

```

1  /* ***** */
2  /* File name:      DroneTimer.h */
3  /* File description: Timer interface header file */
4  /* Author name:     Giacomo Dollevedo, Gustavo Fernandes */
5  /* Creation date:    18nov2020 */
6  /* Revision date:    18dec2020 */
7  /* ***** */
8
9  #ifndef DroneTimer_h
10 #define DroneTimer_h
11
12
13 #include <Arduino.h>
14
15
16
17 class DroneTimer
18 {
19
20 public:
21
22  /*
23  *****
24  */
25  /* Method's name:      initTimer
26  */
27  /* Description:        Initialize hardware timer interruptions
28  */
29  /*
30  */
31  /* Entry parameters:    int frequency -> timer interrupt frequency
32  */
33  /*
34  */
35  /*
36  */
37  /* Return parameters:   n/a
38  */
39  /*
40  *****
41  */
42  void initTimer(int frequency, void (*fn)(void));
43
44
45  /*
46  *****
47  */
48  /* Method's name:      enableTimer
49  */
50  /* Description:        Enable timer interruptions
51  */
52  /*
53  */
54  /* Entry parameters:    n/a
55  */
56  /*
57  */
58  /*
59  */
60  /*
61  */
62  /*
63  */
64  /*
65  */
66  /*
67  */
68  /*
69  */
70  /*
71  */
72  /*
73  */
74  /*
75  */
76  /*
77  */
78  /*
79  */
80  /*
81  */
82  /*
83  */
84  /*
85  */
86  /*
87  */
88  /*
89  */
90  /*
91  */
92  /*
93  */
94  /*
95  */
96  /*
97  */
98  /*
99  */
100 /*
101 *****
102 */

```

```

40 /* Return parameters:      n/a
    */
41 /*
    ****
    */
42 void enableTimer();
43
44
45 /*
    ****
    */
46 /* Method's name:          disableTimer
    */
47 /* Description:            Disable timer interruptions
    */
48 /*
    */
49 /* Entry parameters:       n/a
    */
50 /*
    */
51 /* Return parameters:      n/a
    */
52 /*
    ****
    */
53 void disableTimer();
54
55
56 /*
    ****
    */
57 /* Method's name:          setFrequency
    */
58 /* Description:            Set the timer frequency
    */
59 /*
    */
60 /* Entry parameters:       int frequency -> frequency to be set
    */
61 /*
    */
62 /* Return parameters:      n/a
    */
63 /*
    ****
    */
64 void setFrequency(int frequency);
65
66
67 /*
    ****
    */
68 /* Method's name:          getFrequency
    */
69 /* Description:            Get the timer frequency
    */
70 /*
    */

```

```
71 /* Entry parameters:      n/a
    */
72 /*
    */
73 /* Return parameters:      int -> Timer frequency
    */
74 /*
    ****
    */
75     int getFrequency();
76
77
78 private:
79     /*Hardware timer for system loop time management*/
80     hw_timer_t * timer0 = NULL;
81
82     /*Default frequency to 10 Hz*/
83     int timerFreq = 10;
84
85 };
86
87 #endif
88
```