Commissioning

Relaxc for Modicon Controllers

(Schneider Electric)

John Masse

Relaxc@appedge.com

Version 3

Table des matières

RELAXC

1 Introduction

This documentation describes the Relaxc block in Control Expert tool of Schneider for MODICON controllers
The use of Relaxc is similar to Schneider Lag_filter.

2 What is in This Chapter

This chapter contains the following Topics

3 Description

3.1 Why to use Relaxc.

The Relaxc' approach is:

- Easy to tune, easy to use without complex calculations or specific tools,
- Robust and efficient,
- Generic for all processes.
- Directly tunable without modelling.

Relaxc allows to control **effortless and without overshoot** complex processes including Non minimal phase, small and large pure delay, unstable processes, variable static gain, strong non-linearities, constraints on U, speed saturations, discontinuities, load disturbances, etc. While minimizing the operational costs, the energy peaks of the control U (Urelaxc) at each Step of (SP: set point) with a more reliable control.
We can use Relaxc where the control is not satisfactory resolved by a PID or other control laws. It is advantageous to replace for example all the PID by Relaxc. The maintenance and the predictive maintenance will be easier.
Relaxc stability: Relaxc ensures a good stability by its mathematical nature because it allows to find the best response in closed loop without overshoot. It is a guarantee of robustness.

3.2 Function description

The function block represents Relaxc Controller.
The function block contains the following operating mode:

- Tracking

- Automatic

Operating mode	Tr_S
Tracking	1 (Urelaxc=U_i)
Automatic: (Relaxc works)	0

NOTE: This function block performs an internal initialization in the first program cycle after a warm start or cold start (e.g. application download or power cycle) of the PLC program.
Due to this, you have to make sure that the function block is invoked in the first program cycle. In case of invoking the function block in a later program cycle, the internal initialization will not be performed and the output may deliver a wrong value.

WARNING UNEXPECTED OUTPUT BEHAVIOUR Make sure that the function block is always invoked in the first program cycle.

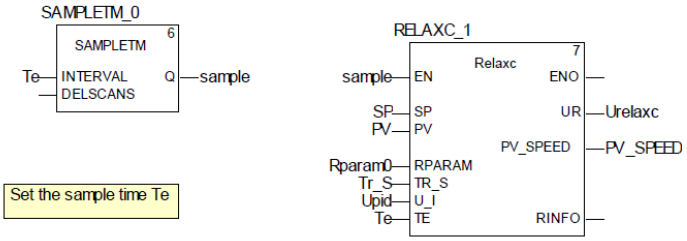
3.3 **Equation of Relaxc**

We just recall briefly the algebraic equation of *Relaxc*:

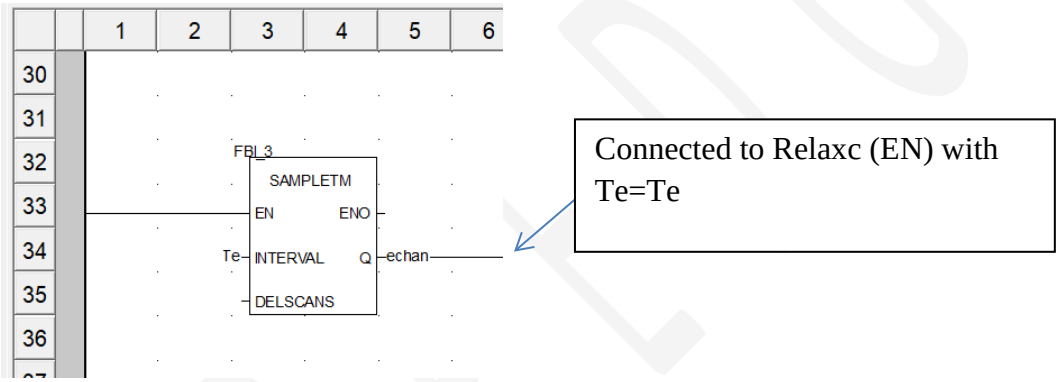
$$u(k)=R(u(k-1),tre)+ks*(tg*s*e+e)$$

- With R the reactivity or Relax function which depends on *tre*
- e=(yref-Pv) with Yref the reference trajectory that depends on the time constant tg and Pv (the output of the process).
- Ks the convergence gain on e
- U (the output of Relaxc)
- (s) Laplace variable.
- U(k) is the output of Relaxc (Urelaxc)

3.4 Representation in FBD



3.5 Representation in LD



3.6 Representation in IL

Remark: Take into account EN/ENO

```
CAL Relaxc0 (Sp := (*REAL*),
          Pv := (*REAL*),
          Rparam := (*Rparam*),
          TR_S := (*BOOL*),
          U_I := (*REAL*),
          Te := (*TIME*),
          URelaxc => (*REAL*),
          Pv_Speed => (*REAL*),
          Rinfo => (*REAL*));
```

3.7 **Parameter description**

Description of input parameters:

Parameter	**Data type**	**Description**
EN	Bool	Enable Input : Relaxc is processed if EN=1 (Edge Rising). Because Relaxc does not use an internal function time
Sp	real	Set point
Pv	real	Measure of the process value
Rparam	Rparam (structure)	Tuning parameters of Relaxc
Tr_s	bool	Initialization Command (0 => Relaxc works, 1=> Urelaxc =U_i)
U_i	real	Initialization of the output U of Relaxc (Urelaxc) on a edge rising of Tr_S or Tr_S=1
Te	time	Sample time with respect to EN

Description of output parameters:

Parameter	**Data type**	**Description**
ENO	Bool	Enable output
UR	real	Output of Relaxc (Urelaxc)
PV_SPEED	real	Display the the speed of the process output (PV).
Rinfo	real	Version of Relaxc.

Description of the internal parameters**: Param**

Parameter	**Data type**	**Description**
tg	real	Time constant of the reference trajectory in seconds
tre	real	Relaxc time or reactivity time (delay+lag) in seconds
Ks	real	Speed convergence gain or throttle between the reference trajectory and PV
Umax, Umin	Real	Upper limit and lower limit on the Relaxc output: (Urelaxc). For example [100 to -100]
tv	real	Time filter of smoothing in second on the derivative (speed) of PV

3.8 Runtime error

For a list of all block error codes and values :- None.

3.9 Sample time Te

The user controls the sample time of Relaxc with Te (Time format example t#10ms). The sample time of Relaxc can be different from the Mast sample time because it is controlled by a SAMPLETM block. (See Figure 1 Relaxc in tracking mode with a PID).

The Sample time (Te) has to be the same for the block SAMPLETM and Relaxc. The output Q of SAMPLETM has to be connected to the input EN of Relaxc.

3.10 Relaxc Initialization

During the cold or warm start, Relaxc initializes the data such as Urelaxc=U_i .

3.11 Minimal time values of tre, tv, tg

These times use a real format. For example to set tre at 100ms we write tre=0.1. For 10 seconds, tre=10.0. It is possible to set tre and tv at zero or under the value of the sample time. For example tre=Te/10.0. We recommend the minimal value tg=8.0*Te or more for Shannon and numerical integration rules.

3.12 Gain ks

Note that the gain ks gives the initial value of Urelaxc. Urelax(t=0 seconde)=ks*SP.

With ks=0, Relaxc does nothing. Urelaxc=0. The value of ks controls the response time or the overshoot in closed loop with respect to tg and tre.

3.13 Normalization of controller range.

The best and the more practicable way to control a process is to normalize the input (PV) between [0,100] or [-100,100] and to do the same thing on Urelaxc and the set point (SP). This facilitates process understanding, process maintenance and the commissioning. This step is necessary if we want to know the process normalized speed (v) to compute the parameter ks.

3.14 Tracking mode TR_S=1 and Pv_speed.

When the mode TR_S=1, thus Relax does not work and Urelaxc=U_I. It is a useful mode when we perform a normalized step of the process independently of SP. It allows measuring the speed (v) of the process. This value is necessary to compute ks.

The value of v is the maximal normalized speed of the process v=measured/U_I and $ks = \frac{1}{v*\tau_g*(\frac{td}{tre}+1)}$

Remark: When TR_S=0(Relaxc works). The Pv_speed is also computed during the control and the speed must be at least close to the measured speed. This rule allows to reach a safe and optimal control without overshoot.

3.15 Constraints on Urelaxc=[Umax, Umin] and on the process speed or acceleration.

Relaxc take into account naturally all these constraints without using complex anti windup scheme or complex saturation tuning. There is nothing to do. Just let Relaxc to work.

3.16 How to tune Relaxc

See the documentation about tuning.

4 Example of Relaxc implementation.

This example shows how to plug Relaxc in parallel to the PID in simulation mode. The goal of this example is to control a first order+delay and to switch the tracking mode functionality between PID and Relaxc controller.

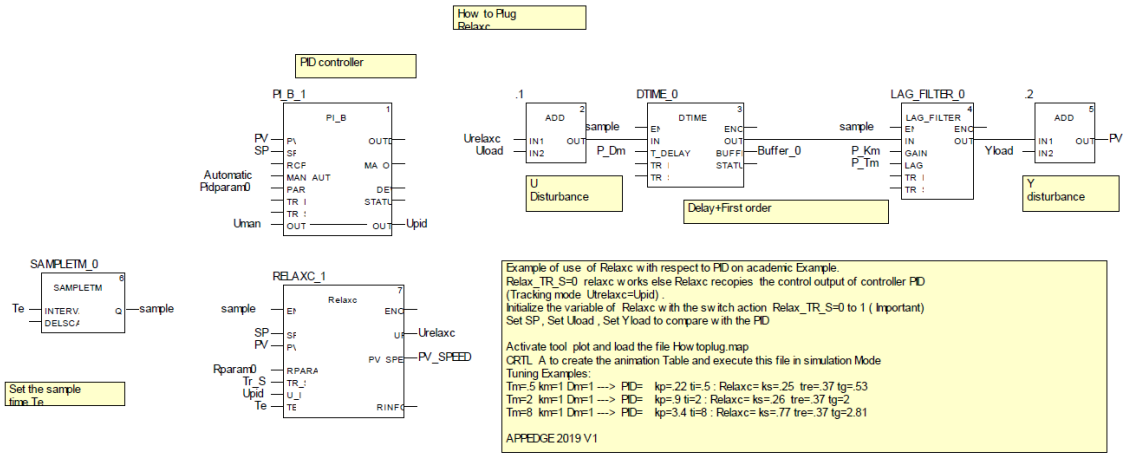
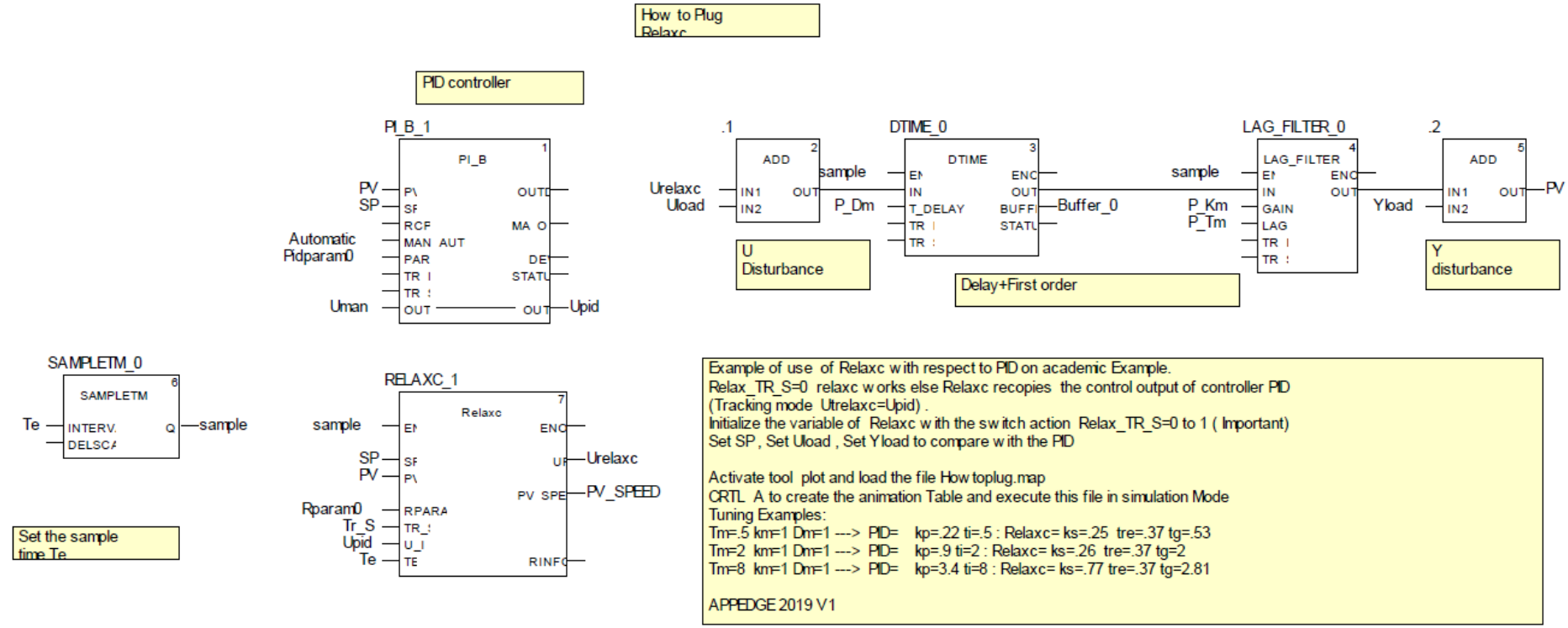


Figure 1 Relaxc in tracking mode with a PID.


How to Plug Relaxc
PID controller
PI_B_1
PI_B
PV
SP
Automatic
Pidparam0
Uman
OUT
Upid
.1
ADD
Urelaxc
Uload
sample
DTIME_0
DTIME
P_Dm
Buffer_0
U Disturbance
Delay+First order
LAG_FILTER_0
LAG_FILTER
P_Km
P_Tm
GAIN
LAG
.2
Yload
PV
Y disturbance
SAMPLETM_0
SAMPLETM
Te
INTERV.
Q
Set the sample time Te
RELAXC_1
Relaxc
SP
PV
Rparam0
Tr_S
Upid
Te
Urelaxc
PV_SPEED
RINFO
Example of use of Relaxc with respect to PID on academic Example.
Relax_TR_S=0 relaxc works else Relaxc recopies the control output of controller PID
(Tracking mode Utrelaxc=Upid) .
Initialize the variable of Relaxc with the switch action Relax_TR_S=0 to 1 (Important)
Set SP , Set Uload , Set Yload to compare with the PID
Activate tool plot and load the file How toplug.map
CRTL A to create the animation Table and execute this file in simulation Mode
Tuning Examples:
Tm=.5 km=1 Dm=1 ---> PID= kp=.22 ti=.5 : Relaxc= ks=.25 tre=.37 tg=.53
Tm=2 km=1 Dm=1 ---> PID= kp=.9 ti=2 : Relaxc= ks=.26 tre=.37 tg=2
Tm=8 km=1 Dm=1 ---> PID= kp=3.4 ti=8 : Relaxc= ks=.77 tre=.37 tg=2.81
APPEDGE 2019 V1