

```

1  /*!
2  * @file      mode_load.c
3  *
4  * @brief      This file contains all the functions for the load LED
5  *
6  * @version    V1.0
7  *
8  * @date      2023-11-04
9  */
10 #include "mode_load.h"
11
12 struct PID_DATA    pidData;
13
14 extern Flag_Check      Flag;
15 Flag_Dm      Flag_Dming;
16 uint8_t SaveStt = 0;
17 uint8_t Check_Stt = 0;
18
19 /***** PID *****/
20 int16_t Val_Ret_PID = 0;
21 float K = 0;
22 float Cov_K = 0;
23
24 float temp_dischar = 0.0;
25
26 uint16_t Save_Temp = 0;           // luu gia tri nhiet do de so sanh voi nhiet do
lan tiep theo
27 uint16_t Update_DowCurr = 0;     // gia tri % se giam Duty khi nhiet do cao
28 uint16_t Curr_Def = 0;           // luu gia tri dong output read tu Flash
29 float Save_Imax = 0;             // luu gia tri dong max 100% -> Dimming
30
31 void Set_PID(void){
32     Init_PID(SCALING_FACTOR*K_P, SCALING_FACTOR*K_I, SCALING_FACTOR*K_D,
&pidData);
33 }
34 void Prt_LOAD(void){
35
36     UPower_DC;
37     TMR_SetCompare2(TMR3, 0);
38     Load_OFF;
39     K = 0;
40 //     Delay_ms(10000);
41     while(Re_Adc_LED >= 2100){           // func Check error LED = ~35V
42         TMR_SetCompare2(TMR3, 0);
43         K = 0;
44         Delay_ms(1000);
45         IWDT_Refresh();
46     }
47     Delay_ms(2000);
48     Load_ON;

```

```

49     if(Ctrl_Program()==1){UPower_Pin;}
50     Flag.Check_LED = Flag_ON;
51     IWDT_Refresh();
52 }
53 void Load_EnableOutputs(void){
54
55     Val_Ret_PID = I_Load - Re_Curr;
56     Cov_K = (float)Val_Ret_PID / 5000;
57     if (Re_Curr > (I_Load-10) && Re_Curr < (I_Load+10)){
58         K = K;
59     }else{
60         K += Cov_K;
61         if (K >= 160){
62             K = 160.00000;
63         } else if (K <= 0){
64             K = 0;
65         }
66         TMR_SetCompare2(TMR3, K);
67     }
68     while(Re_Adc_LED >= 3600){          // LED >= 60V
69         LED1_OFF;
70         Prt_LOAD();
71         return;
72     }
73
74     Fun_Dimming(0,
75     100,60,          // Time 1 duty,time
76     70,120,         // Time 2  "
77     80,135,         // Time 3  "
78     60,150,         // Time 4  "
79     90,Dimming);    // end
80
81     if(sysTick >= 5000){
82         if((temp_dischar = Re_TempNTC()) >= 95){
83             for(uint8_t DisLED=0; DisLED <=160; DisLED++){
84                 K--;
85                 if(K <= 0){
86                     K = 0;
87                     Load_OFF;
88                 }
89                 TMR_SetCompare2(TMR3, K);
90                 Delay(0x2fff);
91                 IWDT_Refresh();
92             }
93             while(1){
94                 IWDT_Refresh();
95                 if((temp_dischar = Re_TempNTC()) <= 85){
96                     Load_ON;
97                     break;
98                 }

```

```

99         LED1_OFF;
100        LED3_OFF;
101        Delay_ms(1000);
102        LED1_ON;
103        LED3_ON;
104        Delay_ms(1000);
105    }
106    }
107    sysTick = 0;
108    }
109
110    if(Re_Curr >= 3255){          // Current >= ~121W/24V -> protect load
111        I_Load = 1700;
112    }
113    IWDT_Refresh();
114}
115void Load_CheckShortCircuit(void){
116    K += 1;
117    if(K >= 100){    // Current =
118        K = 100;
119    }
120    TMR_SetCompare2(TMR3, K);
121    if(Re_Adc_LED >= 3600) {      // LED >= 60V
122        Prt_LOAD();
123        for(uint8_t i = 0; i<5; i++){
124            LED1_ON;
125            Delay_ms(60);
126            LED1_OFF;
127            Delay_ms(60);
128        }
129    }
130    else if(Re_Curr >= 1000) {
131        TMR_SetCompare2(TMR3, 0);
132        K = 0;
133        Flag.Check_LED = Flag_OFF;
134        Delay_ms(2000);
135    }
136    IWDT_Refresh();
137}
138void Load_Enable(void){
139
140    IWDT_Refresh();
141    Prt_LOAD();
142    SaveStt = Ctrl_Program();
143    Check_Stt = SaveStt;
144    I_Load = *(volatile uint32_t*)(0x08007000+28);
145    Curr_Def = I_Load;
146    Update_DowCurr = (10*Curr_Def)/100;
147    Save_Temp = 0;
148    sysTick = 0;

```

```

149     TMR_Enable(TMR14);
150
151     Dimming = 0;
152     Flag_Dming.Flag_Dm1 = Dm_OFF;
153     Flag_Dming.Flag_Dm2 = Dm_OFF;
154     Flag_Dming.Flag_Dm3 = Dm_OFF;
155     Flag_Dming.Flag_Dm4 = Dm_OFF;
156     Flag_Dming.Flag_Dm5 = Dm_OFF;
157
158     while(SaveStt == Check_Stt){
159
160         if(Flag.Check_LED == 1){
161             Load_CheckShortCircuit();
162         }
163         else if(Flag.Check_LED == 0){
164             LED1_ON;
165             Load_EnableOutputs();
166         }
167         Check_Stt = Ctrl_Program();    // save value return
168
169         while(Check_Stt == 3){
170             while(1){
171                 IWDT_Refresh();
172                 UPower_DC;
173                 Load_OFF;
174                 TMR_SetCompare2(TMR3, 0);
175                 if (Ctrl_Program() == 0){
176                     En_PWM_CHG();
177                     if(Re_Adc_BAT >= Recovery_Volt){    // Recovery Voltage
178                         Flag.Cov_ACDC = Flag_OFF;
179                     }
180                     return;
181                 }
182                 else if (Ctrl_Program() == 2){return;};
183
184                 GPIOA->ODATA ^= GPIO_PIN_15; //LED3
185                 Delay_ms(1000);
186             }
187         }
188         while(Check_Stt == 0){
189             Load_OFF;
190             TMR_SetCompare2(TMR3, 0);
191             En_PWM_CHG();
192             if(Re_Adc_BAT >= Recovery_Volt){    // Recovery Voltage
193                 Flag.Cov_ACDC = Flag_OFF;
194             }
195             return;
196         }
197     }
198 }

```

```

199 void Fun_Dimming(uint8_t Tim1, uint8_t SetP1, uint8_t Tim2, uint8_t SetP2, uint8_t
Tim3, uint8_t SetP3, uint8_t Tim4, uint8_t SetP4, uint16_t Tim5, uint8_t SetP5,
uint16_t Check_time){
200
201     if(Check_time >= Tim1 && Flag_Dming.Flag_Dm1 == 0){
202         Save_Imax = Val_CovCurr;
203         I_Load = (((SetP1*Save_Imax)/100)+2.055683594)/3*4096/3.3;
204         //((I_Load*3.3/4096)*3-2.055683594
205         Flag_Dming.Flag_Dm1 = Dm_ON;
206     }
207     else if(Check_time > Tim2 && Flag_Dming.Flag_Dm2 == 0){
208         I_Load = (((SetP2*Save_Imax)/100)+2.055683594)/3*4096/3.3;
209         Flag_Dming.Flag_Dm2 = Dm_ON;
210     }
211     else if(Check_time > Tim3 && Flag_Dming.Flag_Dm3 == 0){
212         I_Load = (((SetP3*Save_Imax)/100)+2.055683594)/3*4096/3.3;
213         Flag_Dming.Flag_Dm3 = Dm_ON;
214     }
215     else if(Check_time > Tim4 && Flag_Dming.Flag_Dm4 == 0){
216         I_Load = (((SetP4*Save_Imax)/100)+2.055683594)/3*4096/3.3;
217         Flag_Dming.Flag_Dm4 = Dm_ON;
218     }
219     else if(Check_time > Tim5 && Flag_Dming.Flag_Dm5 == 0){
220         I_Load = (((SetP5*Save_Imax)/100)+2.055683594)/3*4096/3.3;
221         Flag_Dming.Flag_Dm5 = Dm_ON;
222     }
223     else return;
224 }
225 /**@} end of group Load_Functions */
226

```