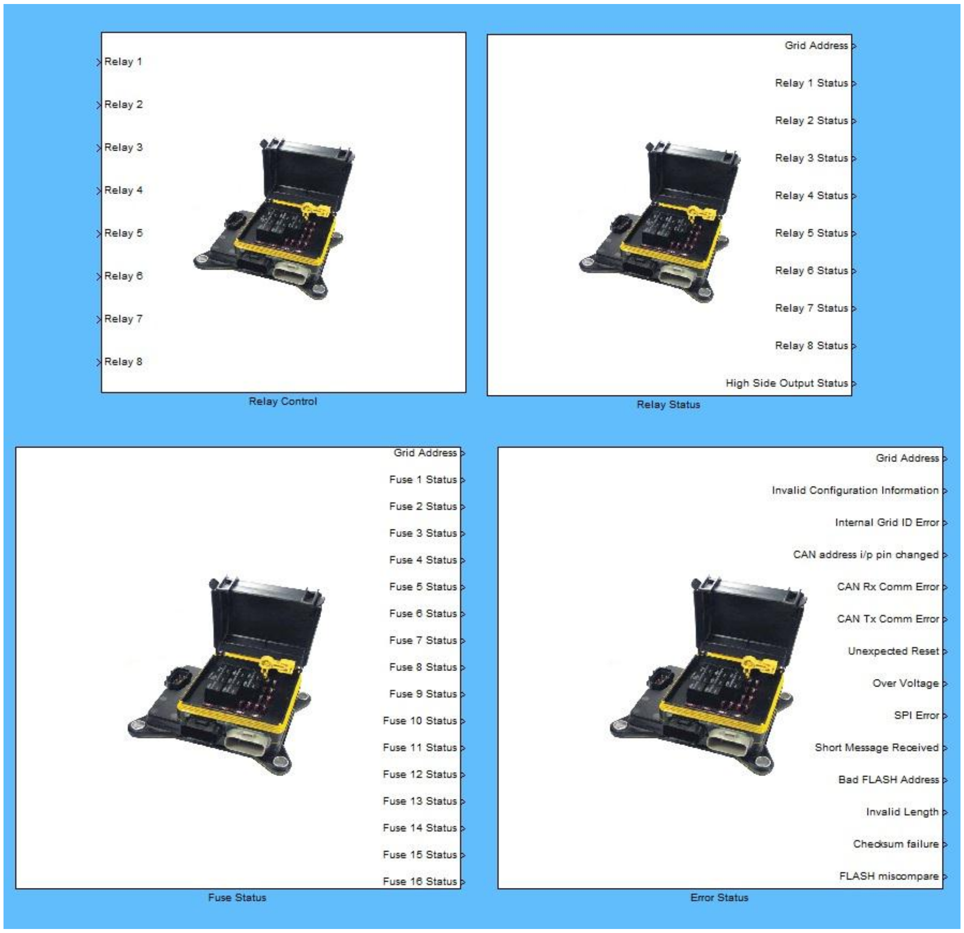
MotoHawk library for Multiplexed Power Distribution Module (or mVEC)

Overview

The objective of this document is to provide some information for getting started with using the MotoHawk CAN library for mVEC. The library contains 4 messages that the system is going to need most likely to function. There are several other CAN messages that you could use to configure, customize and query the mVEC. For more details on those, please refer to the section on CAN in mVEC user manual.



The library

More likely than not, you would have received a zipped file containing a few MATLAB files, a picture and a model file. These are 4 CAN messages written using MotoHawk that allow you to control each of the individual relays and also receive some system status and diagnostic information. Here’s the list:

- **mVEC_error_status.m**

This message is sent by the mVEC every time a system error occurs, or when there is a specific J1939 request message from an external module to obtain System Error Status information.

- **mVEC_fuse_status.m**
  The mVEC sends this message indicating the fault state of its fuses once every 1000 ms or every time the state of a fuse is changed (upto once every 25 ms).

- **mVEC_relay.m**
  This message is used to change the state of relays. The mVEC does not respond to this message.

- **mVEC_relay_status.m**
  The mVEC sends this message indicating the fault state of its relays once every 1000 ms or everytime the state of a relay is changed (no more than once every 25 ms).

- **mVEC_relay.jpg**
  This picture is called by the mask on the library blocks so it should be in the same folder as the other files, unless you would like to change it or remove the mask.

- **ne_mVEC_lib.mdl**
  This is the main library that contains 4 blocks for sending/receiving CAN messages to/from mVEC.

- **slblocks.m**
  This file configures the library blocks to show in Simulink's Library Browser

Saving the files on your computer

Once you have unzipped the folder, you have two options to access it from your MotoHawk model.

1. Place the folder under the 'Blocks' folder of your MotoHawk installation directory. Once you do that, the library will appear in your Matlab Simulink Library Browser with the name 'New Eagle mVEC Library'. The slblocks.m file is there to do just that. Now you can add the blocks to your model just like any other MotoHawk or Simulink block.
2. If you're already working on a MotoHawk project, you could place the folder under the 'Libraries' folder in your project directory. The library would added to Matlab path once you open (or if already open, close out and reopen) your model. The *<model_name>_setup.m* file in your project directory does that. In order to add blocks to your model, you could open the .mdl file in the folder (ne_mVEC_lib.mdl) and drag the desired block over to your model.

Points to note

Initial steps

As soon as you get the block into your model, the first thing to do would be specifying the CAN bus on which it would be installed. In your model, that would translate into looking under the mask of the block you just added to the model and changing the Name parameter to the name of CAN bus (eg. CAN_1, CAN_2 etc.). Also, since the mVEC works with 250 kBaud only, please make sure the CAN bus you're trying to use it on has been defined that way.

Use of multiple mVECs on one system

The CAN messages have been written keeping in mind some of the default settings on the mVEC. The default base source address of the module is set to 0xB0 and default source address offset is 0x00. So the source address (= source address base + source address offset) is also 0xB0 by default. This is reflected in the CAN msg IDs as this module uses J1939 protocol in which source address appears as a part of the CAN msg ID. In order to use more than one mVEC on one CAN bus, you would need to change the source address of each mVEC. Please refer to section 7.1.2 in the manual (page 31) to read about that. As a consequence, you may need to maintain a separate set of m-files that define the CAN message for each unit you use on the bus because the msg IDs would change.

CAN Message Information

Relay Command

This message would be sent to change the state of relays individually. The message currently is written to control upto 8 relays (for the 8 relay MPDM) but could easily be extended to be used with a 12 relay configuration. (Please refer to manual Page 41 for more information about that or contact us at support@neweagle.net)

Each of the inputs Relay 1 through 8 is a 2 bit value that goes from 0 through 3. Below is a table that describes each possible command:

Bit Value	Hex Value	Action
00	0	Turn Relay Off
01	1	Turn Relay On
10	2	Do not change relay state
11	3	Do not change relay state

The 'do not change' relay states are used when multiple modules are controlling the same mVEC to enable you to leave the state of some relays unchanged while changing the state of others with the same message.

Fuse Status

The status of fuses is transmitted by the mVEC every 1000 ms in this message or every time the state of a relay, fuse or error is changed (upto once every 25 ms). It's a 8 byte long message and each fuse status is a 2 bit status in which the meaning of 2 bits is as follows:

Bit Value	Hex Value	Meaning	Option to Disable?
00	0	No Fault	No
01	1	Blown	No
10	2	Not Powered	Yes
11	3	Not Used	No

Relay Status

The status of fuses is transmitted by the mVEC every 1000 ms in this message or every time the state of a relay, fuse or error is changed (upto once every 25 ms). It's a 8 byte long message and each fuse status is a 4 bit status in which the meaning of 4 bits is as follows:

Bit Value	Hex Value	Meaning	Option to Disable?
0000	0	Okay	No
0001	1	Relay Coil Open or Relay Not present	No
0010	2	Coil shorted or failed relay driver	No
0011	3	Normally Open (N.O) contact is open (when a N.O contact is not connected to Common terminal, but should be)	No
0100	4	Normally Closed (N.C) contact is open (when a N.C contact is not connected to the Common terminal, but should be)	No
0101	5	The coil is not receiving power	Yes
0110	6	Normally Open (N.O) contact is shorted (when a N.O contact is connected to the Common terminal, but should not be)	Yes
0111	7	Normally Closed (N.C) contact is shorted (when a N.C contact is connected to the Common terminal, but should not be)	Yes
1000	8	Reserved	No
1001	9	Reserved	No
1010	A	Reserved	No
1011	B	High Side Driver is reporting fault condition	No
1100	C	High Side Driver has an open load	Yes
1101	D	High Side Driver has an over voltage	No
1110	E	Reserved	No
1111	F	Relay location not used	No

System Error Status

The system error status message contains different types of system errors that may cause the MPDM to malfunction. Each fault is a bit, when set, meaning the fault has occurred. Below is a table describing some of the corrective actions that you may try to clear the fault.

Fault	Corrective Action
Invalid Configuration Information	mVEC must be reconfigured by manufacturer
Internal Grid ID Error	mVEC must be serviced by manufacturer
CAN address i/p pin changed	Check harness connections, if no result, contact us at support@neweagle.net
CAN Rx Comm Error	Adjust CAN message count threshold, check harnesses that are sending mVEC messages
CAN Tx Comm Error	Cycle system power, check termination on CAN
Unexpected Reset	Check power and ground connections on CAN connector
Over Voltage	Batt + is higher than 43 V. Reduce Voltage.
SPI Error	Internal Error
Short Message Received	Erase Region command incomplete. Check host application
Bad FLASH Address	Invalid address specified for Erase region or Write Memory command
Invalid Length of data	Invalid data length specified by Write Memory command
Checksum Failure	Invalid Checksum for received data for Write Memory command.
FLASH miscompare	FLASH data doesn't match received data after Write Memory command.