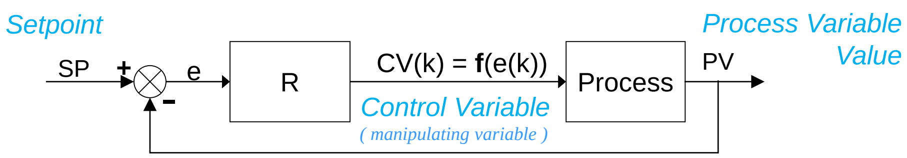
PID Regulator

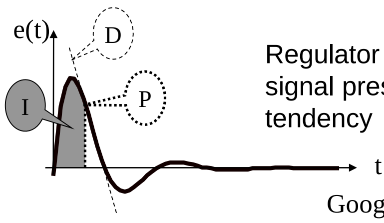
Simple, robust (works also poorly config.) ⇒ the most widely used



$$CV(t) = Kp\left(e(t) + \frac{1}{Ti}\int e(t)dt + Td\frac{de(t)}{dt} \right)$$

P coefficient — *I part* — *D part* — *ideal non-interacting PID algorithm*

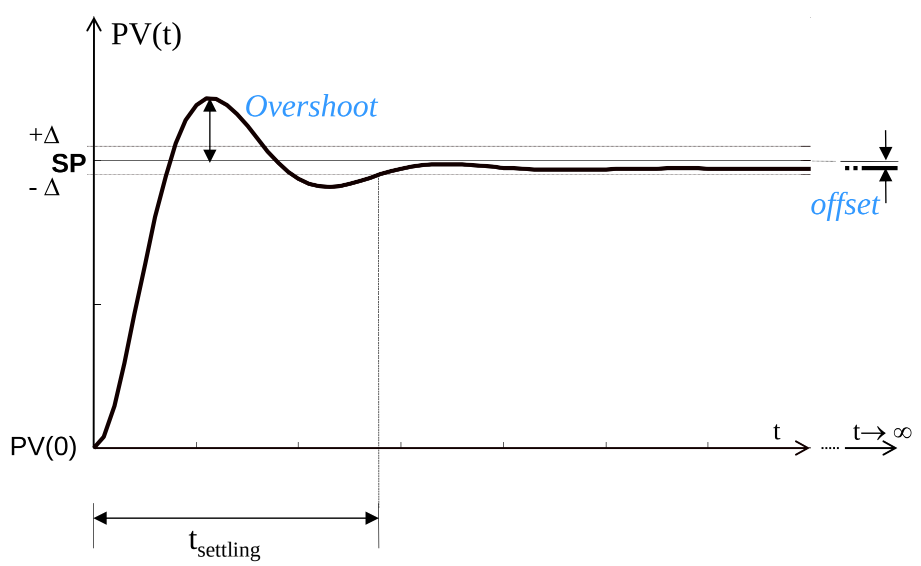
Basics: https://youtu.be/EqEqEdOx-x8



Regulator output (CV) is computed from error signal present and past values and future tendency

Google: **„Control Engineering A guide for beginners“**

Process Variable transient

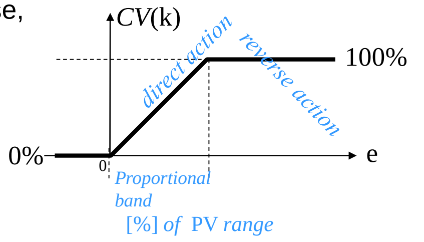


PID controller

P - Proportional action $$CV(k) = Kp \cdot e(k)$$

bigger *Kp* (narrow PB) gives faster response,
P alone leaves offset error

VisiLogic Help > Ladder > FBs Library > PID FB + Auto-tune
> General Background > Inside the PID Function
> Proportional Band / Integral / Derivative Action



I - integral action, reset action

removes static deviation, can add oscillations and "*wind-up*"

CV = const. when e = 0
CV ≠ const. when e ≠ 0

Ti - *integral time*
larger corresponds to reduced I component

D - derivative, rate action

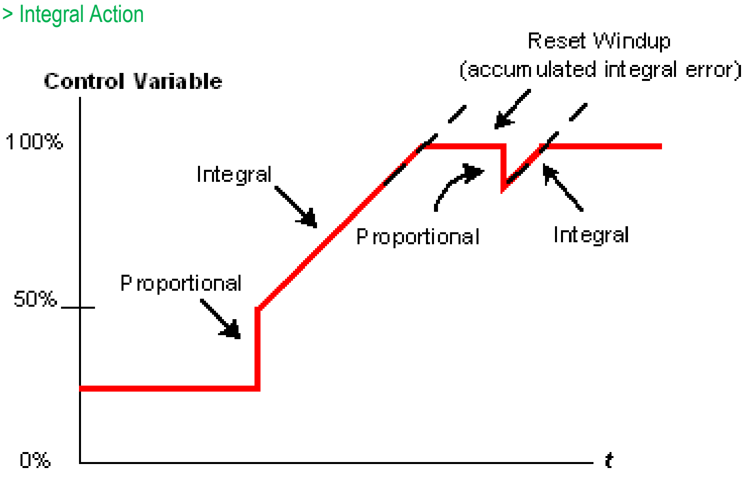
Td - *rate time*
larger corresponds to increased D component

responds to the rate and direction of change in the e
preventing the overshooting and oscillations
allowing to choose stronger P and I

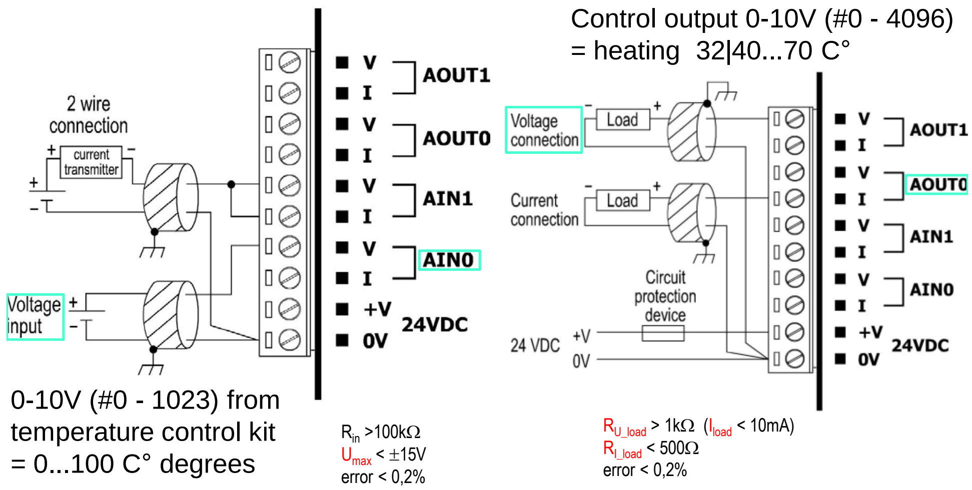
Integral „wind up"

CV output at 100%, a condition called saturation, can take place

VisiLogic Help > Ladder > FBs Library > PID FB + Auto-tune
> General Background > Inside the PID Function
> Integral Action



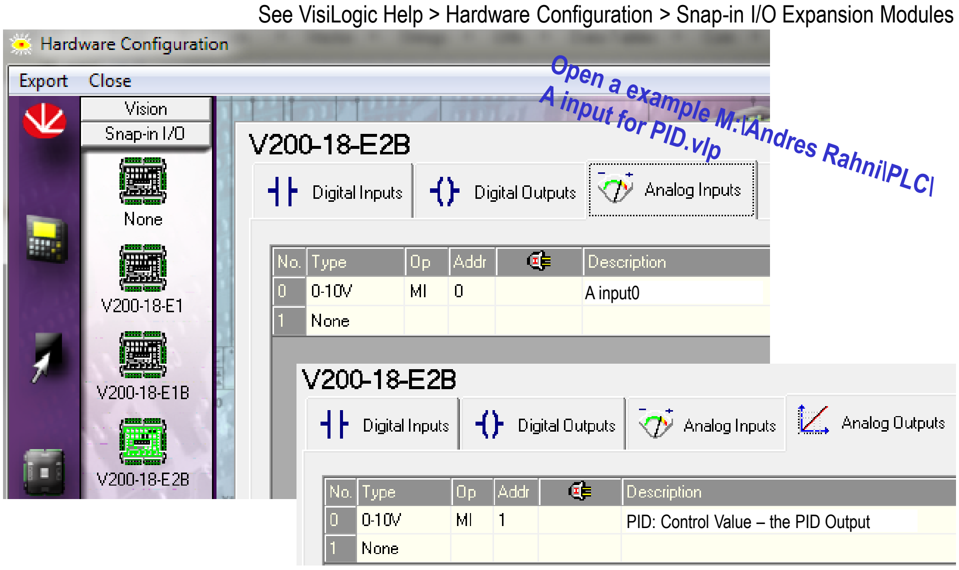
V200-18-E2B Analog I/O



Analog I/O Configuration

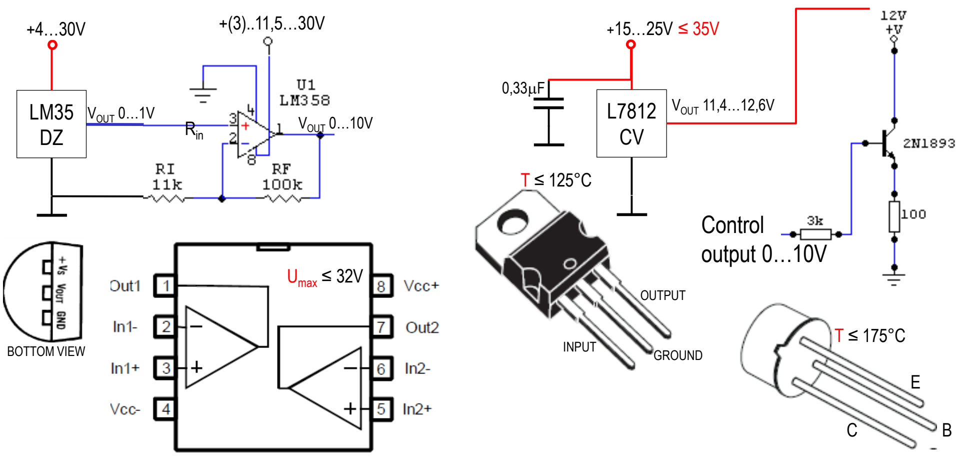
Before you can use an analog input/output in your program, you must link it to an operand. An analog input value can be contained in an MI, ML, or DW.

See VisiLogic Help > Hardware Configuration > Snap-in I/O Expansion Modules



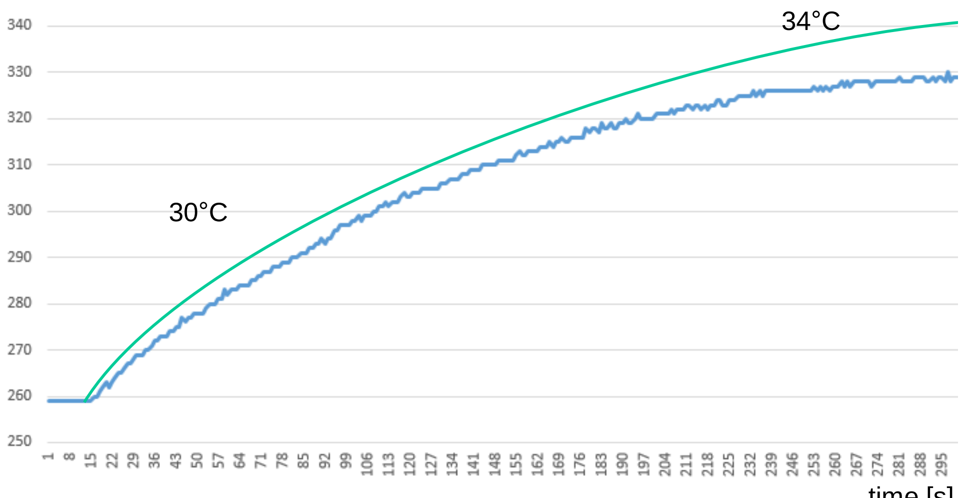
Temperature control kit components

LM35DZ – 2...100 Centigrade Temperature Sensor
(linear 10mV/C°, ± 1.5°C accuracy)
LM358N – Low-power dual operational amplifiers DIP8
L7812CV – 12V voltage regulator - generates heat flow
(w. thermal overload protection)
A transistor amplifies control signal and modulates heating current



Power On Process Self Heating

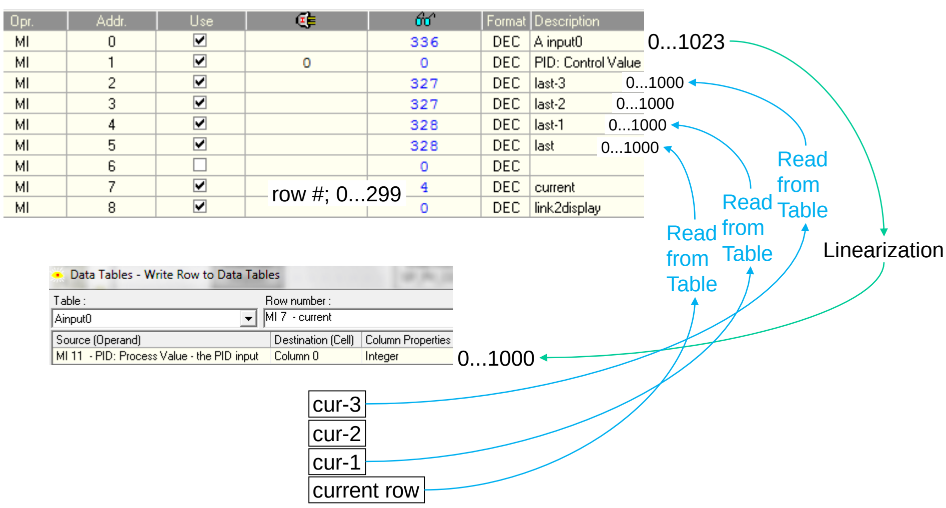
Analog Input0 value – Temperature sensor



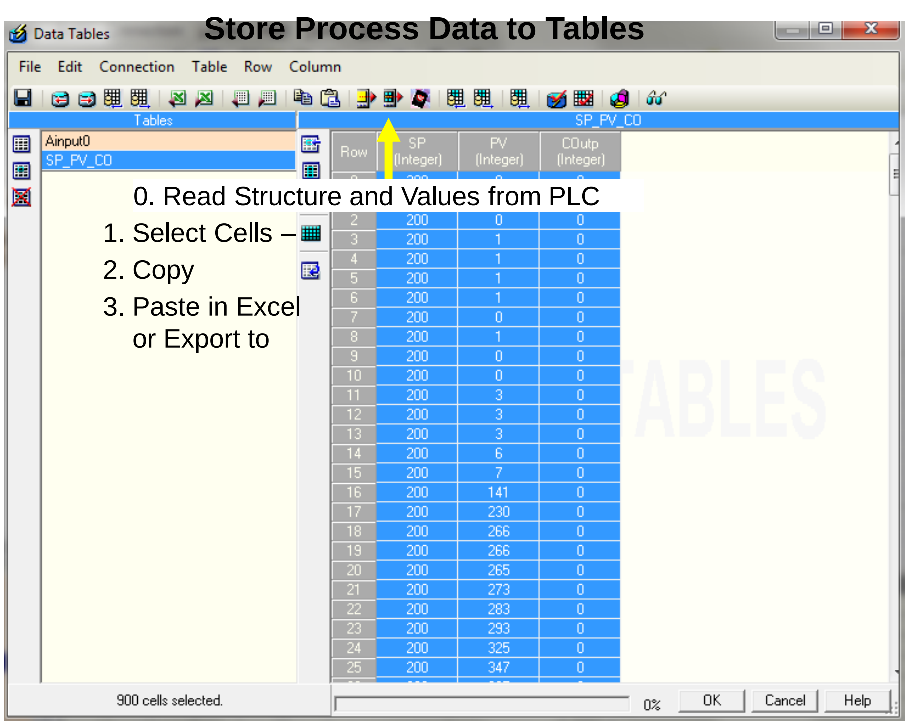
Analog Output0 value = 0; CV =0

Write to Data Tables

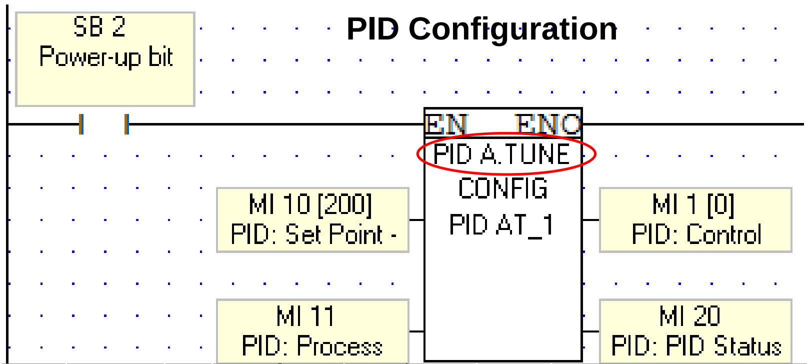
Open example M:\Andres Rahni\PLC\ A input for PID.vlp



Store Process Data to Tables



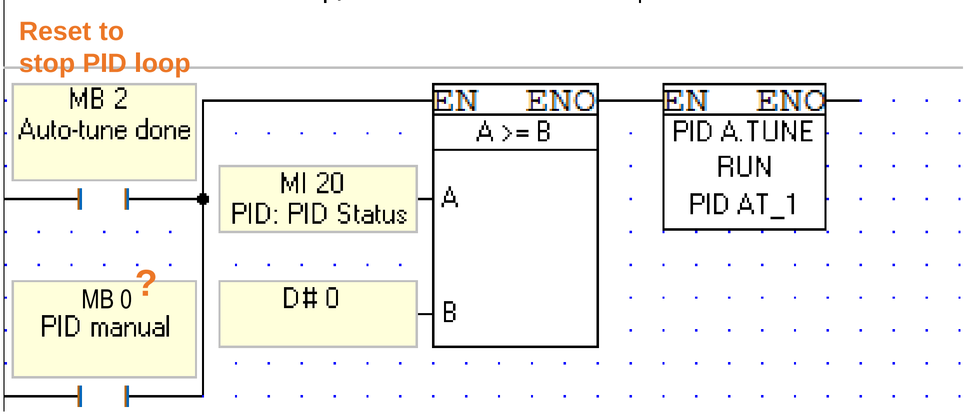
PID Configuration

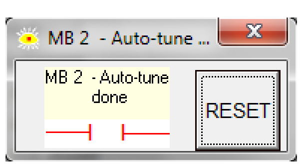


Params	Type	Add			Format	Description
IN	MI	10	350		DEC	PID: Set Point - the target value
	MI	11			DEC	PID: Process Value - the PID input
	MI	12	100		DEC	PID: Proportional band - defined in units of 0.1% (P gain)
	MI	13	10		DEC	PID: Integral time - defined in units of 1 second (I gain)
	MI	14	0		DEC	PID: Derivative time - defined in units of 1 second (D gair
	MI	15	50		DEC	PID: Sample Time - defined in units of 10 mSec .Recomm
	MB	10				PID: Action: 0: Reverse(Heating-default) 1: Direct(Cooling
	MI	16	0		DEC	PID: Input Range - Process Value Low limit
	MI	17	1000		DEC	PID: Input Range - Process Value High limit
	MI	18	0		DEC	PID: Output Range - Control Value Low limit
	MI	19	4096		DEC	PID: Output Range - Control Value High limit
OUT	MI	1	0		DEC	PID: Control Value - the PID output
	MI	20			DEC	PID: PID Status
	MI	200			DEC	Auto-tune parameters, 32 MIs - 1 of 32

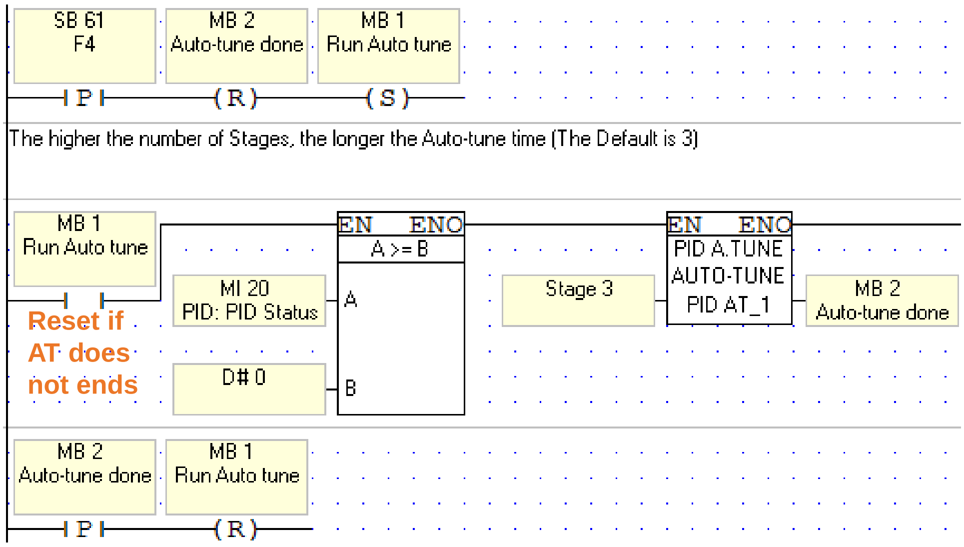
Run PID

In order to Auto-tune the loop, the PID RUN must be suspended

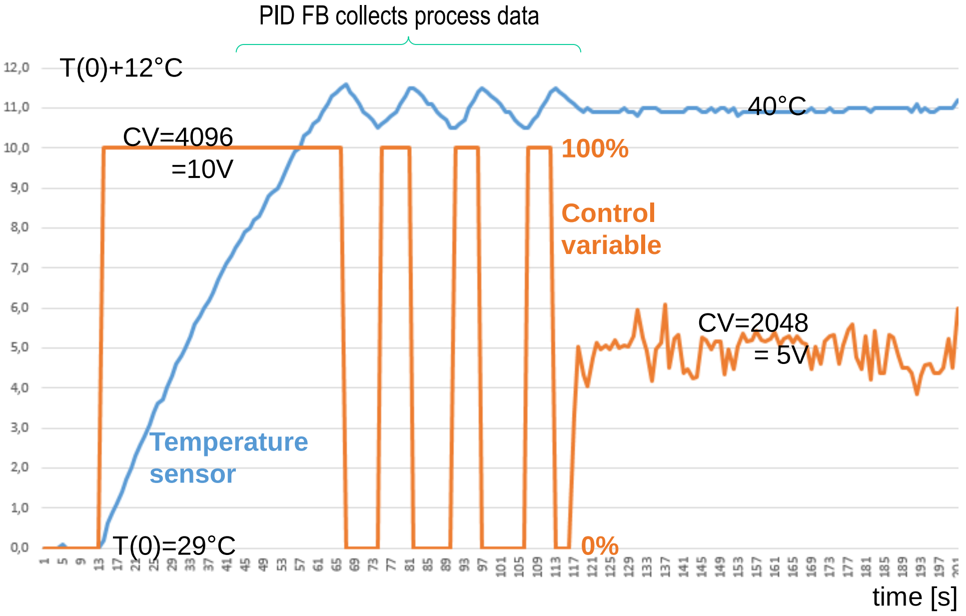




PID Auto-Tune Stages



PID Auto-Tune Stages



PID FB Status Integer

PID error indications are given in the Status Messages MI

See VisiLogic Help > Ladder > FBs Library > PID FB + Auto-tune

0	FB status OK
1, 2, 3	Auto-tune in progress
4	PID running
5, 6	Setpoint change in progress
7	Integral-wind up
8	integral-wind down
9	Pause mode, Integral and Derivative values are not currently being calculated
10, 11	PV exceeds proportional band, no calculation performed
...	

ISS0089 ADVANCED PROGRAMMABLE LOGIC CONTROLLERS

autumn 2018

Homework 3 – Vision 230 and PID regulator

Plant

Vision 230 controller and Temperature control process kit

Tasks:

Program a process PID control algorithm for temperature control in controller. Program must be able to stop the PID control (CV = 0), adjust the temperature setpoint between reasonable limits and run it again.

On controller display User Interface show temperature current value (and short history), control variable (CV) current value, PID current mode and all (manual mode) commands choices: STOP, RUN AT | PID.

In your report list the program, PID controller configuration, user interface manual and a setpoint change response (from self heated temperature level to 45..55°C level) graph. To draw graph, program must be able to store CV, (SP) and PV values for approx. 5 minutes.