Lab 7：室温计

姓名：种晓雅 学号：3120102154

7 室温计

这个实验的目的是理解 uC/OS II 的任务调度方式，编写 uC/OS II 的应用程序，通过寄存器直接操纵 GPIO 来驱动外部设备。

配合课程

第七次：RTOS

实验目的

1. 学习 uC/OS II 的应用程序编写；
2. 理解如何直接操纵 GPIO，体会与 Linux 的不同；
3. 学习单总线设备的访问方式；
4. 学习 7 段数码管的时分复用驱动方式。

实验器材

硬件

- pcDuino v2 板一块；
- 5V/1A 电源一个；
- microUSB 线一根；
- 面包板一块；
- 两位 7 段数码管（共阳）一颗；
- 360Ω 1/8W 电阻 2 颗；
- DHT-11 温湿度传感器 1 个；
- 面包线若干。

以下为自备（可选）器材：

- PC（Windows/Mac OS/Linux）一台；
- USB-TTL 串口线一根（FT232RL 芯片或 PL2303 芯片）；
- 以太网线一根（可能还需要路由器等）；
- 1602 LCD（带配套的 5k 微调电阻）。

软件

- 编译软件；
- Fritzing。

实验步骤

1. 设计输出方案，画连线示意图；
2. 在面包板上连线，完成外部电路；
3. 编写 C/C++程序，测试程序和电路；

   1. 测试、实现 uC/OS II 对 GPIO 的访问；
   2. 实现 DHT-11 数据的读；
   3. 实现以时分复用方式在四位 7 段数码管上依次显示 0000-9999 的数字；
   4. 用两个 uc/OS II 任务，一个定时读 DHT-11 数据，一个轮流驱动数码管，一秒一次显示当前温度和湿度。注意处理好两个任务之间的数据共享。

自选扩展内容

1. 在 LCD 上显示。

实验报告要求

1. 画出你所实际实施的连接示意图；
2. 给出所用的器材的列表；
3. 用 Fritzing 画出外部设备的连线图，附实物照片；
4. 描述所做的实验步骤，给出各步操作的命令和结果；
5. 给出代码并解释；
6. 将所做作品拍摄视频上传到优酷，给出优酷的视频网址；
7. 说明其他所做的扩展内容的情况。

下载 ucos 代码

```
root@Acadia:~# git clone https://github.com/Pillar1989/ucos-ii-for-pcDuino
Cloning into 'ucos-ii-for-pcDuino'...
remote: Counting objects: 128, done.
remote: Compressing objects: 100% (109/109), done.
remote: Total 128 (delta 11), reused 128 (delta 11), pack-reused 0
Receiving objects: 100% (128/128), 1.62 MiB | 96 KiB/s, done.
Resolving deltas: 100% (11/11), done.
```

编译 arduino 动态链接库 libarduino.so

```
root@Acadia:~# cd ucos-ii-for-pcDuino
root@Acadia:~/ucos-ii-for-pcDuino# cd arduino
root@Acadia:~/ucos-ii-for-pcDuino/arduino# make
```

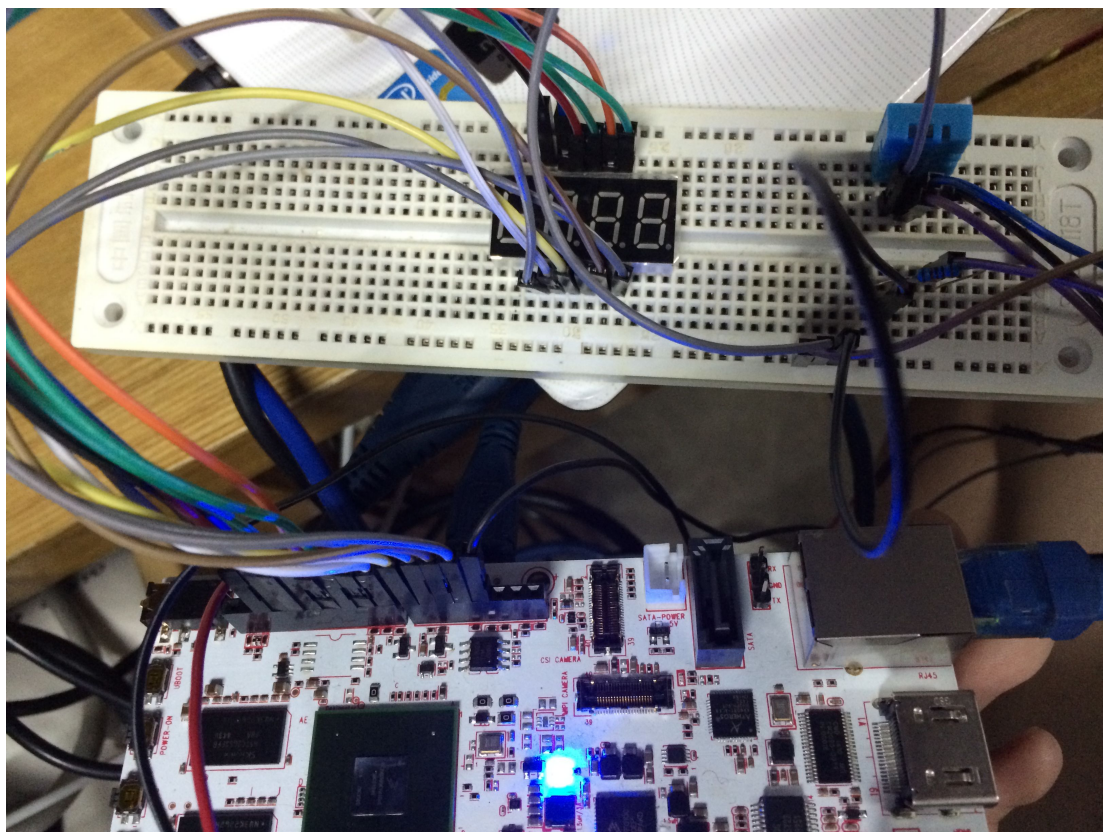再回到上一级目录下，编译整个系统，编译完了之后会生成 ucos_sample

```
root@Acadia:~/ucos-ii-for-pcDuino/arduino# cd ..
root@Acadia:~/ucos-ii-for-pcDuino# make
```

```
root@Acadia:~/ucos-ii-for-pcDuino# ls
Makefile    app      build     ucos                    ucos_sample
README.md   arduino  config.mk  ucos_2.86_original
```

用 Fritzing 画出外部设备的连线图：
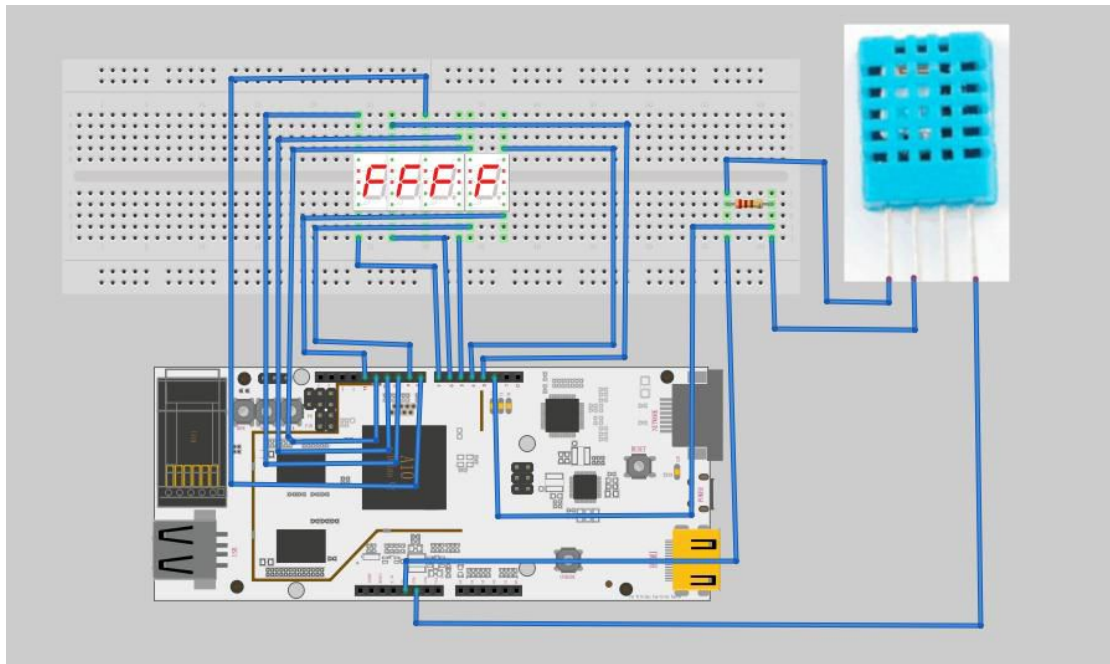
实物图

代码

```
#include <stdio.h>
#include <stdlib.h>
#include "ucos_ii.h"
#include <core.h>
#include <string.h>
#include <Arduino.h>
//引脚定义
#define DHT11PIN 9
const int a=2;
const int b=3;
const int c=4;
const int d=5;
const int e=6;
const int f=7;
const int g=8;
const int dig1=10;
const int dig2=11;
const int dig3=12;
const int dig4=13;
#ifndef dht11_h
#define dht11_h
#define DHT11LIB_VERSION "0.4.1"
```

```c
#define DHTLIB_OK 0
#define DHTLIB_ERROR_CHECKSUM -1
#define DHTLIB_ERROR_TIMEOUT -2
int dht11read(int pin);
//全局变量
int humidity;
int temperature;
int count=0;
#endif
//
// END OF FILE
//
//数码管的一大堆函数，主要是显示 0-9 数字
void display_zero(){
digitalWrite(a,LOW);
digitalWrite(b,LOW);
digitalWrite(c,LOW);
digitalWrite(d,LOW);
digitalWrite(e,LOW);
digitalWrite(f,LOW);
digitalWrite(g,HIGH);
}
void display_one(){
digitalWrite(a,HIGH);
digitalWrite(b,LOW);
digitalWrite(c,LOW);
digitalWrite(d,HIGH);
digitalWrite(e,HIGH);
digitalWrite(f,HIGH);
digitalWrite(g,HIGH);
}
void display_two(){
digitalWrite(a,LOW);
digitalWrite(b,LOW);
digitalWrite(c,HIGH);
digitalWrite(d,LOW);
digitalWrite(e,LOW);
digitalWrite(f,HIGH);
digitalWrite(g,LOW);
}
void display_three(){
digitalWrite(a,LOW);
digitalWrite(b,LOW);
digitalWrite(c,LOW);
```

```
digitalWrite(d,LOW);
digitalWrite(e,HIGH);
digitalWrite(f,HIGH);
digitalWrite(g,LOW);
}
void display_four(){
digitalWrite(a,HIGH);
digitalWrite(b,LOW);
digitalWrite(c,LOW);
digitalWrite(d,HIGH);
digitalWrite(e,HIGH);
digitalWrite(f,LOW);
digitalWrite(g,LOW);
}
void display_five(){
digitalWrite(a,LOW);
digitalWrite(b,HIGH);
digitalWrite(c,LOW);
digitalWrite(d,LOW);
digitalWrite(e,HIGH);
digitalWrite(f,LOW);
digitalWrite(g,LOW);
}
void display_six(){
digitalWrite(a,LOW);
digitalWrite(b,HIGH);
digitalWrite(c,LOW);
digitalWrite(d,LOW);
digitalWrite(e,LOW);
digitalWrite(f,LOW);
digitalWrite(g,LOW);
}
void display_seven(){
digitalWrite(a,LOW);
digitalWrite(b,LOW);
digitalWrite(c,LOW);
digitalWrite(d,HIGH);
digitalWrite(e,HIGH);
digitalWrite(f,HIGH);
digitalWrite(g,HIGH);
}
void display_eight(){
digitalWrite(a,LOW);
digitalWrite(b,LOW);
```

```
digitalWrite(c,LOW);
digitalWrite(d,LOW);
digitalWrite(e,LOW);
digitalWrite(f,LOW);
digitalWrite(g,LOW);
}
void display_nine(){
digitalWrite(a,LOW);
digitalWrite(b,LOW);
digitalWrite(c,LOW);
digitalWrite(d,LOW);
digitalWrite(e,HIGH);
digitalWrite(f,LOW);
digitalWrite(g,LOW);
}
//映射数码管应该显示的值
void choice(int x){
switch(x) {
case 0:display_zero();break;
case 1:display_one();break;
case 2:display_two();break;
case 3:display_three();break;
case 4:display_four();break;
case 5:display_five();break;
case 6:display_six();break;
case 7:display_seven();break;
case 8:display_eight();break;
case 9:display_nine();break;
}
}
//四个数码管
void digitone(){
digitalWrite(dig1,HIGH);
digitalWrite(dig2,LOW);
digitalWrite(dig3,LOW);
digitalWrite(dig4,LOW);
choice(humidity/10);
}
void digittwo(){
digitalWrite(dig2,HIGH);
digitalWrite(dig1,LOW);
digitalWrite(dig3,LOW);
digitalWrite(dig4,LOW);
choice(humidity%10);
```

```
}
void digitthree(){
digitalWrite(dig3,HIGH);
digitalWrite(dig1,LOW);
digitalWrite(dig2,LOW);
digitalWrite(dig4,LOW);
choice(temperature/10);
}
void digitfour(){
digitalWrite(dig4,HIGH);
digitalWrite(dig1,LOW);
digitalWrite(dig2,LOW);
digitalWrite(dig3,LOW);
choice(temperature%10);
}
void digitnone(){
digitalWrite(dig4,LOW);
digitalWrite(dig1,LOW);
digitalWrite(dig2,LOW);
digitalWrite(dig3,LOW);
}
//dht 数据读温度和湿度，这是直接用的库
// Return values:
// DHTLIB_OK
// DHTLIB_ERROR_CHECKSUM
// DHTLIB_ERROR_TIMEOUT
int dht11read(int pin)
{
// BUFFER TO RECEIVE
uint8_t bits[5];
uint8_t cnt = 7;
uint8_t idx = 0;
int i;
// EMPTY BUFFER
for (i=0; i< 5; i++) bits[i] = 0;
// REQUEST SAMPLE
pinMode(pin, OUTPUT);
digitalWrite(pin, LOW);
delay(36);
digitalWrite(pin, HIGH);
delayMicroseconds(40);
pinMode(pin, INPUT);
// ACKNOWLEDGE or TIMEOUT
unsigned int loopCnt = 10000;
```

```
while(digitalRead(pin) == LOW)
if (loopCnt-- == 0) return DHTLIB_ERROR_TIMEOUT;
loopCnt = 10000;
while(digitalRead(pin) == HIGH)
if (loopCnt-- == 0) return DHTLIB_ERROR_TIMEOUT;
// READ OUTPUT - 40 BITS => 5 BYTES or TIMEOUT
for (i=0; i<40; i++)
{
loopCnt = 10000;
while(digitalRead(pin) == LOW)
if (loopCnt-- == 0) return DHTLIB_ERROR_TIMEOUT;
unsigned long t = micros();
loopCnt = 10000;
while(digitalRead(pin) == HIGH)
if (loopCnt-- == 0) return DHTLIB_ERROR_TIMEOUT;
if ((micros() - t) > 40) bits[idx] |= (1 << cnt);
if (cnt == 0) // next byte?
{
cnt = 7; // restart at MSB
idx++; // next byte!
}
else cnt--;
}
// WRITE TO RIGHT VARS
// as bits[1] and bits[3] are allways zero they are omitted in formulas.
humidity = bits[0];
temperature = bits[2];
uint8_t sum = bits[0] + bits[2];
if (bits[4] != sum) return DHTLIB_ERROR_CHECKSUM;
return DHTLIB_OK;
}
//
// END OF FILE
//
//初始化
void hardware_init()
{
pinMode(a, OUTPUT);
pinMode(b, OUTPUT);
pinMode(c, OUTPUT);
pinMode(d, OUTPUT);
pinMode(e, OUTPUT);
pinMode(f, OUTPUT);
pinMode(g, OUTPUT);
```

```
pinMode(dig1, OUTPUT);
pinMode(dig2, OUTPUT);
pinMode(dig3, OUTPUT);
pinMode(dig4, OUTPUT);
}
/* Function common to all tasks */
//任务一，读 dht11 的数据
void MyTask( void *p_arg )
{
#if OS_CRITICAL_METHOD == 3
OS_CPU_SR cpu_sr = 0;
#endif
int val;
while(1)
{
OS_ENTER_CRITICAL();
printf("%d\t",count);
count++;
val=dht11read(DHT11PIN);
switch(val){
case DHTLIB_OK:printf("ok\n");break;
case DHTLIB_ERROR_CHECKSUM:printf("checksum\n");break;
case DHTLIB_ERROR_TIMEOUT:printf("timeout\n");break;
default:printf("unknown\n");break;
}
printf("humidity:%d\ttemperature:%d\n",humidity,temperature);
OS_EXIT_CRITICAL();
OSTimeDly(100);
}/* while */
}
//任务 2，显示数字
void Digital(void *p_arg )
{
#if OS_CRITICAL_METHOD == 3
OS_CPU_SR cpu_sr = 0;
#endif
int i;
while(1)
{
OS_ENTER_CRITICAL();
for(i=0;i<500;i++){
digitone();
OSTimeDly(1);
digittwo();
```

```
OSTimeDly(1);
digitthree();
OSTimeDly(1);
digitfour();
OSTimeDly(1);
digitnone();
}
OS_EXIT_CRITICAL();
OSTimeDly(100);
}/* while */
}
//main 函数
int main (void)
{
/* pthreads allocates its own memory for task stacks. This UCOS linux port needs a
minimum stack size
in order to pass the function information within the port. */
OSInit();
hardware_init();
INT8U Stk1[ OSMinStkSize() ];
INT8U Stk2[ OSMinStkSize() ];
INT8U Stk3[ OSMinStkSize() ];
INT8U Stk4[ OSMinStkSize() ];
INT8U Stk5[ OSMinStkSize() ];
char sTask1[] = "Task 1";
char sTask2[] = "Task 2";
char sTask3[] = "Task 3";
char sTask4[] = "Task 4";
// char sTask5[] = "Task 5";
OSTaskCreate( MyTask, sTask1, (void*)Stk1, 4 );
OSTaskCreate( Digital, sTask2, (void*)Stk2,5 );
// OSTaskCreate( MyTask, sTask3, (void*)Stk3, 6 );
// OSTaskCreate( MyTask, sTask4, (void*)Stk4, 7 );
// OSTaskCreate( MyTask, sTask5, (void*)Stk5, 8 );
OSStart();
return 0;
}
```

运行结果

```
0       timeout
humidity:0      temperature:0
1       ok
humidity:39     temperature:23
2       ok
humidity:39     temperature:23
3       ok
humidity:38     temperature:23
4       ok
```

前两位为湿度，后两位为温度

优酷视频网址

http://v.youku.com/v_show/id_XMTI2MjU5OTY4MA==.html