

1. Description

This option was developed for compressor and pump applications where problems are experienced due to both sags and outages being processed the same way. Previously the DPI ride through timer had to be set up to deal safely with the outages and if a sag occurred it would be treated as if it were an outage. This resulted in a short ride through time ($\pm 350\text{ms}$) that is correct for outages but not for sags where a longer ride through time could have safely been used. By applying the two level detection and timing scheme described below it is possible to maximize the ride through time and reduce the number of unnecessary shut downs due to momentary sags and outages.

The DPI is programmed with two level sensing and ride through times:-

Level 1 - The user adjustable settings for transfer level & timer are used.

Level 2 - A fixed timer set to 200ms triggered when supply drops to 30% of nominal.

The Level 2 values are fixed when programmed but may be specified when ordering. The values used in the generic version were chosen to work with most applications.

When the standard values are used the DPI will function on the Level 1 settings for sags down to 30% of nominal but for sags below 30% or an outage the DPI will drop the controls after 200ms or reset to normal operation if the supply recovers before time out.

A figure of 200ms was chosen because the back EMF from the motor will support the supply for some time during the interruption. A delay of around 150ms before the supply drops to 30% of nominal can be expected for a 200KW motor. If the delay of 150ms is added to the 200ms timer setting then the total delay before time out and the the controls are dropped is 350ms. This is considered safe for large pumps and compressors.

2. Specifications for standard version

Level 1:

Transfer level: 55 - 90% standard - user adjustable

Timer: 0.1 - 3.1 sec standard - user adjustable

Level 2 :

Transfer level : 30% fixed

Timer: 0.2 sec fixed

For use with: All models in 52S and 52L series

Preliminary