```
/**
 1
 2
 3
        @copyright © 2010 - 2020, Fraunhofer-Gesellschaft zur Foerderung der
 4
      * angewandten Forschung e.V. All rights reserved.
 5
 6
      * BSD 3-Clause License
 7
      * Redistribution and use in source and binary forms, with or without
 8
      * modification, are permitted provided that the following conditions are met:
 9
      * 1. Redistributions of source code must retain the above copyright notice,
10
            this list of conditions and the following disclaimer.
11
       2. Redistributions in binary form must reproduce the above copyright
            notice, this list of conditions and the following disclaimer in the
12
            documentation and/or other materials provided with the distribution.
13
14
      * 3. Neither the name of the copyright holder nor the names of its
            contributors may be used to endorse or promote products derived from
15
16
            this software without specific prior written permission.
17
18
      * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
19
      * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
20
      * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
21
      * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE
22
      * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
23
      * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
24
      * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
25
      * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
26
      * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
27
      * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
28
      * POSSIBILITY OF SUCH DAMAGE.
29
      * We kindly request you to use one or more of the following phrases to refer
31
      * to foxBMS in your hardware, software, documentation or advertising
32
      * materials:
33
34
      * ″ This product uses parts of foxBMS&req; ″
35
36
      * ″ This product includes parts of foxBMS&req; ″
37
38
      * ″ This product is derived from foxBMS&req; ″
39
40
      */
41
     /**
42
43
     * @file
               bms cfq.h
      * @author foxBMS Team
44
     * @date
                21.09.2015 (date of creation)
45
46
     * @ingroup ENGINE CONF
      * @prefix BMS
47
48
49
      * @brief bms driver configuration header
50
51
52
     #ifndef BMS CFG H
```

```
53
     #define BMS CFG H
 54
     /*======== Includes ======*/
 55
 56
     #include "general.h"
 57
 58
     #include "contactor.h"
 59
 60
     /*====== Macros and Definitions ==========*/
 61
 62
     #define BMS_REQ_ID_NOREQ 0
 63
     /**
 64
      * @ingroup CONFIG_BMS
 65
 66
      * this is the ID that should be requested via CAN signal to go to STANDBY state (ready, but no contactors closed)
 67
      * \par Type:
      * int
 68
 69
      * \par Default:
 70
 71
     * \par Range:
 72
     * 0<=x
 73
     * /
 74
     #define BMS_REQ_ID_STANDBY 8
 75
 76
     /**
 77
     * @ingroup CONFIG_BMS
78
      * this is the ID that should be requested via CAN signal to go to NORMAL state (contactors closing procedure)
 79
      * \par Type:
 80
      * int
      * \par Default:
 81
 82
 83
     * \par Range:
 84
      * 0<=x
 85
     #define BMS REQ ID NORMAL 3
 86
 87
     /**
 88
 89
     * @ingroup CONFIG_BMS
      * this is the ID that should be requested via CAN signal to go to CHARGE state (contactors closing procedure)
 90
 91
      * \par Type:
 92
      * int
 93
      * \par Default:
 94
      * 4
      * \par Range:
 95
 96
     * 0<=x
 97
     * /
98
     #define BMS_REQ_ID_CHARGE 4
99
                             Add Engine request, which is
100
                             #define BMS_REQ_ID_ENGINE 5
101
102
     #define BMS_BAL_NO_REQUEST 0
103
     /**
104
```

```
* @ingroup CONFIG BMS
105
106
       * this is the ID that should be requested via CAN to deactivate balancing with override
107
       * \par Type:
108
       * int
109
       * \par Default:
110
111
      * \par Range:
112
      * 0<=x
      * /
113
114
      #define BMS_BAL_INACTIVE_OVERRIDE 1
115
      /**
116
      * @ingroup CONFIG_BMS
117
118
       * this is the ID that should be requested via CAN to go out of override mode
119
      * \par Type:
120
      * int
      * \par Default:
121
122
123
      * \par Range:
124
      * 0<=x
125
      * /
126
      #define BMS_BAL_OUT_OF_OVERRIDE 2
127
128
      /**
129
      * @ingroup CONFIG_BMS
      * this is the ID that should be requested via CAN to activate balancing with override
130
131
       * \par Type:
132
      * int
      * \par Default:
133
134
      * \par Range:
135
136
      * 0<=x
137
      */
138
      #define BMS BAL ACTIVE OVERRIDE 3
139
140
141
      /**
142
      * BMS statemachine short time definition in ms
143
144
      #define BMS STATEMACH SHORTTIME MS 1
145
146
      /**
147
      * BMS statemachine medium time definition in ms
148
149
      #define BMS STATEMACH MEDIUMTIME MS 5
150
      /**
151
152
      * BMS statemachine long time definition in ms
153
154
      #define BMS_STATEMACH_LONGTIME_MS 100
155
      /**
156
```

```
* BMS statemachine very long time definition in ms
157
158
159
     #define BMS_STATEMACH_VERYLONGTIME_MS 2000
160
161
     /**
162
     * @ingroup CONFIG_BMS
163
164
      * \par Type:
     * int
165
      * \par Default:
166
167
     * 10
168
     * \par Range:
     * [5,15]
169
     * \par Unit:
170
     * 10*ms
171
172
     * /
173
     #define BMS_SELFPOWERONCHECK_TIMEOUT 10 /* 100ms */
174
175
     /**
176
     * @ingroup CONFIG BMS
177
     * \par Type:
178
     * int
     * \par Default:
179
180
     * 10
181
     * \par Range:
     * [5,15]
182
183
     * \par Unit:
     * 10*ms
184
     */
185
     186
187
188
189
     /**
190
     * @ingroup CONFIG BMS
191
      * \par Type:
192
     * int
     * \par Default:
193
194
     * 50
195
     * \par Range:
196
     * [40,60]
197
     * \par Unit:
     * 10*ms
198
199
     #define BMS_IDLE_TIMEOUT 500 /* 5s timeout to go to sleep or power off in idle state */
200
201
202
     #define BMS_GETSELFCHECK_STATE() BMS_CHECK_OK
                                                                 /* function could return: BMS_CHECK_NOT_OK or OK
     BMS CHECK BUSY */
203
     #define BMS GETPOWERONSELFCHECK STATE() BMS CHECK OK
                                                                /* function could return: BMS CHECK NOT OK or OK
     BMS_CHECK_BUSY */
204
     #define BMS_CHECKPRECHARGE()
                                            BMS_CheckPrecharge() /* DIAG_CheckPrecharge() */
205
     /*
206
```

```
* Mapping the marcos from the contactor-module to
207
       * bms-macros.
208
                              The following are not used and should be commented out and then removed at a later time.
209
       * /
210
      #define BMS ALL CONTACTORS OFF()
                                             CONT_SwitchAllContactorsOff()
211
212
      #define BMS_CONT_MAINMINUS_ON()
                                             CONT_SetContactorState(CONT_MAIN_MINUS, CONT_SWITCH_ON)
213
      #define BMS_CONT_MAINMINUS_OFF()
                                             CONT_SetContactorState(CONT_MAIN_MINUS, CONT_SWITCH OFF)
214
215
      #define BMS CONT MAINPRECHARGE ON()
                                             CONT SetContactorState (CONT PRECHARGE PLUS, CONT SWITCH ON)
216
      #define BMS_CONT_MAINPRECHARGE_OFF()
                                             CONT_SetContactorState(CONT_PRECHARGE_PLUS, CONT_SWITCH_OFF)
217
                                             CONT_SetContactorState(CONT_MAIN_PLUS, CONT SWITCH ON)
218
      #define BMS CONT MAINPLUS ON()
219
      #define BMS_CONT_MAINPLUS_OFF()
                                             CONT_SetContactorState(CONT_MAIN_PLUS, CONT_SWITCH_OFF)
220
221
      #if BS SEPARATE POWERLINES == 1
222
      #define BMS CONT CHARGE MAINMINUS ON()
                                                    CONT SetContactorState (CONT CHARGE MAIN MINUS, CONT SWITCH ON)
223
      #define BMS_CONT_CHARGE_MAINMINUS_OFF()
                                                    CONT_SetContactorState(CONT_CHARGE_MAIN_MINUS, CONT_SWITCH_OFF)
224
225
      #define BMS CONT CHARGE MAINPRECHARGE ON()
                                                    CONT SetContactorState (CONT CHARGE PRECHARGE PLUS, CONT SWITCH ON)
226
      #define BMS CONT CHARGE MAINPRECHARGE OFF()
                                                    CONT SetContactorState (CONT CHARGE PRECHARGE PLUS, CONT SWITCH OFF)
227
228
      #define BMS CONT CHARGE MAINPLUS ON()
                                                    CONT SetContactorState (CONT CHARGE MAIN PLUS, CONT SWITCH ON)
229
      #define BMS CONT CHARGE MAINPLUS OFF()
                                                    CONT SetContactorState (CONT CHARGE MAIN PLUS, CONT SWITCH OFF)
230
      #endif /* BS_SEPARATE_POWERLINES == 1 */
231
      /*----Function Prototypes ----*/
232
233
234
      /*====== Function Implementations ========*/
235
      #endif /* BMS CFG H */
236
237
```