A Freely Available Interactive PID Learning Module

Theorin, Alfred; Johnsson, Charlotta

2014

A Freely Available Interactive PID Learning Module

Alfred Theorin and Charlotta Johnsson
Department of Automatic Control, Lund University, Lund, Sweden

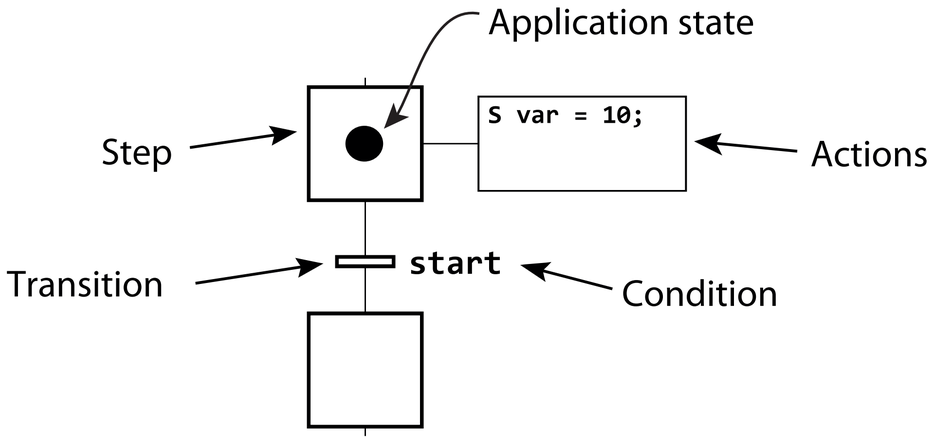
Background

The PID is used in more than 95% of all the billions of control loops and a deep understanding of PID control is thus desirable. Interactive learning tools are great at providing students with hands on experience. One such learning module has been developed in Grafchart and it is freely available.

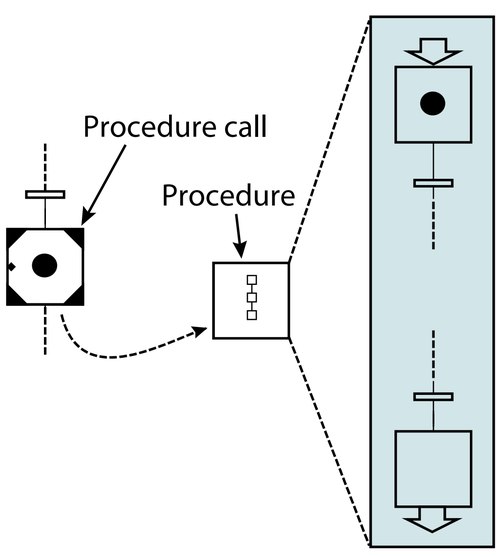
Me

Alfred Theorin
Ph.D. student
alfred.theorin@control.lth.se

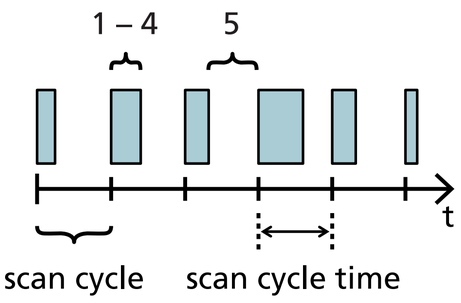
The Grafchart Language



Grafchart is a graphical programming language. The basic building blocks are steps (application states) with actions (what to do when) and transitions (how and when to change applcation state).



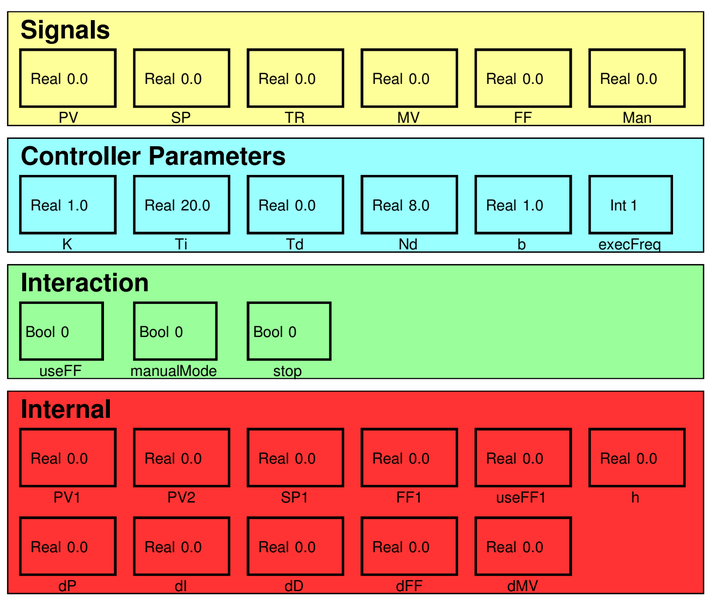
Grafchart procedures enable code reuse.

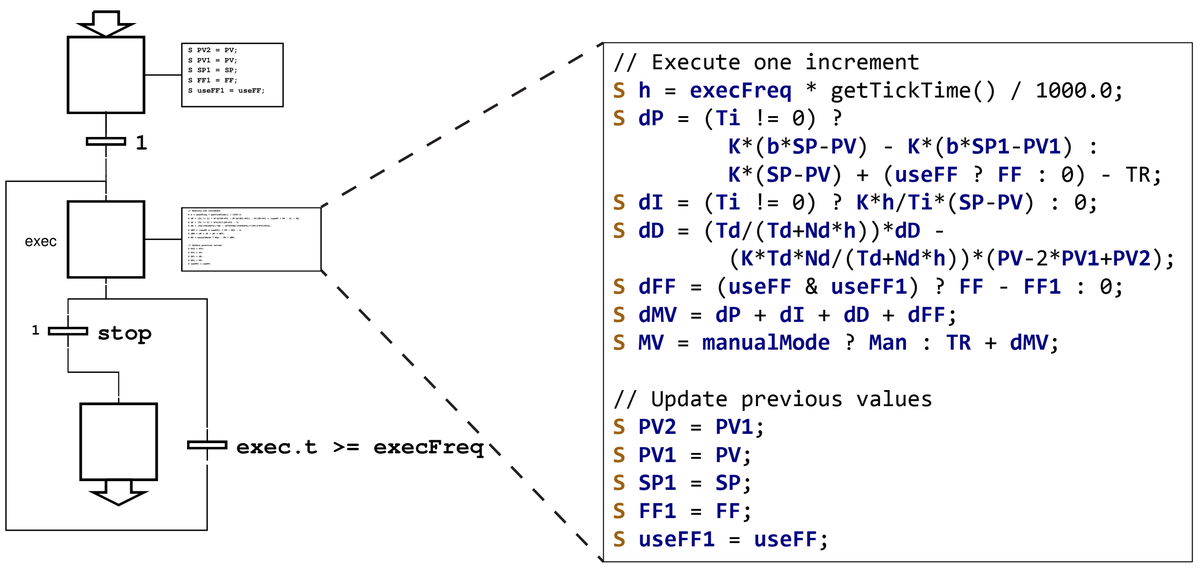


Execution model

1. Read inputs
2. Fire transitions (exeute **X** and **S** actions)
3. Execute **P** actions
4. Update variables subject to **N** actions
5. Sleep until next scan cycle

The PID Procedure





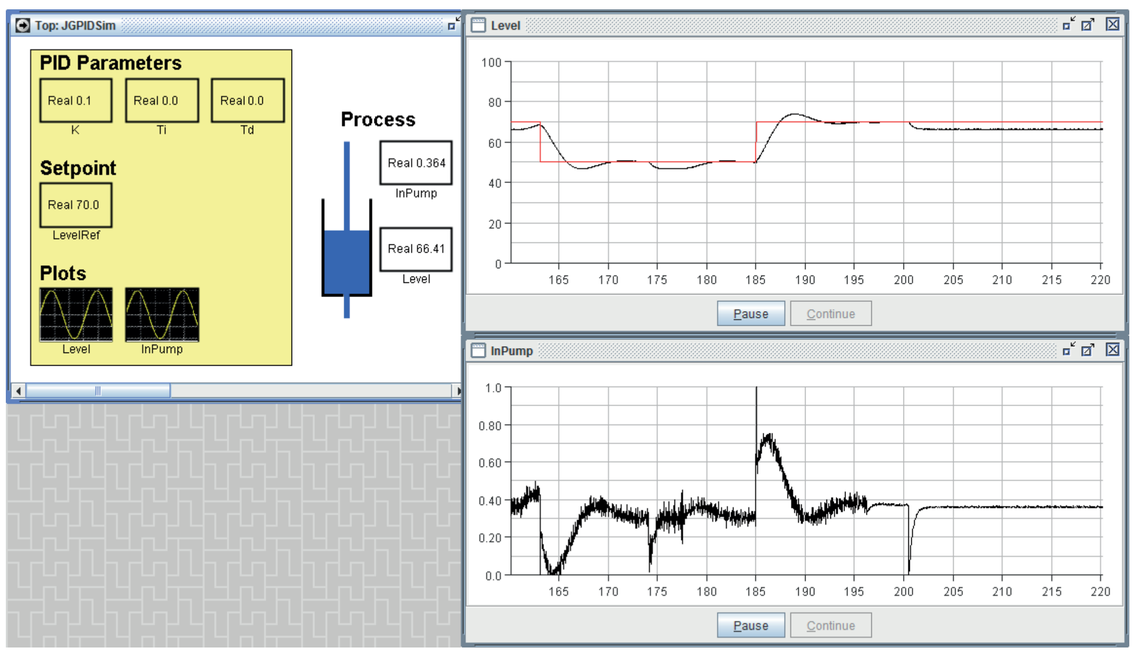

```
// Execute one increment
S h = execFreq * getTickTime() / 1000.0;
S dP = (Ti != 0) ?
         K*(b*SP-PV) - K*(b*SP1-PV1) :
         K*(SP-PV) + (useFF ? FF : 0) - TR;
S dI = (Ti != 0) ? K*h/Ti*(SP-PV) : 0;
S dD = (Td/(Td+Nd*h))*dD -
         (K*Td*Nd/(Td+Nd*h))*(PV-2*PV1+PV2);
S dFF = (useFF & useFF1) ? FF - FF1 : 0;
S dMV = dP + dI + dD + dFF;
S MV = manualMode ? Man : TR + dMV;

// Update previous values
S PV2 = PV1;
S PV1 = PV;
S SP1 = SP;
S FF1 = FF;
S useFF1 = useFF;
```

PV	Process Value
SP	SetPoint
TR	Tracking
MV	Manipulated Variable
FF	FeedForward
Man	Manual control signal
K	Controller gain
Ti	Integral time
Td	Derivative time
Nd	Maximum derivative gain
b	Setpoint weight
execFreq	PID sample time
useFF	Turn feedforward on/off
manualMode	Turn manual mode on/off
stop	Terminate call

A full-fledged PID module has been implemented as a procedure. It supports anti-windup, auto/manual mode, feedforward, tracking, setpoint weighting, and bumpless mode and parameter changes.

It is implemented on incremental form which is easier to get right as most features come for free.

The PID Learning Module



A basic PID learning module has been created. To make it standalone it uses a simulated process. Since the PID procedure uses **S** actions, proper execution was ensured by limiting the control signal and then updating the simulated process with **P** actions in the same step.

To the left, setpoint and controller parameters can be changed during execution and a live animation of the process is shown. To the right, live plots are shown. The upper plot shows process value (black) and setpoint (red) and the lower plot shows the control signal.

Conclusions

The module will be included in future JGrafchart releases. Unlike other PID learning tools JGrafchart is free, based on an industrial control language, and can be used in industry-like environments. Future work includes using it for education as well as adding more features to the PID procedure, for example process value filtering or an auto-tuner.

Download: **http://www.control.lth.se/Research/tools/grafchart.html**

Acknowledgments

Financial support from the VINNOVA-FFI project LISA is gratefully acknowledged. The authors are members of the LCCC Linnaeus Center and the eLLIIT Excellence Center at Lund University.