Code (Mic):

```
#include "mic.h" // for

#include <stdio.h> // for printf()

#include <unistd.h> //for sleep()

#include <signal.h>

#include <sys/time.h>


int is_running = 1;

uint16_t buffer [128];         //define buffer to store captures values

upm::Microphone *mic = NULL;    //create microphone object

void sig_handler(int signo)

{

    printf("got signal\n");

    if (signo == SIGINT) {

        is_running = 0;

    }

}


int main(int argc, char **argv)

{

    // Attach microphone to analog port A0

    mic = new upm::Microphone(0);


    if (signal(SIGINT, sig_handler) == SIG_ERR)

            printf("\ncan't catch SIGINT\n");


    thresholdContext ctx;
    ctx.averageReading = 0;
    ctx.runningAverage = 0;
    ctx.averagedOver   = 2;

    // Infinite loop, ends when program is cancelled
    // Repeatedly, take a sample every 2 microseconds;
    // find the average of 128 samples; and
    // print a running graph of the averages
    while (is_running) {
        int len = mic->getSampledWindow (2, 128, buffer);
        if (len) {
            int thresh = mic->findThreshold (&ctx, 30, buffer, len);
            mic->printGraph(&ctx);

            if (thresh) {
                // do something ....
            }
        }
    }
    printf ("exiting application\n");
    delete mic;
    return 0;
}
```