Lab Assignment-6

Name: Apeksha Sukesh Kallur

Main.cpp:

#include "mbed.h"

#include "rtos.h"

#include "x\_cube\_mems.h"

#include "time.h"

// Initialize variables

volatile float humidity;

volatile float pressure;

volatile float temp;

static X\_CUBE\_MEMS \*mems\_expansion\_board = X\_CUBE\_MEMS::Instance();

Mutex mutex;

// Create a DigitalOut objects for the LED

DigitalOut led(LED1);

// Create a Serial object to communicate via serial port

Serial serialComm(SERIAL\_TX, SERIAL\_RX);

void temperatureNotify(const char\* name, int state) {

mutex.lock(); // Lock Resources

mems\_expansion\_board->hts221.GetTemperature((float \*)&temp);

serialComm.printf("Temperature: %f C\r", temp);

mutex.unlock(); // Release Resources

}

void humidityNotify(const char\* name, int state){

mutex.lock(); // Lock Resources

mems\_expansion\_board->hts221.GetHumidity((float \*)&humidity);

serialComm.printf("\nHumidity: %f %% \r", humidity);

mutex.unlock(); // Release Resources

}

void pressureNotify(const char\* name, int state){

mutex.lock(); // Lock Resources

mems\_expansion\_board->lps25h.GetPressure((float \*)&pressure);

serialComm.printf("\nPressure: %f hPa\r", pressure);

wait(1.0); // sleep for 1 second

serialComm.printf("\033[2J\033[1;1H"); // clear screen

mutex.unlock(); // Release Resources

}

void ledThread(void const \*args){

while (1) {

led = 1; // LED on

wait(0.8);

led = 0; // LED off

wait(0.2);

}

}

void humidityThread(void const \*args) {

while (true) {

humidityNotify((const char\*)args, 0);

Thread::wait(2000); // 2 second wait

}

}

void pressureThread(void const \*args) {

while (true) {

pressureNotify((const char\*)args, 1);

Thread::wait(2000); // 2 second wait

}

}

void tempThread(void const \*args) {

while (true) {

temperatureNotify((const char\*)args, 2);

Thread::wait(2000); // 2 second wait

}

}

int main(){

Thread humidity;

humidity.start(callback(humidityThread, (void \*)"Humidity")); //humidity thread

Thread pressure;

pressure.start(callback(pressureThread, (void \*)"Pressure")); //pressure thread

Thread temp;

temp.start(callback(tempThread, (void \*)"temp"));//temperature thread

Thread blink;

blink.start(callback(ledThread, (void \*)"Blink")); // led blink thread

humidityThread((void \*)"Th 1");

pressureThread((void \*)"Th 2");

tempThread((void \*)"Th 3");

ledThread((void \*)"Th 4");

}