```
#include "mbed.h"
#include "pinout.h"
#include "to_7seg.h"

// seven segment display anodes
// when in a int8_t, they are 0b-GFEDCBA
BusOut      g_seven_seg(SGA_PIN, SGB_PIN, SGC_PIN, SGD_PIN,
                        SGE_PIN, SGF_PIN, SGG_PIN);

// display cathodes
DigitalOut  g_dsr(DSR_PIN);
DigitalOut  g_dsl(DSL_PIN);

// leds
BusOut      g_leds(LDR_PIN, LDM_PIN, LDL_PIN);

//LIT
static AnalogIn LIT(LIT_PIN);

//Interrupciones
static InterruptIn swr(SWR_PIN);
static InterruptIn swl(SWL_PIN);
static InterruptIn swm(SWM_PIN);

static bool volatile swr_fall_evnt;
static bool volatile swl_fall_evnt;
static bool volatile swm_fall_evnt;

static void swr_fall_isr (void){
  swr_fall_evnt = true;
}

static void swl_fall_isr(void){
  swl_fall_evnt = true;
}

static void swm_fall_isr(void){
  swm_fall_evnt = true;
}

// MULTIPLEXACION
static Ticker tick_4ms;
static bool volatile tick_4ms_evnt;
static void tick_4ms_isr (void){
  tick_4ms_evnt = true;
}

//REBOTES
static Timeout tout_4ms_swr;
static bool volatile tout_4ms_swr_evnt;
static void tout_4ms_swr_isr (void){
  tout_4ms_swr_evnt = true;
}

static Timeout tout_4ms_swl;
static bool volatile tout_4ms_swl_evnt;
static void tout_4ms_swl_isr (void){
  tout_4ms_swl_evnt = true;
}

static Timeout tout_4ms_swm;
static bool volatile tout_4ms_swm_evnt;
static void tout_4ms_swm_isr (void){
  tout_4ms_swm_evnt = true;
}

//tiempo de refresco leds
static Ticker tick_10ms;
static bool volatile tick_10ms_evnt;
static void tick_10ms_isr (void){
  tick_10ms_evnt = true;
}

static Timeout tout_led_off;
static bool volatile tout_led_off_evnt;
static void tout_led_off_isr (void){
  tout_led_off_evnt = true;
}

int main (void) {
  uint8_t pulsaciones_m = 50;
  bool mux = false;
  uint16_t brillo = 0;
  uint16_t luz = LIT.read_u16()/656;
```

```
85      uint16_t intensidad = 100.95*luz+5;
86
87      g_dsl = 1;
88      g_dsr = 1;
89
90      g_seven_seg = to_7seg(pulsaciones_m);
91
92      swr.mode(PullUp);
93      swr.fall(swr_fall_isr);
94
95      swl.mode(PullUp);
96      swl.fall(swl_fall_isr);
97
98      swm.mode(PullUp);
99      swm.fall(swm_fall_isr);
100
101     tick_4ms.attach_us(tick_4ms_isr,4000);
102     tick_10ms.attach_us(tick_10ms_isr,10000);
103
104     for (;;) {
105
106       if(tick_4ms_evnt){
107         tick_4ms_evnt = false;
108         mux = !mux;
109
110         if(mux){
111           g_dsl = 0;
112           g_dsr = 1;
113           g_seven_seg = to_7seg(pulsaciones_m%10);
114
115         }else{
116           g_dsl = 1;
117           g_dsr = 0;
118           g_seven_seg = to_7seg(pulsaciones_m/10);
119         }
120       }
121
122       if(swr_fall_evnt){
123         swr_fall_evnt = false;
124         tout_4ms_swr.attach_us(tout_4ms_swr_isr,4000);
125       }
126
127       if(swl_fall_evnt){
128         swl_fall_evnt = false;
129         tout_4ms_swl.attach_us(tout_4ms_swl_isr,4000);
130       }
131
132        if(swm_fall_evnt){
133         swm_fall_evnt = false;
134         tout_4ms_swm.attach_us(tout_4ms_swm_isr,4000);
135       }
136
137       if(tout_4ms_swr_evnt){
138         tout_4ms_swr_evnt = false;
139
140         if(swr == 0){
141           pulsaciones_m = (pulsaciones_m == 0) ? 0 : (pulsaciones_m-1);
142         }
143       }
144
145       if(tout_4ms_swl_evnt){
146         tout_4ms_swl_evnt = false;
147
148         if(swl == 0){
149           pulsaciones_m = (pulsaciones_m == 99) ? 99 : (pulsaciones_m+1);
150         }
151       }
152
153        if(tout_4ms_swm_evnt){
154         tout_4ms_swm_evnt = false;
155
156         if(swm == 0){
157          brillo = intensidad;
158          pulsaciones_m = luz;
159         }
160       }
161
162       if(tick_10ms_evnt){
163         tick_10ms_evnt = false;
164
165         if(pulsaciones_m > 0 and  pulsaciones_m < 51){
166           g_leds = 4;
167           brillo = 198*pulsaciones_m-3.97*(50-pulsaciones_m);
168
```

```
169         }else if (pulsaciones_m > 50 and pulsaciones_m <= 99){
170            g_leds = 1;
171            brillo = 100*pulsaciones_m-106*(99-pulsaciones_m);
172 
173         }else if(pulsaciones_m == 0){
174           g_leds = 0;
175         }
176 
177         tout_led_off.attach_us(tout_led_off_isr, brillo);
178 
179       }
180 
181       if(tout_led_off_evnt){
182         tout_led_off_evnt = false;
183         g_leds = 0;
184       }
185 
186 
187       __disable_irq();
188       if(!tick_4ms_evnt && !swr_fall_evnt && !swl_fall_evnt && !tout_4ms_swr_evnt &&
    !tout_4ms_swl_evnt && !tick_10ms_evnt && !tout_led_off_evnt){
189         __WFI();
190       }
191       __enable_irq();
192 
193 
194     } // for (;;)
195    } // main()
196 
```