# UART

/\*----------------------------------------------------------------------------

LAB EXERCISE - UART interface

SERIAL COMMUNICATION

----------------------------------------

Print text to the PC via UART protocol

Input: None

Output: PC

GOOD LUCK!

\*----------------------------------------------------------------------------\*/

#include "mbed.h"

#include "pindef.h"

// Serial tx, rx connected to the PC via an USB cable

Serial device(UART\_TX, UART\_RX, "COM1");

/\*----------------------------------------------------------------------------

MAIN function

\*----------------------------------------------------------------------------\*/

int main(){

/\*

Set the baudrate to 9600 bps

Print "Hello mbed" to the PC serial monitor

\*/

if (device.writeable())

device.printf("Hello mbed");

}

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*ARM University Program Copyright (c) ARM Ltd 2016\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# I2C

/\*----------------------------------------------------------------------------

LAB EXERCISE - I2C interface

SERIAL COMMUNICATION

----------------------------------------

Access the temperature sensor via I2C interface, print the current temperature

to the PC via UART

Input: temperature sensor

Output: PC

GOOD LUCK!

\*----------------------------------------------------------------------------\*/

#include "mbed.h"

#include "pindef.h"

//I2C interface

I2C temp\_sensor(I2C\_SDA, I2C\_SCL);

Serial pc(UART\_TX, UART\_RX);

//I2C address of temperature sensor DS1631

const int temp\_addr = 0x90;

char cmd[] = {0x51, 0xAA};

char read\_temp[2];

/\*----------------------------------------------------------------------------

MAIN function

\*----------------------------------------------------------------------------\*/

int main(){

while(1){

// Start Convert T command to the sensor

temp\_sensor.write(temp\_addr, &cmd[0], 1, 0);

wait\_ms(1000);

temp\_sensor.write(temp\_addr, &cmd[1], 1, 1);

// Read the 16-bit temperature data

temp\_sensor.read(temp\_addr, read\_temp, 2, 0);

// Convert temperature to Celsius

float temp = (float((read\_temp[0] << 8) | read\_temp[1]) / 256);

// Send data

if (pc.writeable())

pc.printf("%f\r\n", temp);

}

}

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*ARM University Program Copyright (c) ARM Ltd 2016\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*