

```

1 //=====
2 //
3 // Step 40
4 // Clock -> LCD use Ticker
5 // A/Dコンバータを使用して温度計測
6 //
7 //=====
8 #include "mbed.h"
9 #include "TextLCD.h"
10
11 TextLCD lcd(p17, p12, p27, p28, p29, p30); // rs, e, d4-d7
12 AnalogIn ain15(p15); // Analog In P15 <= LM35D
13
14 Ticker tik; // recurring interrupt
15 float tmp; // ondo
16
17 int tikFlg = 0;
18
19 //=====
20 // Ticker
21 //=====
22 void attime(void)
23 {
24     tikFlg = 1;
25 }
26
27 //=====
28 // Main
29 //=====
30 int main(void)
31 {
32     lcd.cls();
33     lcd.locate(0, 0); // x, y
34
35     //=====
36     // RTC check
37     //=====
38     time_t now_time = time(NULL); // today
39     struct tm *s_tm = localtime(&now_time);
40     if(s_tm->tm_year < 118 ) // since 1900
41     { // RTC set
42         struct tm t;
43         t.tm_sec = 0; // 0-59
44         t.tm_min = 0; // 0-59
45         t.tm_hour = 10; // 0-23
46         t.tm_mday = 1; // 1-31
47         t.tm_mon = 7-1; // 0-11
48         t.tm_year = 118; // year since 1900
49
50         set_time(mktime(&t)); // Write RTC
51     }
52
53     tik.attach(&attime, 1); // 1s -> call attime
54     double adt15 = 0.0;
55
56     while(1)
57     {
58         if( tikFlg == 1 )
59         {
60             tikFlg = 0;
61             now_time = time(NULL); // today
62             s_tm = localtime(&now_time);
63             lcd.locate(0, 0);
64             lcd.printf("%02d/%02d %02d:%02d:%02d", s_tm->tm_mon+1, s_tm->tm_mday, s_tm-
65 >tm_hour, s_tm->tm_min, s_tm->tm_sec);

```

```

66         adt15 = 0;
67         for( int i=0; i<1000; i++ )
68         {
69             adt15 += ain15.read();
70         }
71         //-----
72         // Temp
73         //-----
74         //tmp = ain15 * 330;          // ( ain15 / 10mV ) * 3.3V = ain15 * 330    mbed:
0->3.3V --- 0.0->1.0
75         tmp = adt15/1000.0 * 330;
76         lcd.locate(0,1);
77         lcd.printf("Temp=%4.1fdeg", tmp);          // Temp=xx.xdeg
78     }
79 }
80 }

```