APPLICATION NOTE #1

Application Note 1: The PLC and the PID.

This application note is about PID loop controller implementation with Entertron PLCs. The example used is fictitious and should be used only as a guideline.

PID loop controllers are commonly used control algorithms used in various applications. Some of the most common are position and temperature control. PID loops control a process by examining the ERROR signal of the process and reacting with the CONTROL signal accordingly. The ERROR is the difference between where you want a process and where it actually is. The desired location or temperature is called the SETPOINT, the actual temperature or location is called ACTUAL, and their difference is called the ERROR.
In other words, ERROR = SETPOINT - ACTUAL.

PID loops specifically address 3 aspects of the ERROR signal, proportional error, Integral error, and differential error. These aspects of the error are used to direct the CONTROL signal. The CONTROL signal is the numeric value which drives the process. In a motion control situation it may drive a motor, in a temperature control situation it determines the amount of time the heat is called for.

Proportional error correction manipulates the CONTROL signal in proportion to the error. If the ERROR is large then the correction is large. In a temperature control situation the amount of heat called for is directly proportional to the ERROR, hence the term proportional control. Proportional control is easy to implement and can be very useful. The formula below is used for proportional correction.

CONTROL=CONTROL + PK * (ERROR).

Where PK is a constant of correction which is specific to the process. You can write a PHD thesis by describing how to get a good PK, I will not attempt to say any more than to plug in some values, and use the one that works best in the process.

Differential error correction is correction based on the difference of the error from one logic scan to the next. For you calculus buffs it is the first derivative of the ERROR with respect to time. If the difference of the ERROR is large from one scan to the next then the correction will

be large. To calculate the differential correction it is necessary to store the error from one scan to the next. The equations below is used to perform differential correction.

CONTROL=CONTROL + DK * (ERROR-ERRORLAST).
ERRORLAST=ERROR.

Like before DK is a constant of correction which is derived by examining the results.

Integral error correction is based on the history of the error signal from the beginning of the system running. It corrects the CONTROL by the amount of error seen throughout the running of the process. For you calculus buffs it is the definite integral of the error from time zero until the present, integrated with respect to time. As with differential error correction you must store results of each calculation to be used in following calculations. The following 2 equations can be used to calculate integral error correction.

INTERROR = INTERROR + ERROR
CONTROL=CONTROL + IK * INTERROR

At the beginning of the program you must be careful to zero the value INTERROR or results are in error. Also a lot of PID loop controllers zero this register every few minutes to prevent some problems with run away controllers. Integral error correction is best used with caution, and frequent resetting of the integral register.

EXAMPLE

The following registers are used in this example to make the calculations necessary.

R1 CONTROL signal. Typically connected to a driver of some sort.

R2 Analog INPUT. Typically the value of the sensor measuring the process.

R3 SETPOINT. The desired value for the process set by the user program, or one of Entertron's user display devices.

Because the Entertron controller has only integer math, the coefficients of correction are stored as fractions in 2 registers.

R4 DK coefficient numerator.
R5 DK coefficient denominator.

R6 PK coefficient numerator.

R7 PK coefficient denominator.

R8 IK coefficient numerator.
R9 IK coefficient denominator.

R10 Error seen at the last scan, this is used for Differential correction.

R11 Integral error used for integral correction.

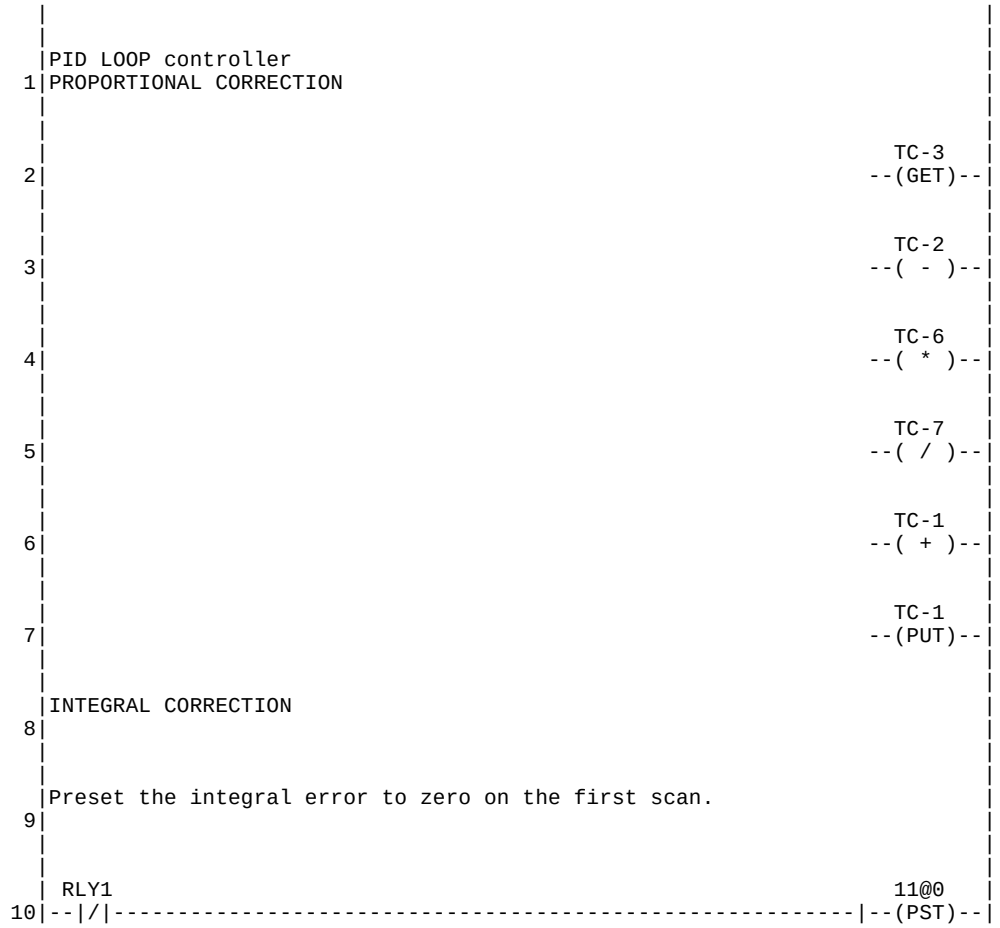
The error signal is (R3 - R2) and is repeated throughout the example.

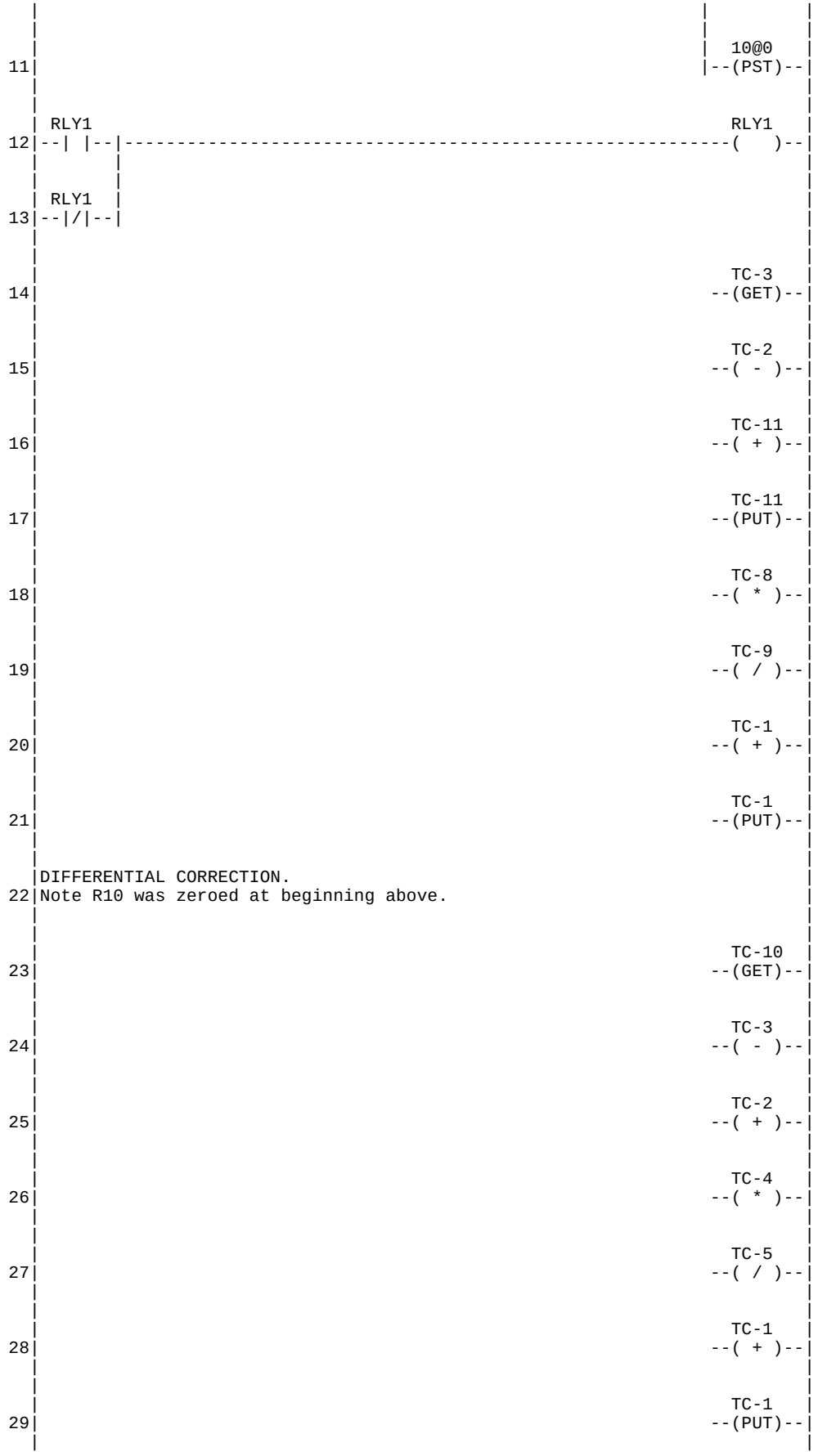
PROPORTIONAL CORRECTION.
R1=R1+(R3-R2)*R6/R7

INTEGRAL CORRECTION.
R11=R11+(R3-R2)
R1=R1+R11*R8/R9

DIFFERENTIAL CORRECTION.
R1=R1+(R10-(R3-R2))*R4/R5
R10=R3-R2

Below you see the ladder logic needed to implement this PID loop controller.




10@0
(PST)
RLY1
RLY1
(   )
RLY1
TC-3
(GET)
TC-2
(-)
TC-11
(+)
TC-11
(PUT)
TC-8
(*)
TC-9
(/)
TC-1
(+)
TC-1
(PUT)
DIFFERENTIAL CORRECTION.
Note R10 was zeroed at beginning above.
TC-10
(GET)
TC-3
(-)
TC-2
(+)
TC-4
(*)
TC-5
(/)
TC-1
(+)
TC-1
(PUT)


30
TC-3
--(GET)--
31
TC-2
--(-)--
32
TC-10
--(PUT)--
33
--(END)--