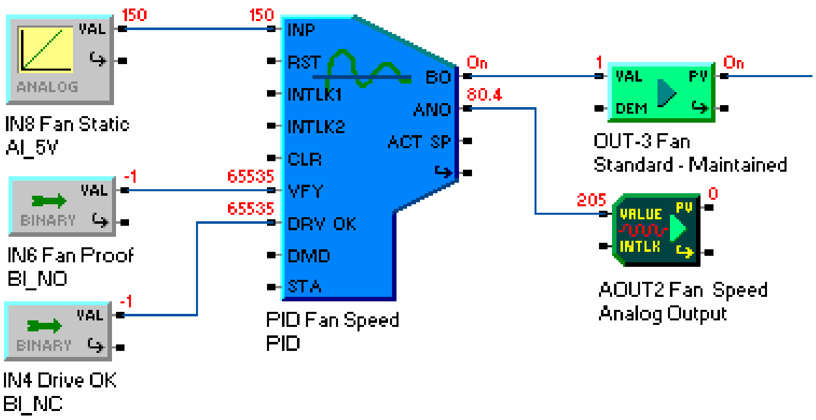
Output Verification: The PID object may be configured to check an input for a signal verifying that the PID's output has in fact turned on the piece of equipment in question.

If the verification is not present after a short wait, the PID object turns off its binary outputs. After a short delay it tries to turn it on again, restarting the interstage timer. This is repeated up to a maximum number of attempts, at which point an alarm is set. Once this Verify Alarm has been set, the user must clear it for the output to be operational.



Both the PID Verify Alarm (Attr-3, LO Bit3) and the Start Counter (Attr-13, HI Byte) must be cleared for the PID loop to restart. The PID Verify Alarm alone can be cleared with Action 3 but does not clear the Start Counter. Action 17,Clear All Alarms, clears all alarms and the Start Counter. Action 18, Fresh Start, Clears all alarms and counters.

The PID Verify Alarm may also be cleared by causing the Clear Verify Alarm Handle to go true after the alarm has been set. This clear the All Alarms and Counters.



Drive OK

Drive OK Interlock: If the PID object sends its output to a driven device, then the PID object may be configured to check a given input to determine if the mechanical drive is operative. If the drive is not operative, then the output is forced to Off Output Value until the drive becomes operative. This is different than Interlock 1 & 2. If Interlock 1 or 2 are enabled and go away, then the output will Step to Minimum before turning off. If Drive OK goes away, the output goes to Off Output Value *Immediately*.

Step to Minimum

Step to Minimum : If enabled, then the analog output steps to minimum before the output turns off. If false, then the analog output immediately goes to minimum when the output turns off. If Drive OK goes away, the output goes to Off Output Value *Immediately*.

Active Control Setpoint

Demand Reset

Reset Value

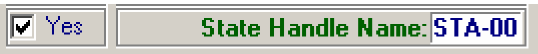
Control Setpoint

PID Active Setpoint Calculation

The Active Setpoint depends on State Sensitive Setpoints and may be reset based on a reset input. In addition the setpoint may be modified under demand limit reset. If the Control Input rises above or below the Active Setpoint by the High or Low Alarm Setpoint value for an Alarm Delay time then a High or Low Alarm is set.

State Sensitive Setpoints

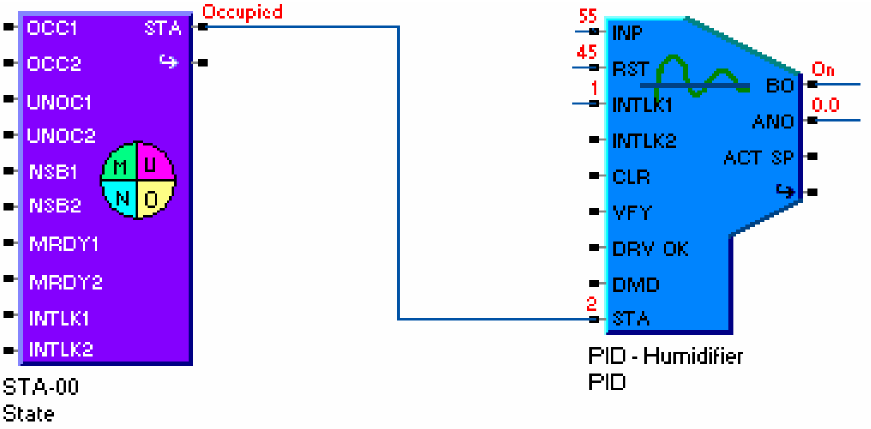
State Setpoints: If Enable by State is set to Yes, different control setpoints are used for each state (Occupied, Unoccupied, Morning Ready, Night Setback).

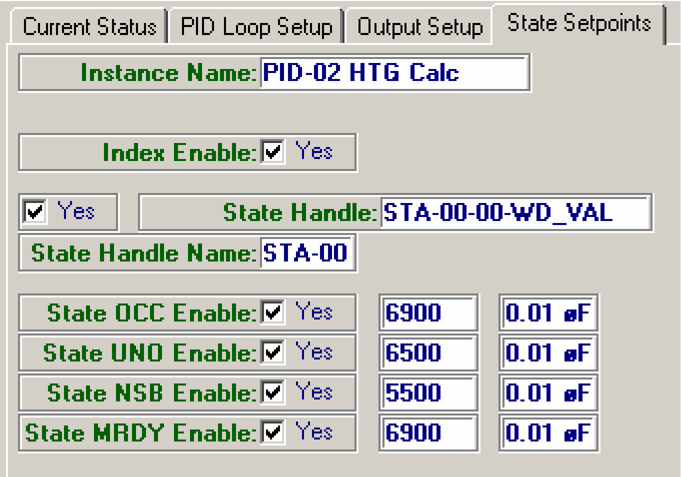


☑ Yes

State Handle Name: STA-00

The PID object checks the State object as identified by the State Handle and sets its control setpoint according to the current state in effect. Optionally, If Enable by State is set to no, state is ignored and the Occupied Control Setpoint is used.





Instance Name: PID-02 HTG Calc

Index Enable: ☑ Yes

☑ Yes

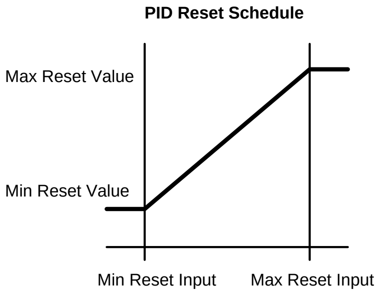
State Handle: STA-00-00-WD_VAL

State Handle Name: STA-00

State OCC Enable: ☑ Yes	6900	0.01 øF
State UNO Enable: ☑ Yes	6500	0.01 øF
State NSB Enable: ☑ Yes	5500	0.01 øF
State MRDY Enable: ☑ Yes	6900	0.01 øF

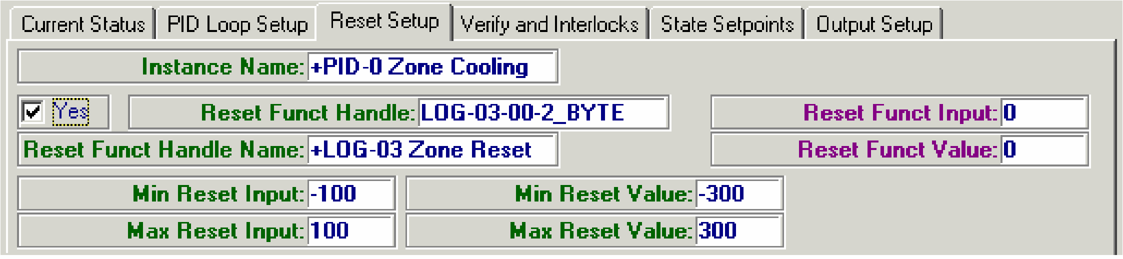
A bug in PID State Enable was fixed, so that it now ignores the value of the State Handle if PID State Enable is ”No”. (FW7/854a1.4)

Reset



Reset Input: The active control setpoint may be modified by an amount determined by the PID Reset Schedule that is built into the PID object based on a value from the Reset Function Handle. If the reset function is enabled, the active control setpoint is adjusted between the Minimum Reset Value and the Maximum Reset Value as the Reset Input goes from Minimum Reset Input to Maximum Reset Input.

If the reset input rises above a Maximum Reset Input, the control setpoint is adjusted by a signed Maximum Reset Value. If the reset input falls below a Minimum Reset Input, then the control setpoint is adjusted by a signed Minimum Reset Value.



Instance Name: +PID-0 Zone Cooling

☑ Yes

Reset Funct Handle: LOG-03-00-2_BYTE

Reset Funct Input: 0

Reset Funct Handle Name: +LOG-03 Zone Reset

Reset Funct Value: 0

Min Reset Input: -100

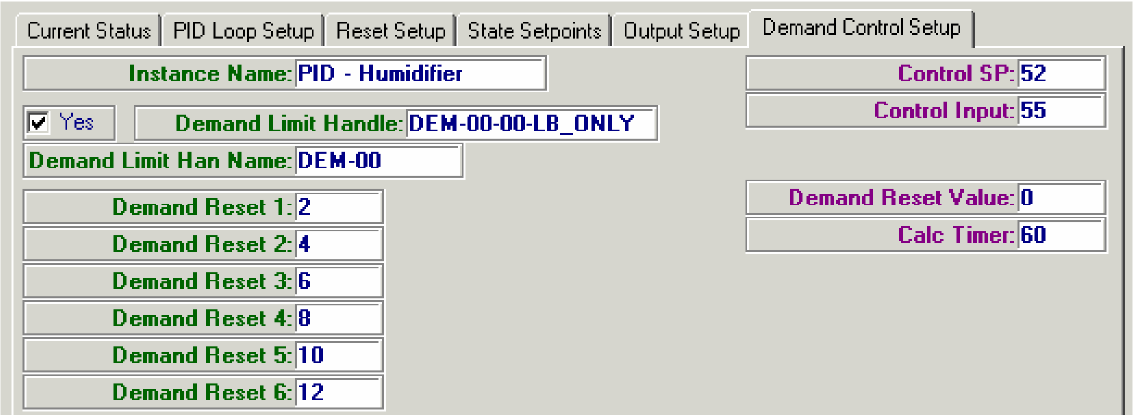
Min Reset Value: -300

Max Reset Input: 100

Max Reset Value: 300

Demand Limit Reset

Demand Limit: The active control setpoint may be modified by an amount determined by 1 of 6 Demand Resets that are built into the PID object based on a value from the Demand Limit Handle. A demand level calculated by the Demand Limit object may be monitored by the PID object. Up to 6 demand level have signed Demand Reset value assigned to them. As the Demand Level changes the corresponding demand reset value is added to the active setpoint. For example, the cooling temperature setpoint can be raised, to reduce electric power consumption as a part of a demand management strategy.



Instance Name: PID - Humidifier

Control SP: 52

Control Input: 55

☑ Yes

Demand Limit Handle: DEM-00-00-LB_ONLY

Demand Limit Han Name: DEM-00

Demand Reset Value: 0

Calc Timer: 60

Demand Reset 1: 2

Demand Reset 2: 4

Demand Reset 3: 6

Demand Reset 4: 8

Demand Reset 5: 10

Demand Reset 6: 12

PID Alarms

High/Low Control Alarm: A High or Low Control Alarm is set when enabled and when the Control Input rises above or below the Active Setpoint by the High or Low Alarm Setpoint value for an Alarm Delay time. The control alarm is cleared when the alarm condition is no longer met. This is available for use by the Alarm object. A bug required a PID Alarm Delay of at least 1 second. Also a sudden transition from High to Low Alarm now works correctly (FW7/854a1.4).

PID Override

In the ASIC/2-7040 the ability to override the PID loop output has been added to facilitate doing the open loop step test. Action 19, Override PID Loop, sets the PID Override Status and stops updating the analog output value. The Analog Output value Attr-1 may then be overridden with a new value. Action 20, Clear PID Override, restores the analog output to its control value.

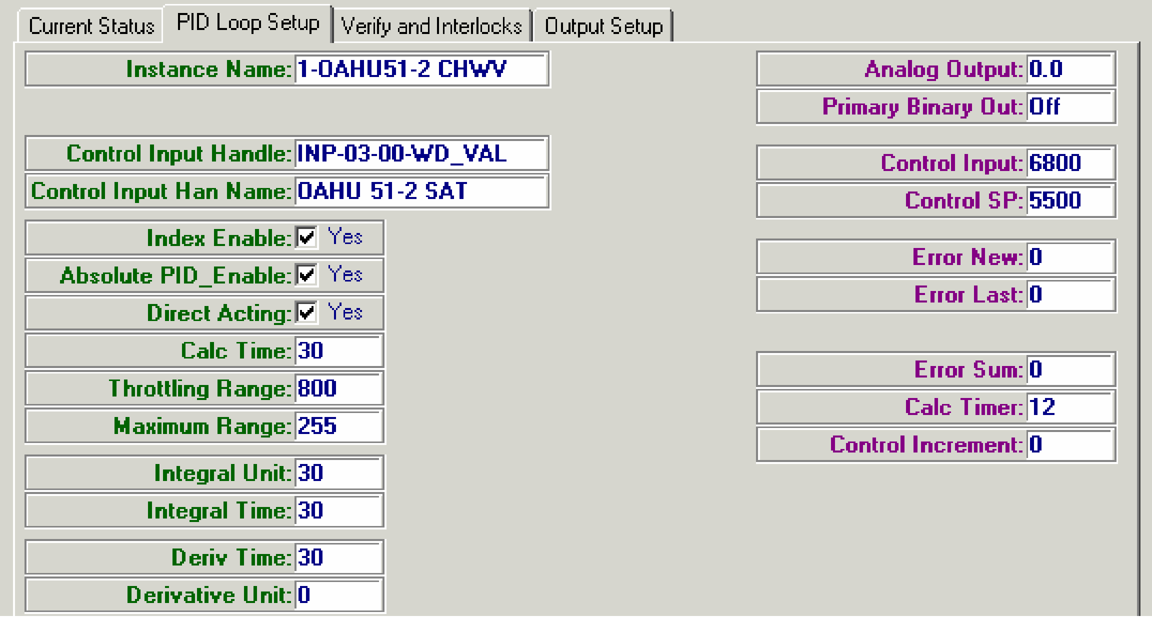
PID Algorithm(Absolute)

An absolute form of the PID control algorithm is used to calculate the output every Calculation Time. If the control is reverse acting then the direction of change is negated. If Absolute PID Enable is set to yes, the output is calculated based on Proportional plus Integral plus Derivative terms. The integral term is limited so that the final output never exceeds the Min and Max Output Limits. The acceleration and deceleration times limit the change in output by the absolute PID control algorithm at any calculation step. The final output is limited by Min and Max Output Limits.

Example: Cooling Temperature Control

A PID Loop is used to calculate the cooling requirement 0..100% (0..255) based on a 4.0 degrees F Throttling range and a Control Setpoint of 74 degrees F.

Parameter	Value
Index Enable	Yes
Direct Acting	Yes
PID Absolute Calc Enable	Yes
Throttling Range	400
Control SP Occupied	7400
Units	0.01 °F



Instance Name: 1-OAHU51-2 CHWV

Control Input Handle: INP-03-00-WD_VAL

Control Input Han Name: OAHU 51-2 SAT

Index Enable: ☑ Yes

Absolute PID_Enable: ☑ Yes

Direct Acting: ☑ Yes

Calc Time: 30

Throttling Range: 800

Maximum Range: 255

Integral Unit: 30

Integral Time: 30

Deriv Time: 30

Derivative Unit: 0

Analog Output: 0.0

Primary Binary Out: Off

Control Input: 6800

Control SP: 5500

Error New: 0

Error Last: 0

Error Sum: 0

Calc Timer: 12

Control Increment: 0

The output is cooling requirement which goes from 0 to 255 (0 to 100%). The Initial Output Value, and Step to Minimum are not used. The Off Output Value is 0.

Parameter	Value
Maximum Range	255, 100.0 %
Maximum Output Limit	255, 100.0 %
Minimum Output Limit	0 0.0 %
Initial Output Enable:	No
Initial Output Value :	0
Step to Minimum Enable:	No
PID Off Output Value:	0,0.0 %

The calculation is done every 10 seconds, with an Integral time of 60 s.

Parameter	Value
Calculation Time (s)	10 s
Integral Time	60 s
Integral Unit	10 s
Derivative Time	0 s
Derivative Unit	60 s
Acceleration Time	60 s
Deceleration Time	60 s
Interstage Time	10 s

Error

The PID Algorithm compares the Control input with the Active Setpoint to determine the error signal, E_n . This is done every Calculation Time.

E_n = (Control Input Value - Control Setpoint)

If Direct Acting is set to yes, then the output increases when the error is positive. If Direct Acting is No, then the output decreases when the error signal is positive.

Direct Acting E_n = (-1)*(Control Input Value - Control Setpoint)

The most recent two errors are saved and used for the calculation.

Error New = E_n
Error Last = E_{n-1}

In addition the Error Sum is also saved.

Error Sum_n = Error $\text{Sum}_{n-1} + E_n$

Calculation Time: The interval between successive PID calculations. In seconds; user-configurable.

Proportional Term

The **Proportional Term** is computed based on the present value of the error. and on the Proportional Gain of the system. The Proportional Gain is the ratio of the Max_Range divided by the Throttling Range.

PROPORTIONAL_n = (Max_Range/Throttle)*E_n = Proportional Gain*E_n

Throttling Range: The throttling range is that value of the sum of the integral, derivative, and proportional terms in the PID algorithm which will drive the PID Calculation from 0 to its maximum output (the PID calculation range). If proportional only control is used it indicates the deviation from setpoint that will result in a maximum output. The Throttling Range must have the same units as the controlled variable. Units depend on application; user-configurable.

Max Range: The maximum value, typically 255, which the PID object may calculate. It represents the value of 100% output in proportional only control when the error equals the throttling range. The actual PID output is between the Minimum Output Limit and Maximum Output Limit. Used internally; user-configurable.

Proportional Gain: Represents the proportional control response of the system and should be about the same size as the gain of the physical system to a change in output.

Proportional Gain = (Max_Range/Throttle)

See discussion of Tuning below.

Integral Term

The **Integral Term** is computed based on the sum of the errors. If the final output is limited by max or min then the integral term is recalculated based on those limits to prevent integral wind-up.

$\mathrm{INTEGRAL}_n$
= $(\mathrm{Max\_Range/Throttle})*(\mathrm{Integral\ Unit})/(\mathrm{Integral\ Time})*(\mathrm{Error\ Sum}_n)$
= $\mathrm{Integral\ Gain}\ *(\mathrm{Error\ Sum}_n)$

where

Integral Gain = (Max_Range/Throttle)*(Integral Unit)/(Integral Time)

Integral Time: Effectively, this is the weighting constant for the integral term used in the PID algorithm. In seconds; user-configurable.

Integral Unit: The time period between successive calculations of the integral term in the PID algorithm. Typically this has the same value as the Calculation Time. If set to zero the Integral term is zero. In seconds..

Integral Weight: The ratio of the Integral Unit to the Integral Time.
Integral Weight = (Integral Unit)/(Integral Time) .

Integral Gain : The product of the Proportional Gain and Integral Weight
Integral Weight = (Integral Unit)/(Integral Time) .
Proportional Gain = (Max_Range/Throttle)
Integral Gain = (Max_Range/Throttle)*(Integral Unit)/(Integral Time)

The integral gain should be sufficiently large that a small error signal will continue to change the output. See discussion of Tuning below.

Note: Ratios, such as (Integral Unit)/(Integral Time) are used because the controller does 16-bit integer math. It can obtain values without losing resolution by multiplying a 16-bit number x16-bit number to give a 32-bit result and then dividing the 32-bit result by a a 16-bit number to give a final 16-bit answer.

Derivative Term

The **Derivative Term** is computed based on the present and previous value of the error.

$\mathrm{DERIVATIVE}_n$
$= (\mathrm{Max\_Range/Throttle})*(\mathrm{Deriv\ Time})/(\mathrm{Deriv\ Unit})*(E_n - E_{n-1})$
$= (\mathrm{Derivative\ Gain})*(E_n - E_{n-1})$

where

(Derivative Gain)= (Max_Range/Throttle)*(Deriv Time)/(Deriv Unit)

Derivative Gain= (Max_Range/Throttle)*(Deriv Time)/(Deriv Unit)

Derivative Unit: The time period between successive calculations of the derivative term in the PID algorithm. Typically this has the same value as the Calculation Time. In seconds; user-configurable.) The derivative unit is also used to determine the maximum rate of change.

Derivative Time: Effectively, this is the proportionality constant for the derivative term in the PID algorithm. The derivative term corrects for rate of change of the error signal. The derivative term should only be used to fine tune the control response. If set to zero the derivative term is zero. In seconds; user-configurable.

PID Output

$\mathrm{Output}_n = \mathrm{PROPORTIONAL}_n + \mathrm{INTEGRAL}_n + \mathrm{DERIVATIVE}_n$

The calculated change is also saved as PID Control Increment

Attr-12 PID Control Increment (Absolute) = Outputn-Outputn-1

Rate of Change

The rate of change in output is limited for each Calculation Time.

Acceleration Time: Represents the minimum amount of time over which the Analog Output may increase by the Maximum Range. In seconds; user-configurable.

Deceleration Time: The minimum time over which the Analog Output is allowed to decrease from the Maximum Range to zero. In seconds; user-configurable.

Max Step Down = - (Calculation Time)*(Max Range)/(Decel Time)

Max Step Up = (Calculation Time)*(Max Range)/(Accel Time)

Max Step Down <= Change <= Max Step Up

Control Increment = Output_n -Output_{n-1}

If Control Increment > Max Step Up, then Control Increment = Max Step Up

If Control Increment < Max Step Down, then Control Increment = Max Step Down

Output_n = Output_{n-1} + Control Increment

Limits

The output is then limited to Max Output Limit, and Min Output Limit.

Min_Output_Limit <= Output_n <= Max_Output_Limit

If the output Value is limited by the Min or Max Output Limits then the Error Sum is recalculated at the limit.

If Max_Output_Limit<= Output_n , then Output_n = Max_Output_Limit
If Output_n <=Min_Output_Limit , then Output_n = Min_Output_Limit

Error Sum_n = (Output_n - PROPORTIONAL_n - DERIVATIVE_n)/Integral Gain
= (Output_n - PROPORTIONAL_n - DERIVATIVE_n)*
(Throttle/Max_Range) *(Integral Time/Integral Unit)

PID Tuning

PID Tuning Parameters

There are three independent parameters that can be used to tune the PID Loop: the Gain, Integral Weight, and Calculation Time.

Gain

The gain is the ratio of the change in output in counts 0..255 = 0..10 Vdc to the change in measured variable.

GAIN = MAX_RANGE/THROTTLE_RANGE.

For example if the cooling coil gives 20.00 F temperature drop when the output changes by 8 Vdc then the MAX Range = 200 the throttle range is 20.00 F = 2000 and the gain is given by

GAIN = 200/2000 = 0.10

Integral Weight

The integral weight gives the relative importance of the integral term to the proportional term in the PI calculation.

INTEGRAL WEIGHT = Integral Unit/Integral Time.

If the Integral Unit is equal to the Calculation Time, for example 30 s, then an Integral Time of 120 seconds would give

INTEGRAL WEIGHT = 30s/120 s = 0.25

An integral weight of 0.1 to 0.33 appears to work in most circumstances.

Output Change

The PID Loop calculates the change in output at each Calculation Time. The change in output is then limited by the acceleration and deceleration times, and by the maximum and minimum output limits. The output change in any Calculation Time is given by

Output Change = GAIN * (Delta_Error + Integral_Weight*Error)

The product of the Gain times the Integral_Weight times the minimum error signal to produce a control action must be greater than 1. .

Minimum Output change = GAIN*Integral_Weight*Minimum_Error > 1

or example if the minimum error is 0.5 F = 55 then and the Integral_Weight is 0.25 then the gain must be at least.

GAIN > 1/(Integral_Weight*Minimum_Error
> 1/0.25*50 = 1/12.5 = 0.08

GAIN = 200/2000 = 0.10 > 0.08

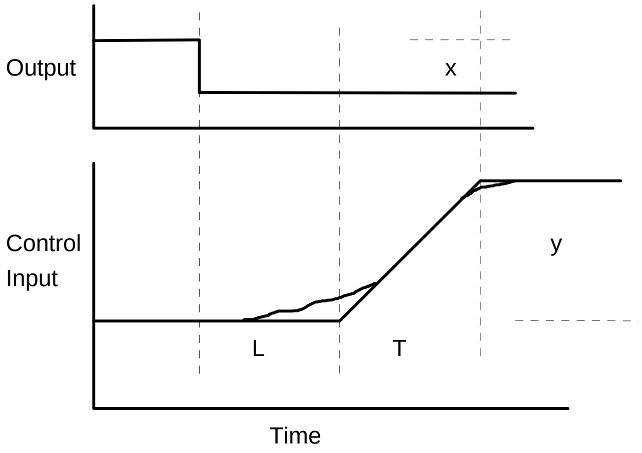
> Note: When using the incremental PID algorithm if the controller is far from setpoint then the integral term is the only one that can bring it back toward setpoint. If the integral term is too small, then the control increment calculates to zero and the controller will never get back to setpoint.

PID Open Loop Step Test

The gain of the system can be determined by using the Open Loop Step Test. The control output is overridden to a value near mid-scale and the controlled variable is measured. For example with an output of 6 Vdc.(150 counts) the Supply Air Temperature might read 58.00 F. The value is allowed to settle down without changing the output. Then the control output value is changed to a lower value, for example.4 Vdc (100 counts) The measured value, Supply Air Temperature is observed at about 10 second intervals and settles out at perhaps 64.00 F.

The ASIC/2 provides Action19, Override PID Loop, to allow overriding the PID control algorithm. If the PID loop is overridden, then the error continues to be calculated at the calculation time, but the Analog output value, and Integral Term is not updated. A new value may be written to the analog PID output value, and the response of the system observed.

Open Loop Step Test



The step test can be performed by overriding the physical output on the Analog Output Screen. The change in the control variable can be observed for example with a 5 second Calculation Time. The successive values of deviation from setpoint are displayed on the PID screen. By writing down this sequence of numbers the step response can be observed.

In the open loop step test there are 4 values that are important:

x Change in output, ΔOutput, in counts that has been imposed on the system:

y Change in control input, ΔControl Input, in raw units that results.

L Delay Time of the system before any significant change occurs, and

T Response Time of the system.

The delay time is estimated by drawing a line tangent the open loop response at or near the point of maximum slope (inflection point) and extrapolating the line to where it crosses the initial and final values as shown.

L, The Delay time is measured from the start time to where the slope crosses the initial value.

T The Response Time is measured from there to where the slope crosses the final value.

The Control Gain is given by the change in control output (in counts) divided by the change in the measured variable (in raw units).

Control Gain =x/y = (150 - 100)/(6400 - 5800) = 50/600 = 0.083

Delay Time = L = 20 seconds

Response Time = T = 25 seconds.

Calculation Time

The open loop step test will also give the time for the system to actually respond to a change in the control output. Large cooling coils can be slow acting - 30 to 60 seconds. Fans can be very fast acting. The PID Calculation Time should be about the same size as the time required for the system to respond. If the Calculation Time is shorter, the controller will over control and hunting will result. If the Calculation Time is too long, the system can move too far before control action brings it back.

The response time of the system is the time for the output value to achieve about 80% of the final value that it settles out at. This can be used to determine the PID Calculation Time. Using this criteria the Calculation Time in the above example would be 45 seconds.

Zigler Nichols Tuning Method

The Zigler Nichols Tuning criteria give the proportional gain, Integral Time and Derivative Time based on the system step response. The proportional gain, Kp = Max_range/Throttle.

Controller Type	Proportional Gain	Integral Time	Derivative Time
P	(x/y)*(T/L)	-	-
P+I	0.9*(x/y)*(T/L)	3.3*L	-
P + I + D	1.2*(x/y)*(T/L)	2*L	0.5*L

PID Tuning Values

Proportional Term : The proportional term is controlled by the throttling range which is equal to the error (offset from setpoint) that will give full output. If the throttling range is very wide, e.g. 35 F, then control response will tend to be very stable. If the throttling range is very narrow, e.g. 5 F, it will be very active and tend to go unstable.

In tuning it is best to start with the throttling range fairly wide, e.g. 15 F. It will then give a control offset, known as droop, which must be corrected using the integral time.

Using the Zigler Nichols PI criteria the Proportional Gain is the product of the Control Gain and the ratio of the Response Time to the Delay Time.

Proportional Gain = Max_range/Throttle
= 0.9* Control Gain*(Response Time/Delay Time)
= 0.9 * (x/y)*(T/L)

Using the above example
Proportional Gain = 0.9* Control Gain*(Response Time/Delay Time)
= 0.9 * (0.083)*(25s/20s) = 0.093

with Max Range = 255,:
Throttle= Max_range/Proportional Gain
Throttle= 2730 (raw)= 27.30 F

Integral Time: The Integral Time is the time over which the contribution due to the integral term for a constant error will equal to the proportional term. It is roughly the time required for the integral term to "wind up" to correct for the control offset. Typical value of Integral time could be 90 seconds. The Integral Units should be set equal to the Calculation Time.

Using the Zigler Nichols PI criteria the Integral Time is the a multiple of the Delay Time. of the Response Time to the Delay Time.

Integral Time = 3.3*Delay Time = 3.3 * L

Using the above example:

Integral Time = 3.3*Delay Time = 3.3 * 20 = 66 s
Integral Unit = Calculation Time = 45 s
Integral Weight = Integral Unit / Integral Time = 45 s/66s = 0.68

Derivative Time: The Derivative Time corrects for rate of change of the error signal. The Derivative Time should initially be set to zero. The Derivative Time should only be used to fine tune the control response. The Derivative Units should be set equal to the Calculation Time.

Using the Zigler Nichols PI criteria the Derivative Time is zero

PID Algorithm (Incremental)

A incremental form of the PID control algorithm is included for backward compatibility and is used to calculate the change in output every Calculation Time. If the control is reverse acting then the direction of change is negated. The change in output is calculated based on Proportional plus Integral plus Derivative terms. The output change between any two samples is limited by an acceleration and deceleration time. The final output is limited by Min and Max Output Limits.

Note: Use of the incremental PID is not recommended. The incremental form has the problem that if the proportional gain and integral weight are not sufficiently large then the controller can get stuck far from setpoint. (See discussion below)

Error

The PID Algorithm compares the Control input with the Active Setpoint to determine the error signal, En. This is done every Calculation Time.

E_n = (Control Input Value - Control Setpoint)

If Direct Acting is set to yes, then the output increases when the error is positive. If Direct Acting is No, then the output decreases when the error signal is positive.

Direct Acting E_n = (-1)*(Control Input Value - Control Setpoint)

The most recent three errors are saved and used for the calculation.

Error New = E_n
Error Last = E_{n-1}
Error Prev = E_{n-2}

If Direct Acting is set to yes, then the output increases when the error is positive. If Direct Acting is No, then the output decreases when the error signal is positive.

Change in Proportional Term

PROP = (E_n - E_{n-1})

Throttling Range: The throttling range is that value of the sum of the integral, derivative, and proportional terms in the PID algorithm which will drive the PID Calculation from 0 to its maximum output (the PID calculation range). If proportional only control is used it indicates the deviation from setpoint that will result in a maximum output. The Throttling Range must have the same units as the controlled variable. Units depend on application; user-configurable.

Calculation Time: The interval between successive PID calculations. In seconds; user-configurable

Max Range: The maximum value, typically 255, which the PID object may calculate. It represents the value of 100% output in proportional only control when the error equals the throttling range. The actual PID output is between the Minimum Output Limit and Maximum Output Limit. Used internally; user-configurable.

Change in Integral Term

Integral Time: Effectively, this is the weighting constant for the integral term used in the PID algorithm. In seconds;

INT = (E_n)*(Integral Unit)/(Integral Time)

Integral Unit: The time period between successive calculations of the integral term in the PID algorithm. Typically this has the same value as the Calculation Time. If set to zero the Integral term is zero. In seconds; user-configurable.

Change in Derivative Term

Derivative Time: Effectively, this is the proportionality constant for the derivative term in the PID algorithm. The derivative term corrects for rate of change of the error signal. The derivative term should only be used to fine tune the control response. If set to zero the derivative term is zero. In seconds; user-configurable. (18,X,27,WORD)

DERIV = $(E_n - 2*E_{n-1} + E_{n-2})$*(Deriv Time)/(Deriv Unit)

Derivative Unit: The time period between successive calculations of the derivative term in the PID algorithm. Typically this has the same value as the Calculation Time. In seconds; user-configurable. The derivative unit is also used to determine the maximum rate of change.

Change in Output

Change = (Max_Range/Throttle)*(PROP+INT+DERIV)

The calculated change is also saved as PID Control Increment

Attr-12 PID Control Increment = Change

Rate of Change

The rate of change in output is limited for each Calculation Time.

Acceleration Time: Represents the minimum amount of time over which the Analog Output may increase by the Maximum Range. In seconds; user-configurable. (18,X,28,WORD)

Deceleration Time: The minimum time over which the Analog Output is allowed to decrease from the Maximum Range to zero. In seconds; user-configurable. (18,X,29,WORD)

Max Step Down = - (Calculation Time)*(Max Range)/(Decel Time)
Max Step Up = (Calculation Time)*(Max Range)/(Accel Time)

Note: In the ASIC/2-7040 FW740A.. the Max Step Up and Down is based on the Calculation Time . In the ASIC/2-7000 FW700A..I it was based incorrectly on the Derivative Unit

Max Step Down <= Change <= Max Step Up

$\text{Output}_n = \text{Output}_{n-1}$ + Change

Limits

The output is then limited to Max Output Limit, and Min Output Limit.

Min_Output_Limit <= Output_n <= Max_Output_Limit

PID Glossary

PID Parameters

Absolute PID Enable

If set to Yes, Absolute PID Enable causes the PID algorithm to use a absolute PID calculation rather than the incremental calculation. This calculation has protection to limit the size of the integral term. . Yes, No; user-configurable. (18,X,18,LOBIT1)

Acceleration Time

Represents the minimum amount of time over which the Analog Output may increase by the Maximum Range. In seconds; user-configurable. (18,X,28,WORD)

Alarm Delay

The conditions for a low or high alarm must exist for Alarm Delay for the alarm to be declared. They must also be absent for Alarm Delay for the alarm to be cleared. In seconds; user-configurable. (18,X,41,LO_BYTE)

Alarm Delay Timer

Used to time Alarm Delay. Reads in seconds; not user-changeable. (18,X,17,LO_BYTE)

Analog Output

The output of the PID algorithm. Varies between Minimum Output Limit and Maximum Output Limit; not user-changeable. (18,X,1,WORD)

Calculate Now

Indicates that a PID calculation should be performed now. Yes, No; not user-changeable. (18,X,12,LOBIT6)

Calculation Time

The interval between successive PID calculations. In seconds; user-configurable. (18,X,39,LO_BYTE)

Calculation Timer

Used to measure calculation time. Reads in seconds; not user-changeable. (18,X,14,LO_BYTE)

Clear Verify Alarm Enable

Enables Clear Verify Handle

Yes, No; user-configurable. (18,X,19,HIGH BYTE BITS 2)

Clear Verify Alarm Handle

Points to the attribute used to clear the Verify Alarm. Previously used for Interlock 3. User-configurable. (18,X,60,2 WORDS)

Control Increment

The change which needs to be added to the analog output to bring it current. 0 to 100%; not user-changeable. (18,X,12,WORD)

Control Input

The value of the independent control variable on which the PID algorithm depends. Units depend on application; not user-changeable. (18,X,10,WORD)

Control Input Handle

Points to the attribute containing the value to be used as the control input. Typically points to an Input index. User-configurable. (18,X,52,2 WORDS)

Control Setpoint - Active

The active setpoint for the control parameter. The control parameter is that parameter read by the control input (examples static pressure, fan speed). The Control Setpoint-Active is the control setpoint for the current state (MRDY, NSB, OCC, UNOCC) as modified by the reset input and the demand limit. The PID algorithm is used to maintain the control variable at the active control setpoint. Units depend on the application; not user-changeable. (18,X,9,WORD)

Control Setpoint - MRDY

The control setpoint used when the control is in Morning Ready state. Units depend on application; user-configurable. (18,X,24,WORD)

Control Setpoint - NSB

The control setpoint used when the control is in Night Setback state. Units depend on application; user-configurable. (18,X,23,WORD)

Control Setpoint - OCC

The control setpoint used when the control is in Occupied state. Also used as the default state if Enable by State has been set to "Disabled". Units depend on application; user-configurable. (18,X,22,WORD)

Control Setpoint - UNOCC

The control setpoint used when the control is in Unoccupied state. Units depend on application; user-configurable. (18,X,21,WORD)

Deceleration Time

The minimum time over which the Analog Output is allowed to decrease from the Maximum Range to zero. In seconds; user-configurable. (18,X,29,WORD)

Demand Limit Enable

Enables use of the demand limit feature. The demand limit feature adds Demand Reset X (X = 1 -> 6) from the Active Control Setpoint. X is the Demand Limit Level; it is read from the attribute pointed to by the Demand Limit Handle. Enabled, Disabled; user-configurable. (18,X,19,HIBIT5)

Demand Limit Handle

Points to the attribute containing the Demand Limit Level (typically this will be in Object 11 - Demand Limit). User-configurable. (18,X,68,2 WORDS)

Demand Reset Value

The actual demand reset value to be added to the active control setpoint chosen from Demand Reset X (1 to 6). Units depend on the application the PID object is used for; not user-changeable. (18,X,16,WORD)

Demand Reset X

X is an integer from 1 to 6; it is the Demand Limit Level. For non-zero Demand Limit Levels, the corresponding Demand Reset X is added to the Active Control Setpoint (if the demand limit feature is enabled). Note that Demand Reset X is a signed value. Units depend on the application the PID object is used for; user-configurable. (18,X,44->49,WORD)

Derivative Gain

Ratio of The Derivative Time to Derivative Unit . Derivative Gain is a Calculated Value Used to determine the derivative contribution to output.

Derivative Time

Effectively, this is the proportionality constant for the derivative term in the PID algorithm. In seconds; user-configurable. (18,X,27,WORD)

Derivative Unit

The time period between successive calculations of the derivative term in the PID algorithm. In seconds; user-configurable. (18,X,35,WORD)

Direct Acting

If set to Yes, Direct Acting causes the PID algorithm to increase the signal should the error increase and visa-versa. Yes, No; user-configurable. (18,X,18,LOBIT0)

Drive Alarm

A Drive Alarm will be recorded if the Drive OK input reads false and Drive Alarm has been enabled. In the presence of a Drive Alarm, the analog output will be set to zero and the primary binary output will be off. OK, Alarm; not user-changeable. (18,X,3,LOBIT3)

Drive Alarm Enable

Enables the drive alarm. Enabled, Disabled; user-configurable. (18,X,19,LOBIT4)

Drive OK

Represents whether the physical output which the PID's Analog Output controls is responding properly. Drive OK is the value read by the Drive OK Handle. Yes, No; user-configurable. (18,X,2,LOBIT7)

Drive OK Handle

Points to the attribute which Drive OK is to be read from. Typically this will be an input object which reads a sensor on the physical output PID's Analog Output is controlling; the sensor verifies that the output is operating properly. User-configurable. (18,X,66,2 WORDS)

Enable By State

Enabling this feature causes PID Object operations to be enabled or disabled based on state. The parameter State X Enable is looked at to decide whether the PID operations should occur. If Enable by State is set to "Disabled", the PID Object will examine its interlocks to determine whether to perform operations or not. X is the current control state (either UNOCC, OCC, MRDY, or NSB). Enabled, Disabled; user-configurable. (18,X,19,LOBIT1)

Error Last

The value which Error New was equal to immediately before the current value was read. Units depend on application; not user-changeable. (18,X,6,WORD)

Error New

The error most recently read. Error is equal to the Active Control Setpoint minus the Control Input Value. Units depend on application; not user-changeable. (18,X,5,WORD)

Error Previous

The value of Error New which was read immediately before Error Last when using the Incremental Algorithm. Units depend on application; not user-changeable. (18,X,7,WORD)

Error Sum

The sum of the Errors when using the Absolute Algorithm. Units depend on application; not user-changeable. (18,X,7,WORD)

High Alarm Enable

Enables the High Control Alarm for operation. Enabled, Disabled; user-configurable. (18,X,19,LOBIT6)

High Alarm Setpoint

The trigger setpoint for the High Control Alarm Units depend on application; user-configurable. (18,X,43,WORD)

High Control Alarm

If this alarm is enabled, it will be set if the control input rises above the High Alarm Setpoint. OK, Alarm; user-configurable. (18,X,3,LOBIT1)

Index Enable

Enables this index of the PID Object for operation. For any index to be enabled, the PID Object must also be enabled in Object 1, the System object. Enabled, Disabled; user-configurable. (18,X,19,LOBIT0)

Initial Value Enable

Used by to enable Initial Value Feature (18,X,18,LO Bit 2)

Initial Output Value

When the PID Loop starts, if enabled, the Analog Output is maintained at the Initial Output Value for an Interstage Time. (18,X,62,WORD)

Initial Value Enable

Enables the Start up Initial Output Value feature. Yes, No; user-configurable. (18,X,18,LOBIT2)

Integral Gain

The product of the Proportional Gain and the Integral Weight. Multiplies the Error Sum to determine the integral term of the output. Integral Gain is a calculated value used to determine the integral contribution to output.

Integral Time

Effectively, this is the weighting constant for the integral term used in the PID algorithm. In seconds; user-configurable. (18,X,26,WORD)

Integral Unit

The time period between successive calculations of the integral term in the PID algorithm. No seconds; user-configurable. (18,X,34,WORD)

Integral Weight

The ratio of the Integral Unit to the Integral Time. Should be about 0.25 . Determines the relative importance of the integral term.   Integral Weight is a calculated value used to determine the relative integral contribution to output.

Interlock X Enable

Enables interlocks 1 and 2 to prevent or allow operation of the PID object based on the value of the interlock. Yes, No; user-configurable. (18,X,19,HIGH BYTE BITS 0..1)

Interlock X Handle

Points to the attribute containing Interlock X. X = 1 to 2 User-configurable. (18,X,56...60,2 WORDS)

Interlock X OK

Tells whether Interlock X is present. X = 1 to 2. Yes, No; not user-changeable. (18,X,2,HIBIT2)

Interlock X OK

Tells whether Interlock X is present. X = 1 to 2. Yes, No; not user-changeable. (18,X,2,HIBIT2)

Interlocks OK

Tells whether all interlocks are present or not. Yes, No; user-configurable. (18,X,2,LOBIT2)

Interstage Time

The delay which must elapse between turning on any two pieces of electrical equipment. Used by the Initial Value Option to determine the time to remain at the initial value. Not used by the Incremental Algorithm object but available here for access by others. In seconds; user-configurable. (18,X,39,HI_BYTE)

Interstage Timer

Used to track the interstage time. The Interstage Timer is loaded with Interstage Time when the Primary Binary Output turns on. Not user-changeable. (18,X,14,HI_BYTE)

Low Alarm Enable

Enables the Low Control Alarm for operation. Yes, No; user-configurable. (18,X,19,LOBIT5)

Low Alarm Setpoint

The trigger setpoint for the Low Control Alarm. Units depend on application; user-configurable. (18,X,42,WORD)

Low Control Alarm

If enabled, this alarm will be recorded when the Control Input value drops below the Low Alarm Setpoint. OK, Alarm; not user-changeable. (18,X,3,LOBIT0)

Maximum Output Limit

The maximum value which the Analog Output may attain. Units depend on application; user-configurable. (18,X,37,WORD)

Maximum Reset Input

If the Reset Input exceeds this value, the Maximum Reset Value is added to the Active Control Setpoint. Units depend on application; user-configurable. (18,X,30,WORD)

Maximum Reset Value

The signed value which is added to the Active Control Setpoint if the Reset Input exceeds the Maximum Reset Input Value. Units depend on application; user-configurable. (18,X,32,WORD)

Maximum Starts

The maximum number of starts of the equipment controlled by the PID Index which may be attempted without attaining Drive OK verification. After Maximum Starts has been attained, the PID Output is set to zero and the object must be reset via the communications line to become operational. Integer value; user-configurable. (18,X,40,HI_BYTE)

Minimum Output Limit

The Minimum Value the Analog may be equal to, irrespective of its calculated value. Units depend on application; user-configurable. (18,X,38,WORD)

Minimum Reset Input

If the Reset Input falls below this value, the Minimum Reset Value is added to the Active Control Setpoint. Units depend on application; user-configurable. (18,X,31,WORD)

Minimum Reset Value

If the Reset Input falls below the Reset Minimum Input, then Minimum Reset Value is added to the Active Control Setpoint. Signed. Units depend on application; user-configurable. (18,X,33,WORD)

Off Output Value

Analog output value used when PID Loop is Off. (18,X,63,WORD) New!

Output On

Designates whether the Primary Binary Output is on or not. Yes, No; not user-changeable. (18,X,2,LOBIT3)

Output Verify Fault

Indicates that after a number of attempts equal to Maximum Starts, the Primary Binary Output was turned on but output verification was not attained for any attempt. Yes, No; not user-configurable. (18,X,2,LOBIT5)

Output Verify Handle

Points to the Input index which reads the output verification sensor. User-configurable. (18,X,64,2 WORDS)

Output Verify Status

Indicates whether verification of the Primary Binary Output has been attained. Yes, No; user-configurable. (18,X,2,LOBIT4)

PID Action

Writing the value given below to this attribute causes the corresponding event to occur:

1 Clear Low Control Alarm
2 Clear High Control Alarm
3 Clear Verify Alarm
4 Clear Drive Alarm
,,,
n Clear Alarm bit n-1
..
17 Clear All Alarms and Start Counters
18 Fresh Start - clear attributes 0,2,3 and Start Counters
19 Override PID Analog Output
20 Clear PID Override !

PID Operation Requested

Indicates whether PID Object operation has been enabled for the current state. Yes, No; user-configurable. (18,X,2,LOBIT0)

PID Pause Enable

Converts Interlock 2 Handle to pause when true PID Attr-19, HIBit6, (740/840E2.0)

PID Paused

Shows the PID is paused when PID Pause Enable is Yes and Interlock 2 Handle is true. PID Attr-2 HiBit5 (740/840E2.0)

Previous Output Value

- Absolute Algorithm (18,X,12,Word) New FW740A..

Primary Binary Output

The current status of the Primary Binary Output. On, Off; not user-changeable. (18,X,0,LOBIT0)

Primary Binary Output Active

Indicates whether the Primary Binary Output may be used. (Could be locked-out due to verify fault.) Internal flag. Yes, No; not user-changeable. (18,X,2,LOBIT1)

Proportional Gain

Ratio of the Maximum Range to Throttling Range. Used to determine the proportional contribution to output. Integral Gain is a calculated value used to determine the integral contribution to output.

Reset Function Enable

Enables use of the Reset Function. Yes, No; user-configurable. (18,X,19,LOBIT2)

Reset Function Handle

Points to the attribute the Reset Function Input is to be obtained from. User-configurable. (18,X,54,2 WORDS)

Reset Function Input

This value is obtained from an external object and is used to alter the Active Control Setpoint. Typically, if the PID is used to control a duct temperature, then the value of the Outdoor Air Temperature could be used to reset the Active Control Setpoint to a more applicable value. Units depend on application; not user-changeable. (18,X,8,WORD)

Reset Function Value

An intermediate step in the Reset Function sequence, this parameter holds a value between the Maximum Reset Value and Minimum Reset Value, depending on the value of the Reset Function Input. It is added to the Active Control Setpoint at the beginning of each calculation period. Units depend of application; user-configurable. (18,X,11,WORD)

Retry/Start Timer

Times the delay periods used in the Drive OK and Output verification functions. Reads in seconds; not user-configurable. (18,X,13,LO_BYTE)

Start Counter

Counts the number of times the PID has attempted to turn on the Primary Binary Output without success. Integer value; not user-changeable. (18,X,13,HI_BYTE)

Starts Total

The number of times the PID Object has turned on the Primary Binary Output. It is zeroed only on reset of the controller. Integer value; not user-configurable. (18,X,15,WORD)

State Handle

Points to the attribute the PID Object is to attain the current control state from. Typically points to the State object. User-configurable. (18,X,70,2 WORDS)

State MRDY Enable

Designates if the PID Object should be operational for Morning Ready State under the Enable by State feature. Yes, No; user-configurable. (18,X,20,LOBIT3)

State NSB Enable

Designates if the PID Object should be operational for Night Setback State under the Enable by State feature. Yes, No; user-configurable. (18,X,20,LOBIT2)

State OCC Enable

Designates if the PID Object should be operational for Occupied State under the Enable By State feature. Yes, No; user-configurable. (18,X,20,LOBIT1)

State UNOCC Enable

Designates if the PID Object should be operational for Unoccupied State under the Enable By State feature. Yes, No; user-configurable. (18,X,20,LOBIT0)

Step to Minimum Enable

Under the Step to Minimum feature, the PID will not abruptly move the Analog Output to minimum when is turned off, but instead will gradually reduce the value, based on the deceleration time. Yes, No; user-configurable. (18,X,19,HIBIT4)

Throttling Range

The throttling range is that value of the sum of the integral, derivative, and proportional terms in the PID algorithm which will drive the PID Calculation from 0 to its maximum output (the PID calculation range). Units depend on application and must be the same as the control input; user-configurable. (18,X,25,WORD)

Units

An integer code which tells ASI VISTA what units are to be used in conjunction with this index of the PID object. The choices are given in the Object Definition. User configurable. (18,X,50,LO_BYTE)

Verify Alarm

If Output Verify Status does not indicate a verified output for a period of time equal to Verify Delay, the Verify Alarm will be set (if it is enabled). OK, Alarm; user-configurable. (18,X,3,LOBIT2)

Verify Alarm Enable

Enables the Verify Alarm for operation. Yes, No; user-configurable. (18,X,19,LOBIT3)

Verify Delay

The delay used in conjunction with the Verify Alarm. In seconds; user-configurable. (18,X,40,LO_BYTE)

PID Properties

This object is used to implement modulating proportional plus integral plus derivative control loops, such as those required for air handling units.

Object Name	PID
Object Number	= 18
Data Type	= WORD
Index	= 0..N as allocated
Attribute	= 72 (0..71)
DYNAMIC Attributes	= 18 (0..17)
STATIC Attributse	= 54 (18..71)
	= 62 (18..79) FW740C..

PID Firmware Revision

ASIC/2-7540 FW754a Ver 1.4m Release 2006-04-07
ASIC/2-8540 FW854a Ver 1.4m Release 2006-04-07

o Fixes PID State Enable. Now ignores State Handle if not set.
o Fixes PID Alarm Delay. Always at least 1 second. Fixes the High to Low Alarm transition.
o Fixes PID Interlock to add at least 1 second interstage to recalculate integral term so that it continues smoothly..

ASIC/2-7540 FW754A Rev 1.0 Forthcoming 2005

o As in FW740E2.0.

ASIC/2-7040 FW740E Rev 2.0 Released 12/15/2000 CHK 0x181F
ASIC/2-8040 FW840E Rev 2.0 Released 12/15/2000 CHK 0xF6C1

o **PID Pause Enable** Attr-19, HIBit6,
converts Interlock 2 Handle to pause when true.
PID Paused Attr-2 HiBit5
o PID Pause Enable converts Interlock 2 Handle to pause when true.
PID Action Pause no longer winds up.

ASIC/2-7040 FW740C Rev 1.0 Released 01/29/97

o Instance Name Labels for PID, Attr-72..79

ASIC/2-8040 FW840A Rev 1.0, Released March 1996

ASIC/2-7040 FW740A/B Rev 1.8 Released 12 April 95

o Fixed PID Override Feature so that it allows override of output value.
o Fixed Step To Minimum. It does Not work in FW700A Rev 1.5, 1.6 or 1.7 it gets lost stepping down and cycles.

ASIC/2-7040 FW740B Rev 1.6 released 2 Dec 1994

o Improved overflow/underflow protection for Error Sum Calculation
o PID Step to Minimum did not work and has been fixed.

ASIC/2-7040 FW740A Rev 1.0 Released (31 March 1994)

Added absolute PID algorithm option

New or Changed Parameters:

- Absolute PID Enable, Attr-18,LO Bit 1
- Error Sum, Attr-7 (redefined)
- Interlock 3 Handle , Attr-60,61 (Redefined) Clear Verify Alarm Handle
- Interlock 4 Handle , Attr-62,63 Removed
- Initial Output Value, Attr-62
- Off Output Value, Attr-63
- Previous Output Value, Attr-12 (redefined)
- Interstage Timer, Attr-14 Hi Byte (new use)
- Interstage Time, Attr- 39 Hi Byte(new use)

ASIC/2-7000 Ver. 1.0 FW700A 5 Dec 1991

Identical with SC/1 version

SC/1 Ver 1.0 FW907C 21 April 1992

Fixed power reset bug.

PID DYNAMIC Properties

Attr-0 Binary Output Status

LO bit 0 - **Primary Binary Output**
LO bit 1 - Spare

HI Byte – Spare

Attr-1 **Analog Output**

> Note: The Attr-36 Max Range, Attr-37 Max and Attr-38 Min Output Limits are Word Value. The Analog Output Value Attr-1 is also an unsigned word.

Attr-2 Status

LO bit 0 - **PID Operation Request**
True if enabled for the current state.

LO bit 1 - **Interstage Active** (internal flag)
True when interstage timer nonzero

LO bit 2 - **Interlocks OK**
True if all enabled interlocks are true.
False if any enable interlock is false.

LO bit 3 - **Output ON**
True if the primary binary output is on.

LO bit 4 - **Output Verify Status**
True if the assigned verify input is non zero. If enabled, this is used to verify that the binary output has really come on.

LO bit 5 - **Output Verify Fault**
True if Binary Output is ON and the assigned verify input not true with in a Verify Delay Time.(Attr-40,LO) The device is locked out.

LO bit 6 - **Calculate Now**
True when it is time to update the calculation.

LO bit 7 - **Drive OK**
True when assigned Drive OK input is true,
or if Drive OK is not enabled.
Jams Analog Output to *(Off Output Value)* if not true.

HI Byte - Interlock Status

The Interlocks are OK true, if either the assigned input is non-zero, or if the interlock has not been enabled.

HI bit 0 - **Interlock 1 OK**
True when Interlock not enabled, or
enabled assigned interlock is true.

HI bit 1 - **Interlock 2 OK**

HI bit 2 - **Verify Reset On** (740A..)
Interlock 3 OK (700A..)

HI bit 3 - **PID Override Status** (740A..)
Set/Cleared by PID Action 19 or 20
Interlock 4 OK (700A..)

HI bit 4 - Internal Flag (740A..) (AH_PROOF_IN)
Used to identify that the Clear Verify Alarm Handle is going positive.
HI bit 5 - **PID Paused** [AH_PAUSED] (FW740E2.0, 840E2.0)
HI bit 6 - Spare
HI bit 7 - Spare

Attr-3 Alarm

LO bit 0 - **Low Control Alarm**
If the unit is OFF, No Low Alarm is set.

LO bit 1 - **High Control Alarm**

LO bit 2 - **Verify Alarm**

True if verify is enabled, and the Binary Output has started and the Output Verify Input does not become true within a Verify Delay, and the Start Count equals Maximum Starts, then the Verify Alarm is set. Once set there will be no further retries until this alarm is cleared by communications, or unless the controller is reset. May also be reset on any Clear Verify Alarm Handle going true after alarm is set.

LO bit 3 - **Drive Alarm**

True if Drive OK is enabled, and Drive OK input is false. When true, the analog output is set to Off Output Value. No other action is taken.

LO bit 4 - Spare
LO bit 5 - Spare
LO bit 6 - Spare
LO bit 7 - Spare

HI Byte - Spare

Attr-4 **Action**

0 = No Operation

1..16 = Reset Attr-3 Alarm bit 0..F

17 = Clear All Alarms - Clear Attr-3 Alarm, and the Start Counter (Attr-13).

18 = Fresh Start - Clear Attr-0,2,3 and the Start Counter (Attr-13)

Note: If Verify Alarm is set, it must be cleared before the Binary output can be started. It is cleared by writing to the Action entry, 18-PID, Index=0, Attr-4 Action = 17, or by resetting the controller.

Note: The Clear Verify Alarm Handle can also be used to clear the verify fault through the sequence. In ASIC/2-7000 FW700A..I, only communication or reset can clear the verify fault.

19 = Override PID Loop

20 = Clear PID Loop Override.

Two New Actions 19, 20 are added to stop updating the analog output value to allow performing an open loop step test. The error continues to be updated but the final analog output value and integral term is not updated. A new value may be written to the Analog output value. When the override is cleared the algorithm continues.

21 = Set Index Enable (Attr-19,LO Bit0= 1).

22 = Clear Index Enable (Attr-19,LO Bit0 = 0).

PID Error

The new and previous two values of error signal, difference between the control input and the active setpoint are saved and used in the PID calculation.

Attr-5 **Error New** - Control Parameter, E_n

Attr-6 **Error Last** - Control Parameter, E_{n-1}

Attr-7 **Error Sum** (Absolute Algorithm) (New Use ASIC/2-7040), or
Error Previous - Control Parameter, E_{n-2} (Incremental Algorithm)

Attr-8 **Reset Function Input**

The value of the assigned Reset Function Input is read and saved. This value is used to calculate the Active Reset Value based on Minimum and Maximum Reset Input and Minimum and Maximum Reset Value.

Attr-9 **Active Control Setpoint**

This is the Active Control Setpoint based on State, as modified by the Reset Value.

Attr-10 **Control Input**

This is the value of the assigned Control Input. Smoothing is done in the Input control block.

Attr-11 **Reset Function Value**

The calculated Reset Function Value is based on the present value Reset Function Input and on the Minimum and Maximum Reset Input and Minimum and Maximum Reset Values.

Attr-12 **Control Increment**

Present Change in Analog output based on the .Incremental Algorithm
The Control Increment is still displayed here. (Absolute Algorithm)

Attr-13 Timer and Counter

LO Byte - **Retry/Start Timer**
On Start it represents Verify Delay before fault
On Stop it represents Retry Delay before restart

HI Byte - **Start Counter**

Attr-14 Timers

LO Byte - **Calculation Timer**

Used to time calculation interval.

HI Byte - **Interstage Timer**

Attr-15 **Starts Total**

This counter is incremented each new start of binary control output. It is never zeroed except on reset.

Attr-16 **Demand Reset Value**

This is a signed value in the units of the control input. If Demand Limit Enable is Set, the value is selected from the Demand Reset 1..6 based Attr-44..49, based on the current demand level as identified by the Demand Limit Handle.

Attr-17 Timers

LO Byte - **Alarm Delay Timer** (sec)
Used only on Low and High Control Alarms. If the binary output is off, there is no Low Control Alarm.

HI Byte - Spare

PID STATIC Properties

Attr-18 Setup

Attr-18 LO bit 0 - **Direct Acting Enable**

- This bit is set if an increase in the error should result in a increase in the output. "(AH_REVERSE)"

Attr-18 LO bit 1 - **Absolute PID Enable** (AH_DIRECT_EN)

Attr-18 LO bit 2 - **Initial Value Enable**
LO bit 3 -Spare
LO bit 4 - Spare
LO bit 5 - Spare
LO bit 6 - Spare
LO bit 7 - Spare

HI Byte - Spare

Attr-19 Enable 1
LO Byte -

Attr-19 LO bit 0 - **Index Enable**
This flag is set to enable this index. (AH_MASTER)

Attr-19 LO bit 1 - **Enable by State**
If Enable by State is true, then the value of STATE as identified by the State Handle and the State UNOCC, OCC, NSB, MRDY Enable options are examined to determine if operation should be requested.

If Enable by State is false, Request to Operate, Attr-2, LO Bit 0 is set true and the operation is determined by the interlocks. Occupied Setpoints will be used for control function. (AH_STATE_EN)

Attr-19 LO bit 2 - **Reset Function Enable**
If true, then the assigned Reset Function Handle is examined, and the calculated Reset Function Value is added to the Active Setpoint.
If false, no modification of Active Setpoint is made. (AH_RESET_EN)

Attr-19 LO bit 3 - **Verify Alarm Enable**
If Verify Alarm Enable is set, then the assigned Output Verify Handle Input is examined, and used to verify proper control binary output start.
If false, Output Verified , Attr-2, LO Bit 4 is set true. (AH_PROOF_EN)

Attr-19 LO bit 4 - **Drive Alarm Enable**
If true, then the Drive OK Handle is examined, and used to confirm Variable Speed Drive operation. If false, Drive Alarm, Attr-3, LO Bit 3 is set true, and Analog Output is clamped at 0. (AH_VERIFY_EN)

Attr-19 LO bit 5 - **Low Alarm Enable**
If true, allows Low static pressure alarm. The Low Control Alarm is locked out when the output is OFF. If false, then there is no low Alarm. (AH_LO_EN)

Attr-19 LO bit 6 - **High Alarm Enable**
If true, allows High Control Alarm. If false, then there is no High Alarm. (AH_HI_EN)

LO bit 7 - Spare

HI Byte -

Attr-19 HI bit 0 - **Interlock 1 Enable**
If true, then the assigned Interlock Input is examined, and must be true before operation is permitted. If false, then interlock does not inhibit operation. (AH_LK_1_EN)

Attr-19 HI bit 1 - **Interlock 2 Enable**
If true, then the assigned Interlock Input is examined, and must be true before operation is permitted. If false, then interlock does not inhibit operation. (AH_LK_2_EN)

Attr-19 HI bit 2 - **Clear Verify Alarm Enable** (740A..)

New Function for ASIC/2-7040 . Clear Verify Alarm Handle is used to reset the Verify Alarm.

Interlock 3 Enable (700A..) If true, then the assigned Interlock Input is examined, and must be true before operation is permitted. If false, then interlock does not inhibit operation (AH_CLEAR_EN)

Attr-19 HI bit 3 - Spare Not used in ASIC/2-7040 (AH_SPARE_1)

Interlock 4 Enable(700A..)

If true, then the assigned Interlock Input is examined, and must be true before operation is permitted. If false, then interlock does not inhibit operation.

Attr-19 HI bit 4 - **Step to Minimum Enable**

If true, then the analog output steps to minimum before the output turns off. If false, then the analog output immediately goes to minimum. (AH_STEP_EN)

Attr-19 HI bit 5 - **Demand Limit Enable**

If true, then the active setpoint is reset based on the Demand reset corresponding to the active demand level, after reset by calculated reset value. (AH_DEMAND_EN)

Attr-19 HI bit 6 - **PID Pause Enable** [AH_PAUSE_EN] (740E2.0, 840E2.0)

Converts Interlock 2 Handle to pause when true

Attr-19 HI bit 7 - Spare

Attr-20 Enable Flags

LO Byte - State Enable Flags

If Enable by State is false, then these flags are ignored. If Enable by State Enable is true and if any of these enables are true, then operation is be permitted when that State is active. If not true, then operation is not permitted.

LO bit 0 - **State UNOCC Enable**
LO bit 1 - **State OCC Enable**
LO bit 2 - **State NSB Enable**
LO bit 3 - **State MRDY Enable**

LO bit 4 - Spare
LO bit 5 - Spare
LO bit 6 - Spare
LO bit 7 - Spare

HI Byte - Spare

State Dependent Setpoints

These setpoints are in the same units as the control input. If State has been enabled, then the appropriate setpoint is used together with the current resets to determine the active setpoint.

Attr-21 **Control Setpoint UNOCC**
Attr-22 **Control Setpoint OCC**
Attr-23 **Control Setpoint NSB**
Attr-24 **Control Setpoint MRDY**

PID Parameters

Attr-25 **Throttling Range**

The Throttling Range is in the same units as the control input.

Attr-26 **Integral Time** (seconds)

The time for a constant error over which the integral term will have the same magnitude as the proportional term of the PID algorithm.

Attr-27 **Derivative Time** (seconds)

Attr-28 **Acceleration Time** (seconds)

The minimum time over which the Analog Output is allowed to increase from the zero to Maximum Range. In seconds; user-configurable.

= (Calculation Time)*(Max Range)/(Accel Time)

> Note: Max Step Up /Down in the ASIC/2-7040 FW740A.. is now calculated using Calculation Time. In the ASIC/2-7000 it was calculated using the Derivative Unit.

Attr-29 **Deceleration Time** (seconds)

The minimum time over which the Analog Output is allowed to decrease from the Maximum Range to zero. In seconds; user-configurable. (18,X,29,WORD) Max Step Down = - (Calculation Time)*(Max Range)/(Decel Time)

Reset Parameters

Attr-30 **Maximum Reset Input**

If the reset Function input is equal to or greater than the Max Reset Input, then the Reset Function Value will be equal to the Max Reset Value.

Attr-31 **Minimum Reset Input**

If the reset Function input is equal to or less than the Min Reset Input, then the Reset Function Value will be equal to the Min Reset Value.

Attr-32 **Maximum Reset Value**

The maximum reset of the control setpoint when the Reset Function Input is equal to or greater than the Max Reset Input.

Attr-33 **Minimum Reset Value**

The minimum reset of the control setpoint when the Reset Function Input is equal to or less than the Min Reset Input.

Attr-34 **Derivative Unit**

This is typically the PID Calculation Time and is a divisor used to determine the weighting of the Derivative term of the PID control algorithm. The larger the Derivative Unit, the smaller the weighting of the derivative term. The Derivative Unit must be set to the Calculation Time for the algorithm to work.

Attr-35 **Integral Unit**

This is typically the PID Calculation Time and is a multiplier used to determine the weighting of the Integral term of the PID control algorithm. The larger the Integral Unit, the larger the weighting of the Integral term.

Output Limits

Attr-36 **Maximum Range**

Used to determine the maximum range of the PID
control output. In Proportional Only control, the output equals the Maximum Range, when the error is equal to the Throttling Range. Typically it is set to 255, representing a range of 0..255 bits. The final output is 0..255 representing 0 to 255%. (Unsigned Word Value)

Attr-37 **Maximum Output Limit**

Used to limit the maximum analog output value in the range 0 to Max Range. (Unsigned Word Value)

Attr-38 **Minimum Output Limit**

Used to limit the minimum analog output value in the range 0 to Max Range. (Unsigned Word Value)

Attr-39 Times

LO BYTE **Calculation Time** (seconds)
This value is used to time how often the PID calculation is updated. All other input/output functions continue at 1 second intervals.

HI BYTE - **Interstage Time** (seconds)
In the ASIC/2-7040 FW740A.. it is used to time Initial Output Value
Available for use by others to inhibit other input/output activity.

Attr-40 Verify Parameters (AH_PROOF_TIME)

LO BYTE - **Verify Delay** (seconds)
Used for delay while waiting for start verification, and for off time while waiting for retry.

HI BYTE - **Max Starts**
Total number of starts before declaring start verify fault. If 0, no retries are attempted.

Alarms

Attr-41 Alarm Delay

LO BYTE - **Alarm Delay** (seconds)
When the alarm conditions exist, the time to wait before declaring a low or high alarm. If in alarm and alarm conditions go away, the time to wait before clearing the alarm.

HI BYTE - Spare

Attr-42 **Low Alarm SP**

If the Input Value is less than the Active Setpoint by the Low Alarm SP, then the Low Alarm is set.

Attr-43 **High Alarm SP**

If the Input Value is greater than the Active Setpoint by the High Alarm SP, then the High Alarm is set.

Demand Reset

Demand Resets are signed values in same units as setpoints

Attr-44 **Demand Reset 1**
Attr-45 **Demand Reset 2**
Attr-46 **Demand Reset 3**
Attr-47 **Demand Reset 4**
Attr-48 **Demand Reset 5**
Attr-49 **Demand Reset 6**

Attr-50 Input ID

LO BYTE - **Units**
HI BYTE - Spare

Attr-51- Spare

Attr-52,53 **Control Input Handle**

Attr-54,55 **Reset Function Handle**

Points to the Reset Function Input

Attr-56,57 **Interlock 1 Handle**

This value, if enabled, is used to interlock the request for operation of the device.

Attr-58,59 **Interlock 2 Handle**

Note: **PID Pause Enable** redefines the operation of Interlock 2. (FW740E2.0, 840E2.0)

Attr-60,61 **Clear Verify Alarm Handle** (FW740A..)

This is a new function in the ASIC/2-7040
Interlock 3 Handle(FW700)

Note: The ASIC/2-7040 uses Attr-60,61 as the Clear Verify Alarm Handle to clear the alarm through the sequence. The ASIC/2-7000 uses Attr-60,61 as the Interlock 3 Handle.

Note: The ASIC/2-7040 uses Interlock 4 Handle for ***Initial Output Value* and *Off Output Value*** parameters

Attr-62 **Initial Output Value**

If enabled, when PID Loop turns on, the analog output is set to the Initial Output Value for an Interstage Time. (AH_INITIAL_VALUE)
Attr-62,63 Interlock 4 Handle (FW700A., FW907A..)

Attr-63 **Off Output Value**

When the PID Loop is Off, the analog output is set to the Off Output Value. Usually set to the Minimum Output Limit. (AH_OFF_VALUE)

Attr-64,65 **Output Verify Handle**

This value, if enabled, is used to verify operation of the binary output device. If the output fails to verify, it turns the output off, waits a Verify Delay, and retries until the Start Counter equals the Maximum Starts. (AH_PROOF_HDL)

Attr-66,67 **Drive OK Handle**

This value, if enabled, is used to confirm operation of the analog output device (variable speed drive). If the drive OK fails, the outputs are turned off and the Drive fault alarm is set. (AH_VERIFY_HDL)

Attr-68,69 **Demand Limit Handle**

If Demand Limit is Enabled, then the LO Byte pointed to by the handle gives the active Demand Level and the HI Byte gives the active Demand Rotate Group.

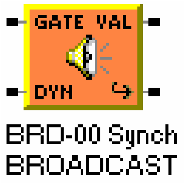
Attr-70,71 **State Handle**

If Enable by State is true, then the LO Byte pointed to by the handle is examined for a number 0 = None, 1 = UNOCC, 2 = OCC, 3 = NSB, and 4 = MRDY. This then is used by the PID Object to select which setpoints to use for control. IF a value other than 1..4 is found, or if the Enabled by State is not true, then OCC is assumed.

Attr-72..79 **PID Loop Name** 16 characters

Object 19 - Broadcast

Broadcast Summary



Overview: The broadcast object is used by to configure the ASIC/2 or SINC/3 controller to send messages on the local communications bus either periodically or based on a trigger. Each index of the 19-BROADCAST represents a different message to be broadcast to a specific device, group, or global address

Programmable messages can be broadcast to device, group, or global addresses on the Local bus ONLY. A Broadcast Trigger/Gate Handle points at the data to be examined to determine if the message should be broadcast. A Data Source handle points to the data that is to be inserted into the broadcast message. The message is totally defined within the bytes of the index of the Broadcast object.

The triggered or periodic broadcasts may be initiated by inputs from a local contact closure, internal logic, or by reception of specific data on a Remote Point.

Periodic Broadcasts: If the Type of Broadcast is periodic, and **Use Gate** is No, the message will be broadcast every **Broadcast Interval** (1..255 minutes). If **Use Gate** is Yes, the status of the **Trigger/Gate Handle** is examined. While the **Trigger/Gate Value** is non zero, the message will be broadcast every **Broadcast Interval** (1..255 minutes) while the gate conditions are satisfied. It begins broadcasting immediately upon the trigger conditions being met. If the **Gate Sense** is LO, it will broadcast while the trigger is zero. If the **Gate Sense** is HI, it will trigger while the trigger is non-zero. If the Trigger on Any Change is set, then it will always broadcast.

Triggered Broadcast: If the **Type of Broadcast** is Trigger the message will be broadcast only upon the trigger conditions being satisfied. The broadcast of a message on the local bus is initiated by a trigger event identified by **the Trigger/Gate Handle**. The Trigger/Gate Handle is examined each second. If the Trigger Sense is LO, it will broadcast on transition to zero. If the Trigger Sense is HI, it will trigger on transition to non- zero. If Trigger on Any Change is set, then the message will be broadcast on any change in value of the Trigger Source. Triggered broadcasts only transmit the message on change of value. If it is desired to continue transmitting the message while the trigger condition is true, then periodic broadcast should be selected.

Static Message Bytes: The Broadcast object is designed to transmit short messages. The device address, the ASI message type and up to to 4 static data bytes are maintained in an index of the broadcast object.

Dynamic Message Bytes: Up to 15 dynamic message bytes starting with the value identified by the **Dynamic Message Handle** may be broadcast as part of the message.

Typically the Dynamic Message Handle returns one or two data bytes. The Dynamic Message Bytes always follow the Static Message Bytes.

Destination Address: The destination address may be any device, global, or group address on the local bus. Group and global messages are always broadcast three times on the local bus.

Forced Broadcast: The user may force an index of the Broadcast object to send its message at any time using the Broadcast Action.

Local Bus Assignment: The SINC/3 allows assigning the Broadcast to Local 1, or Local 2.

Broadcast Message

Destination Address

The address (device, group, or global address) is Attr-15 and Attr-16. To broadcast a Terminal Unit Global use address 23130 (5A 5A hex).

When the message is to be broadcast, the controller constructs a message using the destination address, its own local bus address as the source address, the Message Type, the Static Byte Count (0..4) of Static Message Bytes, followed by the Dynamic Byte Count (0..15) of data identified by the Dynamic Message Handle.

Local Bus Assignment

The SINC/3 allows assigning the Broadcast to Local 1, or Local 2.

Message Type

The Message Type. If message type 38h of 3Fh is used, then the current time from the clock object will be broadcast. Note: ASI Message Type is always in hex.

The checksum is computed internally as the message is sent.

Static Data

The static data bytes to be transmitted are located in Attr-18,..., Attr-27 of the index. The number of fixed data bytes to be included in the message is assignable starting with Attr-14 up to a maximum of 10 data bytes. Note: In the ASIC/2-7040 FW740A.., the number of static data bytes has been increased to 10 to allow for object write, 90h, messages.

Dynamic Message Bytes

The **Dynamic Message Handle** points at the dynamic data to be included as the last bytes of the message. The **Dynamic Byte Count** give the number of dynamic data bytes to be include in the message. Typically the Dynamic Message Handle returns one or two data bytes.

To send messages to a configurable controller, the body of the message may be placed in an index of the utility object which is referred to by the Dynamic Message Handle.

If the Dynamic Message Handle is "None", then no dynamic data is included in the message.

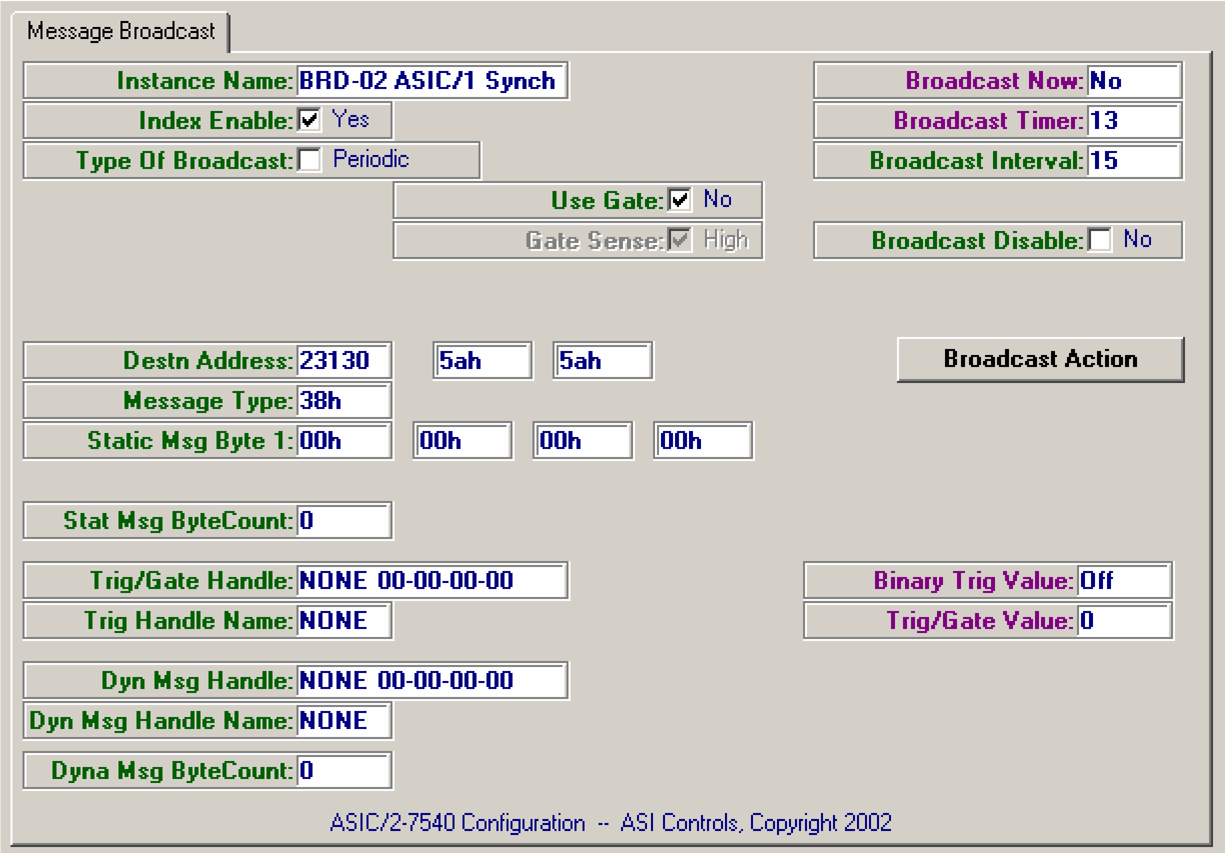
Periodic Broadcast

If the Broadcast Type is Periodic, then the message will be broadcast when the Broadcast Timer counts down to zero. The broadcast timer is then reloaded with the Broadcast Interval (1..255 minutes).

If **Use Gate** is Yes, the status of the **Trigger/Gate Handle** is examined. While the **Trigger/Gate Value** is non zero, the message will be broadcast every **Broadcast Interval** (1..255 minutes) while the gate conditions are satisfied. It begins broadcasting immediately upon the trigger conditions being met. If the **Gate Sense** is LO, it will broadcast while the trigger is zero. If the **Gate Sense** is HI, it will trigger while the trigger is non-zero. If the Trigger on Any Change is set, then it will always broadcast.

Periodic ASIC/1 Time Broadcast

Typically time is broadcast to ASIC/1 controllers to group address 23130 (5A5Ah) every 15 minutes. Time is broadcast using a Message Type, 38h. Only the message type needs to be configured. The rest of the message is automatically generated using the clock.



Instance Name: BRD-02 ASIC/1 Synch

Index Enable: ☑ Yes

Type Of Broadcast: □ Periodic

Use Gate: ☑ No

Gate Sense: ☑ High

Broadcast Now: No

Broadcast Timer: 13

Broadcast Interval: 15

Broadcast Disable: □ No

Destn Address: 23130 5ah 5ah

Message Type: 38h

Static Msg Byte 1: 00h 00h 00h 00h

Stat Msg ByteCount: 0

Trig/Gate Handle: NONE 00-00-00-00

Trig Handle Name: NONE

Binary Trig Value: Off

Trig/Gate Value: 0

Dyn Msg Handle: NONE 00-00-00-00

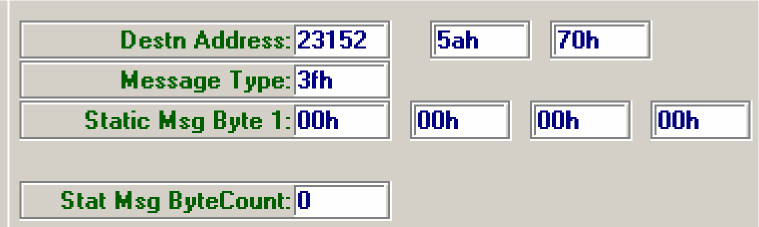
Dyn Msg Handle Name: NONE

Dyna Msg ByteCount: 0

ASIC/2-7540 Configuration -- ASI Controls, Copyright 2002

Periodic ASIC/2 Time Broadcast

Time is broadcast using a Message Type,3Fh. Typically time is broadcast to ASIC/1 controllers to group address 23152 (0x5A70) every 15 minutes. Only the message type needs to be configured. The rest of the message is automatically generated using the clock



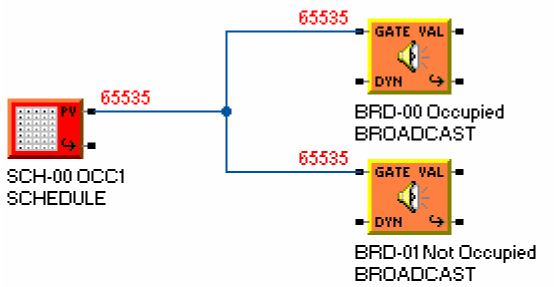
Destn Address: 23152 | 5ah | 70h

Message Type: 3fh

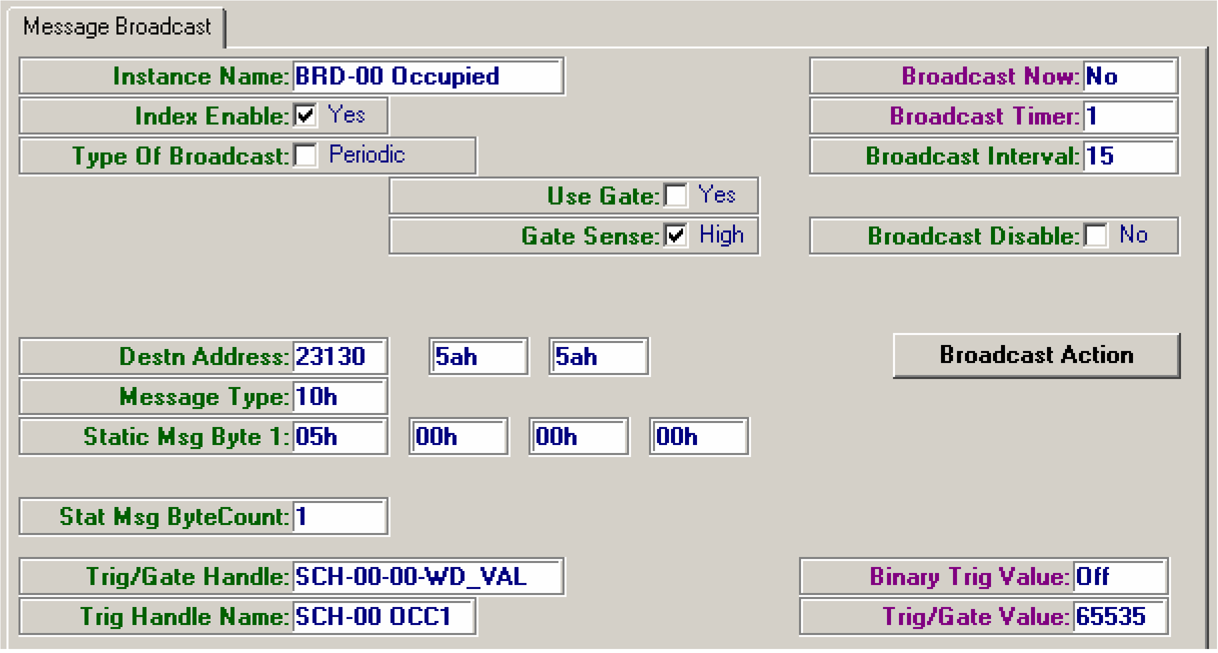
Static Msg Byte 1: 00h | 00h | 00h | 00h

Stat Msg ByteCount: 0

State Broadcast with Gate



Override to occupied can be broadcast based on a binary condition such as the status of a schedule. When the schedule goes true and every 15 minutes while the schedule is true, the controller broadcasts the ASIC/1 override to occupied message. MT=10h, M1 = 5



Instance Name: BRD-00 Occupied

Index Enable: ☑ Yes

Type Of Broadcast: □ Periodic

Use Gate: □ Yes

Gate Sense: ☑ High

Broadcast Now: No

Broadcast Timer: 1

Broadcast Interval: 15

Broadcast Disable: □ No

Destn Address: 23130 | 5ah | 5ah

Message Type: 10h

Static Msg Byte 1: 05h | 00h | 00h | 00h

Stat Msg ByteCount: 1

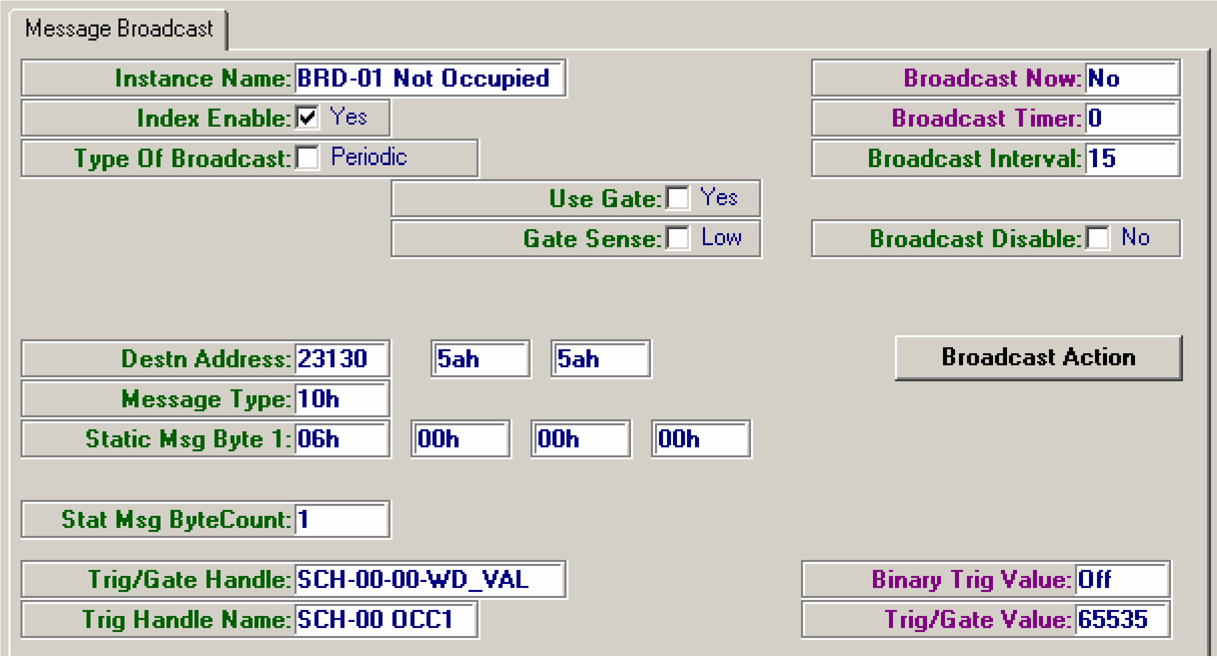
Trig/Gate Handle: SCH-00-00-WD_VAL

Trig Handle Name: SCH-00 OCC1

Binary Trig Value: Off

Trig/Gate Value: 65535

When the schedule goes false and every 15 minutes while the schedule is false, the controller broadcasts the ASIC/1 override to Unoccupied message. MT=10h, M1 = 6.



Message Broadcast

Instance Name: BRD-01 Not Occupied

Index Enable: ☑ Yes

Type Of Broadcast: □ Periodic

Use Gate: □ Yes

Gate Sense: □ Low

Broadcast Now: No

Broadcast Timer: 0

Broadcast Interval: 15

Broadcast Disable: □ No

Destn Address: 23130 | 5ah | 5ah

Broadcast Action

Message Type: 10h

Static Msg Byte 1: 06h | 00h | 00h | 00h

Stat Msg ByteCount: 1

Trig/Gate Handle: SCH-00-00-WD_VAL

Trig Handle Name: SCH-00 OCC1

Binary Trig Value: Off

Trig/Gate Value: 65535

Note: The State message will also work for ASIC/2 controllers provided the Override State Parameter in Utility, index 0, Attr-0 is configured to override the State object in the controller.

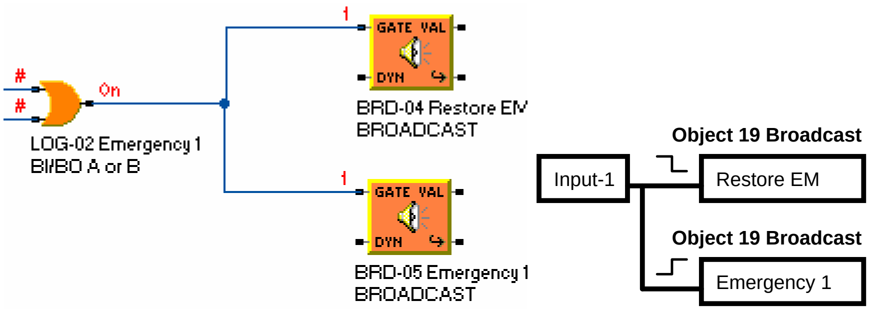
Triggered Broadcast

If the **Type of Broadcast** is Trigger the message will be broadcast only upon the trigger conditions being satisfied. The broadcast of a message on the local bus is initiated by a trigger event identified by **the Trigger/Gate Handle**. The Trigger/Gate Handle is examined each second. If the **Trigger Sense** is LO, it will broadcast on transition to zero. If the **Trigger Sense** is HI, it will trigger on transition to non- zero. If **Trigger on Any Change** is yes, then the message will be broadcast on any change in the Trigger/Gate Value.

Triggered broadcasts only transmit the message on change of value. If it is desired to continue transmitting the message while the trigger condition is true, then periodic broadcast should be selected.

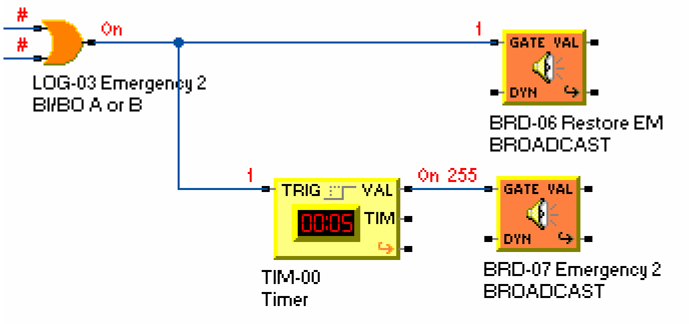
Emergency 1 Broadcast

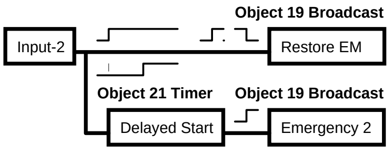
Sometimes it is necessary to configure to broadcast an emergency 1 message based on a status such as a contact on Input-1. When the contact opens the emergency can be restored. Note: That both EM1, EM2 and Restore Emergency are write to STATIC MEMORYand should only be broadcast once.

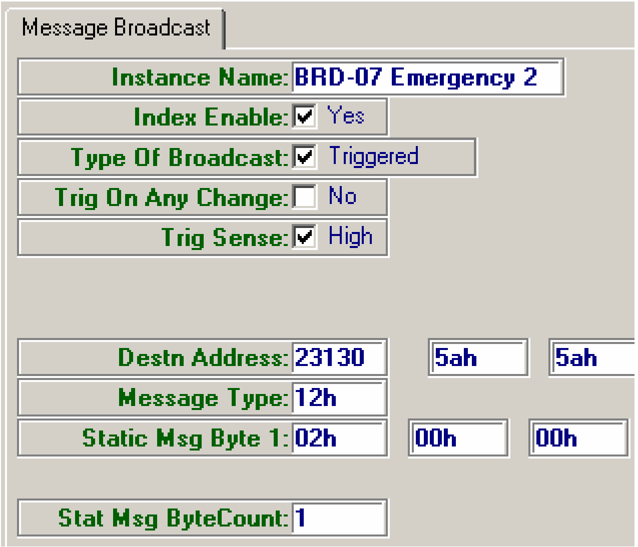


Emergency 2 Broadcast

Sometimes it is necessary to configure to broadcast an emergency 2 message based on a status such as a contact on Input-2. Note: that EM1 must be cleared When the contact opens the emergency can be restored. This can be accomplished by triggering the restore message when the trigger goes true and when it goes false, and triggering the EM2 from a delayed start timer that waits 5 s before sending the message.

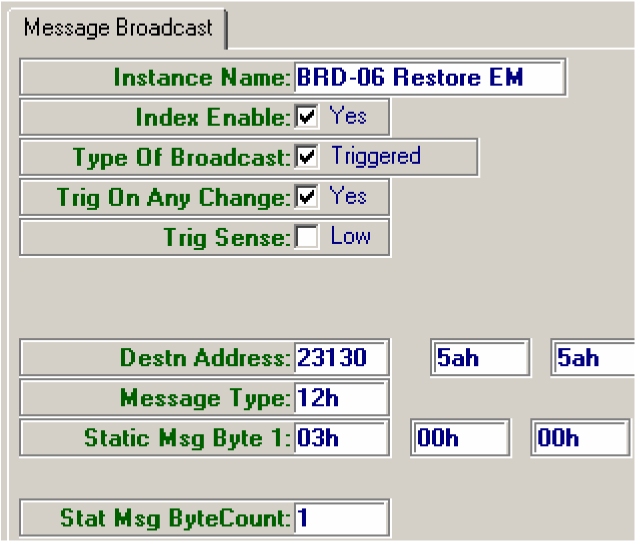






Message Broadcast

Instance Name:	BRD-07 Emergency 2		
Index Enable:	☑ Yes		
Type Of Broadcast:	☑ Triggered		
Trig On Any Change:	□ No		
Trig Sense:	☑ High		
Destn Address:	23130	5ah	5ah
Message Type:	12h		
Static Msg Byte 1:	02h	00h	00h
Stat Msg ByteCount:	1		



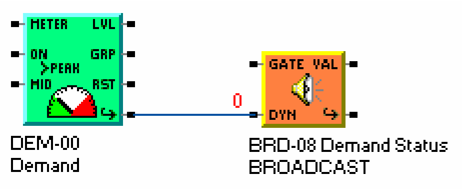
Message Broadcast

Instance Name:	BRD-06 Restore EM		
Index Enable:	☑ Yes		
Type Of Broadcast:	☑ Triggered		
Trig On Any Change:	☑ Yes		
Trig Sense:	□ Low		
Destn Address:	23130	5ah	5ah
Message Type:	12h		
Static Msg Byte 1:	03h	00h	00h
Stat Msg ByteCount:	1		

Dynamic Data Broadcast

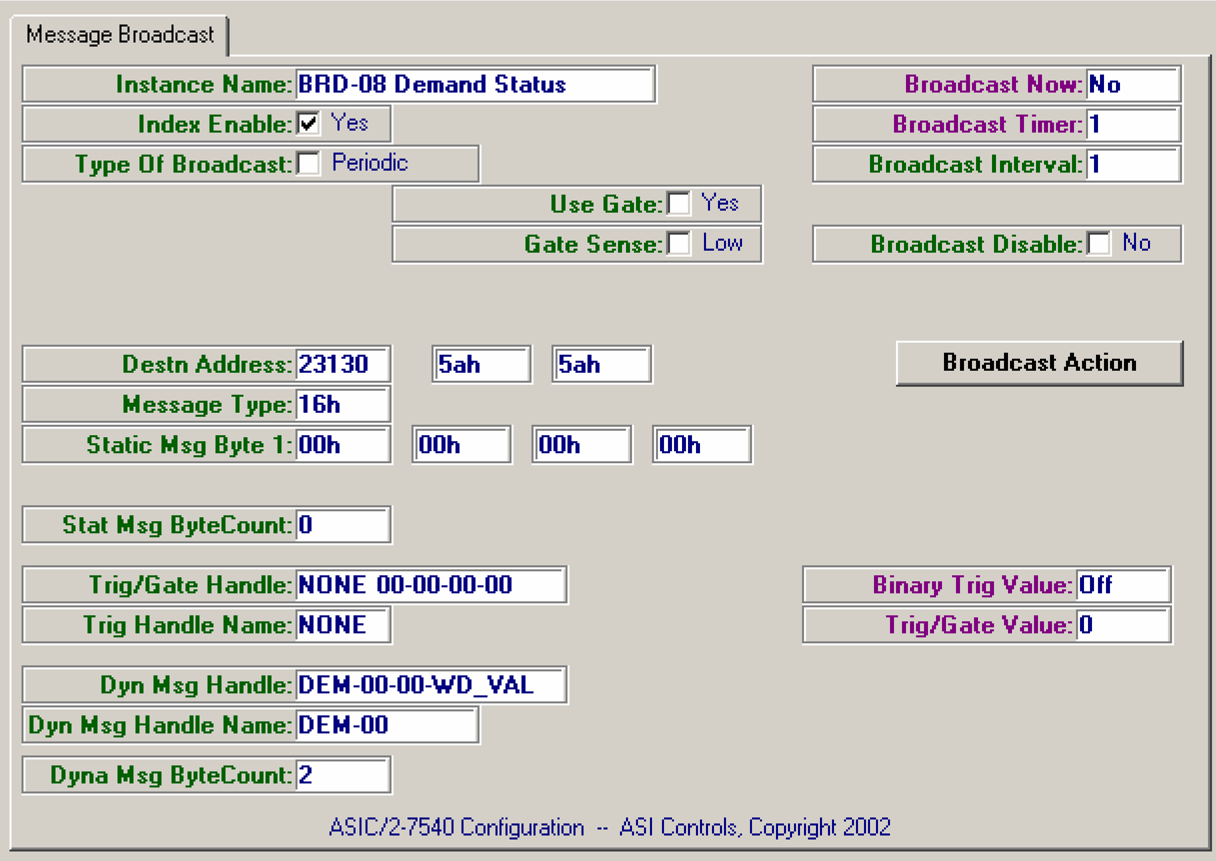
The **Dynamic Message Handle** points at the dynamic data to be included as the last bytes of the message. The **Dynamic Byte Count** give the number of dynamic data bytes to be include in the message. Typically the Dynamic Message Handle returns one or two data bytes.

Demand Level



The Active Demand status can be broadcast periodically once per minute on the local bus. The Dynamic Data handle points at the Active Demand Status. Which has LO Byte = Demand Level, HI Byte = Rotate Group.

The Set/Reset Demand Status, Message Type = 0x16, has M1 = Demand Level, M2 = Rotate Group. Two Bytes of Dynamic Data are used for the broadcast. Consult the ASIC/1 Engineering guide for a discussion of ASIC/1 commands.
Static Byte Count is 0, and the Dynamic Byte Count is 2



Instance Name: BRD-08 Demand Status

Index Enable: ☑ Yes

Type Of Broadcast: □ Periodic

Use Gate: □ Yes

Gate Sense: □ Low

Broadcast Now: No

Broadcast Timer: 1

Broadcast Interval: 1

Broadcast Disable: □ No

Destn Address: 23130 5ah 5ah

Message Type: 16h

Static Msg Byte 1: 00h 00h 00h 00h

Stat Msg ByteCount: 0

Trig/Gate Handle: NONE 00-00-00-00

Trig Handle Name: NONE

Binary Trig Value: Off

Trig/Gate Value: 0

Dyn Msg Handle: DEM-00-00-WD_VAL

Dyn Msg Handle Name: DEM-00

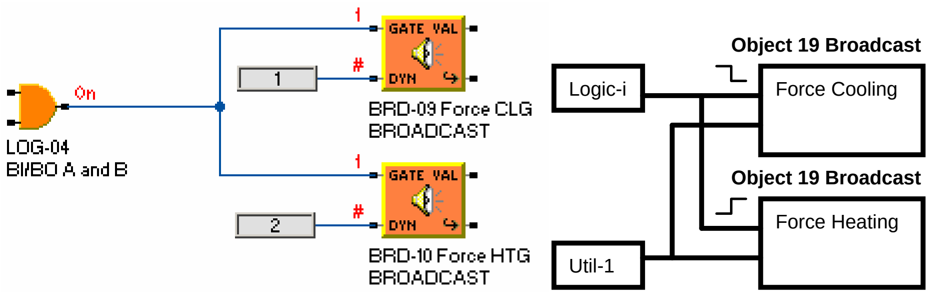
Dyna Msg ByteCount: 2

ASIC/2-7540 Configuration -- ASI Controls, Copyright 2002

Static ASIC/2 Broadcast

The Static Messages to an ASIC/2 controller may broadcast on the local bus using the object write message, MT=0x90.

For example, an ASIC/2 controller on the local bus uses Utility Index 2 Attribute-0 to force the controller into cooling or heating mode. For cooling broadcast a value 1 when a logic block goes low, or for heating broadcast a value 2 when the logic block goes high.

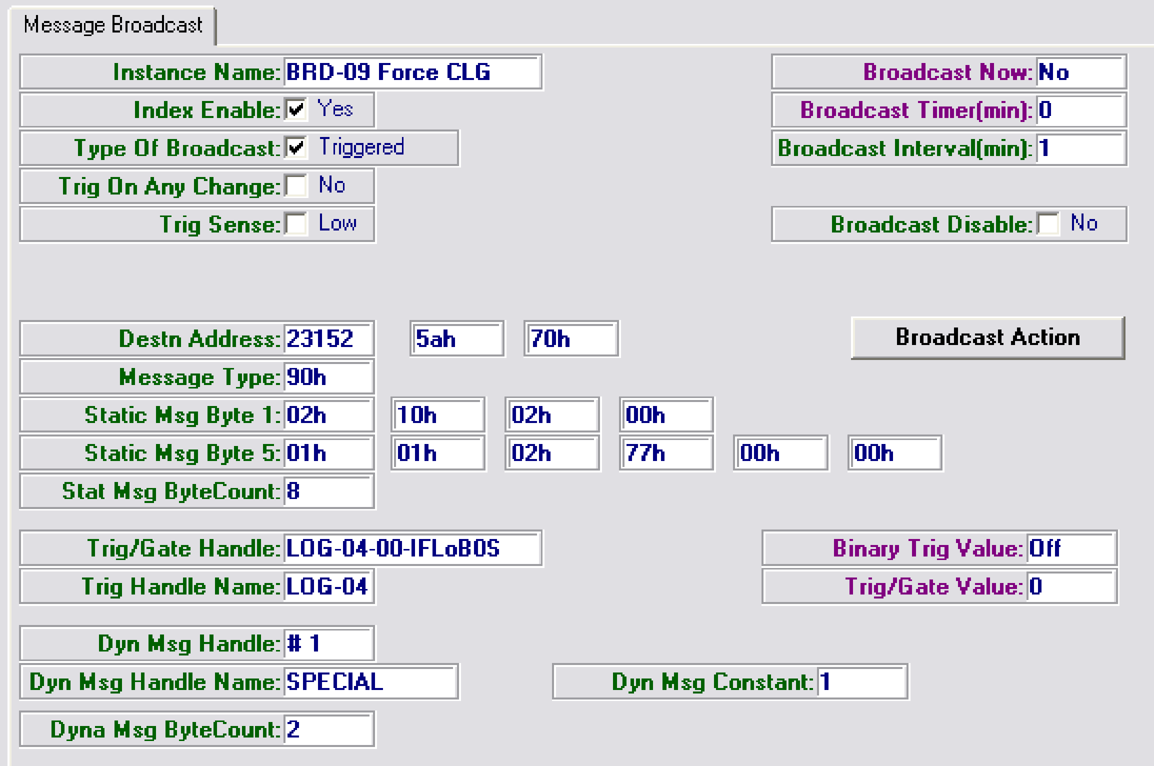


Message Type: 90h, Object Write

Static Message Bytes:

M1	M2	M3	M4	M5	M6	M7	M8	M9	10
2	0x10	1	0	1	1	2	77	0	0
Word	Obj	Index	Attr	Nidx	N_Ar	Bytes	FR	LO	HI

The fixed part of the message is up to the Framing Byte. The Static Byte Count is 8.
Note that the object number 16 is entered in Hex (0x10)
The value to be broadcast is obtained from the Dynamic Data handle and the Dynamic Byte Count is 2



Message Broadcast

Instance Name: BRD-09 Force CLG
Index Enable: ☑ Yes
Type Of Broadcast: ☑ Triggered
Trig On Any Change: □ No
Trig Sense: □ Low

Broadcast Now: No
Broadcast Timer(min): 0
Broadcast Interval(min): 1
Broadcast Disable: □ No

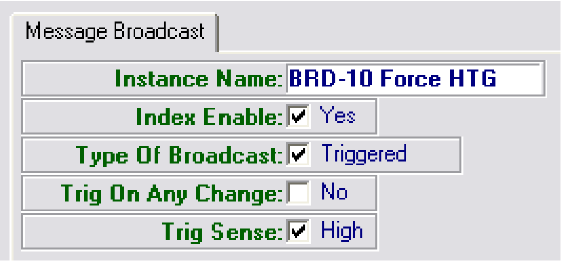
Destn Address: 23152 5ah 70h
Message Type: 90h
Static Msg Byte 1: 02h 10h 02h 00h
Static Msg Byte 5: 01h 01h 02h 77h 00h 00h
Stat Msg ByteCount: 8

Broadcast Action

Trig/Gate Handle: LOG-04-00-IFLoBOS
Trig Handle Name: LOG-04
Binary Trig Value: Off
Trig/Gate Value: 0

Dyn Msg Handle: # 1
Dyn Msg Handle Name: SPECIAL
Dyn Msg Constant: 1
Dyna Msg ByteCount: 2

The Force Heating message is broadcast when the Trigger Sense is high.



Message Broadcast

Instance Name: BRD-10 Force HTG

Index Enable: ☑ Yes

Type Of Broadcast: ☑ Triggered

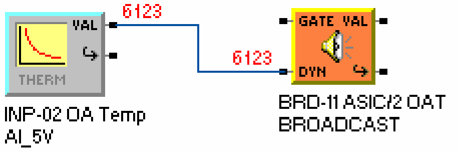
Trig On Any Change: □ No

Trig Sense: ☑ High

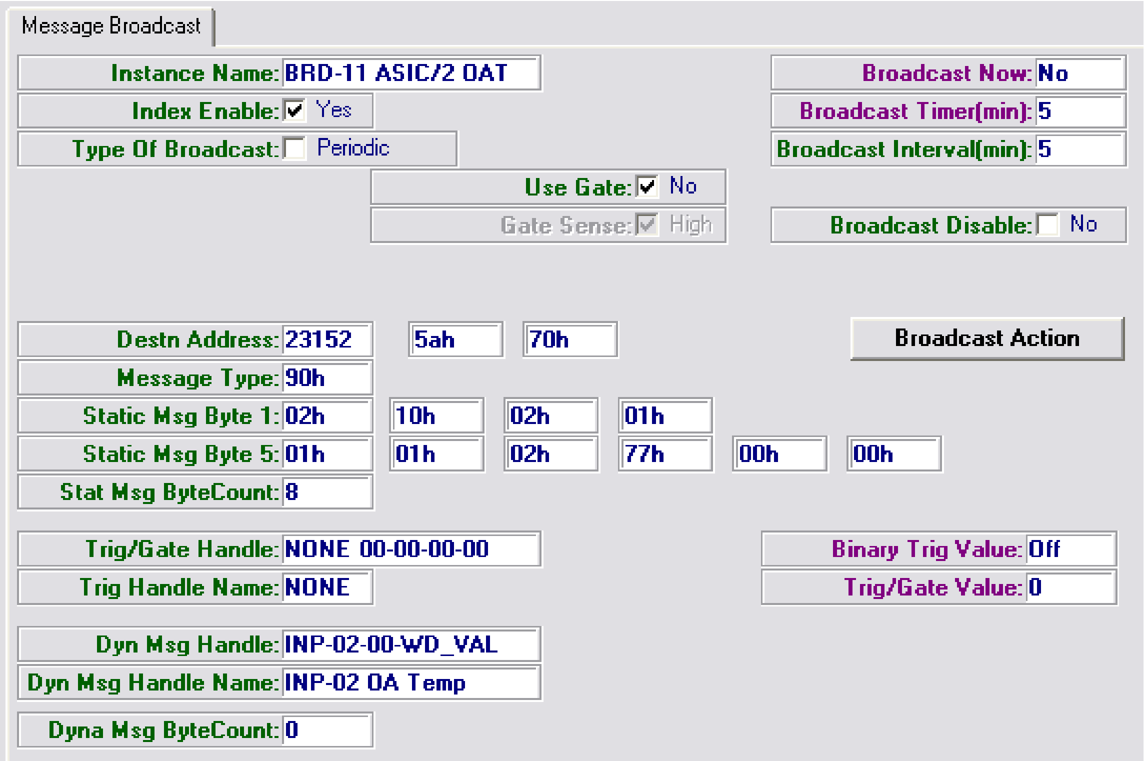
Dynamic ASIC/2 Broadcast

For example, an ASIC/2 controller on the local bus uses Utility Index 2 Attribute-1 to save the Outside Air Temperature broadcast from another ASIC/2 controller on the local bus using the an object write message.

The outdoor air temperature on Input, index 2, attr-0 could be broadcast every 5 minutes to other controllers as Utility index 2, Attr-1.



The static message is configured to set up the message header. The Dynamic Message handle would point at the outdoor air temperature on Input, index 2.



Message Broadcast

Instance Name: BRD-11 ASIC/2 OAT

Index Enable: ☑ Yes

Type Of Broadcast: □ Periodic

Use Gate: ☑ No

Gate Sense: ☑ High

Broadcast Now: No

Broadcast Timer(min): 5

Broadcast Interval(min): 5

Broadcast Disable: □ No

Destn Address: 23152 | 5ah | 70h

Broadcast Action

Message Type: 90h

Static Msg Byte 1: 02h | 10h | 02h | 01h

Static Msg Byte 5: 01h | 01h | 02h | 77h | 00h | 00h

Stat Msg ByteCount: 8

Trig/Gate Handle: NONE 00-00-00-00

Trig Handle Name: NONE

Binary Trig Value: Off

Trig/Gate Value: 0

Dyn Msg Handle: INP-02-00-WD_VAL

Dyn Msg Handle Name: INP-02 OA Temp

Dyna Msg ByteCount: 0

Broadcast Glossary

Broadcast Parameters

Binary Trigger Value

The most recently calculated binary trigger. Trigger/Gate Value is calculated from Trigger Source (which is pointed to by Trigger/Gate Handle). Not User-changeable; "Zero", "Non-zero". (19,X,0,BIT 5)

Broadcast Action

Writing the listed value to this parameter (19,X,4,BYTE) causes the corresponding action to occur:

0	No effect
1	Enable Broadcast object
2	Disable Broadcast object
3	Perform a broadcast immediately

Broadcast Interval

Applies to periodic broadcast mode only. This is the time interval between successive broadcast messages. User-changeable; in minutes. (19,X,14,BYTE)

Broadcast Now

Internal flag that indicates that the index is currently performing a broadcast. Automatically set to "No" upon completion of a broadcast. Not user-changeable; "Yes", "No". (19,X,0,BIT 6)

Broadcast Timer

Used to track Broadcast Interval. Not user-changeable; reads in minutes. (19,X,3,BYTE)

Destination Address

The destination address to be used in the index's broadcast message. User-changeable; integer from 0 to 65535. (19,X,15,2 BYTES)

Dynamic Message Byte Count

The number of dynamic message bytes which are to be included in the index's broadcast message. Integer from 1 to 4; user-changeable. (19,X,22,HIGH NIBBLE) Moved to Attr-28, FW740A..

Dynamic Message Bytes Handle

Points to the location of the block of dynamic bytes which is to be included in the index's message. User-changeable. (19,X,10,4 BYTES)

Index Enable

Enables the index for operation. If not enabled, the index will not send any broadcast messages. "Yes", "No"; user-changeable. (19,X,5,BIT 0)

Index Enable Status

Indicates whether the index is enabled for operation. "Yes", "No"; user-changeable via Index Enable. (19,X,0,BIT 0)

Local Bus Assignment

The SINC/3 allows assigning the Broadcast to Local 1, Local 2, or Both. 0, 1 = Local 1 only, 2 = Local 2 only. FW300B (19,X,5,Bits45)

Message Type

The "message type" byte which is to be included in the actual broadcast message sent by the index. User-changeable; hexadecimal value. (19,X,17,BYTE)

Static Byte Count

The number of static message bytes which are to be included in the index's broadcast message. Integer from 1 to 4; user-changeable. (19,X,22,LO_NBL)- Moved to Attr-28, FW740A..

Static Message Byte X

The static message bytes to be included in the index's broadcast message are stored here. Not user-changeable. X is an integer from 1 to 4. (19,X,18 -> 21,BYTE) Increased to 10 with FW740A..

Trigger on Any Change

Applies to once-only broadcasts only. Configures the index to broadcast upon any change in Trigger Value. User-changeable; "Yes", "No". (19,X,5,BIT 3)

Trigger Sense

Determine when the trigger condition is true. For Trigger Sense "High" the trigger condition is true only for non-zero Trigger Value. For Trigger Sense "Low" the trigger condition is true only for Trigger/Gate Value equal to zero. User-changeable; "High", "Low". (19,X,5,BIT 2)

Trigger Value

The value of the parameter pointed to by the Trigger/Gate Handle, as it was upon last reading. Not user-changeable; integer. (19,X,1,2 BYTES)

Trigger/Gate Handle

Points to the parameter from which the value for the trigger is to be obtained. User-changeable. (19,X,6,4 BYTES)

Type of Broadcast

Configures this index to perform its broadcast on a once only or a periodic basis. User-changeable; "Periodic", "Once-Only". (19,X,5,BIT 1)

Type of Broadcast - Status

Indicates whether the index performs its broadcasts on a once-only or a periodic basis. Mirrors the value chosen via Broadcast Type. Not user-changeable; "Periodic", "Once-Only". (19,X,0,BIT 1)

Broadcast Properties

The BROADCAST object defines the present values and setup parameters used by the broadcast control block. Each ASIC/2-7040 or SC/1-9040 controller maintains a Broadcast Object. Each index of the 19-BROADCAST represents a different message to be broadcast to a specific device, group, or global address.

BROADCAST

Object Number = 19	
Data Type	= BYTE
Index	= 6 or As allocated
Attribute	= 23 (0..22) (FW907..)
	= 29 (0..28) (FW740A..,300B)
DYNAMIC Attribute	= 5 (0..4)
STATIC Attribute	= 18 (6..22) (FW907..)
	= 24 (6..28) (FW740A..,300B..)

Broadcast Firmware Revision

ASIC/2-7540 FW754A Rev 1.0 Forthcoming 2005

o as in FW740E.

ASIC/3-3000 FW300B Rev 1.0 Released 10/06/1999

o Add Local Bus Assignment for Local 1 and Local 2

ASIC/2-7040 740C 2.4 Released 06/09/99 CHK 0x68E1
ASIC/2-8040 FW840C Rev 1.4 Released 06/09/99 CHK
Fixes problem when number of Broadcast indexes >127.

ASIC/2-7040 FW740B Rev 1.2 Released 08/18/95
ASIC/2-7040 FW740A Rev 2.1 Released 08/16/95

o NOTE: The Broadcast object must have a valid Trigger/Gate Handle, even if it is not used, otherwise periodic broadcast will not happen.

ASIC/2-7040 FW740A Rev 1.7 8 Dec 1994

o Fixed Pass through bug. Broadcast interferes with pass through.

ASIC/2-7040 FW740A Rev 1.6 Released 2 Dec 1994

o Fixes Broadcast to responding controllers. If the broadcast address is an individual controller, the destination will respond.
The Local bus now accepts this response but does not do anything with the response. Broadcast to responding controllers appears to work.
o Periodic broadcasts must have a valid handle.
o To broadcast all of the time, the On Any Change must be yes.

ASIC/2-7040 FW740A Rev 1.4Release 7 Sept 94

o Fixed Triggered Broadcast to insure broadcasts are not missed.

7040 FW740A Rev. 1.0 Jan 1993

o Remove String Message
o Add 3F time broadcast.
o Added 6 additional Static Message Bytes Attr-22..27
o Moved Byte Counts from Attr-22 to Attr-28

SC/1 FW907E Rev. 1.0, Released 3 Dec 1992

o Upgrade Broadcast,
o Fixed Broadcast repeat if responded to

SC1 FW907A Rev 1.0, Released 25 July 1991

Broadcast DYNAMIC Properties

Attr-0 Present value
bit 0 - **Index Enable - Status**;
0 = disabled, 1 = enabled.
bit 1 - **Type Of Broadcast**;
0 = periodic, 1 = triggered.
bit 2 - Spare
bit 3 - Spare
bit 4 - Spare

bit 5 - Binary Trigger Value;
Internal Flag indicating current binary value of trigger
bit 6 - Broadcast Now;
Internal Flag indicating that it is time to broadcast
bit 7 - Broadcast Poll
Internal flag

Attr-1,2 **Trigger/Gate Value** Word
Current Trigger/Gate Value - low byte.
Attr-1 LO BYTE
Attr-2 HI BYTE .

Attr-3 **Broadcast Timer** (minutes)
When the timer counts to zero, then a flag is set to initiate broadcast of the message.
Used for sustained (periodic) broadcast messages.

Attr-4 **Broadcast Action**
0 - No Operation
1 - Broadcast Index Enable.
2 - Broadcast Index Disable
3 - Broadcast now

Broadcast STATIC Properties

Attr-5 Setup

bit 0 - **Index Enable**

0 = disabled, 1 = enabled.

bit 1 - **Type Of Broadcast**;

0 = periodic, 1 = triggered.

bit 2 - **Trigger Sense** (Triggered)

1 = HI trigger zero to Non-zero;
0 = Low trigger non-zero to zero

Gate Sense (Periodic)

0=Lo, 1= Hi

bit 3 - **Trigger on Any Change** (Triggered)

0 = No, 1 = Yes
If Trigger on Any Change is No then Trigger Sense is examined.

Use Gate (Periodic)

1 = No, 0 = Yes
If Use Gate is No, then Gate Sense is ignored.
If Use Gate is Yes, then Gate Sense is used and it must have a valid Trigger/Gate handle.

bits 45 - **Local Bus Assignment -(** FW300B..)
assigns Broadcast to Local 1, Local 2, or Both. 0, 1 = Local 1 only, 2 = Local 2 only.
bit 6 - Spare
bit 7 - Spare

Attr-6,7,8,9 **Trigger/Gate Handle**

Attr-6 Trigger/Gate Handle - Object
Attr-7 Trigger/Gate Handle - Index
Attr-8 Trigger/Gate Handle - Attribute
Attr-9 Trigger/Gate Handle - Select

Attr-10,11,12,13 **Dynamic Message Handle**

Attr-10 Data Source Handle - Object
Attr-11 Data Source Handle - Index
Attr-12 Data Source Handle - Attribute
Attr-13 Data Source Handle - Select

Attr-14 **Broadcast Interval**

Broadcast interval in minutes, 1..255.

Attr-15,16 **Destination Address**

Attr-15 Destination Address LO
Attr-16 Destination Address HI

Static Message Data

Attr-17 **Message Type**

Communication Message Type, if Message Type is 38h, then it will take the current time and day from clock object for time broadcast.

Attr-18 **Static Message Byte 1** (FW907A.,FW740A..)
Attr-19 **Static Message Byte 2** (FW907A.,FW740A..)
Attr-20 **Static Message Byte 3** (FW907A.,FW740A..)
Attr-21 **Static Message Byte 4** (FW907A.,FW740A..)
Attr-22 **Static Message Byte 5** (FW740A..)
Attr-23 **Static Message Byte 6** (FW740A..)
Attr-24 **Static Message Byte 7** (FW740A..)
Attr-25 **Static Message Byte 8** (FW740A..)
Attr-26 **Static Message Byte 9** (FW740A..)
Attr-27 **Static Message Byte 10** (FW740A..)

Attr-28 Byte Count (FW907A.. Attr-22 Byte Count)

LS_NBL (bits0..3) - **Static Byte Count**

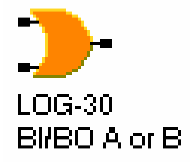
Number of Static Message Bytes (0..4) to include in the message from Attr-18..Attr-21 of this index.

MS_NBL (bits4..7) - **Dynamic Byte Count**

(0..15) Number of Dynamic Message Bytes to include after static message bytes in the message from the Data Source Handle.

Object 20 - Logic

Logic Summary



Overview: The Logic object performs a sequence of Boolean operations on its inputs to obtain its output.

Handles: The three input handles, A, B, C may be used to return any attribute of any index of any object in memory. The attributes may be masked so as to contain only a given bit. Attributes may also be converted to read in as 0000h or FFFFh depending on whether the attribute is zero or non-zero. The Input C Handle is used only by select type of logic blocks, and then only if the Selector Source has been chosen to be Input C Handle. In the ASIC/2-8040 FW840A.. the Input C Handle is always used for selectors and Selector Source is ignored.

Outputs: The output may be analog or binary according to the operation performed. The output is a Binary Value, or a 2 byte Analog value depending on the Logic Class used.

Operations: Three classes of operations are available: Binary inputs to binary output: Analog inputs to binary output, Analog inputs to analog output:

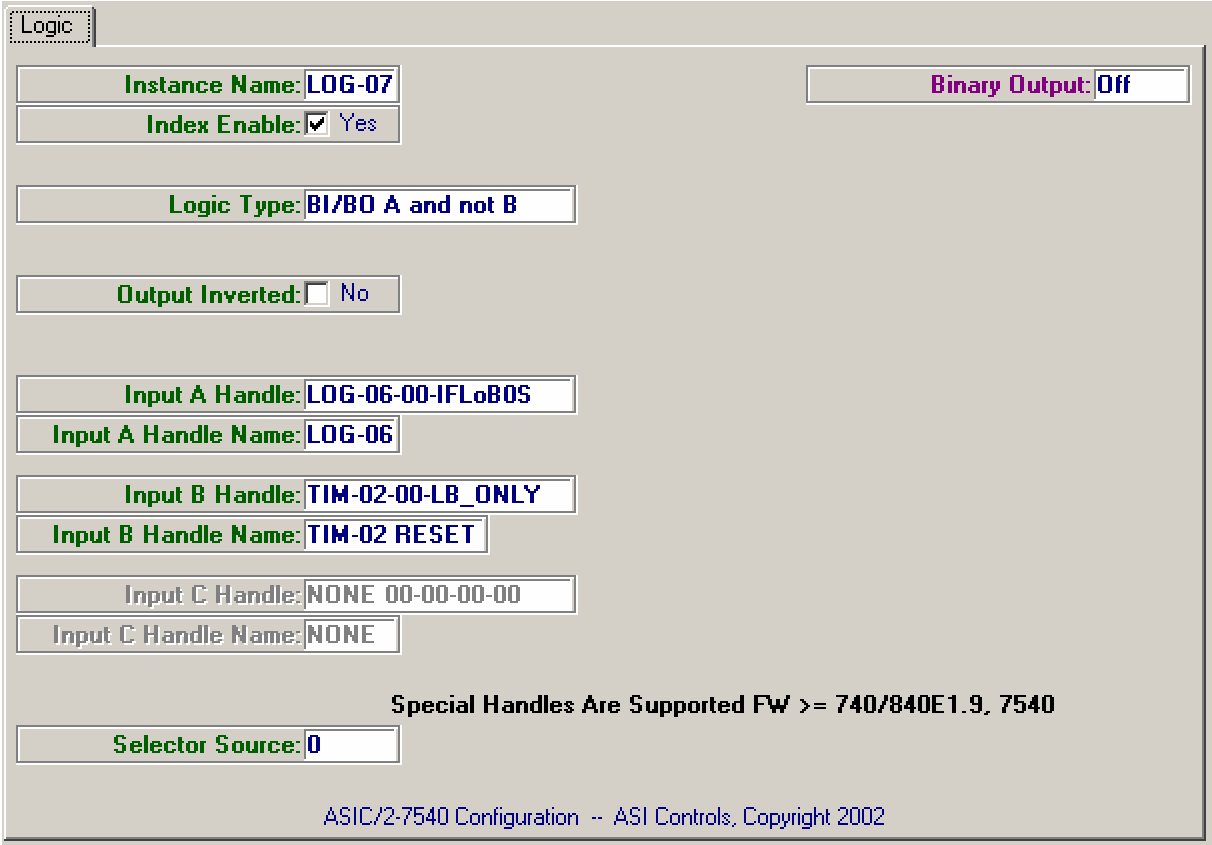
Binary inputs to binary output: The Boolean operators "and", "or", and "exclusive or" are available; either or both of the inputs may be inverted (the "not" operator). Optionally, the logic object may be configured to select one of the inputs, Select A or Select B depending on the logical state of the selector in which case the gate acts as a switch between the value of A or the value of B.

Analog inputs to binary output: The comparison operators "greater than", "greater than or equal to", "equal to", "less than", "less than or equal to" may be applied to the 2 analog inputs; the output is yes (non-zero) if statement is true, or no (zero) if the statement is false.

Analog inputs to analog output: The greater or the lesser of the 2 analog inputs may be chosen to be output. The inputs may be considered as either signed or unsigned values. Optionally, the logic object may be configured to select one of the inputs, Select A or Select B depending on the logical state of the selector. Note that when a handle identifies the analog output of the logic block the handle select "2 Adjacent Bytes" should always be used.

Special Handle: Any logic handle can be made Special which allow a constant value to be entered into the handle. (FW740E2.0, 840E2.0, 300B2.0, 754A,,)

Logic Operation



Logic

Instance Name: LOG-07

Index Enable: ☑ Yes

Binary Output: Off

Logic Type: BI/BO A and not B

Output Inverted: □ No

Input A Handle: LOG-06-00-IFLoBOS

Input A Handle Name: LOG-06

Input B Handle: TIM-02-00-LB_ONLY

Input B Handle Name: TIM-02 RESET

Input C Handle: NONE 00-00-00-00

Input C Handle Name: NONE

Special Handles Are Supported FW >= 740/840E1.9, 7540

Selector Source: 0

ASIC/2-7540 Configuration -- ASI Controls, Copyright 2002

Binary Inputs/Binary Outputs (BI/BO)

A class of logical operations is available that takes a two binary inputs and gives a binary output. This group has Logic Class = 0 (Attr-2 Setup, bits 4,5)

The Setup Parameter (Attr-2) for Logic Class = 0, is given by

Attr-2 Setup = Type, bits 0..3
+0 , bits 4,5 Logic Class
+0 , bit 6, Inverted Output
+128, bit 7, Index Enable

BI/BO Logic Types

A full set of Boolean operations are available. Note: ~A means "Not A", etc.

Logic Type		**Logical Operation**
128	A. and. B;	True if both A and B are both True
129	~A .and. ~B;	True if both .not. A and .not. B are True
130	~A .or. ~B;	True if either .not. A or .not. B are True
131	A .or. B;	True if either A or B are True
132	~A .and. B;	True if both .not. A and B are True
133	A .and. ~B;	True if both A and .not. B are True
134	A .or. ~B;	True if either A or .not. B are True
135	~A .or. B;	True if either .not. A or B are True
136	A .xor. B;	True if either A or B are True, but not both
137	~(A .xor. B);	Not True if either A or B are True, but not both
138	Select A when Hi;	Value of A to output when Selector non-zero(HI)
139	Select B when Hi;	Value of B to output when Selector non-zero(HI)
140	Triple OR ;	True if either A or B or C is true.
141	Triple AND ;	True only if A and B and C are all true.

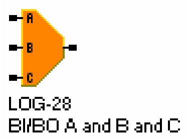
And Type Logic

LOG-29
BI/BO A and B

AND type Boolean logic blocks include: A .and. B; ~A .and. ~B; ~A .and. B; and A .and. ~B

128	A .and. B;	True if both A and B are both True
129	~A .and. ~B;	True if both .not. A and .not. B are True
132	~A .and. B;	True if both .not. A and B are True
133	A .and. ~B;	True if both A and .not. B are True

Triple And



The Triple AND is true if and only if all three input values are true. The triple And always uses the value returned by the C Handle

141	Triple AND ;	True only if A and B and C are all true.

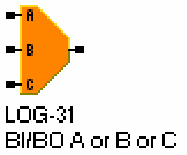
Or Type Logic

LOG-30
BI/BO A or B

OR type Boolean logic blocks include: A .or. B; ~A .or. ~B A .or. ~B; ~A .or. B; A .xor. B; and ~(A .xor. B):

130	~A .or. ~B;	True if either .not. A or .not. B are True
131	A .or. B;	True if either A or B are True
134	A .or. ~B;	True if either A or .not. B are True
135	~A .or. B;	True if either .not. A or B are True
136	A .xor. B;	True if either A or B are True, but not both
137	~(A .xor. B);	Not True if either A or B are True, but not both

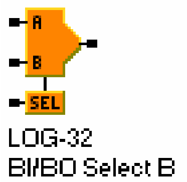
Triple OR



The Triple OR is true if any of the three input values is true. The triple Or always uses the value returned by the C Handle.

140	Triple OR ;	True if either A or B or C is true.

BI/BO Selector Type Logic



In BI/BO, Logic Class 0, if the Logic type of comparison is Select, 138 or 139, then the Source of the Selector must be identified. The Selector Source for FW840A.. is always the Input C handle. For FW740A.. it may be another logic block, using the Source of Selector Signal or it may be the value identified by Input C Handle.

138 Binary Select A when Hi; else select B
Value of A to output when Selector non-zero(HI)
139 Binary Select B when Hi; else select B
Value of B to output when Selector non-zero(HI)

Source of Selector Signal (FW840A..)

The Selector Source for FW840A.. is always the Input C handle. The Source of Selector Signal is ignored.

Source of Selector Signal (FW700I..)

If the Source of Selector Signal is used then the Selector Source = 0, 1 or ,2 points at another logic block and the Selector Index gives the Index of Logic block which is to be used. The Selector Source identifies which input or output of the Logic block used as the source of select signal: 0; Use Present Value(Attr-0) ; 1; Use Input A Source ; or 2; Use Input B Source. Note: Only logic blocks with index 0..63 may be selected in this manner.

Selector Index Logic Index (0..63) to examine for Selector Source = 0, 1, or 2. (Attr-3, bits 0..5)

Selector Source Value to examine for select.(Attr-3, bit 6,7)
0; Use Present Value of Selector Index as source
1; Use Input A of Selector Index as source
2; Use Input B of Selector Index as source
3; Use Input C Handle

If the Selector Source = 3, then value of the selector is obtained using the full Input C Handle which may point at any parameter in the controller and is not restricted to another logic block. If the value returned by Input C Handle is zero then the selector is false. If the value returned by Input C Handle is non-zero then the selector is true and the alternate value, for example Select A, is transferred to the output.

Note: The Input C Handle has been added with FW740A., FW840A.

Analog Inputs/Binary Outputs (AI/BO)

An AI/BO class of logical operations is available that takes a two analog inputs and gives a binary output. This group has Logic Class = 1 (Attr-2 Setup, bits 4,5)

The Setup Parameter (Attr-2) for Logic Class = 1, is given by
Attr-2 Setup = Type, bits 0..3
+16 , bits 4,5 Logic Class = 1
+0 , bit 6, No Inverted Output
+128, bit 7, Index Enable

Note: In signed comparison 2 is larger than -10 . In the arithmetic of computers a signed negative value looks like a number greater than 32767 and consequently when using unsigned comparison negative numbers always appear larger.

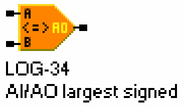
AI/BO Logic Types

144	A .gt. B;	True if signed A is greater than B
145	A .ge. B;	True if signed A is greater than or equal B
146	A .eq. B ;	True if signed A is equal to B
147	A .lt. B;	True if signed A is less than B
148	A .le. B;	True if signed A is less than or equal to B
149	A > B;	True if unsigned A is greater than B
150	A >= B; True if unsigned A is greater than or equal B	
151	A < B;	True if unsigned A is less than B
152	A <= B;	True if unsigned A is less than or equal to B

Hysteresis

Hysteresis is always applied in Logic Class 1 (AI/BO) in going from true to false. It is signed or unsigned depending on the Type of Comparison The Hysteresis value is contained in Attr-12, Lo Byte, and Attr-13, Hi Byte.

Analog Inputs/Analog Outputs (AI/AO)



An AI/AO class of logical operations is available that takes a two analog inputs and gives an analog output. This group has Logic Class = 2 (Attr-2 Setup, bits 4,5)

Note: The analog output value consists of two bytes Attr-0 = Lo and Attr-1 = HI and will be returned by a handle as a Word Value.

The Setup Parameter (Attr-2) for Logic Class =2, is given by
Attr-2 Setup =   Type, bits 0..3
+32, bits 4,5 Logic Class = 2
+0 , bit 6, No Inverted Output
+128, bit 7, Index Enable

AI/AO Logic Types

160	highest unsigned;	Value is unsigned greater of A or B
161	lowest unsigned;	Value is unsigned smaller of A or B
162	largest signed;	Value is signed greater of A or B
163	smallest signed;	Value is signed smaller of A or B

AI/AO Selector Types

A
AO
B
SEL
LOG-35
AI/AO Select A

In AI/AO, Logic Class 2, if the Logic type of comparison is Select, 164 or 165, then the Source of the Selector must be identified. The Selector Source for FW840A.. is always the Input C handle. For FW740A.. it may be another logic block, using the Source of Selector Signal or it may be the value identified by Input C Handle. .

164   Select A when Hi;
Value of A to output when Selector is non-zero (HI)
165   Select B when Hi;
Value of B to output when Selector is non-zero (HI)

Source of Selector Signal (FW840A..)

The Selector Source for FW840A.. is always the Input C handle. The Source of Selector Signal is ignored.

Source of Selector Signal (FW700IA..)

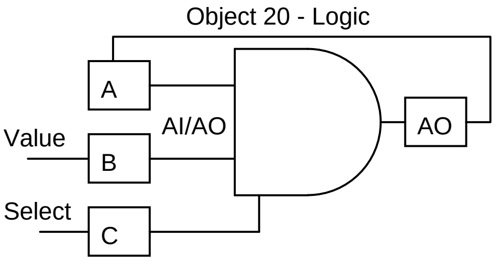
Selector Index gives the Index of Logic block which is to be used.   The Selector Source identifies which input or output of the Logic block used as the source of select signal: 0; Use Present Value(Attr-0) ; 1; Use Input A Source ; or 2; Use Input B Source. Note: Only logic blocks with index 0..63 may be selected in this manner.

If the Selector Source = 3, Use Input C Handle, then value of the selector is obtained using the full Input C Handle which may point at any parameter in the controller and is not restricted to another logic block. If the value returned by Input C Handle is zero then the selector is false. If the value returned by Input C Handle is non-zero then the selector is true and the alternate value, for example Select A, is transferred to the output.

Note: The output of an analog select is a word of 2 bytes consisting of Attribute 0 and 1. When pointing at this analog logic output value it is necessary to use a Handle Select for 2 bytes.

Logic Configuration

Sample and Hold



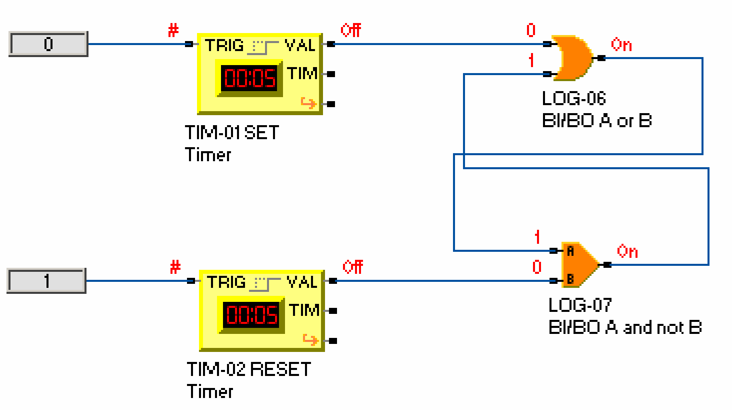
Type 165 Select B used as Sample and Hold

Sample and Hold can be constructed using an Analog or Binary Select Block. When the Select handle goes true, the value at Input B is sampled by the output of the Logic block. When the Select handle goes false, the last value is held.

The output of an analog select is a word of 2 bytes consisting of Attribute 0 and 1. Make sure that when pointing at this analog logic output value to use a Handle Select for 2 Adjacent Bytes.

Latching Logic

Latching Logic can be constructed using two logic gates. This is used when one wishes to set a value when a certain condition occurs, but only clear the condition by a specific intervention.



The order of indexes is important here. The logic block with the reset signal must have an index number higher than that of the logic block with the set signal.

Logic Glossary

Logic Parameters

Analog Output

For analog to analog type comparisons . The result of the logical comparison. Not user-changeable; integer word. (20,X,0,2 BYTES)

Binary Output

For analog to binary, or binary to binary type comparisons: The result of the logical comparison. Not user-changeable; "True", "False". (20,X,0,BIT 0)

Hysteresis

For analog to binary type comparisons: If Binary Output has become "True" as the result of an analog comparison (example: A >= B), then for it to become "False", the comparison must move in the opposite direction by at least Hysteresis (example: A < [B - Hysteresis]).   (20,X,12,2 BYTES)

Index Enable

Enables the index for operation. "Yes", "No"; user-configurable. (20,X,2,BITS 6)

Input A Constant

The numeric value returned when the Input A Handle is Special. which Logic Value User-configurable. (20,X,6,2 BYTES)

Input A Handle

Points to the parameter from which Logic Value A is to be obtained. User-configurable. (20,X,4,4 BYTES)

Input B Constant

The numeric value returned when the Input B Handle is Special. which Logic Value User-configurable. (20,X,10,2 BYTES)

Input B Handle

Points to the parameter from which Logic Value B is to be obtained. User-configurable. (20,X,8,4 BYTES)

Input C Constant

The numeric value returned when the Input C Handle is Special. which Logic Value User-configurable. (20,X,16,2 BYTES)

Input C Handle

Points to the parameter from which Select Value is to be obtained, for Select Logic in FW840A.., or in FW740A.., if the Selector Source is 3, Use Input C Handle. . User-configurable. (20,X,14,4 BYTES)

Logic Class

Designates if the logic comparison is to be analog inputs to analog outputs, binary inputs to binary outputs, or analog inputs to binary outputs. See the Logic Object Definition for details. User-configurable; integer code. (20,X,2,BITS 4,5)

Logic Type

Designates the type of values being compared (analog inputs to analog outputs, binary inputs to binary outputs, or analog inputs to binary outputs) he type of logic comparison to be made. Composed of the Logic Class, Type of Comparison, and Index Enabled. For enabled indexes has a value from 128 to 165.: and   See the Logic Object Definition for details. User-configurable; integer code. (20,X,2,Byte)

Output Inverted

Applies only to those Logic indexes with binary outputs. "On" designates that the comparison result is to be inverted before being placed in Binary Output; "Off" designates that the comparison result is to be output as is. The value of Output Inverted is also equal to Output Value for those Logic indexes which are disabled (Output Inverted = "On" implies the Output Value will be "Off", Output Inverted = "Off" implies Output Value = "Off"). User-configurable; "High", "Low". (20,X,2,BIT 7) This was been removed from FW740A.. It has been reinstated in FW840A..

Selector Index

For select type comparisons (either analog to analog or binary to binary). If the Selector Source = 0, 1 or ,2 then selector points at another logic block and the Selector Index gives the Index of Logic object which is to be used. Note: Only logic blocks with index 0..63 may be selected in this manner. (20,X,3,Bits 0..5) FW700I.
Not used in FW740/840.

Selector Source

For select type comparisons (either analog to analog or binary to binary). The Selector Source identifies which input or output of the Logic block used as the source of select signal: 0; Use Present Value(Attr-0) ; 1; Use Input A Source ; or 2; Use Input B Source. User-configurable; integer. If the Selector Source = 3, Use Input C Handle, then value of the selector is obtained using the full Input C Handle which may point at any parameter in the controller and is not restricted to another logic block. (20,X,3,Bits 6,7) FW700I.
Not used in FW740/840.

Type of Comparison

This code designates the type of logical comparison to be performed. See the Logic Object Definition for details. User-configurable; integer code. (20,X,2,BITS 0..3)

Logic Properties

The Logic Object allows generation of a binary or analog output based on logical comparison of two inputs. One of two inputs can also be selected based on the output of another logical block or parameter.

The LOGIC object defines the present values and setup parameters used by the broadcast control block.

Object Name	LOGIC
Object Number	= 20
Data Type	= BYTE
Index	= 16 As allocated
Attributes	= 14 (0..13)
DYNAMIC Attribute	= 2 (0..1)
STATIC Attribute	= 12 (2..13) (FW907A..,FW700A..)
Attributes	= 18 (0..17)
DYNAMIC Attribute	= 2 (0..1)
STATIC Attribute	= 16 (2..17) (FW740A...FW840A..)

Firmware Revision - Logic

ASIC/2-7540 FW754A Rev 1.0 Forthcoming 2005
o As in FW740E2.0.

ASIC/2-7040 FW740E Rev 1.9 Unreleased 11/17/2000
ASIC/2-8040 FW840E Rev 1.9 Unreleased 11/17/2000
o Adds Special Handles generally to objects such as Logic
Note: Not implemented in FW300B.

ASIC/2-7040 FW740C Rev 2.4 Released 06/09/99 CHK 0x68E1
ASIC/2-8040 FW840C Rev 1.4 Released 06/09/99 CHK 69B7h
o Fixes problem with Inverted Logic, for example Logic Type 192, 196, etc.

ASIC/2-7040 FW740C Rev 1.0 Released 01/29/97
Restore Output Inverted Attr-2 LOBit6
Always use Input C handle for Select.

ASIC/2-8040 FW840A Rev 1.0, March 1996
Restore Output Inverted Attr-2 LOBit6
Always use Input C handle for Select.

ASIC/2-7040 FW740A Rev 1.0 Alpha Test (31 March 1994)
Add C Handle Attr-14..17 and Selector Source = 3
Delete Output Inverted Attr-2 LOBit6

ASIC/2-7000 FW700A.. Rev 1.0 Released 12/05/91
SC/1-9040 FW907A Rev 1.0 Released 07/25/91

Logic DYNAMIC Properties

Attr-0 **Analog Output - LO Byte**
Note: Any handle that points at this analog value should select 2 Adjacent Bytes, so that the word value will be obtained.

Attr-0 Binary Output
Attr-0 bit 0 - **Binary Output Status**
0 = OFF, 1 = ON
bit 1..7 undefined

Attr-1 **Analog Output - HI Byte**
Binary Output - Spare

Logic STATIC Properties

Attr-2 Setup

Attr-2 LONBL (bits 0..3) **Type of Comparison**
(Depends on Class)
See Below

Attr-2 LO bits 4,5 - **Logic Class**
0 = (BI/BO) Binary Inputs, Binary Outputs
1 = (AI/BO) Analog Inputs, Binary Outputs
2 = (AI/AO) Analog Inputs, Analog Outputs

Attr-2 LO bits 6 - **Output Inverted**
This has been removed from FW740A.., 740B, & 740D.
It ha been restored with FW740C.

For FW700A..I, FW840A.. for Binary outputs, Logic Class 0 and 1 the output can be inverted; 1 = Yes, 0 = No. Applies to Logic indexes with binary outputs only.
Output Inverted = Yes designates that the comparison result is to be inverted before being placed in Binary Output; Output Inverted = No designates that comparison result is to be output as is.
Output Inverted also designates the Output Value to be used for Logic indexes which are disabled, Index Enable = No.
Output Inverted = Yes, "On"
Output Inverted = No, "Off"

Attr-2 LO bits 7 - **Index Enable**
0 = No, Disabled
1 = Yes, Enabled,

Attr-3 Source of Selector Signal FW740A.., FW700A..
Not used by FW840A..
The selector looks only at an another logic block. In AI/AO, Logic Class 2, if Type of Comparison is Selector = 4,5 , or in BI/BO, Logic Class 0, if Type of Comparison is Selector = 10,11 , then the Source of Selector Signal gives the index of Logic block which is to be used as the source for the select.

Attr-3 bits 0..5 - **Selector Index**
Index of Logic Block to examine for Selector(0..63)

Attr-3 bit 6,7 - **Selector Source**
Word To Examine
0; Use Present Value of Selector Index as Source
1; Use Input A Value of Selector Index as Source
2; Use Input B Value of Selector Index as Source
3; Use Input C Handle Value as Source ***New! (FW740A..)***
ignore selector handle

Attr-4,5,6,7 **Input A Handle**
The Logic A is the sensed variable that is used for the input A to the logical operation or comparison.
Attr-4 Input A Object
Attr-5 Input A Index
Attr-6 Input A Attribute
Attr-7 Input A Select

Note: Handle Select is used to identify which data the handle should retrieve and not the same as the logical Selector as discussed above.

Attr-8,9,10,11 **Input B Handle**
The Logic B is the sensed variable that is used for the input B to the logical operation or comparison.
Attr-8 Input B Object
Attr-9 Input B Index
Attr-10 Input B Attribute
Attr-11 Input B Select

Attr-12,13 **Hysteresis**

Hysteresis is always applied in Logic Class 1 (AI/BO) in going from true to false. It is in the same units as the Logic Value A and B. It is signed or unsigned depending on the Type of Comparison
Attr-12 Hysteresis- LO Byte
Attr-13 Hysteresis- HI Byte

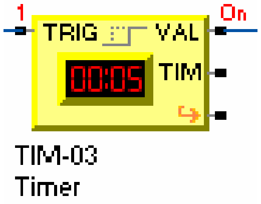
Attr-14,15,16,17 **Input C Handle**

The Logic C is the sensed variable that is used for the select if the **Selector Source** is 3, Use C Handle (Attr-3 bit 6,7)
Always used by FW840A for Select.

Attr-14 Input C Object
Attr-15 Input C Index
Attr-16 Input C Attribute
Attr-17 Input C Select

Object 21 - Timer

Timer Summary



Overview: The Timer object is used to keep track of user-defined delays following a triggering event. Each index of the timer may be configured to one of the following four timer-types:

1) One Shot (Positive or Negative Trigger or both)

2) Integrating (Positive or Negative Trigger or both)

3) Delayed Start on Positive Trigger

4) Delayed Stop on Negative Trigger

Input: The only input to the timer is the trigger value, which may be configured to be read from any attribute of any index of any object. The input may be bit masked and the bit converted to a 0000h or FFFFh. The input may also be inverted.

Triggers: A Positive-type trigger occurs when the trigger value changes from a zero-value to a non-zero value. A Negative-type trigger occurs when the trigger value changes from a non-zero value to a zero-value.

Output: The output is 0 if the timer is disabled; otherwise, the timer functions as follows, depending on the timer-type:

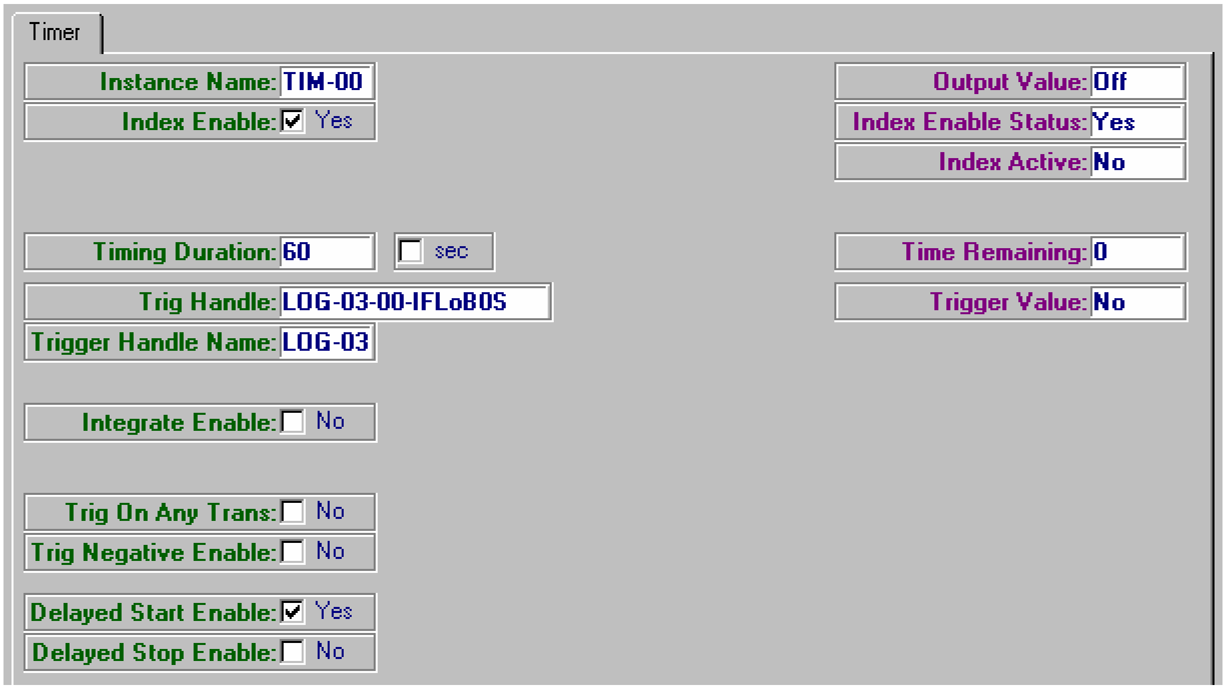
One-Shot Timer: Under this timer-type, a Positive trigger, a Negative trigger, or both triggers, may be designated to start the timer. Upon the beginning of timing, the timing duration is loaded into the timer and the output goes high until the timer has timed-out. Upon time-out, the output goes low.

Integrating Timer: Similar to One-Shot, except if subsequent triggers of the designated type occur while the timer is timing, the timer is re-loaded with the timing duration on each occasion. In this fashion, the timer will only-time out after a timing duration has fully elapsed with no triggers occurring.

Delayed Start Timer: The timer is loaded with the timing duration upon the occurrence of a Positive trigger, but the output is held to low until the timer times-out. The output is then held high until the trigger value returns to zero.

Delayed Stop Timer: The timer's output goes high upon occurrence of a positive trigger but the timer only begins timing upon the subsequent negative trigger (when the trigger value falls back to zero). The output is then continued to be held high until the timer times-out.

Timer Operation



Timer

Instance Name: TIM-00

Index Enable: ☑ Yes

Timing Duration: 60 □ sec

Trig Handle: LOG-03-00-IFLoB0S

Trigger Handle Name: LOG-03

Integrate Enable: □ No

Trig On Any Trans: □ No

Trig Negative Enable: □ No

Delayed Start Enable: ☑ Yes

Delayed Stop Enable: □ No

Output Value: Off

Index Enable Status: Yes

Index Active: No

Time Remaining: 0

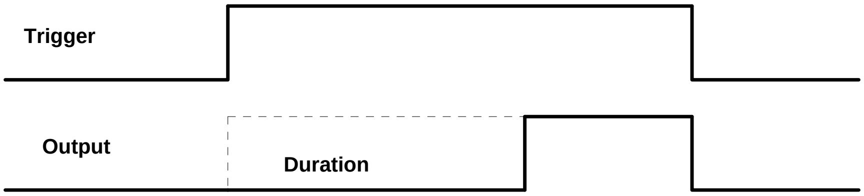
Trigger Value: No

The timer if enabled examines the value pointed at by the Trigger Handle to determine if a valid trigger exists. The Trigger Handle may point to a whole attribute, or an individual bit of any index of any object; Input, Logic, Output, Timer, etc. If the timer is disabled, Attr-0 has a value of 00h.

Delayed Start Timer

If **Delayed Start Enable** is yes the Delayed Start Timer is started by a positive trigger. The timer will begin counting with the occurrence of the trigger, but the output will not go high (FFh) until the timer counts to zero. It will then remain high as long as the input trigger is high. When the input trigger drops to zero, the output will immediately drop to zero. On power reset the delayed start timer begins counting if the input is true, but its output will not go high until the timer counts to zero.

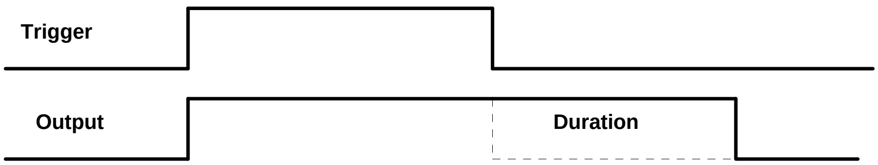
Delayed Start (Positive Trigger Only)



Delayed Stop Timer

If **Delayed Stop Enable** is yes, The Delayed Stop Timer is started by a negative trigger. When the input trigger goes high, the output will immediately go high, FFh. It will then remain high as long as the input trigger is high. The timer will begin counting with the occurrence of the negative trigger, but the output will not go low (00h) until the timer counts to zero.

Delayed Stop (Negative Trigger Only)



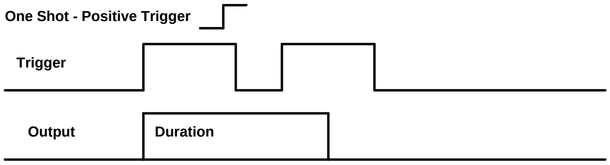
A timer can to be both Delayed Start and Delayed Stop. If that function is required then two timers in series, first a Delayed Start, followed by a delayed stop are required.

One Shot Timer

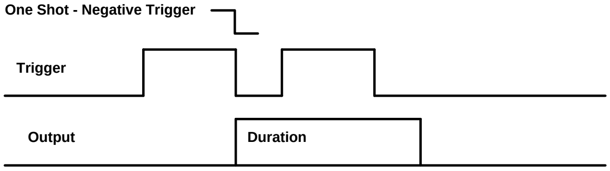
If both Delayed Start Enable and Delayed Stop Enable are No then it is a One Shot Timer. It can be started by a Positive Trigger, a transition from zero to non-zero value. Or if **Trigger Negative Enable** is Yes, it can be started by a Negative Trigger, a transition from non-zero to zero value. Or if **Trigger on Any Transition** is Yes, it can be started by both , . When a valid trigger is received, the Timer Duration is loaded into the Timer. The Timer then counts down in either seconds, or minutes. When the One shot timer is running, Attr-0 has a value of 255 (FFh). When the timer is not running or if the timer is disabled, Attr-0 has a value of 00h.

If a non-integrating One Shot timer is selected, **Integrate Enable** = No, Timer Duration is reloaded only if the timer is zero and a valid trigger is received.

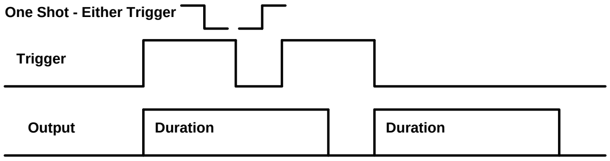
One Shot - Positive Trigger



One Shot - Negative Trigger

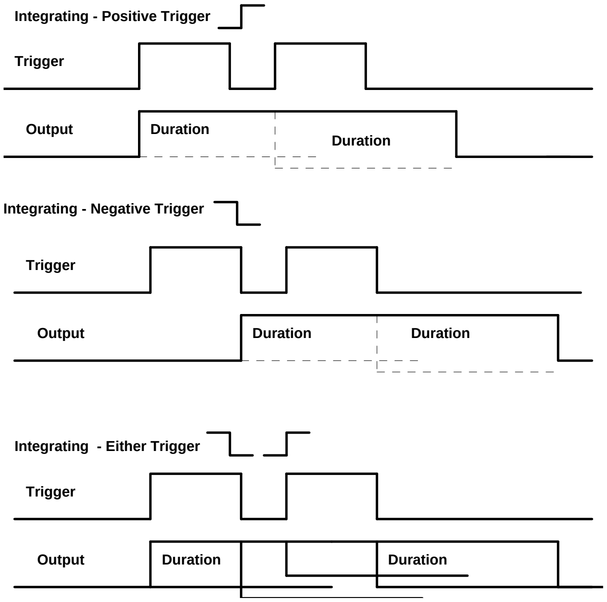


One Shot - Either Trigger



Integrating One Shot Timer

If **Integrate Enable** is yes, it is an Integrating One Shot timer and the Timer Duration is reloaded whenever a valid trigger is received.

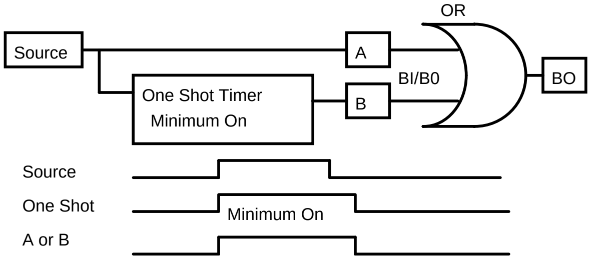


Integrating - Positive Trigger
Trigger
Output
Duration
Duration
Integrating - Negative Trigger
Trigger
Output
Duration
Duration
Integrating - Either Trigger
Trigger
Output
Duration
Duration

Timer Configuration

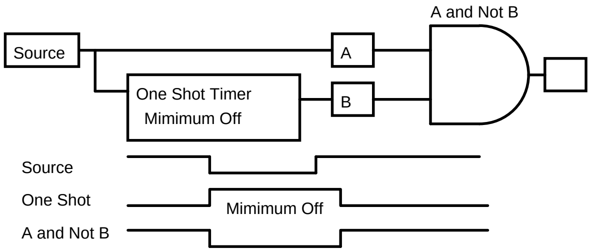
Example: Minimum ON Timer

Trigger One Shot Timer on. Output is held On while Timer is running.



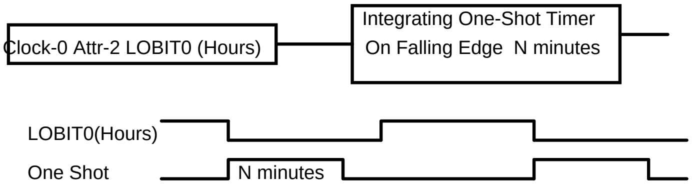
Example: Minimum Off Timer

Trigger One Shot Timer on Off. Output is held Off while Timer is running.



Example: Duty Cycle Timer Triggered on Clock

Trigger One Shot Timer on falling clock bit. Timer will start every 1, 2, 4, 8, 16 minutes, and run for N minutes.

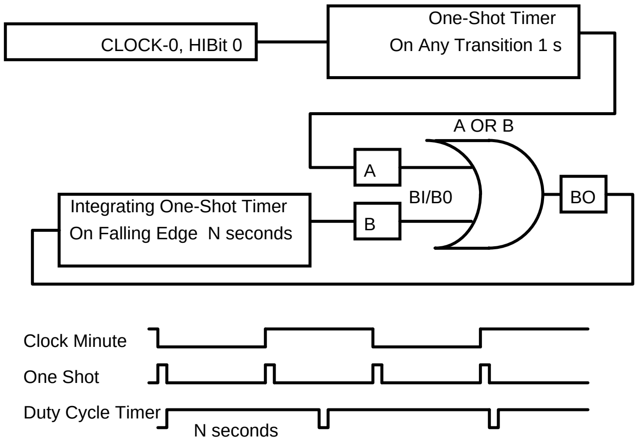

Clock-0 Attr-2 LOBIT0 (Hours)
Integrating One-Shot Timer
On Falling Edge N minutes
LOBIT0(Hours)
One Shot
N minutes

Example: Free Running Duty Cycle Timer

CLOCK minute bit is used to initiate the Integrating Timer by starting a falling trigger when the Timer is initially off after power reset. If the end of the Timer corresponds exactly with the minute transition then the start of the next cycle will be delayed by 1 second

To Set up free running timer with a period of N+1 seconds

LOG-n Free Running Timer A or B
TIM-i Startup Timer
TIM-i+1 Integrating One Shot



Example: Proportional Duty Cycle Timer

1) Setting up free running timer for variable transitions.
LOG-07 Free Running Timer
TIM-06 Startup Timer

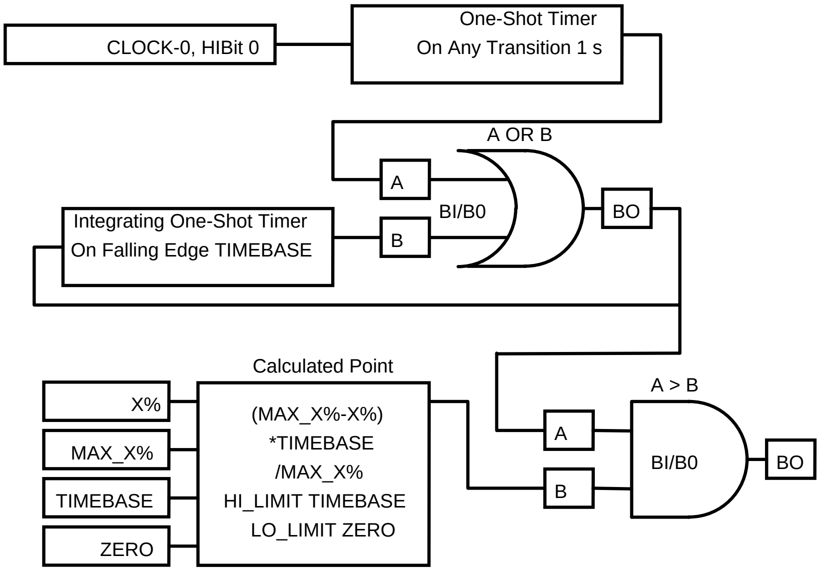
TIM-07 Repeating Timer for TIMEBASE

CLOCK minute bit is used to initiate the Integrating Timer by starting a falling trigger when the

Timer is initially off after power reset. If the end of the Timer corresponds exactly with the minute transition then the start of the next cycle will be delayed by 1 second

2) Setup Calculated point to calculate the transition from On to Off.
Off Time = (MAX_X%-X%)*TIMEBASE/MAX_X%

3) Setup Logic for Timer > Off Time.



Timer Glossary

Timer Parameters

Delayed Start Enable

If the index is configured to be of delayed-start type, the timer is started upon occurrence of a positive trigger, but Output Value is held "Low" until the timer times-out; at that point, the timer sets Output Value to "High" and holds it at this value until the trigger returns to zero. Delayed Start Enable takes precedence over Delayed Stop Enable. "Yes", "No"; user-configurable. (21,X,3,BIT 5)

Delayed Stop Enable

If index is configured to be of delayed-stop type, Output Value goes "High" upon occurrence of a positive trigger. The timer begins timing upon the trigger falling back to zero (a negative trigger) and Output Value does not fall to "Low" until the timer times-out. "Yes", "No"; user-configurable. (21,X,3,BIT 6)

Index Active

Indicates if the index is currently performing a timing sequence. "Yes", "No"; not user-changeable. (21,X,1,BIT 1)

Index En*a*ble Status

Indicates if the index has been enabled for operation. Set by configuring Index Enable. Not user-changeable; "Yes", "No". (21,X,1,BIT 0)

Index Enable

Enables the index for operation. "Yes", "No"; user-configurable. (21,X,3,BIT 0)

Integrate Enable

If set to "Yes", then the index is an Integrating type timer, and the timer is re-started (Time Remaining is reloaded with Timing Duration) if another trigger is received while timing is occurring. If set to "No", the index is a One-Shot type timer and performs a complete timing sequence but once per trigger. User-configurable; "Yes", "No". (21,X,3,BIT 1)

Negative Trigger Enable

Applies to Integrating and One-Shot type timers only. Usually a Positive type trigger is employed and timing sequences are initiated upon zero to non-zero transitions of Trigger Value. If Negative Trigger Enable is set to "Yes", a Negative type trigger is used instead and timing sequences are initiated on a non-zero to zero transition of Trigger Value. "Yes", "No"; user-configurable. (21,X,3,BIT 3)

Output Value

This value is set to "High" (= FFFFh) or "Low" (0000h) in accordance with the logical sequence associated with the type of timer this index is configured for (One-Shot, Integrating, Delayed-Start, or Delayed-Stop). "High", "Low"; not user-configurable. (21,X,0,BYTE)

Time Remaining

Indicates the amount of time remaining until completion of the current timing sequence. In seconds or in minutes depending on the value of Timing Units. Not user-changeable. (21,X,2,BYTE)

Timer Units

Timing may be performed in terms of minutes or in terms of seconds. "Minutes", "Seconds"; user-configurable. (21,X,3,BIT 2)

Timing Duration

The amount of time to be loaded into Time Remaining upon commencement of a timing sequence. In minutes or seconds depending on Timing Units. User-configurable. (21,X,4,BYTE)

Trigger Handle

Points to the location from which Trigger Value is to be obtained. Object, Index, Attribute; Select user-configurable. (21,X,7, 4 Bytes)

Trigger On Any Transition

Applies to Integrating and One-Shot type timers only

Indicates that any change in Trigger Value will cause the initiation of a timing sequence. User-configurable; "Yes", "No". (21,X,3,BIT 4)

Trigger Source Handle

Points to the parameter containing the value to use for Trigger Source. Object, Index, Attribute, Select. User-configurable. (21,X,5,4 BYTES)

Trigger Value

The most recently obtained value for the trigger. "High", "Low"; not user-changeable. (21,X,1,BIT 2)

Timer Properties

The TIMER Object is available to time or delay binary actions. Each index of the timer object may be one of four types of timer:

1) One Shot on Positive or Negative Trigger, or both;
2) Integrating One Shot on Positive or Negative Trigger, or both;
3) Delayed Start on Positive Trigger;
4) Delayed Stop on Negative Trigger.

It has a single input handle and single byte output value which is 00h when off and is FFh when on .

Object Name	TIMER
Object Number	= 21
Data Type	= BYTE
Index	= 16 or as allocated
Attribute	= 0..8
DYNAMIC Attribute	= 3 (0..2)
STATIC Attribute	= 6 (3..8)

Timer Firmware Revision

ASIC/2-7540 FW754A Rev 1.0 Forthcoming 2005

o As in FW740E2.0.

SINC/3-3000 FW300B Rev 1.0 Released 10/06/1999 CHK 0x8FB7

ASIC/2-7040 FW740B Rev 1.2 Released 08/18/95
ASIC/2-7040 FW740A Rev 2.1 Released 08/16/95

o Max Number of Timers increased from 128 to 256

ASIC/2-8040 FW840A Rev 1.0, Forthcoming March 1996

ASIC/2-7040 FW740A Rev 1.0 31 March 1994
SC/1-9040 1.0 FW907A, Released 8 March 1991

Timer DYNAMIC Properties

Attr-0 **Output Value**

00h when Off; FFh when On.

Attr-1 Dynamic Status

Attr-1 bit 0 - **Index Enable Status**
Attr-1 bit 1 - **Index Active**
Attr-1 bit 2 - **Trigger Value**
Attr-1 bit 3 -
Attr-1 bit 4 -
Attr-1 bit 5 -
Attr-1 bit 6 -
Attr-1 bit 7 -

Attr-2 **Time Remaining**

When triggered, the value of Timing Duration is loaded into the Timer. The value will count down every second or minute until it reaches zero.

Timer STATIC Properties

Attr-3 Setup

Attr-3 bit 0 - **Index Enable**
0 = Disable Timer for this index
1 = Enable Timer for this index

Attr-3 bit 1 - **Integrate Enable** (One Shot Only)
0 = Non-Integrating One Shot ,
Once timer has started it will not be restarted until it counts down to zero.
1 = Integrating One Shot
Timer is reloaded with Duration whenever a valid trigger is received.

Attr-3 bit 2 - **Timer Units**
0 = Seconds, Duration and Timer in Seconds
1 = Minutes, Duration and Timer in Minutes

Attr-3 bit 3 - **Trigger Negative Enable** (One Shot Only)
0 = Positive Trigger,
transition from zero to non-zero
1 = Negative Trigger,
transition from non-zero to zero

Attr-3 bit 4 - **Trigger on Any Transition** (One Shot Only)
0 = use bit 3;
1 = Either transition will trigger.

Attr-3 bit 5 - **Delayed Start Enable**
Has priority over Delayed Stop and One Shot Type Timer.

Attr-3 bit 6 - **Delayed Stop Enable**
Has priority over One Shot Type Timer

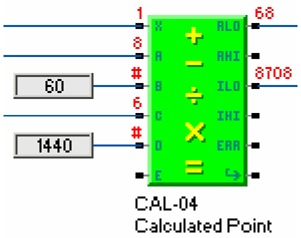
Attr-3 bit 7 - Spare

Attr-4 **Timing Duration**
Value to be loaded into the timer in minutes or seconds depending on Units (Attr-3,Bit 2).

Attr-5,6,7,8 **Trigger Handle**
The Trigger Handle is examined once per second to see if the conditions for a valid trigger exist.

Object 22 - Calculated Point

Calculated Point Summary



Overview: The calculate point object is used to perform arithmetic operations on data from various other objects.

Inputs: There are six full handles that reference any attribute of any index of any object in the controller: Handle X , Handle A, Handle B, Handle C, Handle D and Handle E.

Outputs: There are four output words that may be referenced by other objects.

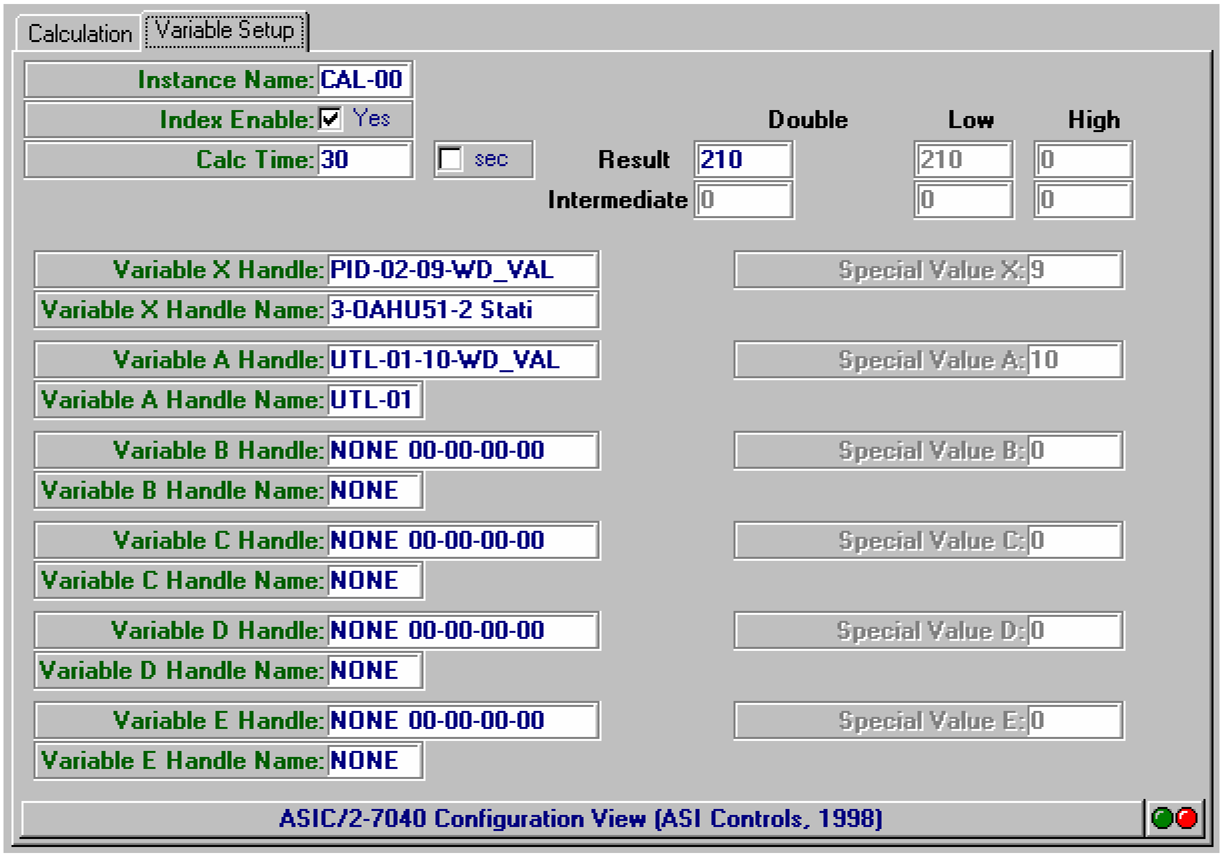
Operations: Calculations are done using full double precision 32 bit integer arithmetic. Up to four independent calculation steps can be configured for a single index of the calculated point. Each step has two sources, a function, and a destination. Functions that can be performed include: Add, Subtract, Negate, NOT, Multiply, Divide, Multiply Unsigned, Divide Unsigned, Swap High Word of Source to the Low Word, and Put.

Calculations are done using full double precision 32 bit integer arithmetic. Up to four independent calculation steps can be configured for a single index of the calculated point. Each step has two sources, a function, and a destination.

Note: That all calculations are done with signed integer arithmetic using numbers between -32766 and 32767. For example, a temperature of 74.52 F, is represented as the signed integer 7452.

Calculated Point Operation

Variable Setup



Instance Name: CAL-00

Index Enable: ☑ Yes

Calc Time: 30 ☐ sec

	Double	Low	High
Result	210	210	0
Intermediate	0	0	0

Variable X Handle:	PID-02-09-WD_VAL	Special Value X:	9
Variable X Handle Name:	3-OAHU51-2 Stati		
Variable A Handle:	UTL-01-10-WD_VAL	Special Value A:	10
Variable A Handle Name:	UTL-01		
Variable B Handle:	NONE 00-00-00-00	Special Value B:	0
Variable B Handle Name:	NONE		
Variable C Handle:	NONE 00-00-00-00	Special Value C:	0
Variable C Handle Name:	NONE		
Variable D Handle:	NONE 00-00-00-00	Special Value D:	0
Variable D Handle Name:	NONE		
Variable E Handle:	NONE 00-00-00-00	Special Value E:	0
Variable E Handle Name:	NONE		

ASIC/2-7040 Configuration View (ASI Controls, 1998)

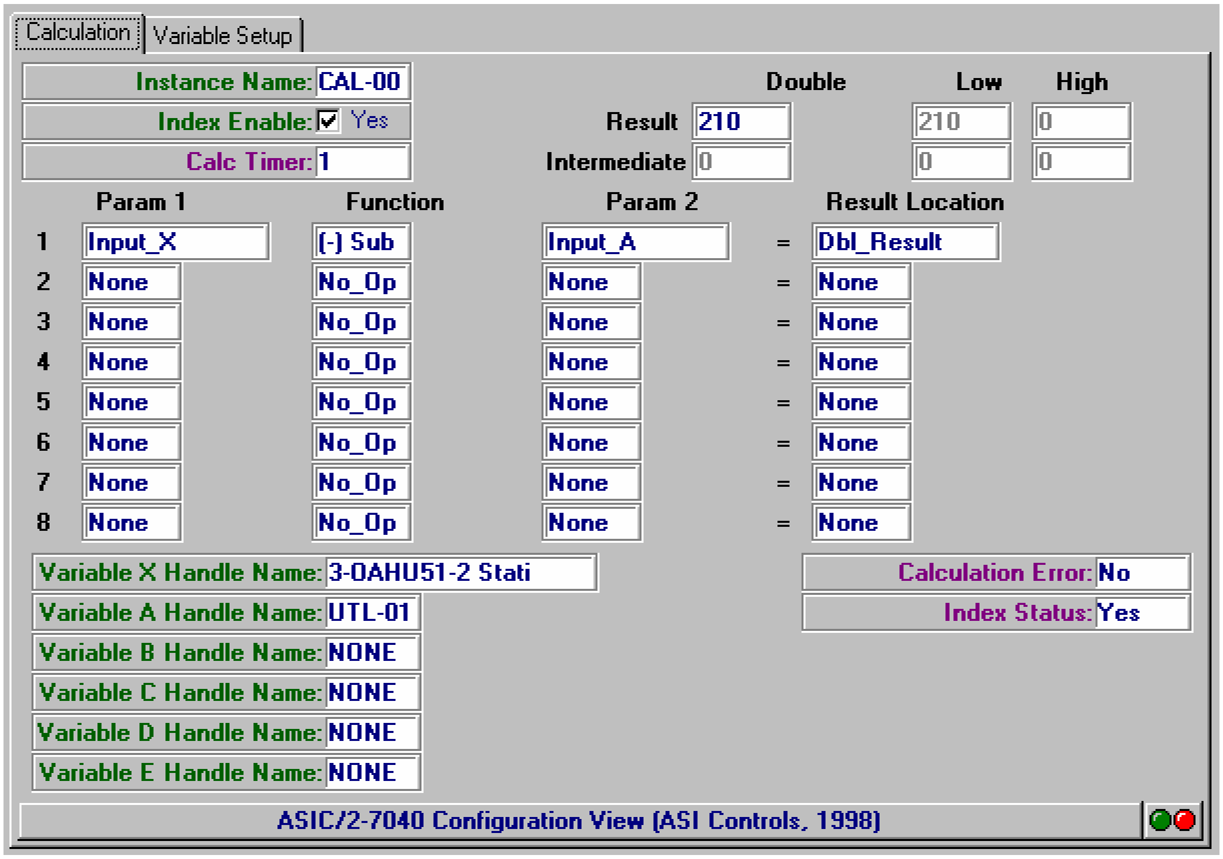
The Calculated point has handles for up to 6 variables, X, A, B, C, D, and E which can fetch data from any instance of any object in the controller.

The handle can also be identified as Special Data where the number to be used is kept in the last two bytes of the handle. Special Data is useful for storing constants such as 2, 10, 60, etc. which are referenced in the calculation. If the source Handle Object = 255 (Special), then the second word of the handle, Handle Attribute (LO Byte) and Handle Select (HI Byte), contains the data to be used.

Registers

The Calculated point has two 32-bit double word registers, Result and Intermediate, which can be used as the destination for multiplication, division, addition, or subtraction. And can be used as the source for division. The high and low words of these registers, LO Result, HI Result, LO Intermediate, and HI Intermediate, that can also be referenced as the source or destination for a calculation step.

Calculation



Instance Name: CAL-00

Index Enable: ☑ Yes

Calc Timer: 1

	Double	Low	High
Result	210	210	0
Intermediate	0	0	0

	Param 1	Function	Param 2		Result Location
1	Input_X	(-) Sub	Input_A	=	Dbl_Result
2	None	No_Op	None	=	None
3	None	No_Op	None	=	None
4	None	No_Op	None	=	None
5	None	No_Op	None	=	None
6	None	No_Op	None	=	None
7	None	No_Op	None	=	None
8	None	No_Op	None	=	None

Variable X Handle Name: 3-0AHU51-2 Stati

Variable A Handle Name: UTL-01

Variable B Handle Name: NONE

Variable C Handle Name: NONE

Variable D Handle Name: NONE

Variable E Handle Name: NONE

Calculation Error: No

Index Status: Yes

ASIC/2-7040 Configuration View (ASI Controls, 1998)

The calculation proceeds one step at a time. The calculated point loads the value designated by Parameter 1 into an internal 32 bit register. If a single precision (16 bit) source is indicated, the high word is filled with 0000h (or, FFFFh if signed <0).

The value designated by Parameter 2 is then loaded into a second internal 32 bit register. The function is then performed, and stored in one of several internal register.

The internal registers loaded from Parameter 1 and Parameter 2 are then combined based on the Step Function which is then placed in a 32 bit answer.

If the destination is double precision then the entire 32 bit answer is transferred to the destination. If the destination is single precision only the LO word of the answer is transferred to the destination. The exception to this is the Swap function, where the high word is transferred to the Lo Word of the destination, with the high word being padded with 0000h (or, FFFFh if signed <0).

Step N Functions To Perform

Note: If a calculation step is not to be used, then the Step Function, Destination, Parameter 1 Source and Parameter 2 Source must all be zero.

0 = No-op No Operation.

1 = Add (Double Precision) Adds Parameter 2 to Parameter 1.

2 = Subtract (Double Precision) Subtracts Parameter 2 from Parameter 1.

3 = Negate (Double Precision) Negates Parameter 1

4 = NOT (Double Precision) Takes the 1's Complement of Parameter 1.

5 = Multiply (Single Precision to Double Precision)
Multiplies Low Word of Parameter 1
by Lo word of Parameter 2.

6 = Divide (Double Precision to Single Precision)
Divides Double Precision Parameter 1 by
Lo word of Parameter 2.
LO WORD contains Result,
HI WORD contains Remainder

7 = Multiply Unsigned (Single Precision to Double Precision)
Multiplies Low Word of Parameter 1
by Lo word of Parameter 2.

8 = Divide Unsigned (Double Precision to Single Precision)
Divides Double Precision Parameter 1
by Lo word of Parameter 2.
LO WORD contains Result,
HI WORD contains Remainder

9 = Swap (Single or Double Precision)
Moves the High Word of Parameter 1
to the Low Word of Destination
High word is padded with 0000h
(or, FFFFh if signed <0).

10 = Put (Single Precision)
Moves the Low Word of Parameter 1
to the Destination

11 =Hi Limit (Single Precision)
Limits the signed value of Source1 to be no greater than Parameter 2
to the Destination.

12 =*Lo Limit* (Single Precision)
Limits the signed value of Source1 by to be no less than Parameter 2
to the Destination.

13 = Spare

14 = Spare

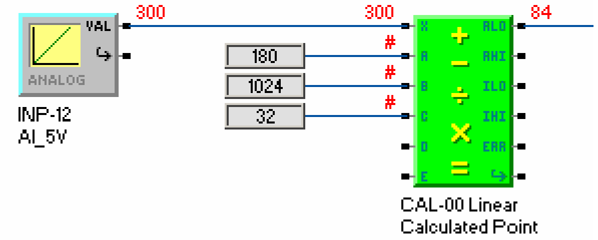
15 = Min/Max/Average (Single Precision)(FW740A..)
This takes the sum of each enabled handle and places it in Double Word Result. The lowest is placed in low intermediate. The highest in high intermediate. When all have been examined, the Double Word Result is divided by the number of values with the Lo Word Result having the Average, and Hi Word Result having the remainder.

Parameter Names for Source and Destination

1 = Result	(Double Precision)	
2 = Intermediate	(Double Precision)	
3 = Intermediate LO	(Single Precision)	
4 = Intermediate HI	(Single Precision)	
5 = Value of Handle X	(Single Precision)	Not a valid destination
6 = Value of Handle A	(Single Precision)	Not a valid destination
7 = Value of Handle B	(Single Precision)	Not a valid destination
8 = Value of Handle C	(Single Precision)	Not a valid destination
9 = Result LO	(Single Precision)	
10 = Result HI	(Single Precision)	
11 = Value of Handle D	(Single Precision)(FW740A..)	Not a valid destination
12 = Value of Handle E	(Single Precision)(FW740A..)	Not a valid destination
13 = Spare		
14 = Spare		
15 = Spare		

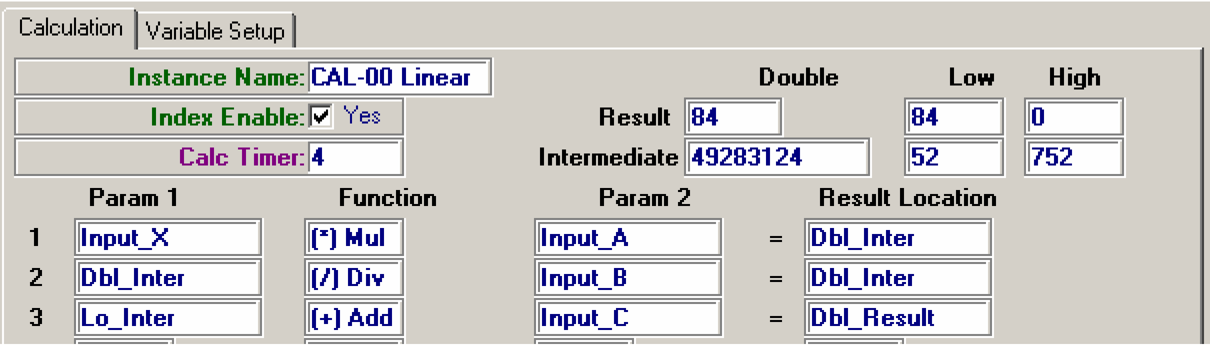
Calculated Point Configuration -

Example: Linear Conversion.



Linear conversion is often used to translate one value into another. For example, a transmitter may generate a 0 to 5 Vdc signal (0..1024) which corresponds to 32 to 212 ^{o}F. The general form of the Linear Conversion is Result = A*X/B + C

Note: When doing Multiplication and division the result **must** be less than 65,535 for unsigned arithmetic, and less than 32767 for signed arithmetic.



Instance Name: CAL-00 Linear

Index Enable: ☑ Yes

Calc Timer: 4

	Double	Low	High
Result	84	84	0
Intermediate	49283124	52	752

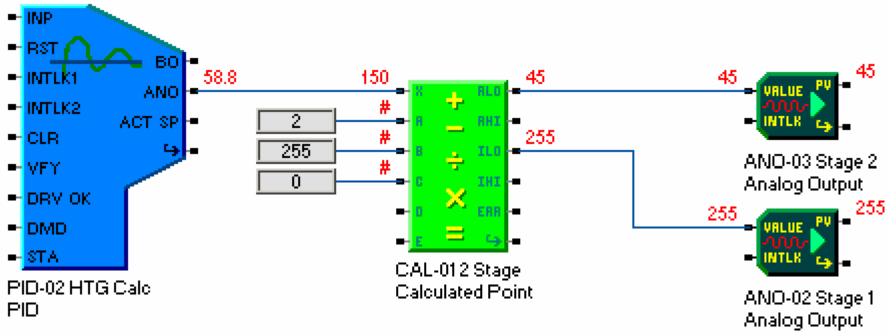
	Param 1	Function	Param 2		Result Location
1	Input_X	(*) Mul	Input_A	=	Dbl_Inter
2	Dbl_Inter	(/) Div	Input_B	=	Dbl_Inter
3	Lo_Inter	(+) Add	Input_C	=	Dbl_Result

In Step 1 the Input_X is multiplied by Input_A and the result is placed in Intermediate Double Result.

In Step 2, the Intermediate Double Result is divided by Input_B to give a 16-bit result (Low Intermediate) and a 16 bit remainder (High Intermediate)

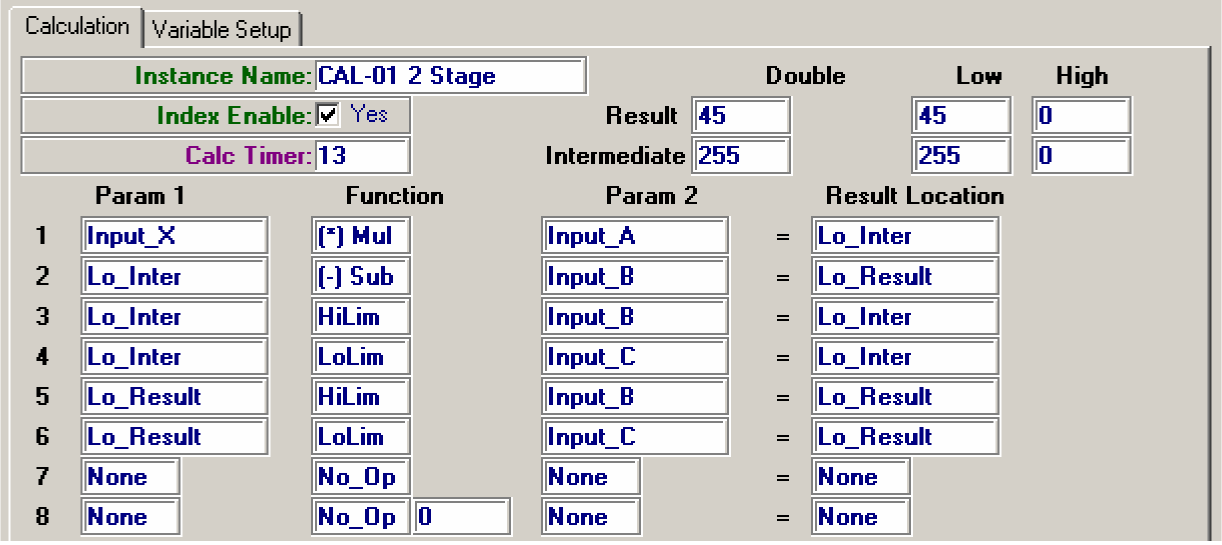
In Step 3, the Low Intermediate is added to the Input_C to give the final result placed in Double Result. The Low Result word has the value of interest.

Example: Sequencing two devices.



The

The Calculated point can take a value 0..255 and break it into two stages. 0..127 for stage1 and 128 to 255 for stage 2.



Instance Name: CAL-01 2 Stage

Index Enable: ☑ Yes

Calc Timer: 13

	Double	Low	High
Result	45	45	0
Intermediate	255	255	0

	Param 1	Function	Param 2		Result Location
1	Input_X	(*) Mul	Input_A	=	Lo_Inter
2	Lo_Inter	(-) Sub	Input_B	=	Lo_Result
3	Lo_Inter	HiLim	Input_B	=	Lo_Inter
4	Lo_Inter	LoLim	Input_C	=	Lo_Inter
5	Lo_Result	HiLim	Input_B	=	Lo_Result
6	Lo_Result	LoLim	Input_C	=	Lo_Result
7	None	No_Op	None	=	None
8	None	No_Op 0	None	=	None

In Step 1, the PID Output (0.255) is multiplied by Input_A (2). Since the result is never more than 510 it can be placed in the Low Intermediate word.

In Step 2, Input_B (255) is subtracted from the Low Intermediate word and placed in the Low Result Word.

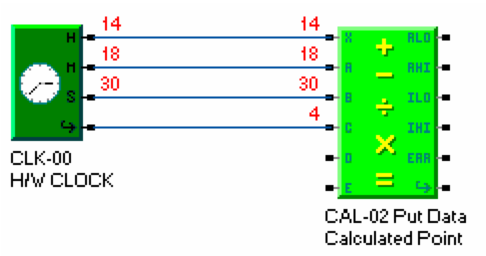
In Steps 3 and 4, the Lo_Inter (Stage 1) value is limited between 0..255.

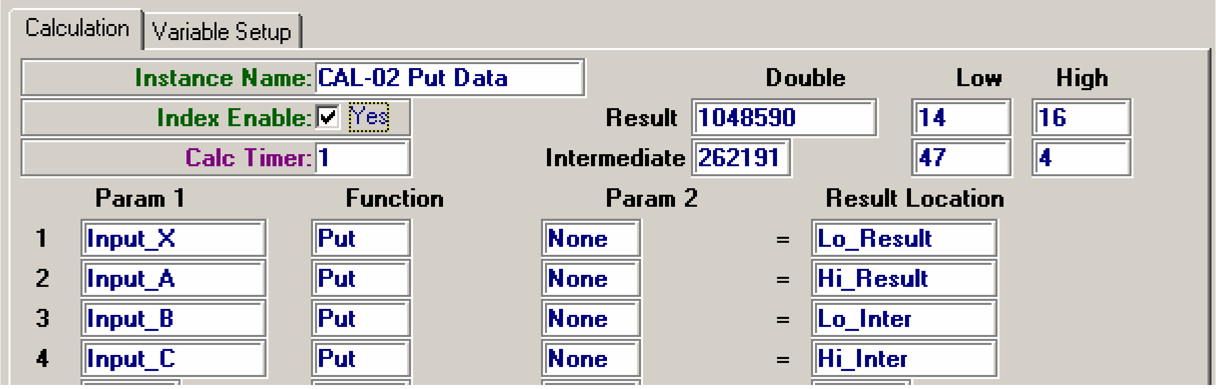
In Steps 5 and 6, the Lo_Result (Stage 2) valueis limited between 0..255.

> Note: In Steps 3 & 4 the value Lo_Inter is modified and written back into itself. In Steps 5 & 6 the value Lo_Result is modified and written back into itself. The steps are executed in sequence so that temporary results can be held in the Intermediate or Result values and then modified. The last step to change the value gives the final answer.

Example: Put Data.

The calculated point can be used to group points from various locations in a single object. Note that the result of this data is Read Only. The data can be changed only by changing the data in the originating objects. This is useful when building up a calculation step by step to verify that you have gotten the correct data.





Instance Name: CAL-02 Put Data

Index Enable: ☑ Yes

Calc Timer: 1

	Double	Low	High
Result	1048590	14	16
Intermediate	262191	47	4

	Param 1	Function	Param 2		Result Location
1	Input_X	Put	None	=	Lo_Result
2	Input_A	Put	None	=	Hi_Result
3	Input_B	Put	None	=	Lo_Inter
4	Input_C	Put	None	=	Hi_Inter

Calculated Point Glossary

Calculated Point Parameters

Calculation Error

Indicates that at some point during the calculation a value was produced which could not be stored in the space provided by Intermediate Storage. "Error", "OK"; not user-changeable. (22,X,2,LOW BIT 1)

Calculation Interval

Designates whether the calculation is to be performed every minute or every second. "Minute", "Second"; user-configurable. (22,X,5,LO BIT 1)

Calculation Time

Indicates how often the calculation takes place. In seconds or minutes depending on Calculate Interval. User changeable. (FW740A..) (22,X,5,HI Byte)

Calculation Timer

Internal counter to keep track of how often the calculation takes place. In seconds or minutes depending on Calculate Interval. Not User changeable. (FW740A..) (22,X,2,HI Byte)

Index Enable

Enables the index for operation. "Yes", "No"; user-configurable. (22,X,5,LO BIT 0)

Index Status

Indicates if the index has been enabled for operation. "Enabled", "Disabled"; not user-changeable. (22,X,2,LO BIT 0)

Intermediate Storage

Used internally to store intermediate results in multi-step calculations. Not user-changeable. (22,X,3,2 WORDS)

Result High Word

The high word of the result of the index's most recent calculation. Not user-changeable. (22,X,1,WORD)

Result Low Word

The low word of the result of the index's most recent calculation. Not user-changeable. (22,X,0,WORD)

Special Data

Special Data is useful for storing constants such as 2, 10, 60, etc. which are referenced in the calculation. If the source Handle Object = 255 (Special), then the second word of the handle, Handle Attribute (LO Byte) and Handle Select (HI Byte), contains the data to be used.

Step N - Function

Designates the type of calculation which is to be performed for step N. See the Calculation Point Object Definition for details. User-configurable; integer code. (22,X,6 -> 9,LOW BYTE LOW NIBBLE)

Step N - Parameter 1 Source

Designates what is to be used for Parameter 1 in step N of the calculation. N is from 1 to 4. The possibilities are

Variables A,B,C,X or Intermediates 1,2,3 or Result). See the Calculation Point Object Definition for details. User-configurable. (22,X,6 -> 9,HIGH BYTE LOW NIBBLE)

Step N - Parameter 2 Source

Designates what is to be used for Parameter 1 in step N of the calculation. N is from 1 to 4. The possibilities are

Variables A,B,C,X or Intermediates 1,2,3 or Result. See the Calculation Point Object Definition for details. User-configurable. (22,X,6 -> 9,HIGH BYTE LOW NIBBLE)

Step N - Result Destination

Designates where the result of step N in the calculation is to be placed. N is from 1 to 4. The possibilities are

Variables A,B,C,X or Intermediates 1,2,3 or Result. See the Calculation Point Object Definition for details. (22,X,6 -> 9,LOW BYTE HIGH NIBBLE)

Variable A Handle

Points to the parameter from which Variable A is to be obtained. User-configurable. (20,X,12,4 BYTES)

Variable B Handle

Points to the parameter from which Variable B is to be obtained. User-configurable. (20,X,14,4 BYTES)

Variable C Handle

Points to the parameter from which Variable C is to be obtained. User-configurable. (20,X,16,4 BYTES)

Variable D Handle

Points to the parameter from which Variable C is to be obtained. User-configurable. (20,X,18,4 BYTES)

Variable E Handle

Points to the parameter from which Variable C is to be obtained. User-configurable. (20,X,20,4 BYTES)

Variable X Handle

Points to the parameter from which Variable X is to be obtained. User-configurable. (20,X,10,4 BYTES)

Calculated Point Properties

The Calculated Point Object has Data Type WORD; It is used to perform arithmetic to calculate points which combine data from different objects. Calculate uses Double Word precision arithmetic.

Object Name	CALCULATED POINT	
Object Number	22	
Data Type	WORD	
Index	0..n as allocated	
Attributes	18 (0..17)	
DYNAMIC Attributes	5 (0..4)	
STATIC Attributes	13 (5..17)	
	21 (5..25)	FW740A.., FW840A..

Calculated Point Firmware Revision

ASIC/2-7540 FW754A Rev 1.0 Forthcoming 2005

o As in FW740E.

ASIC/2-8040 FW840A Rev 1.0, March 1996

ASIC/2-7040 FW740A/B Rev 1.8 Released 12 April 95

o Fixed Calculated Point Min/Max Ave to correctly calculate average.

o Added Ave Number, Attr-2 LO_MS_NBL, to count number of handles that are active in doing the average.

ASIC/2-7040 FW740A Rev 1.5 15 Sept 94

o Fixed Calculated Point Min/Max to ignore Object 0 handles.

ASIC/2-7040 FW740A Rev 1.0 Alpha Test (31 March 1994)

o Added Handle D and Handle E.

o Added Calculation Timer

o Added Calculation Steps 5,6,7, & 8

SC/1-9040 FW907A Rev 1.0 Alpha Test (25 July 1991)

Calculated Point DYNAMIC Properties

Attr-0 **Calculation Result - LO WORD**

Attr-1 **Calculation Result - HI WORD**

Attr-2 Calculated Status
LO BYTE
LO bit 0 - Status (?)
1 = Yes; 0 = No
LO bit 1 - **Calculation Error**
Over/Underflow Error, 1 = Yes; 0 = No
LO bit 2 -
LO bit 3 -
LO bit 4 -
LO bit 5 -
LO bit 6 -
LO bit 7 -
HI BYTE - **Calculation Timer** (FW740A..)

Attr-3 **Intermediate Data - LO WORD**

Attr-4 **Intermediate Data - HI WORD**

Calculated Point STATIC Properties

Attr-5 Calculated Setup
- LO BYTE
  - LO bit 0 - **Index Enable**
  - LO bit 1 - **Calculate Interval**
    - 1, in minutes
    - 0, in seconds
  - LO bit 2 -
  - LO bit 3 -
  - LO bit 4 -
  - LO bit 5 -
  - LO bit 6 -
  - LO bit 7 -
- HI BYTE - **Calculation Time** (FW740A..)

Attr-6 **Step 1**
- LOLSNBL Step1 Function (0..15)
- LOMSNBL Step 1 Result Destination (0..15)
- HILSNBL Step 1 Parameter 1 Source (0..15)
- HIMSNBL Step 1 Parameter 2 Source (0..15)

Attr-7 **Step 2**
- LOLSNBL Step 2 Function (0..15)
- LOMSNBL Step 2 Result Destination (0..15)
- HILSNBL Step 2 Parameter 1 Source (0..15)
- HIMSNBL Step 2 Parameter 2 Source (0..15)

Attr-8 **Step 3**
- LOLSNBL Step 3 Function (0..15)
- LOMSNBL Step 3 Result Destination (0..15)
- HILSNBL Step 3 Parameter 1 Source (0..15)
- HIMSNBL Step 3 Parameter 2 Source (0..15)

Attr-9 **Step 4**
- LOLSNBL Step 4 Function (0..15)
- LOMSNBL Step 4 Result Destination (0..15)
- HILSNBL Step 4 Parameter 1 Source (0..15)
- HIMSNBL Step 4 Parameter 2 Source (0..15)

Attr-10,11 **Variable X Handle**

Attr-12,13 **Variable A Handle**

Attr-14,15 **Variable B Handle**

Attr-16,17 **Variable C Handle**

Attr-18,19 **Variable D Handle**
- Attr-18, LO - Variable D Object
- Attr-18, HI - Variable D Index
- Attr-19, LO - Variable D Attribute
- Attr-19, HI - Variable D Select

Attr-20,21 **Variable E Handle**
- Attr-20, LO - Variable E Object
- Attr-20, HI - Variable E Index
- Attr-21, LO - Variable E Attribute
- Attr-21, HI - Variable E Select

Attr-22 **Step 5** (FW740A..)
- LOLSNBL Step5 Function (0..15)
- LOMSNBL Step 5 Result Destination (0..15)
- HILSNBL Step 5 Parameter 1 Source (0..15)
- HIMSNBL Step 5 Parameter 2 Source (0..15)

Attr-23 **Step 6** (FW740A..)

LOLSNBL Step 6 Function (0..15)
LOMSNBL Step 6 Result Destination (0..15)
HILSNBL Step 6 Parameter 1 Source (0..15)
HIMSNBL Step 6 Parameter 2 Source (0..15)

Attr-24 **Step7** (FW740A..)

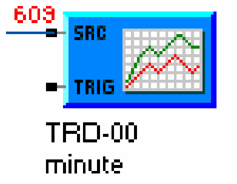
LOLSNBL Step1 Function (0..15)
LOMSNBL Step 1 Result Destination (0..15)
HILSNBL Step 1 Parameter 1 Source (0..15)
HIMSNBL Step 1 Parameter 2 Source (0..15)

Attr-25 **Step 8** (FW740A..)

LOLSNBL Step1 Function (0..15)
LOMSNBL Step 1 Result Destination (0..15)
HILSNBL Step 1 Parameter 1 Source (0..15)
HIMSNBL Step 1 Parameter 2 Source (0..15)

Object 23 - Trend

Trend Summary



Overview: The Trend object provides a method of collecting data in the controller and storing it in volatile RAM memory.

Input: A full handle points to a Word or Byte in any object in the controller.

Output: The output of trend is data stored in dynamic memory in the controller. The trend object can be configured to record up to 256 Bytes of data or 128 Words of data.

Operation: The trend object will perform a single trend starting at a given target time, and a continuous trend that updates the Trend Date Stamp at the end of each loop. A trend may be stopped temporarily and restarted. The trend frequency may be seconds, minutes, half-hour, or hour intervals. It may also trend once at the same time each day. Trend data is stored in volatile memory which is lost with loss of power. Important data should be uploaded for storage at frequent intervals.

Note: The contents of volatile RAM memory are lost when power is lost. See Object 27, Static Trend for application where data must be preserved if power is lost.

Trend Operation

The Trend Source Handle points at the data to be trended. Up to 128 words of data may be saved for each index. If byte values are trended, then up to 256 entries in subsequent bytes, Lo byte first, are saved.

The Trend Interval timer can be in seconds, minutes, half-hours, hours, or each day. Trend saves data before other modules are run so that accumulated data from previous day gets saved. Trend can be single, or continuous. If each day, then the value will be saved at the time given by the Trend Time Stamp, or at midnight if the time stamp is zero..

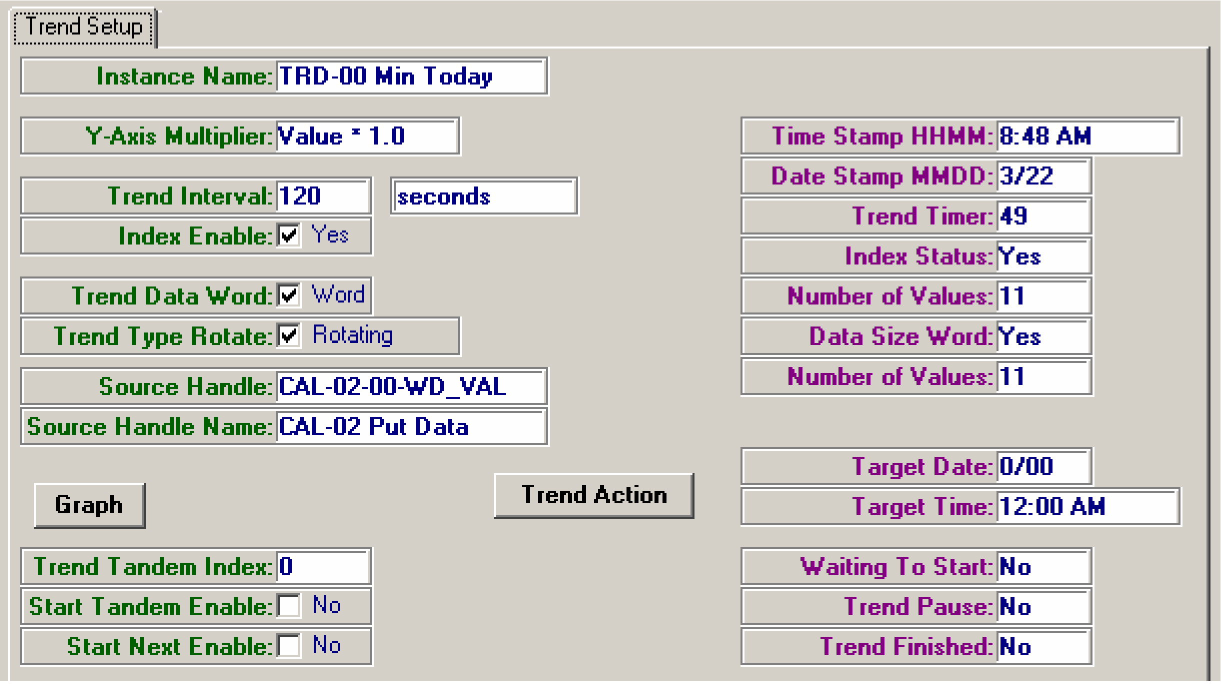
Single Trend

For single trending, trending does not begin until the first time period that matches the date stamp.

To Initiate Single Trend use the action, Attr-4 = 1, Initiate Single Trend.

If Attr-5,6 are written with Target Date and Time, trend will begin at that time.

If Target Date and Time are zero, Trend will cause the current date and time to be written to the Trend Time Stamp at the first trend interval. It will stop trending after one loop and keep the Trend Time Stamp.



Instance Name: TRD-00 Min Today

Y-Axis Multiplier: Value * 1.0

Trend Interval: 120 seconds

Index Enable: ☑ Yes

Trend Data Word: ☑ Word

Trend Type Rotate: ☑ Rotating

Source Handle: CAL-02-00-WD_VAL

Source Handle Name: CAL-02 Put Data

Trend Tandem Index: 0

Start Tandem Enable: □ No

Start Next Enable: □ No

Time Stamp HHMM: 8:48 AM

Date Stamp MMDD: 3/22

Trend Timer: 49

Index Status: Yes

Number of Values: 11

Data Size Word: Yes

Number of Values: 11

Target Date: 0/00

Target Time: 12:00 AM

Waiting To Start: No

Trend Pause: No

Trend Finished: No

Continuous Trend

Continuous trending sets the date stamp when trend pointer resets to 0 and on power-up. As the trend buffer becomes full with 128 words, or 256 bytes, the trend pointer resets to 0 and a new date stamp is written. All data will be read up and wrapped at the trend pointer for continuous trending. If power is lost all trend data is lost, therefore host a should save trend data frequently.

To Initiate Continuous Trend use the action, Attr-4 = 2, Initiate Continuous Trend.

If Attr-5,6 are written with aTarget Date and Time, trend will begin at that time.

If Target Date and Time are zero, Trend will cause the current date and time to be written to the Trend Time Stamp at the first trend interval. It will renew the Target Date and Time Stamp after each loop

Stop Trend (Pause)

A trend in progress may be temporarily stopped using the action, Attr-4 = 3, Stop Trend. If the trend is paused and restarted, then the Trend Time Stamp will not be accurate. No trace of the pause is left in the trend record.

A Stopped trend may be restarted using the action, Attr-4 = 4, Restart Stop Trend. It will resume trending at the next appropriate time.

Tandem Trend Feature

With FW740A Rev 1.2 a tandem trend feature has been added. A new parameter Trend Tandem Index, Attr-141 LO Byte, has been added which identifies another trend index, typically with a lower numerical value.

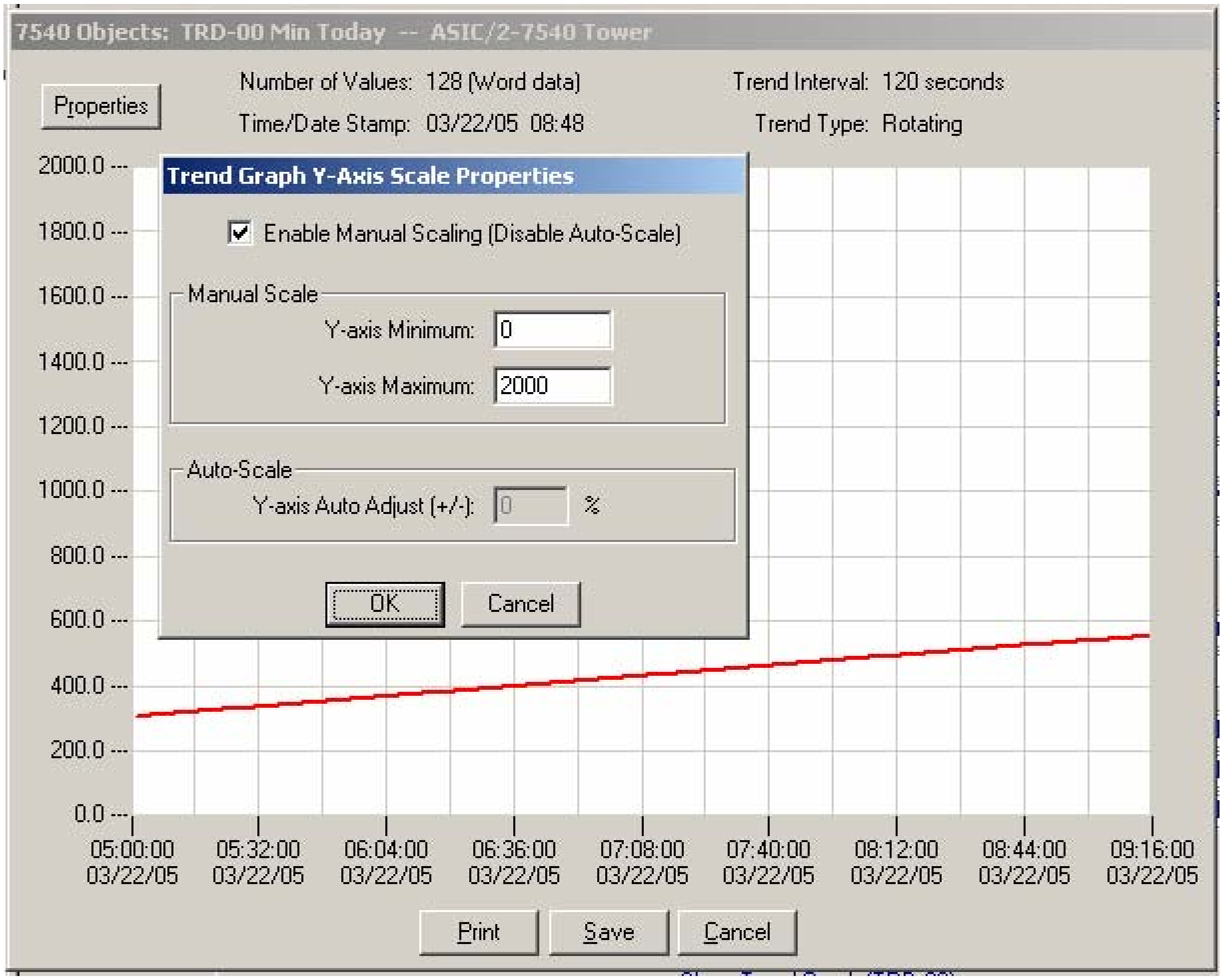
If the Start Tandem Enable flag , Attr-141 HI Bit 0 is set, then this trend will start only when the trend identified by the Tandem Trend Index starts. This allows starting multiple trends at the same time in tandem. When the trend starts the date stamp will be initialized. If Start Tandem Enable is set then Start Next Enable is ignored.

If the Start Next Enable flag , Attr-141 HI Bit 1 is set, then this trend will start only when the trend identified by the Tandem Trend Index finishes. This allows stacking trends to have a longer trend interval than can be achieved by a single trend index. When the trend starts the date stamp will be initialized.

Trend Graph

Expert has the ability upload and display the current trend data by selecting the Graph button. Normally the graph is Auto-Scaled between Minimum and Maximum values, however by selecting Properties check Enable Manual Scaling and set the Y-axis Minimum and Maximum Values.

Note: Manual Scaling is an Expert setting and applies to all graphs of instances of Dynamic Trend and Static Trend .



7540 Objects: TRD-00 Min Today -- ASIC/2-7540 Tower

Number of Values: 128 (Word data)

Trend Interval: 120 seconds

Time/Date Stamp: 03/22/05 08:48

Trend Type: Rotating

Trend Graph Y-Axis Scale Properties

☑ Enable Manual Scaling (Disable Auto-Scale)

Manual Scale

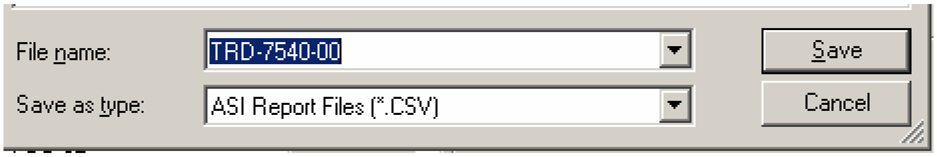
Y-axis Minimum: 0

Y-axis Maximum: 2000

Auto-Scale

Y-axis Auto Adjust (+/-): 0 %

Print sends the graph to the Default Printer. Save saves the data values to an ASI Report file (*.CSV) file in the Expert Project folder, with a default file name TRD-ADDR-IDX, where ADDR is the controller address, and IDX is the instance number. If the file already exists, it will append the data to the file.

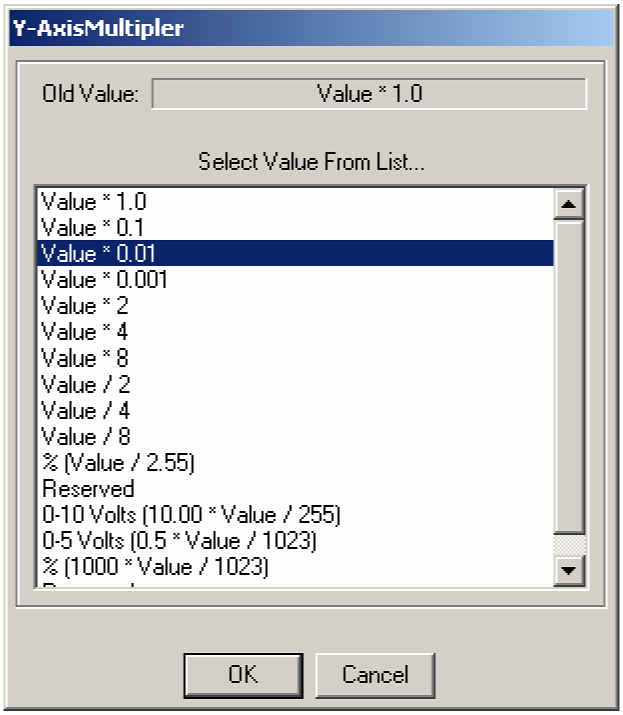


File name: TRD-7540-00

Save as type: ASI Report Files (*.CSV)

Y-axis Multiplier

Trend data is stored as a raw word or byte value. It may be necessary to convert the value for display purposes, for example multiply Value*0.01 to convert to degrees. The Y-axis Multiplier is stored in the controller for each instance of trend and is used to convert the Trend graph and saved data for this trend.



Old Value: Value * 1.0

Value * 1.0
Value * 0.1
Value * 0.01
Value * 0.001
Value * 2
Value * 4
Value * 8
Value / 2
Value / 4
Value / 8
% (Value / 2.55)
Reserved
0-10 Volts (10.00 * Value / 255)
0-5 Volts (0.5 * Value / 1023)
% (1000 * Value / 1023)

Trend Glossary

Trend Parameters

Data Size Status

Indicates if the data to be taken by the trend is to be of byte size or of word size. User-configurable; "1 = Byte","0 = Word". (23,X,2,LOW BIT 2)

Index Enable

Used to enable or disable this index of the Trend object for operation. "Enable","Disable"; user-configurable. (23,X,137,LOW BIT 0)

Index Enable Status

Indicates if the Trend object has been enabled for operation. Mirrors the value chosen in Index Enable. "Enabled","Disabled"; not user-changeable. (23,X,2,LO BIT 0)

New Trend Finish

Used to identify that the new trend started based on the Trend Tandem Index, has run to completion. (23,X,7,LO BIT 1)(FW740A Rev 1.2)

New Trend Start

Used to identify that the new trend has started based on the Trend Tandem Index. (23,X,7,LO BIT 0)(FW740A Rev 1.2)

Number of Data Taken

Indicates the index (in terms of bytes) into the trend storage block where the last piece of trend data taken was placed. Integer 0 to 256; user-changeable (not advised). (23,X,2,HIGH BYTE)

Start Next Enable

If the Start Next Enable flag is set, then this trend will start only when the trend identified by the Tandem Trend Index finishes. This allows stacking trends to have a longer trend interval than can be achieved by a single trend index. When the trend starts the date stamp will be initialized. (23,X,141,HI BIT 1)(FW740A Rev 1.2)

Start Tandem Enable

If the Start Tandem Enable flag is set, then this trend will start only when the trend identified by the Tandem Trend Index starts. This allows starting multiple trends at the same time in tandem. When the trend starts the date stamp will be initialized. If Start Tandem Enable is set then Start Next Enable is ignored. (23,X,141,HI BIT 0)(FW740A Rev 1.2)

Trend Action

Writing the listed value to this parameter (23,X,4,WORD) causes the corresponding action to occur:

1 Initiate Single Trend: Initiates the reading and saving of data each Trend Interval until the trend storage block is filled, at which point the trend is ended. The data is obtained from the parameter pointed at by Trend Source Handle.

2 Initiate Continuous Trend: Initiates the reading and saving of data each Trend Interval from the parameter pointed at by Trend Source Handle. Once the trend storage block is filled, trending does not stop; each new piece of data taken is simply overwritten over the oldest existing piece of data.

3 Stop Trend: Stops the current trend.

4 Restart Trend: Restarts a stopped trend. If no trend was stopped with Ac**tion** 3, this action has no effect.

Trend Data Byte X

Byte-size pieces of trended data X = 1 to 256 are stored in these locations. If Trend Data Size is equal to "Word", then 2 of these bytes are needed to store each data piece. User-changeable (but not advised); integer 1 -> 256. (23,X,9 -> 136,LOW BYTE,HIGH BYTE alternating)

Trend Data Size

Designates the size for trend data taken: either byte-size or word-size. User-configurable; "1 = Byte","0 = Word". (23,X,137,LOW BIT 1)

Trend Date Stamp

For single trends: This value is set equal to the date at which the first piece of data was taken for the currently active or most recent trend. For continuous trends: This value is set equal to the date at which Trend Pointer was last equal to zero. LO Byte Month, Integer 1..12. HI Byte Day of Month, Integer 1..31; . (23,X,0,WORD)

Trend Finished

Applies to single-type trends only: indicates that the data storage area has been filled (256 bytes of data have been taken). "Yes","No"; not user-changeable. (23,X,2,LOW BIT 4)

Trend Interval Length

The number of Trend Interval Units between the taking of successive pieces of trended data. Integer; user-configurable. (23,X,138,WORD)

Trend Interval Units

Designates the units to use for Trend Interval Length. "0 = Seconds","1 = Minutes","2 = HalfHours","3 = Hours","4 = Each Day"; user-configurable. (23,X,137,HIGH BYTE)

Trend Source Handle

Points to the parameter from which the trended data is to be obtained. User-configurable. (23,X,139,4 BYTES)

Trend Tandem Index

Identifies another trend index, typically with a lower numerical value, that is used to initate trending for this index based on the settings of the Start Trend Enable or Start Next Enable flags.(23,X,141,LO BYTE) (FW740A Rev 1.2)

Trend Target Date Stamp

The date at which a trend initiated via the Action parameter is to begin. If Trend Target Time Date is less than the current date, then the trend is begun upon arrival of the next Trend Interval. User-configurable; LO Byte Date, integer 1..31. HI Byte Month, integer 1..12. (23,X,5,WORD)

Trend Target Time Stamp

The hour and minute of the day at which a trend initiated via the Action parameter is to begin. If Trend Target Time is less than the current time, then the trend is begun upon arrival of the next Trend Interval. User-configurable; LO Byte Hours, integer 1..24. HI Byte, Minutes; integer 1..59. (23,X,6,WORD)

Trend Time Stamp

For single trends: This value is set equal to the time at which the first piece of data was taken for the currently active or most recent trend. For continuous trends: This value is set equal to the hour at which Trend Pointer was last equal to zero. Not user-changeable. LO Byte Minutes, Integer 1..59. HI Byte Hours, Integer 1..24; (23,X,1,WORD)

Trend Timer

Used to time Trend Interval. Not user-changeable; reads in units designated by Trend Interval Units. (23,X,3,WORD)

Trend Type

Designates whether a continuous or single trend will be performed. "Single","Continuous"; user-configurable. (23,X,137,LOW BIT 2)

Trend Type Status

Indicates whether a continuous or single value trend will be performed. Mirrors the value chosen via Trend Type. "Single", "Continuous"; not user-changeable. (23,X,2,LOW BIT 1)

Trending Now

Applies to single-type trends only: indicates that Trend is currently in the process of filling the data storage area. "Yes","No"; user-configurable. (23,X,2,LOW BIT 5)

Waiting to Start

Indicates that a trend has been initiated via the Action parameter, but the designated time to take the first piece of data has not yet arrived. User-configurable; "Yes","No". (23,X,2,LOW BIT 3)

Y-Axis Multiplier

Used by ASI Expert (06/24/19999) to scale graphs. Not used in the controller. (23,X,141,HIMSNBL) DT=23, Class 83.

Trend Properties

The TREND Object has Data Type WORD; It is used to save data to RAM on a sequential bases. This data may then be uploaded for analysis.

Object Name	TRD, TREND
Object Number	23
Data Type	WORD
Index	0..n as allocated
DYNAMIC Attributes	137 (0..136)
STATIC Attributes	5 (137..141) FW740A....
STATIC Attributes	7 (137..143) FW840A....

Trend Firmware Revision

ASIC/2-7540 FW754A Rev 1.0 Forthcoming 2005

o As in FW740E.

SINC/3-3000 FW300B Rev 2.9 Released 05/29/2003 CHK 0xA80A

o Fixed 23-Trend and 27-Static Trend broken in 300b2.8
o Fixed 23-Trend rollover that added extra entry.
o Fixed 27-Static Trend Date Stamp at rollover.

ASIC/2-7040 FW740C Rev 2.5 Released 07/08/99 CHK 0x6613
ASIC/2-8040 FW840C Rev 1.5 Released 07/08/99 CHK 0x66EB

o Adds Y-Axis Multiplier Used by ASI Expert (06/24/1999)
o Fixes STD and TRD against stray bits being set

ASIC/2-7040 FW740D Rev 1.4 Released 11/05/96

o Object 23 Dynamic Trend fixed to automatically start on reset of power. Tandem parallel trends also start.

ASIC/2-8040 FW840A Rev 1.0 (March 1996)

o Add Trend Trigger Handle, Trend Trigger Enable, and Trigger Slope.

ASIC/2-7040 FW740A Rev 1.3 (16 June 1994)

o Eliminate Auto-Start on Power Reset

ASIC/2-7040 FW740A Rev 1.2 (14 June 1994)

o Add Multiple Trend Start Feature

ASIC/2-7040 FW740A Rev 1.0 (31 March 1994)

ASIC/2-7000 FW700A.. Rev 1.0 (12/05/91)

SC/1-9040 FW907A Rev 1.0 (07/25/91)

Trend DYNAMIC Properties

Attr-0 **Trend Date Stamp -** (mo,day)
LO BYTE- Trend Time Stamp - Month
HI BYTE - Trend Time Stamp - Day

Attr-1 **Trend Time Stamp - Time** (hr,min)
LO BYTE- Trend Time Stamp - Hour
HI BYTE - Trend Time Stamp - Minute

Attr-2 Trend Status LO BYTE

LO bit 0 - **Index Enable Status**

LO bit 1 - **Trend Data Size Status**
1 = word, 0 = byte

LO bit 2 - **Trend Type Status**
0 = single trend, 1 = rotating trend

LO bit 3 - **Waiting To Start**

LO bit 4 - **Trend Finished**

LO bit 5 - **Trending Now**

LO bit 6 - **Trend Pause** (action 3)

LO bit 7 - Spare

HI BYTE - **Number of Values**

Current Number of Values that has been trended. 0 indicates waiting for first trend. 255 indicates trend complete.

Attr-3 **Trend Timer**

Attr-4 **Static Trend Action**

1 = Initiate Single Trend.

If Target Date and Time are not zero, the trend will begin at that time. If Target Date and Time are zero, Trend will cause the current date and time to be written to the Trend Time Stamp at the first trend interval. It will stop trending after one loop and keep the Trend Time Stamp.

2 = Initiate Continuous Trend.

If Attr-5,6 written with Target Date and Time, trend will begin at that time. If Target Date and Time are zero, Trend will cause the current date and time to be written to the Trend Time Stamp at the first trend interval. It will renew the Target Date and Time Stamp after each loop

3 = Stop Trend (Pause).

Stop a trend in progress. If the trend is paused and restarted, then the Trend Time Stamp will not be accurate. No trace of the pause is left in the trend record.

4 = Restart Trend

Restart a stopped trend.

Attr-5 **Trend Target Time** - Date (mo,day)
LO BYTE- Trend Target Time - Month
HI BYTE - Trend Target Time - Date

Attr-6 **Trend Target Time** - Time (hr,min)
LO BYTE- Trend Target Time - Hour
HI BYTE - Trend Target Time - Minute

Attr-7 Trend Status
LO Bit 0 **New Trend Start**
LO Bit 1 **New Trend Finish**
LO Bit 2 Spare
LO Bit 3 Spare
LO Bit 4 Spare
LO Bit 5 Spare
LO Bit 6 Spare
LO Bit 7 Spare
HI Byte Spare

Attr-8 Spare

Attr-9 **Trend Data 1**
LO BYTE- Trend Data Byte 1
HI BYTE - Trend Data Byte 2

Attr-10 **Trend Data 2**
LO BYTE- Trend Data Byte 3
HI BYTE - Trend Data Byte 4

...

Attr-136 **Trend Data 128**
LO BYTE- Trend Data Byte 255
HI BYTE - Trend Data Byte 256

Trend STATIC Properties

Attr-137 Trend Setup

LO BYTE

LO bit 0 - **Index Enable**
LO bit 1 - **Trend Data Size**
1 = word, 0 = byte
LO bit 2 - **Trend Type**
0 = single trend, 1 = rotating trend
LO bit 3 - **Trigger Mode Enable** (840A..)
LO bit 4 - **Trigger Slope** (840A..)
LO bit 5 - Spare
LO bit 6 - Spare
LO bit 7 - Spare

HI BYTE - **Trend Interval Units**
0 = in seconds
1 = in minutes
2 = in half-hours
3 = in hours
4 = at each new day
Will trend at Trend Time Stamp, or at midnight if Time Stamp is 0.

Attr-138 **Trend Interval**
In Trend Interval Units.

Attr-139,140 **Trend Source Handle**
Attr-139 LO,Trend Source - Object
Attr-139 HI,Trend Source - Index
Attr-140 LO,Trend Source - Attribute
Attr-140 HI,Trend Source - Select

Attr-141 TandemTrend Setup
LO Byte **Trend Tandem Index**
HI Bit 0 **Start Tandem Enable**
HI Bit 1 **Start Next Enable**
HI Bit 2 Spare
HI Bit 3 Spare

HIMSNBL - **Y-Axis Multiplier** Used by ASI Expert (06/24/1999)
0 = Units , Value*1.0
1 = Deci- , Value*0.1.
2 = Centi-, Value *0.01
3 = Milli- , Value*0.001
4 = Value*2
5 = Value*4
6 = Value*8
7 = Value/2
8 = Value/4
9 = Value/8
10 = Percent, Value/2.55 , e.g. Analog Output
11 = Reserved
12 = Volts 10.00*Value/255 for Analog Output, 0 to 10.00 volts
13 = Volts 0 to 5.00 volts; .500*Value/1023 10bit Input.(FW740A..)
14 = Percent = 1000*Raw/1023 , Value*1.0

Attr-142,143 **Trend Trigger Handle** (FW840A..)
Attr-142 LO,Trend Trigger - Object
Attr-142 HI,Trend Trigger - Index
Attr-143 LO,Trend Trigger - Attribute
Attr-144 HI, Trend Trigger - Select