

```

#include <stdio.h>
#include <stdint.h>
#include <stdarg.h>
#include <string.h>
#include <math.h>
#include "library.h"

extern void seed_random(uint32_t);
extern uint32_t random(void);
extern void setupLEDs(void);
extern void setLEDs(uint32_t, uint32_t);
extern void setup_button(void);
extern uint32_t button(void);
extern void wait_for_press(void);
extern void wait_for_release(void);
extern void delay(uint32_t);

int main(void) {
    uint32_t msec, t0, t1;

    InitializeHardware(HEADER, "Reaction test");
    seed_random(0x12345678); // Seed the random bit generator
    setupLEDs(); // Configure the GPIO port G bits for driving the LEDs
    setup_button(); // Configure the GPIO port A bits for reading the push button
    while (1) {
        setLEDs(1,0); // Set the red LED on, green LED off
        msec = random(); // Construct a random delay time between 1 and 5.095 seconds
        msec = msec & 0x000000FFF;
        msec = msec + 1000;
        delay(msec); // Delay program by a random wait time
        setLEDs(0,1); // Set the green LED on, red LED off
        if (button()) { // Check for the push button pressed too early
            printf("False start\n");
        }
        else {
            t0 = GetClockCycleCount(); // Record clock cycle count before and after
            button pressed
            wait_for_press();
            t1 = GetClockCycleCount();
            printf("Reaction time: %lu msec\n", (t1 - t0) / 180000); // Display reaction
            time
        }
        delay(50); // Wait for button switch bounce
        wait_for_release(); // Reset after button is released
    }
}

```