I^2 t Protection

Explanation & Examples

Application Note

agito

www.agito-akribis.com

Member of Akribis Systems group

Revision History

Version	Description	Date
1.0	Initial Release	1 March 2023

Contact Information

Manufacturer	Agito Akribis Systems Ltd., Member of Akribis Systems Group
Address	6 Yad-Harutsim St., P.O.Box 7172, Kfar-Saba 4464103
Telephone	+972-9-8909797
Website	www.agito-akribis.com

Copyright Notice

Products Rights

AGDx, AGCx, AGMx, AGAx, AGIx, and AGLx are products designed by Agito Akribis Systems Ltd. in Israel. Sales of the products are licensed to Akribis Systems Pte Ltd. under intercompany license agreement.

Agito Akribis Systems Ltd. has full rights to distribute above products worldwide.

Disclaimer

This product documentation was accurate and reliable at the time of its release.

Agito Akribis Systems Ltd. reserves the right to change the specifications of the product described in this manual without notice at any time.

Trademarks

Agito PCSuite is a trademark of Agito Akribis Systems Ltd..

Warranty

This product is warranted to be free of defects in material and workmanship and conforms to the specifications listed in this manual, for a period of 12 months from the shipment date from factory.

Contents

1 Introduction

1.1 Background

The I^2T protection feature is implemented in all Agito products. This protection is based on the principle that if we inject current to a motor which is as room temperature, the motor can withstand the peak current for a certain amount of time until it reaches the motor's critical temperature.

Although the protection is primarily meant for protecting the motor from reaching its maximal, the I^2T is also used for driver protection. Drivers can also maintain peak current for a short time. This is especially true in cases where the motor can reach a higher current than the driver current.

In addition, the motor temperature also depends on cooling, air flow and motion profile. If you want accurate temperature sensing, we recommend using a temperature sensor like PT100.

The I^2T algorithm limits the current that can be injected into the motor to the continuous current, based on an estimation of the motor coil's thermal load state. The motor current can remain higher than the continuous current for a certain amount of time before being limited to the continuous current. This time period depends on the difference between the motor current and the specified continuous current.

1.2 Scope

This application note seeks to introduce how the protection works, and how to configure its parameters in PCSuite.

A simulation in excel will be attached to this document to demonstrate how the protection works with different parameters.

For a demonstration on how I^2T works with different parameters, click here.

In this application note, AGM800 + AGA102 are used. Other Central-i and integrated controllers also support this function.

1.3 Physics basis and how it works

The connection between the current and the motor's temperature can be represented as follows:

$$Temp \propto Curr^2 * \left(1 - e^{-\frac{t}{\tau}}\right)$$

$t - time$

$\tau - time\ constant\ of\ the\ system$

This is based on the power input to the motor over time.

By inputting the continuous current of the motor – given that the current injection to the motor is constant for infinite time, starting from a cold system - the motor will reach its maximal temperature.

If we inject the motor with the peak current, we will reach the maximal temperature much

faster than by injecting the continuous current.

We can determine the equation for τ.

$$PeakCurr^2 * \left(1 - e^{-\frac{tpeak}{\tau}}\right) = ContinousCurr^2 \rightarrow$$

$$\tau = \frac{-tpeak}{\ln\left(1 - \frac{ContinousCurr^2}{PeakCurr^2}\right)}$$

Once we've solved for τ, we can implement the protection by substituting the contiuous current with the motor current in the previous formula, and tracking whether it's higher than the continuous current squared - as follows:

$$If\ \ Curr^2 * \left(1 - e^{-\frac{t}{\tau}}\right) > ContinousCurr^2\ then\ activate\ the\ protection$$

If this condition is met, the current will be limited to the continuous current.

1.4 Simulation

To visually see how the protection works, please refer to the next pictures:

MotorCurr – The current measurement that flows through the motor coils.

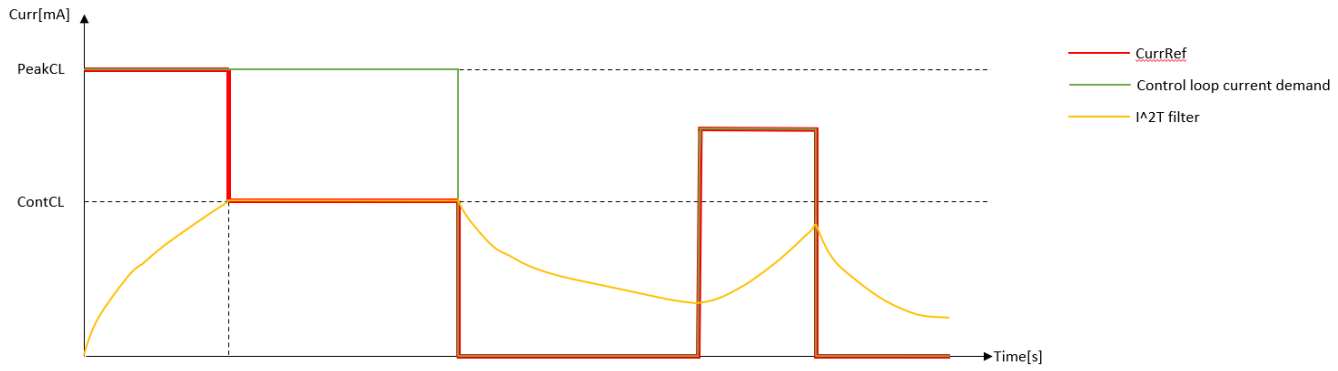

Figure 2. I^2T example of how the feature operates – Temperature simulation.

I^2T filter – The I^2T algorithm can be considered like a"filter" which fills up and activates the protection.

CurrRef – The current required for performing the required movement based on the motion profile.

PeakCL – The peak current allowed to flow through the motor's coils.

ContCL – The continuous current allowed to flow through the motor's coils.

PeakTime – The amount of time that the Peak current is allowed to flow through the motor.

Control loop current command – The current the control loop demand for the amplifier to output.

The I^2T "filter" fills up when the motor current squared is higher than the peak current squared , and once the filter reaches its maximal limit at peak time –the current is constrained to the continuous

current. After which, the filter empties when the current drops to zero or any current below the continuous current.

Afterwards, the “filter” can continue to fill up after it empties enough.

As the “filter” empties, the amount of time that the motor can be injected with the peak increases the emptier the “filter” is.

To check the value at which the protection is activated in your system, see the Excel simulation linked here.

As you can see in figure 3:
you can insert the required current for your application motion profile.

Type the continuous current and the peak current in the correct cells in the table, and type the amount of time you will allow for the peak current to be injected into the motor.

After inserting the parameters mentioned above, the simulation will display the time until the protection is activated.

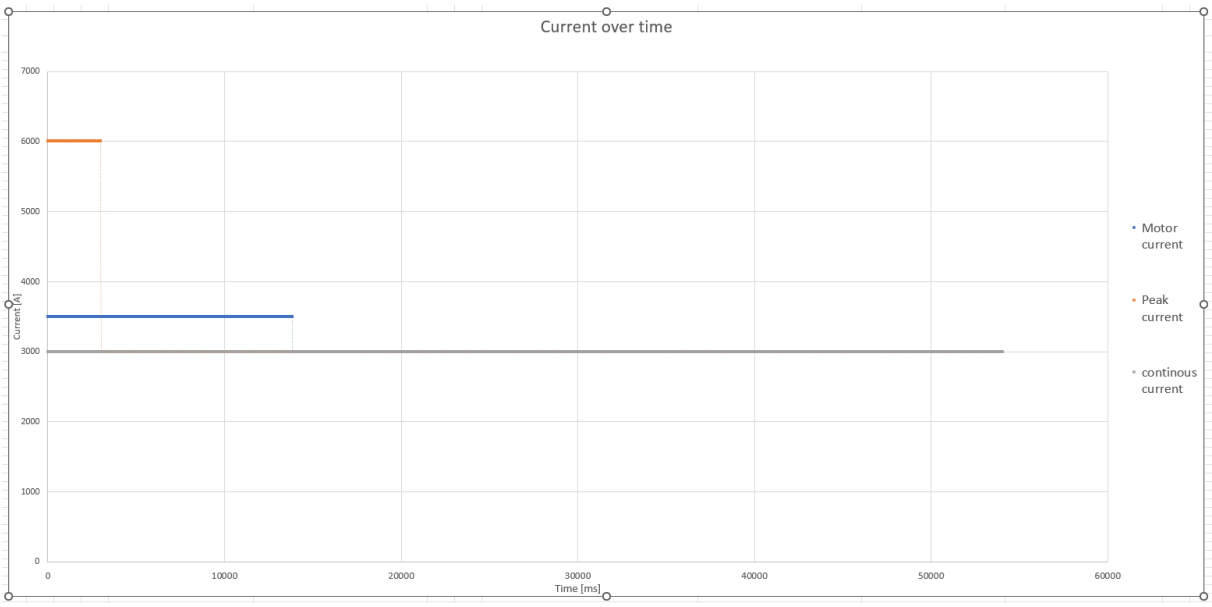

Figure 3. Excel simulation.

2 Setup

2.1 Configuration

COFIG 

/POW 

Set Continuous limitation to the motor's rated current.

Set Peak limitation to the motor's peak current.

Set the peak limitation time to the amount of time the motor is allowed to be at the maximum current to flow to the motor. Contact the motor manufacturer to get this information.

In Akribis motors the allowed time to remain in the peak current is 1 sec

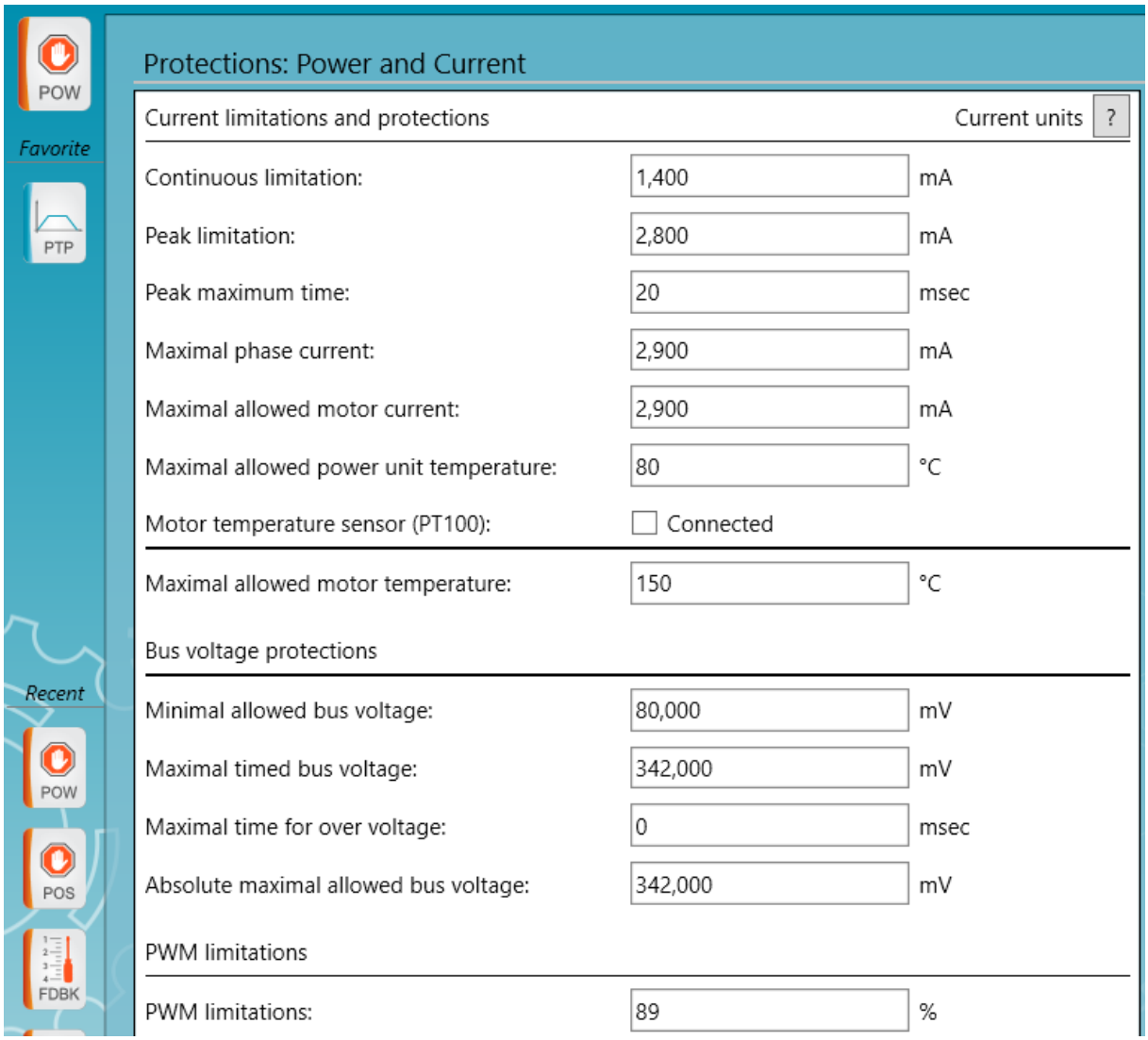

Figure 4. Configure the I^2T parameters.

The protection will be automatically implemented after you input the parameters mentioned above. In addition the protection **cannot be disabled**.

The I^2T is only a limitation, it is not a fault and will not turn off the motor. (It will not set MotorOn=0)

To monitor using PCSuite keywords, you can insert the command StatReg and check the bit 24 in the number it returns.

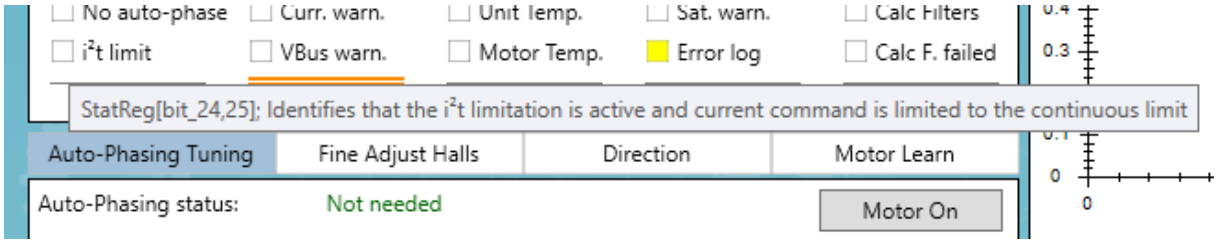

Figure 5.Statreg bit number that monitors the I^2T warning.

For FW version older than 3.0.5

If you want the motor to be turned off when the I^2T warning turns on, you can use the following user program (attached to this document) which can turn off the motor when the warning pops up, and presents a Confault representing the activation of the I^2T warning.

```
14  // Main task (Task 1)
15  // ------------------
16  //
17  // This is the start point of the program as it is
18  //
19  // Dynamic allocation ranges: [Tasks],[Functions],[
20
21  main ([10, 30], [5, 20], [800, 1000])
22      while (1)
23          if ((AStatReg >> 23)&1)
24              AMotorOn = 0
25              AWaitTime, 100
26              AConFlt =  - 17 // I^2T fault
27              break
28          end
29      end
30  endofmain
31
32
33
```

Position 6,318,495 Velocity: 0
Pos. error: 0 Motor current: -0.012 A
Status: No Motion Temperatures: 38 / NA / NA °C
Fault by user: -17 (User program)
Motion ended: Unexpected Motor Off (see MotorRea

Figure 6. I^2T user program to monitor the warning and turn off the motor.

For FW version newer than 3.0.5:

There is a checkbox to disable the motor when the I^2T protection is activated, otherwise it will behave the same as mentioned in the simulation.

You can access this checkbox through the keyword – ControlMode and setting the value in bit 3 to 1.

Figure 7. I^2T disable motor checkbox tooltip.

Figure 8. I^2T disable motor checkbox.

2.2 Real example

In this example, the continuous current is set to 200 mA and the peak current to 300 mA with a peak time of100 ms.

I ran the motor in jog mode and recorded the motor current and the current reference.

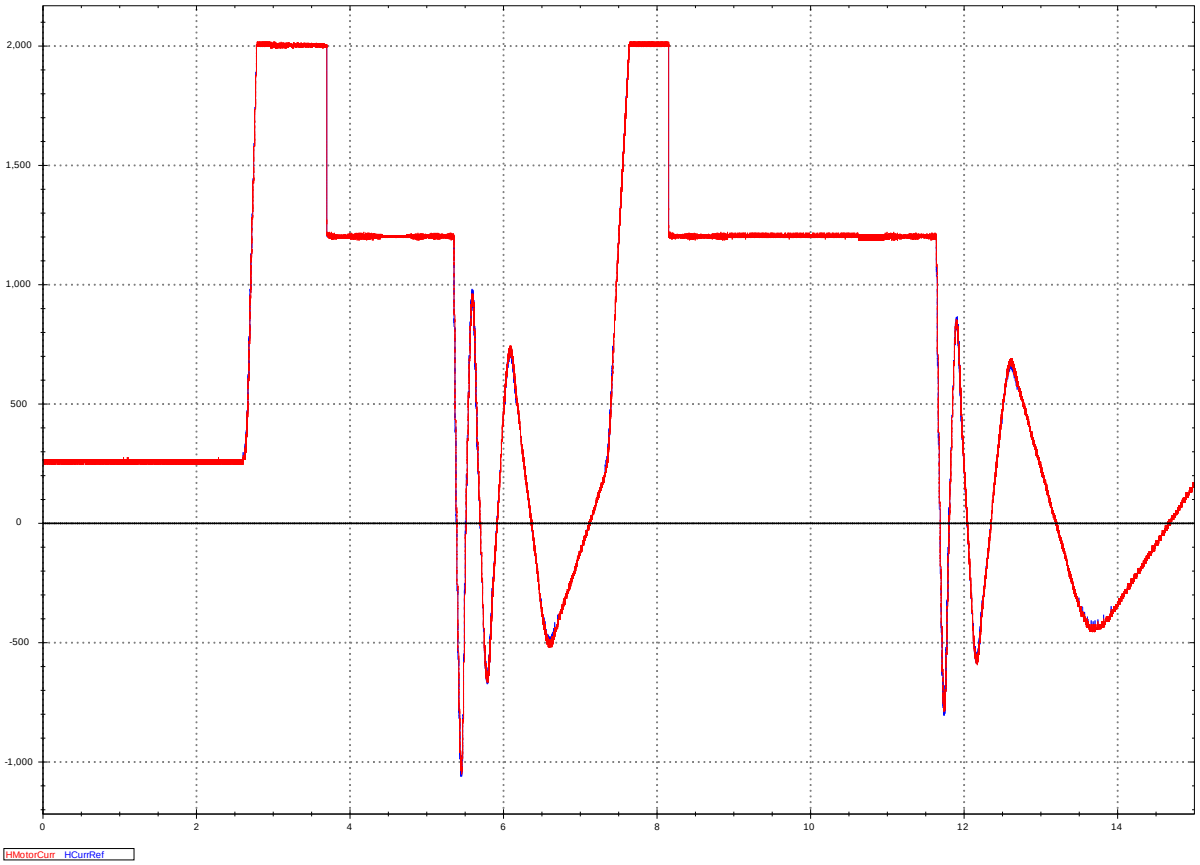

Figure 7. I^2T example recorded.

As you can see the current is bounded to 200mA after the motor current stays at 300mA for 100ms, and after the current rises, the filter empties and then refills.

In addition, when the protection is activated the controller will alert that it is active, and an LED in PCSuite will be lit, as can be seen in Figure 8.

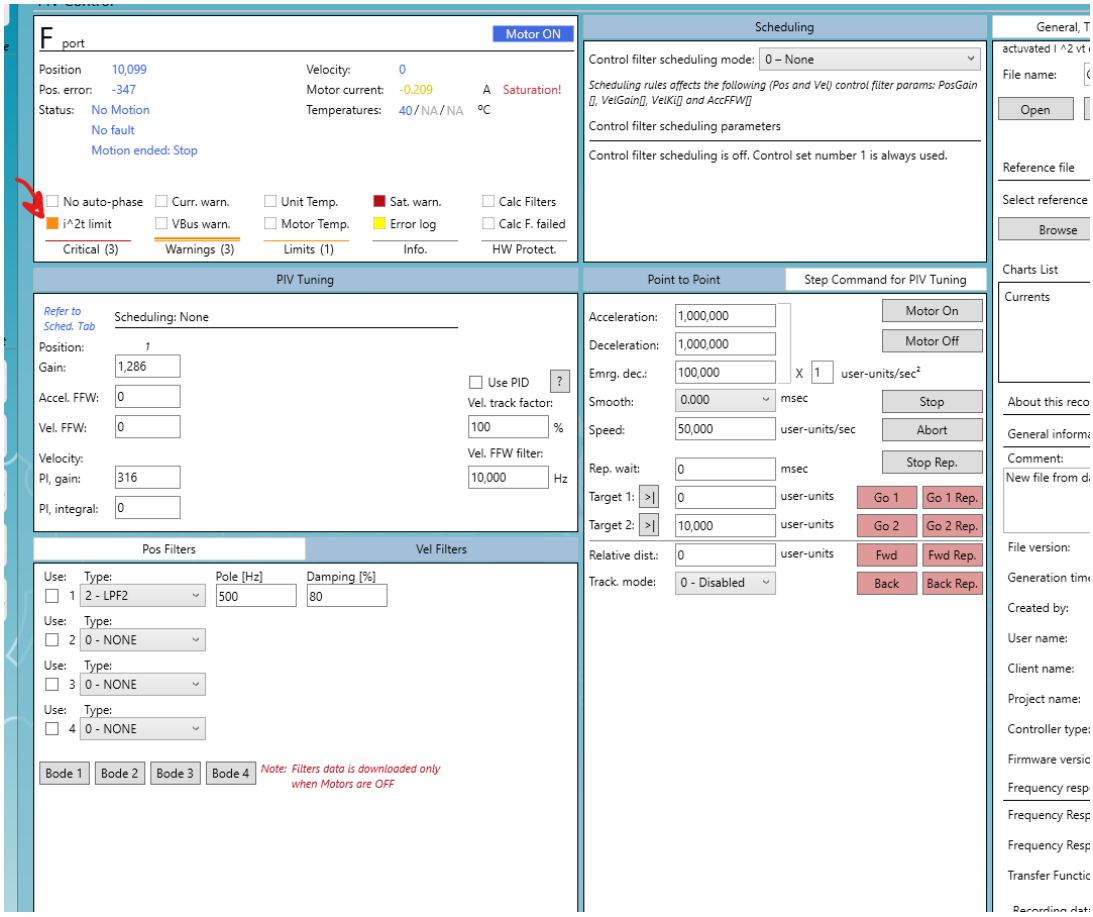

Figure 8. I^2T LED activated when the protection is active.