PMU-16

POWER MANAGEMENT UNIT

TECH NOTE

Improved Power Control Model for PMU16.

Revision 1.0

1. Introduction.

Some devices, such as electric motors, draw much more current during startup than they do when they are operating under steady-state conditions. This additional power consumption is referred to as inrush current.
The PMU16 allows inrush current to be higher than the current rating of the output (15A or 25A) for a very short time.

This way, for example, you can supply current through one 25A output to a device that draws and inrush current up to 55A for 3s, even though the device may only draw 10A during normal operating conditions. By allowing for inrush current, it is not necessary to use two outputs for this device.

2. Requirements.

PMU firmware: version 0.030.0 or later.

3. Power diagnostics in PMU16.

PMU16 is an intelligent power management system that constantly monitors the current through all output channels.
When power output to a channel exceeds predefined limits, that channel is turned off to prevent damage. Such a situation is referred to as Over Current in the PMU16 software.

PMU16 uses the following parameters to define limits of power outputs:
- **Inrush current** - if the current exceeds this limit, the output will be turned off immediately.
- **Max current** – operating current should remain below this threshold.
- **Inrush time** - an inrush current situation should not extend past this time interval.

If any one of these three conditions is met, the output goes into Over Current state and is turned off:
A. **Hard limit condition** - when the current exceeds *Inrush current*.
B. **Time-based condition** - when current above Max current is drawn longer than *Inrush time*.
C. **Heating model condition** - when emitted heat of the terminal exceeds 100% of allowed maximum (emitted heat is simulated by heating model)

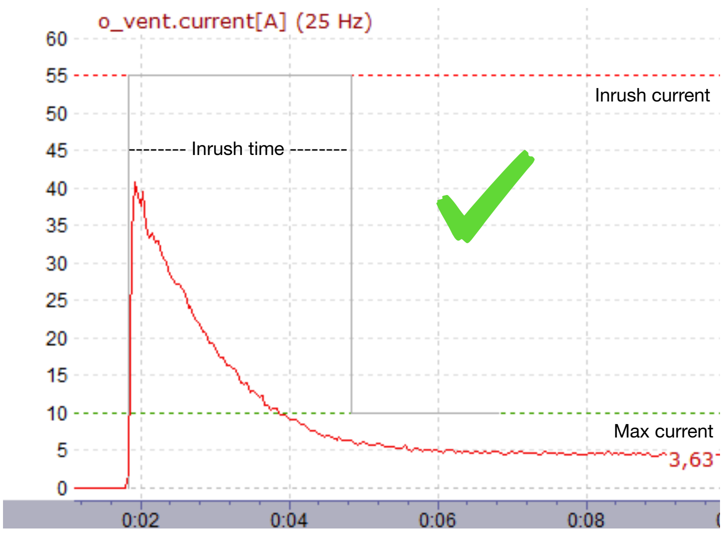


Fig1. Sample current trace and parameters

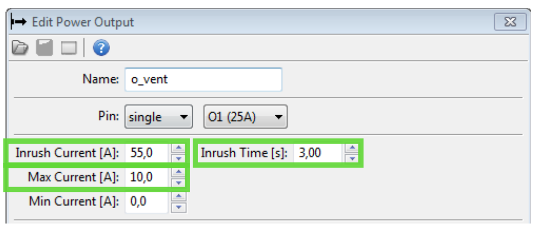


Fig2. Defining Power Output parameters in PMU16

4. Detailed explanation of all Over Current conditions.

A. Hard limit condition.

When current exceeds the *Inrush current* limit, the output goes into Over Current state and is turned off.

For the example below, defined *Inrush current* is too small. The current trace does not fit below the *Inrush current* limit. The *Inrush current* parameter should be increased to allow the device to work without interruption.

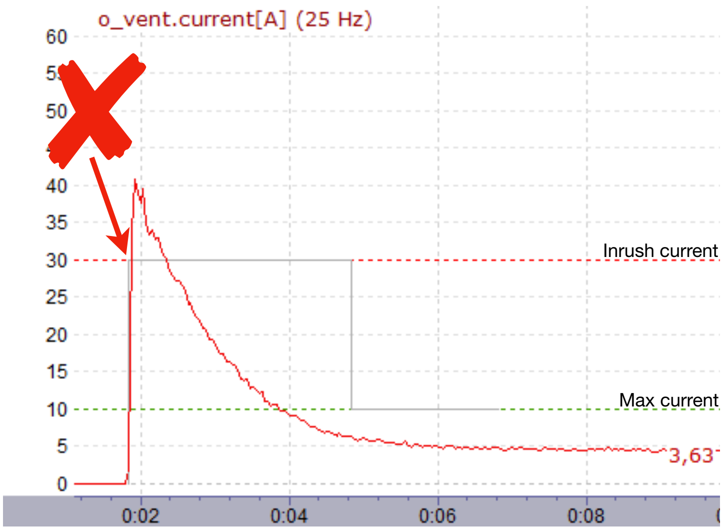


Fig3. Over Current by exceeding Inrush current

(This figure is NOT captured from PMU16 GraphLog. On a real PMU16, the output will be turned off immediately after exceeding 30A.)

B. Time-based condition.

When current exceeds the *Max current* value for longer than the *Inrush current* period, the output goes into Over Current state and is turned off.

For the example below, the defined *Inrush time* is too short. The current trace exceeds the *Max current* value. The *Inrush time* parameter should be increased to allow the device to work without interruption.

The time above *Max current* can be monitored on timeAboveMax channel (eg. o_heat.timeAboveMax).

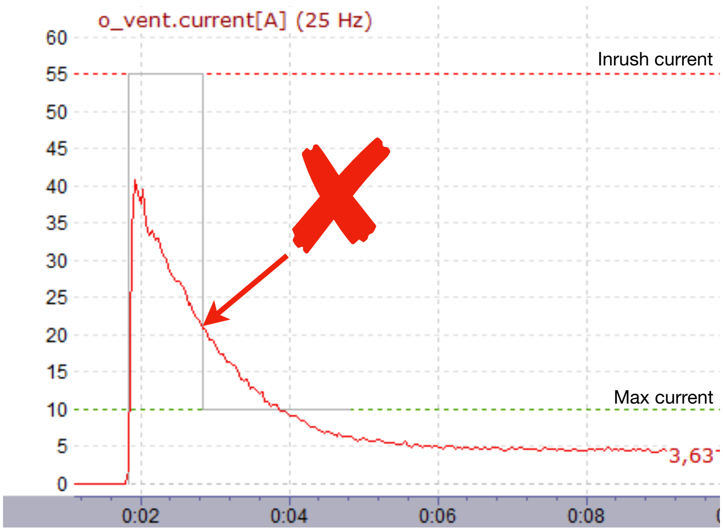


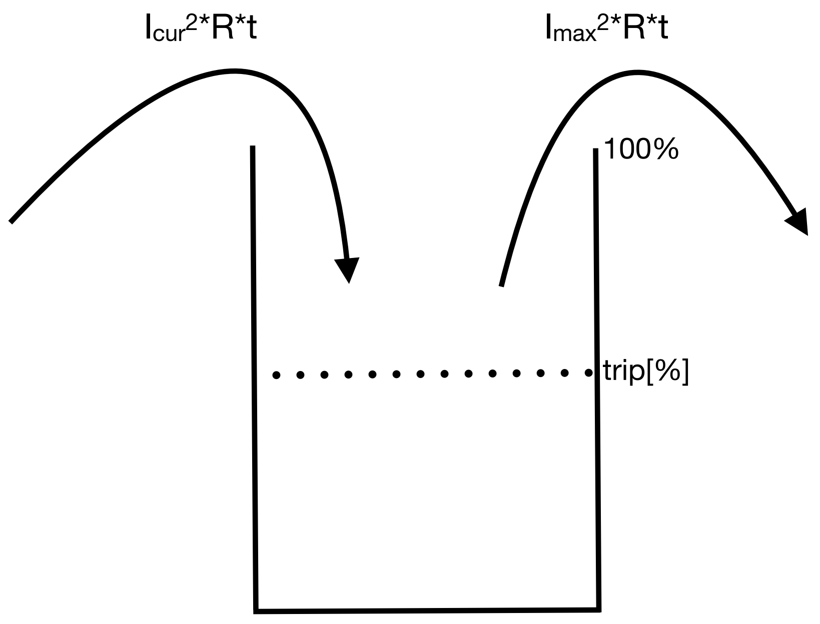
Fig4. Over Current by exceeding Inrush time

(This figure is NOT captured from PMU16 GraphLog. On a real PMU16, the output will be turned off 1 second after exceeding *Max current.*)

C. Heating model condition.

The heating model is used to prevent overheating caused by repeated inrush current conditions (current exceeding and falling below the *Max current* value multiple times).

Imagine that the heat generated by each output channel on the PMU16 device is modeled as a container. The PMU16 is designed to dissipate heat over time, but the total amount of heat generated by the device cannot be larger than the amount of heat the device is able to dissipate, or the device will overheat. With this heating model, each time current exceeds the *Max current* value, heat is added to the container. When the total current is less than *Max current* value, the amount of heat in the container will lessen as the heat is dissipated. The normalized level of heat in this virtual container is called the **trip** level in the PMU16 software and is measured as a percentage ([%]). When the *trip* parameter exceeds 100%, the output goes into Over Current state and is turned off.



The amount of heat arriving into the container is based on Joule's law:

$$Q = I_{cur}^2 * R * t$$

where ***Q*** is the amount of heat, ***Icur*** is electric current, ***R*** is some constant resistance and ***t*** is an amount of time.

The total capacity (**Qmax**) of the container is roughly equal to:

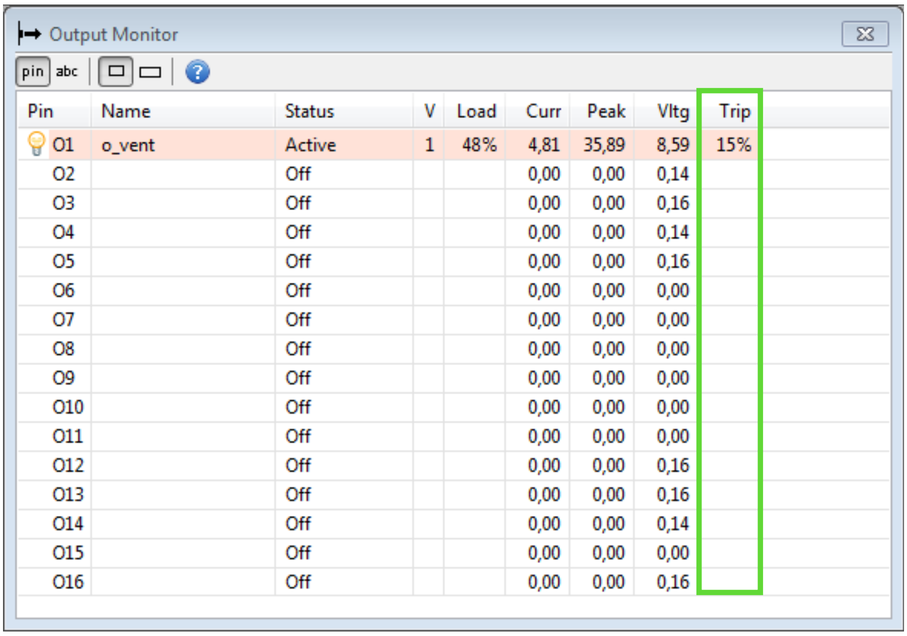
$$Q_{max} = I_{inrush}^2 * R * t_{inrush}$$

And the **trip** is normalized value:

$$trip = 100\% * Q / Q_{max}$$

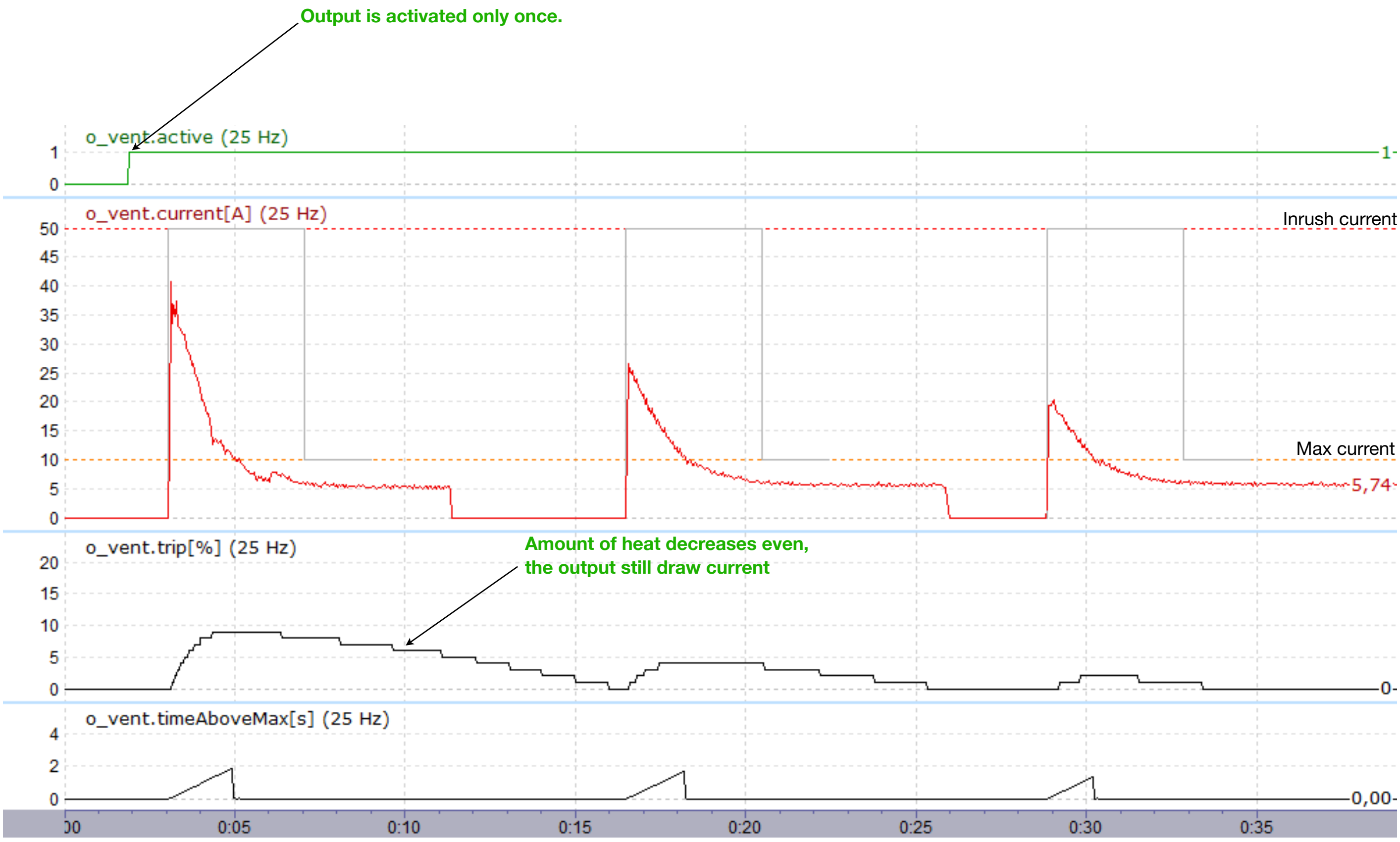
Increasing *Inrush current* will increase Q_{max}.
Increasing *Inrush time* will increase Q_{max}.

The **trip** parameter of the output can be monitored on the Output Monitor:



Pin	Name	Status	V	Load	Curr	Peak	Vltg	Trip
O1	o_vent	Active	1	48%	4,81	35,89	8,59	15%
O2		Off			0,00	0,00	0,14	
O3		Off			0,00	0,00	0,16	
O4		Off			0,00	0,00	0,14	
O5		Off			0,00	0,00	0,16	
O6		Off			0,00	0,00	0,00	
O7		Off			0,00	0,00	0,00	
O8		Off			0,00	0,00	0,00	
O9		Off			0,00	0,00	0,00	
O10		Off			0,00	0,00	0,00	
O11		Off			0,00	0,00	0,00	
O12		Off			0,00	0,00	0,16	
O13		Off			0,00	0,00	0,16	
O14		Off			0,00	0,00	0,14	
O15		Off			0,00	0,00	0,00	
O16		Off			0,00	0,00	0,16	

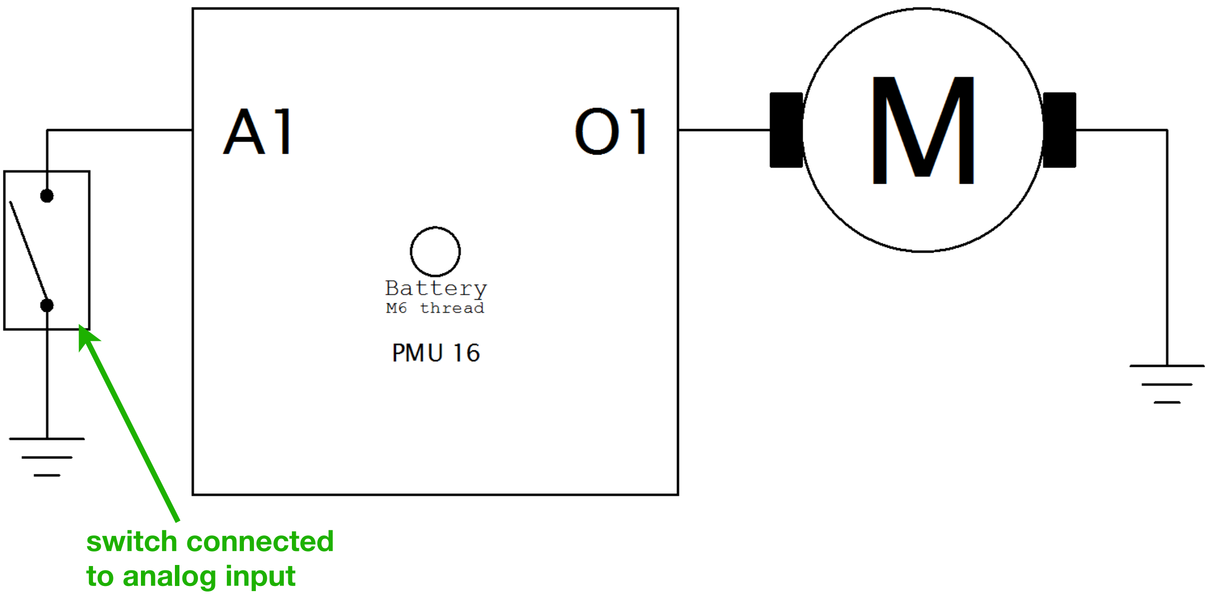
5. Graph Log example of output controlled by PMU16 Firmware FW 0.030.



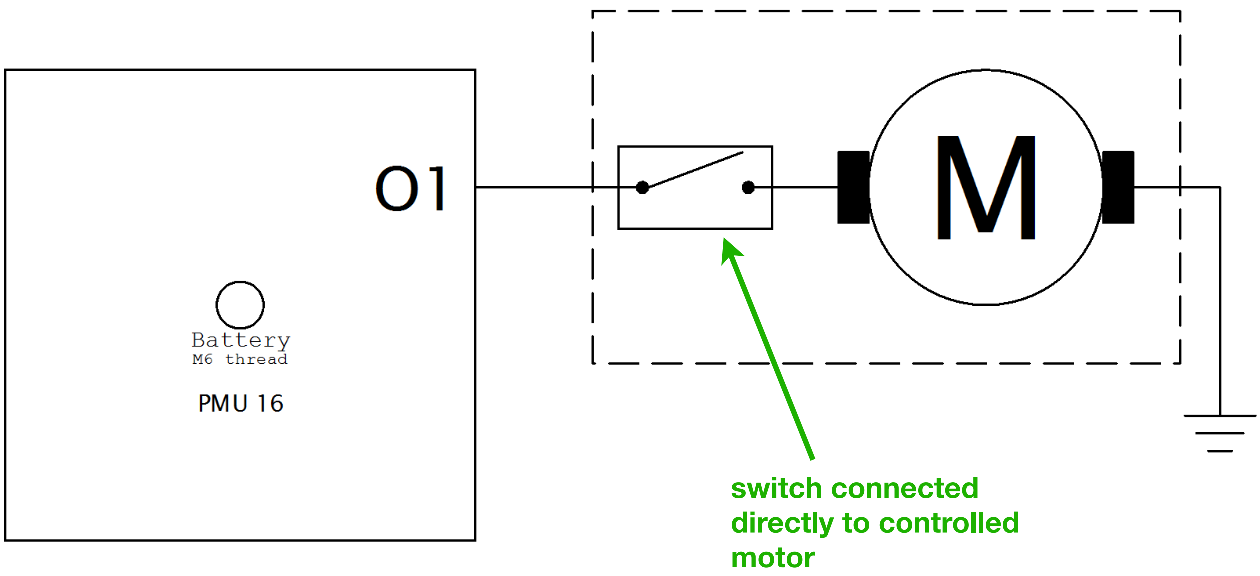
6. Explanation for modeling improvements in FW 0.030.

There are two changes In PMU16 firmware FW 0.030:
1. The heating model condition (C) has been added.
2. Also, we have changed when the timer starts counting for time-based conditions (B). Before FW 0.030, the timer started counting against Inrush time immediately after turning on the output. In FW 0.030 this timer starts counting after exceeding *Max current*.

The prior model works well for configuration with an output controlled by switch connected to PMU16 through an analog input. See configuration below:



There are some use cases that require the improved model from FW 0.030.
One of these examples is powering a motor through a switch, as shown below.
Another real-world example is an ABS system in a car. When the car starts, the ABS unit draws a small amount of current (about 1A). During emergency braking, the current can exceed the current rating of a single output of the PMU16 (25A). With the new model, both these use cases can be implemented with the PMU16 using a single output.



7. Summary.

With firmware 0.030 the power monitoring model of the PMU16 has been generalised to support more use cases. All setups from previous versions should work without any modification. There are no additional options in new model that need to be configured.