

File "/PFS\_rcvr/main.cpp" printed from mbed.org on 7/8/2015

```
1 #include "mbed.h"
2 #include "TextLCD.h"
3 #include "BufferedSerial.h"
4 #include "definitions.h"
5 #include "interrupts.h"
6 PwmOut pwm(PTE30);
7 DigitalOut lcdR_W(PTA12);
8 TextLCD lcd(PTA4, PTD4, PTE20, PTE21, PTE22, PTE23, TextLCD::LCD20x4);
9
10 //
11 // global variables, ec.
12 //
13 #define NumberOfMenus 4
14 uint32_t menu_number;
15
16 #define DiffBufferSize 32
17 int diff_buffer_pointer = 0;
18 int diff_buffer[DiffBufferSize];
19 char Pattern[21] = {0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20};
20 char phase_Pattern[21];
21
22 float diff_average(int v) {
23     int i, sum;
24
25     if ((v > -10) && (v < 10)) {
26         diff_buffer[diff_buffer_pointer++] = v;
27         diff_buffer_pointer %= DiffBufferSize;
28     }
29
30     sum = 0;
31     for (i = 0; i < DiffBufferSize; i++) sum += diff_buffer[i];
32
33     return (float) sum / (float) DiffBufferSize;
34 }
35
36 void phase_shift_display(int shift, int line) { // shifts the pattern shift characters to the right on second line of the
37 // if shift is negative, the pattern is shifted to the left
38 #define CharPerLine 20
39     int i, j;
40
41     shift %= CharPerLine;
42
43     lcd.locate(0, line);
44     for (i = 0; i < CharPerLine; i++) {
45         j = i - shift;
46         if (j < 0) j += CharPerLine;
47         j %= CharPerLine;
48         lcd.putc(Pattern[j]);
49     }
50 }
51 void initialize(void);
52
53
54 int main() {
55     int i;
56
57     initialize();
58
59     while (1) {
60
61         if (oven_sense == 1) ovenLEDOn() else ovenLEDOff();
62
63         if (gps_locked == 1) gpsLEDOn() else gpsLEDOff();
64
65         if menuPB_Pressed {
66             while menuPB_Pressed;
67             menu_number++;
68             menu_number %= NumberOfMenus;
69             lcd.cls();
70
71             switch (menu_number) { // it just does these once when the menu button has been pressed
72                 case 0: lcd.locate(0,3);
```

```
72         lcd.printf("Zulu Time:");
73         break;
74
75     case 1: lcd.locate(0,3);
76             lcd.printf("Tune with helipot");
77             lcd.locate(0,0);
78             lcd.printf(Pattern);
79             lcd.locate(0,1);
80             break;
81
82     case 2: break;
83     case 3: break;
84     default: break;
85 }
86
87 if (RMC_rcvd == 1) { // this will occur once per second
88     switch (menu_number) {
89     case 0: lcd.locate(12, 3);
90             if (gps_locked == 1) for (i = 0; i < 6; i++) lcd.putc(gps_string[i + 7]);
91             break;
92
93     case 1: //
94             phase_shift_display(gps_count, 0);
95             phase_shift_display(ocxo_count, 1);
96             if (gps_count_number == CountInterval) {
97                 lcd.locate(0, 0); lcd.printf("%d %f", final_gps_count, diff_average(final_gps_count - final_
98                 lcd.locate(0, 1); lcd.printf("%d", final_ocxo_count);
99             }
100            break;
101
102     case 2: pwm.write(1.0);
103             phase_shift_display(gps_count, 0);
104             phase_shift_display(ocxo_count, 1);
105             if (gps_count_number == CountInterval) {
106                 lcd.locate(0, 0); lcd.printf("%d %f", final_gps_count, diff_average(final_gps_count - final_
107                 lcd.locate(0, 1); lcd.printf("%d", final_ocxo_count);
108             }
109            break;
110
111     case 3: pwm.write(0.25);
112             phase_shift_display(gps_count, 0);
113             phase_shift_display(ocxo_count, 1);
114             if (gps_count_number == CountInterval) {
115                 lcd.locate(0, 0); lcd.printf("%d %f", final_gps_count, diff_average(final_gps_count - final_
116                 lcd.locate(0, 1); lcd.printf("%d", final_ocxo_count);
117             }
118            break;
119
120     default: break;
121     }
122     RMC_rcvd = 0;
123 }
124
125 void initialize(void) {
126
127     lcdR_W = 0; // needed because TextLCD does not read the LCD ... it is always just written to
128
129     /*
130     lcd.printf("Testing");
131     lcd.locate(0,1);
132     lcd.putc(0xff);
133     */
134     menu_number = 0;
135     lcd.cls();
136     lcd.locate(0, 3);
137     lcd.printf("Zulu Time:");
138
139     pwm = 0.5;
140
141     gps.baud(9600);
142     gps.attach(&gps_serial_interrupt_routine);
143
144     gps_pps.rise(&gps_pps_routine);
145     ocxo_pps.rise(&ocxo_pps_routine);
146 }
```

```
145 rotary.fall(&rotary_interrupt_routine);
146
147 tic_gps.start();
148 tic_ocxo.start();
149
150 }
151
```

---

152  
153 "PFS\_rcvr/main.cpp" printed from mbed.org on 7/8/2015

```
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
```