

```

1  /* ***** */
2  /* */
3  /* Nome do arquivo:    aquecedorECooler.h */
4  /* */
5  /* Descricao:        Arquivo Header contendo a declaracao */
6  /*                   das funcoes de interface do microcontrolador */
7  /*                   com a resistencia de aquecimento e cooler do kit */
8  /* */
9  /* Autores:          Gustavo Lino e Giacomo Dollevedo */
10 /* Criado em:         21/04/2020 */
11 /* Ultima revisao em: 24/07/2020 */
12 /* ***** */
13
14 #ifndef SOURCES_COOLER_HEATER_
15 #define SOURCES_COOLER_HEATER_
16
17
18 /* ***** */
19 /* Nome do metodo:     PWM_init */
20 /* Descricao:          Inicializa os registradores para funcionamento do PWM */
21 /*                   entre 5 e 20Hz */
22 /* */
23 /* Parametros de entrada:  n/a */
24 /* */
25 /* Parametros de saida:   n/a */
26 /* ***** */
27 void PWM_init(void);
28
29
30 /* ***** */
31 /* Nome do metodo:     PWM_clearCounter */
32 /* Descricao:          Reseta o contador TPM1_CNT para nao haver overflow */
33 /* */
34 /* Parametros de entrada:  n/a */
35 /* */
36 /* Parametros de saida:   n/a */
37 /* ***** */
38 void PWM_clearCounter(void);
39
40 /* ***** */
41 /* Nome do metodo:     coolerfan_init */
42 /* Descricao:          Configura a liberacao do sinal PWM no pino PTA13 */
43 /* */
44 /* Parametros de entrada:  n/a */
45 /* */
46 /* Parametros de saida:   n/a */
47 /* ***** */
48 void coolerfan_init(void);
49
50
51 /* ***** */
52 /* Nome do metodo:     heater_init */
53 /* Descricao:          Configura a liberacao do sinal PWM no pino PTA12 */
54 /* */
55 /* Parametros de entrada:  n/a */
56 /* */
57 /* Parametros de saida:   n/a */
58 /* ***** */
59 void heater_init(void);
60
61 /* ***** */
62 /* Nome do metodo:     coolerfan_PWMDuty */
63 /* Descricao:          Configura o Duty Cycle do PWM para o cooler */
64 /* */
65 /* Parametros de entrada:  fCoolerDuty -> valor entre 0 e 1 indicando o Duty Cycle */
66 /* */

```

```
67  /* Parametros de saida:    n/a                               */
68  /* ***** */
69  void coolerfan_PWMDuty(float fCoolerDuty);
70
71  /* ***** */
72  /* Nome do metodo:        heater_PWMDuty                      */
73  /* Descricao:            Configura o Duty Cycle do PWM para o cooler */
74  /* ***** */
75  /* Parametros de entrada: fHeaterDuty -> valor entre 0 e 1 indicando o Duty Cycle */
76  /* ***** */
77  /* Parametros de saida:    n/a                               */
78  /* ***** */
79  void heater_PWMDuty(float fHeaterDuty);
80
81  #endif /* SOURCES_COOLER_HEATER_ */
```