Technical Note 124

Calculating the derivative of speed with Drive Application Programming

Programming example

This tech note provides an example for calculating the derivative of a drive's speed. This is often useful for creating a feed forward torque demand to compensate for the inertia of a fixed roll or for web handling applications. This example is based on ACS880 firmware. However, this method can be implemented on any ABB all-compatible drive.

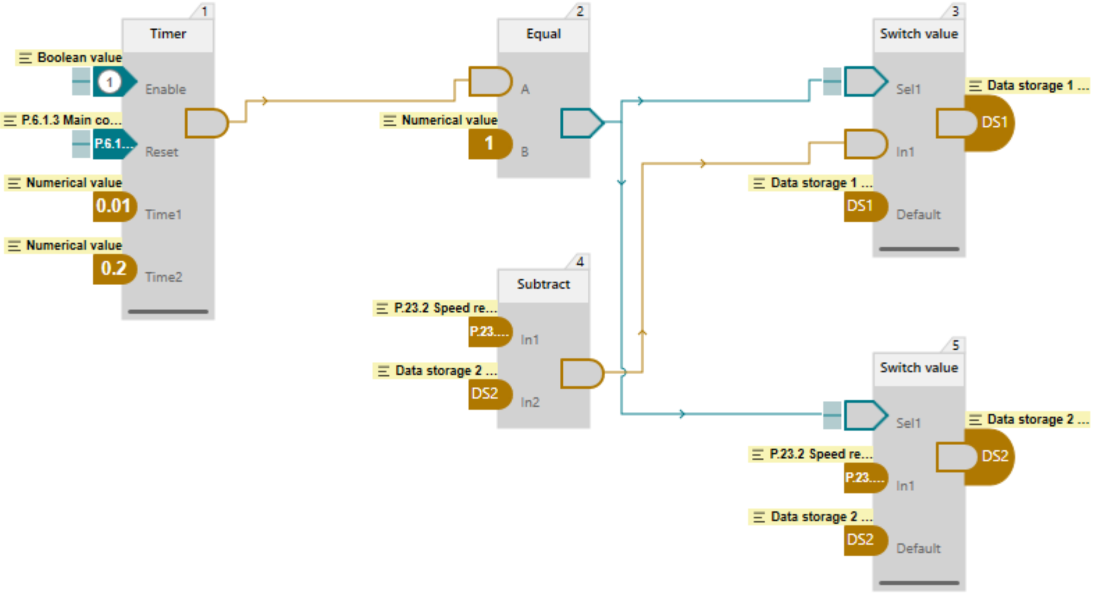
This application programming (AP) logic example has been developed as an alternate way to create inertia compensation. It is a standalone method to replace functions found within the ACS800 inline based firmware. This example is configured for use with the ACS880 demo case and can also be used in a real-world application.

Description of operation

The program calculates the rate of change of the drives speed reference. By default, it is calculated every 200ms, but this can be adapted to other needs by modifying the values embedded in the program. It uses two storage registers to hold the previous speed reference and the current rate of change in the reference. These data storage locations can be used to point at so you can use this calculation in other areas of the drive logic.

AP program

This example requires the AP blocks as shown:



Program description

Block 1, the Timer block, is the key to operation. The timer consists of two values, Time1 and Time2. The block is always enabled and is reset with the drive running status bit in parameter 6.1 bit 3. Time 1 value is set to 10ms (0.01 seconds) and block 2 at 200ms (0.2 seconds). The timer block 1 outputs a 1 for 10ms followed a 2 for 200ms, cycling back to 1 and repeating endlessly.

Time1 is the initialization block, and it runs for one AP program scan which is 10ms. During each initialization cycle (output of block 1 = 1), blocks 3 & 4 select the values specified contained at their IN1 inputs and store this to the output destination (DS1 or DS2). The output is fed back into the Default input pin of the corresponding block such that when the Sel1 input = 0, they form a latch to hold the In1 value for the 200ms period.

Effectively, block 3 takes a snapshot of the current speed reference and block 4 holds the current delta speed during each 200ms window created by the timer in block 1. The subtraction of current speed minus previous speed is done in block 4. The result is that DS1 is constantly updated with the derivative of the speed every 200ms.

Testing with demo case

Download the supplied "*100ms derivative.dcparamsbak*" file using Drive Composer. The acceleration and deceleration time of the demo program is set to 18 seconds. With the default value of 1800 rpm maximum speed, the expected speed derivative is 1800 / 18 / 5 samples per seconds for a resultant value of 20 which should appear in DS1 when accelerating the drive. When the drive is decelerated back to zero speed, the resulting DS1 value will be -20. Changing the accel/decel rates will be reflected in the value shown in DS1. Should the application require faster or slower sampling, the constant in the Timer block Time2 can be changed as required.

Reference documents

This document references both the ACS880 firmware manual and Adaptative programming guide.

- 3AXD50000085967 – ACS880 primary control program Firmware manual (Rev W)
- 3AXD50000028574 – Application guide Adaptive programming (Rev C)