

```

1 #include "timer.h"
2
3
4
5 tUserTimer tim;
6
7 void SetupTimer(tUserTimer *t, uint32_t interval)
8 {
9     t->interval = interval;
10    t->flag = 0;
11    t->flag2 = 0;
12    t->event_cnt = 0;
13}
14
15 void StartTimer(tUserTimer *t)
16 {
17     t->target_tick = HAL_GetTick() + t->interval;
18     t->flag = 1;
19}
20
21 void StopTimer(tUserTimer *t)
22 {
23     t->flag = 0;
24}
25
26 void ClearTimer(tUserTimer *t)
27 {
28     t->event_cnt = 0;
29}
30
31 void TimerProcess(tUserTimer *t)
32 {
33     if (t->flag && HAL_GetTick() >= t->target_tick)
34     {
35         t->event_cnt++;
36         t->target_tick = t->target_tick + t->interval;
37     }
38}
39
40 uint32_t isTimerEventExist(tUserTimer *t)
41 {
42     return t->event_cnt;
43}
44
45 void User_Timer_Callback(void)
46 {
47     TimerProcess(&tim);
48     // Add additional timer processing if more user timers
49}

```

所定の時間経過するごとに t->event\_cnt をカウントアップする。

rc.c 内の関数 HAL\_SYSTICK\_Callback() から 1ms 毎に呼び出される。