

## 齐鲁工业大学 2017/2018 学年第二学期《无线传感器网络》

## 期末考试试卷 (A 卷)

(本试卷共 8 页)

(适用班级: 物联网工程 2016 级)

题号	一	二	三	总分
得分				

得分	
阅卷人	

## 一、填空题(每题 2 分, 共 40 分)

1. 无线传感器网络的发展历史可分为三个阶段, 第一阶段是传统的传感器系统, 第二阶段是\_\_\_\_\_, 第三阶段是\_\_\_\_\_。
2. 在传感器节点不能和网关直接通信的时候, 路由器节点就起到了连接\_\_\_\_\_和\_\_\_\_\_通信的目的。
3. 根据覆盖目标不同, 目前覆盖算法可以分为面覆盖、\_\_\_\_\_和\_\_\_\_\_。
4. 基于节点度算法的核心思想是给定节点度的上限和下限需求, 动态调整节点的\_\_\_\_\_, 使得\_\_\_\_\_落在上限和下限之间。
5. 目前无线传感器网络采用的主要传输介质包括无线电、\_\_\_\_\_和\_\_\_\_\_等。
6. 设计传感器网络物理层时需要考虑\_\_\_\_\_, \_\_\_\_\_和通信频段问题。
7. 平面路由协议中的 Flooding and Grossing 协议主要有两种, 一种是\_\_\_\_\_, 另一种是\_\_\_\_\_。
8. 在传感器网络节点定位算法中, 根据节点是否需要测量实际节点间的距离将定位算法分为\_\_\_\_\_的定位和\_\_\_\_\_的定位算法。
9. 在传感器网络节点的节能策略中, 休眠机制的主要思想是, 当节点周围没有感兴趣的事件发生时, 计算与通信单元处于\_\_\_\_\_, 把这些组件关掉或调到\_\_\_\_\_的状态, 即休眠状态。
10. 无线传感器网络数据融合方法主要有: 直接对数据源操作的方法、\_\_\_\_\_和\_\_\_\_\_。

11. 质心定位算法首先确定包含\_\_\_\_\_, 计算这个区域的\_\_\_\_\_, 并将其作为未知节点的位置。
12. 制定 IEEE 1451 标准的目的是通过定义一套通用的通信接口, 以使变送器(传感器/执行器)能够独立于网络, 并与现有基于微处理器的系统、仪器仪表的现场总线网络相连, 解决不同网络之间的\_\_\_\_\_问题, 并最终能够实现变送器到网络的\_\_\_\_\_。
13. IEEE 802.15.4 标准主要包括\_\_\_\_\_和\_\_\_\_\_层的标准。
14. WiFi 标准具有四个特点: 无线电波覆盖范围广、\_\_\_\_\_, 厂商进入该领域的门槛低和\_\_\_\_\_。
15. 接入 WSN 的方式有面向以太网的 WSN 接入、\_\_\_\_\_和\_\_\_\_\_。
16. 为道路参数监测节点选择光强度传感器时, 主要考虑因素时传感器的\_\_\_\_\_, 供电电压、光波长、强度测量范围、封装、\_\_\_\_\_以及传感器价格等。
17. 智能家居可以定义为一个\_\_\_\_\_或者一个\_\_\_\_\_。
18. RFID 的主要技术包括\_\_\_\_\_和\_\_\_\_\_。
19. RFID 与传感器整合时把 RFID 标签装上\_\_\_\_\_, 这样可以为标签提供\_\_\_\_\_。
20. 物联网的关键技术主要有\_\_\_\_\_, RFID 标签和\_\_\_\_\_。

得分	
阅卷人	

## 二、简答题(每题 5 分, 共 40 分)

1. 传感器硬件节点设计的主要内容时什么?

姓名

学号

专业班级

学院、系

线

封

密

2. 拓扑控制的意义是什么？

3. 无线传感器网络路由协议考虑的因素有哪些？

4. 传感器网络的安全性需求包括哪些内容？

5. RSSI 测距的原理是什么？

6. WSN→Internet 数据包转换的思想是什么？

7. 基于 TinyOS 的 WSN 定位系统的设计原则有哪些？

8. 智能家居网关应用软件设计主要有哪些内容？

线

封

密

得分	
阅卷人	

### 三、程序功能分析(每题 10 分, 共 20 分)

1. f8wconfig.cfg 文件是 Zstack 协议栈的配置文件, 在此文件中可以设置 Zigbee 使用的信道和 Zigbee 网络的 PANID。根据以

下代码, 说明 Zigbee 当前使用的信道和网络的 PANID。

```
// 信道设置
// 0      :    868MHz    0x00000001
// 1 - 10  :    915MHz    0x000007FE
// 11 - 26 :    2.4GHz    0x07FFF800
// -DMAX_CHANNELS_868MHz    0x00000001
// -DMAX_CHANNELS_915MHz    0x000007FE
// -DMAX_CHANNELS_24GHz     0x07FFF800
// 以下为信道 11-26 的设置
// -DDEFAULT_CHANLIST = 0x04000000 // 0x1A
// -DDEFAULT_CHANLIST = 0x02000000 // 0x19
// -DDEFAULT_CHANLIST = 0x00800000 // 0x17
// -DDEFAULT_CHANLIST = 0x00400000 // 0x16
// -DDEFAULT_CHANLIST = 0x00200000 // 0x15
// -DDEFAULT_CHANLIST = 0x00100000 // 0x14
// -DDEFAULT_CHANLIST = 0x00080000 // 0x13
// -DDEFAULT_CHANLIST = 0x00040000 // 0x12
// -DDEFAULT_CHANLIST = 0x00020000 // 0x11
// -DDEFAULT_CHANLIST = 0x00010000 // 0x10
// -DDEFAULT_CHANLIST = 0x00008000 // 0x0F
// -DDEFAULT_CHANLIST = 0x00004000 // 0x0E
// -DDEFAULT_CHANLIST = 0x00002000 // 0x0D
// -DDEFAULT_CHANLIST = 0x00001000 // 0x0C
// -DDEFAULT_CHANLIST = 0x00000800 // 0x0B
```

```
-DZDAPP_CONFIG_PAN_ID = 0x1206
```



线

封

密

2. 已知 Zstack 协议栈数据的发送和接受程序的代码如(1)和(2)所示。

(1) 信息发送程序代码如下:

```
void MySendTest_SendPeriodicMessage(void){
    // 发送的数据
    char theMessageData[] = "*****";
    if(AF_DataRequest(
        &MySendTest_Periodic_DstAddr, //发送目的地址
        &MySendTest_epDesc, //发送的端点描述符
        &MySendTest_PERIODIC_CLUSTERID, //簇 ID 号
        (uint16)osal_strlen(theMessageData)+1, //发送的字节长度
        (uint8 *)theMessageData, //发送的数据
        &MyFirstAppCoordManage_TransID, //发送的数据 ID 序号
        AF_DISCV_ROUTE, //设置路由发现
        AF_DEFAULT_RADIUS //设置路由域
    ) == ZSUCCES){
    } else {

    }
}
```

(2) 信息接受程序的代码如下:

```
void SampleApp_MessageMSGCB(aflIncomingMSGPacket_t * pkt){
    uint16 flashTime;
    //数据的接受通过判断 clusterId
    switch(pkt->clusterId){ //判断接受的输入簇 ID
        case SAMPLEAPP_PERIODIC_CLUSTERID:
            if( (pkt->cmd.Data[0] == 'L')
                &&(pkt->cmd.Data[1] == 'E')
                &&(pkt->cmd.Data[2] == 'D')
                &&(pkt->cmd.Data[3] == '1')){//判断是否收到"LED1"
```

```

        HalLedBlink(HAL_LED_1, 4, 50, 500); //LED1 闪烁
    } else {
        HalLedBlink(HAL_LED_0, 4, 50, 500); //LED0 闪烁
    }
    break;
case SAMPLEAPP_FLASH_CLUSTERID:
    break;
}
}

```

试分析若 theMessageData[] = "LED1" 和 theMessageData[] = "LED0"时(假设发送和接受程序的簇 ID 相同)，数据接受程序的执行结果。