```
#include "RIMS.h"
volatile unsigned char TimerFlag = 0;
void TimerISR() { TimerFlag = 1; }
enum GP_States { GP_Start, GP_Pulse } GP_State;
void GP_Tick() {
static unsigned char i, pulse;
... // Standard switch statements for SM
}

enum RE_States { RE_Start, RE_Wait, RE_LED_ON, RE_LED_OFF }
RE_State;
void RE_Tick() {
static unsigned char echoes;
... // Standard switch statements for SM
}

void main() {
B = 0;
TimerSet(200);
TimerOn();
GP_State = GP_Start;
RE_State = RE_Start;
while (1) {
        GP_Tick();
        RE_Tick();
        while (!TimerFlag){} // Wait for timer period
                TimerFlag = 0; // Lower flag raised by timer
        }
}
```