

物联网应用实践

——关键技术进阶指南

主编 付忠勇

WULIANWANG YINGYONG SHIJIAN

GUANJIAN JISHU

JINJIE ZHINAN



北京邮电大学出版社
BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS PRESS

物联网应用实践

——关键技术进阶指南

主编 付忠勇



北京邮电大学出版社
www.buptpress.com

内 容 简 介

本书介绍了物联网关键技术中的程序开发、传感器技术基础、无线通信原理共三个方面的内容。

关于编写代码,本书较为详细地介绍了编程的基本规范,为了便于开发后期的集成测试与系统联调,更加注重程序的可测性,注重代码效率、质量保证、代码测试和维护;传感器技术基础,主要说明了传感器的分类和典型应用;无线通信原理部分,解释了无线通信基本概念和原理,包括无线信道、分集技术、无线通信多址技术、资源分配及算法、语音编码技术等。

本书适合于从事物联网应用开发的技术人员、工程师、教师及掌握了基础知识的大学生作为进阶学习的参考手册。

图书在版编目(CIP)数据

物联网应用实践:关键技术进阶指南/付忠勇主编.--北京:北京邮电大学出版社,2014.8
ISBN 978-7-5635-4129-4

I. ①物… II. ①付… III. ①互联网络—应用②智能技术—应用 IV. ①TP393.4②TP18
中国版本图书馆CIP数据核字(2014)第196563号

书 名:物联网应用实践——关键技术进阶指南

著作责任者:付忠勇 主编

责任编辑:刘 颖

出版发行:北京邮电大学出版社

社 址:北京市海淀区西土城路10号(邮编:100876)

发 行 部:电话:010-62282185 传真:010-62283578

E-mail:publish@bupt.edu.cn

经 销:各地新华书店

印 刷:北京鑫丰华彩印有限公司

开 本:787 mm×1 092 mm 1/16

印 张:19.25

字 数:479

版 次:2014年8月第1版 2014年8月第1次印刷

ISBN 978-7-5635-4129-4

定 价:42.00 元

• 如有印装质量问题,请与北京邮电大学出版社发行部联系 •

前 言

2013 年年初,国务院在《关于推进物联网有序健康发展的指导意见》中提出,到 2015 年,实现物联网在经济社会重要领域的规模示范应用,突破一批核心技术,初步形成物联网产业体系。

以计算机网络为基础,利用射频识别、无线数据通信、计算机等技术,构造一个物与物相联的智能网络,即物联网。物联网内每个物品都有一个唯一的电子识别码(EPC),将 EPC 附在被标识产品上,可以被信息处理软件识别、查询、处理和传递,进而形成各种信息服务。

《物联网应用实践——关键技术进阶指南》是近年来笔者及多位同事从事物联网技术教学的心得,也包含着合作企业多位专家的技术经验总结,希望能给读者一些切实有效的帮助。

本书选取了物联网关键技术中三个方面的内容,包括编写程序、传感器技术基础和无线通信原理。

编程部分包括 C 语言规范、嵌入式 C 开发、IAR-AVR C 编译器指南,书中较为详细地介绍了编程的基本规范,为了便于开发后期的集成测试与系统联调,该部分更加注重代码效率、质量保证、代码测试和维护;传感器技术基础部分,主要说明了传感器的分类和典型应用;无线通信原理部分,则解释了无线通信基本概念和原理,包括无线信道、分集技术、无线通信多址技术、资源分配及算法、语音编码技术等。

本书适合从事物联网应用开发的技术人员、工程师、教师 and 有一定基础的学生作为进阶学习的参考手册。

感谢来自行业一线的李羿、刘勇等工程专家,是他们促成了本书的完成。

鉴于互联网技术更新很快,鉴于自身能力和见识的不足,疏漏、偏颇和错误之处在所难免,敬请读者批评指正。

作 者

目 录

第 1 章 C 语言编程规范	1
1.1 排版	1
1.2 注释	5
1.3 标识符命名	13
1.4 可读性	15
1.5 变量、结构	17
1.6 函数、过程	23
1.7 可测性	31
1.8 程序效率	36
1.9 质量保证	40
1.10 代码编辑、编译、审查	47
1.11 代码测试、维护	48
1.12 宏	49
第 2 章 嵌入式 C 进阶之道	51
2.1 良好的编程风格	51
2.1.1 排版	51
2.1.2 注释	53
2.1.3 标识符	55
2.1.4 表达式和基本语句	57
2.1.5 杂项	59
2.2 模块化编程	59
2.3 不可不用的关键字	69
2.3.1 static 关键字	69
2.3.2 const 关键字	73
2.3.3 volatile 关键字	74
2.3.4 struct 与 typedef 关键字	76

2.4 MISRAC:2004	78
2.4.1 开发环境	79
2.4.2 语言外延	80
2.4.3 文档化	81
2.4.4 字符集	83
2.4.5 标识符	84
2.4.6 类型	87
2.4.7 常量	88
2.4.8 声明与定义	88
2.4.9 初始化	90
2.4.10 数值类型转换	92
2.4.11 隐式和显式类型转换	92
2.4.12 隐式转换的类型	92
2.4.13 危险的类型转换	94
2.4.14 基本类型	95
2.4.15 复杂表达式	97
2.4.16 隐式类型转换,显式类型转换和整数后缀	98
2.4.17 指针类型转换	104
2.4.18 表达式	105
2.4.19 控制语句表达式	113
2.4.20 控制流	116
2.4.21 switch 语句	120
2.4.22 函数	122
2.4.23 指针和数组	124
2.4.24 结构与联合	127
2.4.25 预处理指令	132
2.4.26 标准库	138
2.4.27 运行时错误	141
第3章 IAR-AVR C 编译器简要指南	144
3.1 数据类型	144
3.1.1 整型数据	144
3.1.2 浮点数据	145
3.1.3 指针类型	145
3.1.4 数据指针	145

3.1.5	函数指针	145
3.2	关于扩展关键字	145
3.2.1	扩展关键字的使用	145
3.2.2	函数扩展关键字	146
3.2.3	其他特别的關鍵字	146
3.3	EEPROM 常用类型的具体操作方法	146
3.3.1	EEPROM 区域数据存储	146
3.3.2	与__eeprom 有关的指针操作	147
3.3.3	指向 EEPROM 空间的指针:EEPROM 指针(控制属性)	148
3.3.4	存储于 EEPROM 空间的指针数据指针(就像存储于 EEPROM 空间的数据一样)控制存储	148
3.3.5	控制数据和指针存放的__eeprom 定义必须是全局变量,控制属性(好像只有指针)可以是局部变量	148
3.3.6	__root 关键字	149
3.3.7	EEPROM 操作宏取函数	149
3.3.8	自动生成.eep 文件	150
3.4	FLASH 常用类型的具体操作方法	153
3.4.1	FLASH 区域数据存储	153
3.4.2	与__flash 有关的指针操作	155
3.4.3	指向 FLASH 空间的指针 FLASH 指针(控制类型属性)	155
3.4.4	存储于 FLASH 空间的指针数据指针	155
3.4.5	控制数据和指针存放的__flash 定义必须是全局变量,控制类型属性(好像只有指针)可以是局部变量	156
3.4.6	__root 关键字	156
3.4.7	FLASH 操作宏函数	156
3.4.8	自编程函数	158
3.5	SARM 数据类型的具体操作方法	158
3.5.1	定义变量在工作寄存器	158
3.5.2	定义变量的绝对地址	159
3.5.3	定义没有存储特性的绝对地址变量	160
3.5.4	带存储特性的关键字定义变量的绝对地址	160
3.5.5	void main(void)	161
3.5.6	关键字 volatile	161
3.5.7	强大的位操作	161
3.5.8	在 C 语言里对位的操作	161

3.5.9	IAR 编译器对位的支持更强大	161
3.5.10	位变量定义	162
3.5.11	bool 数据类型	163
3.6	函 数	163
3.6.1	中断函数	163
3.6.2	内在函数	164
3.6.3	延时函数	164
3.6.4	中断指令	164
3.6.5	从 FLASH 空间指定地址读取数据	164
3.6.6	乘法函数	165
3.6.7	半字节交换指令	165
3.6.8	MCU 控制指令	165
3.7	头文件	165
3.8	嵌入汇编语言	166
第 4 章	传感器技术基础	167
4.1	传感器概述	167
4.1.1	什么是传感器	167
4.1.2	传感器的分类	167
4.1.3	传感器的性能和评价指标	167
4.2	应变式传感器	169
4.2.1	电阻应变效应	169
4.2.2	应变计的灵敏度系数	169
4.2.3	横向效应	170
4.2.4	应变计的动态特性	170
4.2.5	其他特性参数	170
4.2.6	应变计的粘贴	171
4.2.7	直流电桥的特性方程及平衡条件	171
4.2.8	直流电桥的电压灵敏度	172
4.2.9	交流电桥的平衡条件和电压输出	172
4.2.10	温度误差产生的原因	173
4.2.11	温度补偿方法	174
4.2.12	电阻应变仪	175
4.2.13	应变式传感器	176
4.2.14	应变式测力与称重传感器	176

4.2.15	应变式压力传感器	177
4.2.16	与几种新型的微应变式传感器相关的概念	177
4.3	光电式传感器	178
4.3.1	光电效应	178
4.3.2	热释电效应	178
4.3.3	光传感器的特性表示法	178
4.3.4	光电管	179
4.3.5	光电倍增管	180
4.3.6	光敏电阻器	180
4.3.7	光电二极管和光电三极管	181
4.3.8	光电池	183
4.3.9	光电耦合器	185
4.3.10	电荷耦合器件图像传感器	186
4.4	光纤传感器	186
4.4.1	光纤的结构	186
4.4.2	光在光纤中的传播	187
4.4.3	光纤的几个重要参数	187
4.4.4	光纤的类型	188
4.4.5	反射式光纤位移传感器	188
4.4.6	相位调制型光纤传感器基本原理	189
4.4.7	光纤(强度型)干涉仪	189
4.4.8	光纤传感器的应用举例	189
4.5	变磁阻式传感器	190
4.5.1	工作原理	190
4.5.2	输出特性	191
4.5.3	差动电感传感器	192
4.5.4	测量电路	192
4.5.5	差动变压器式传感器	192
4.5.6	Ⅱ形差动变压器的输出特性	192
4.5.7	螺管形差动变压器	193
4.5.8	变磁阻式传感器的应用	193
4.6	压电传感器	194
4.6.1	压电效应	194
4.6.2	压电材料	194
4.6.3	压电加速度传感器的工作原理	194

4.6.4	压电加速度传感器的参数	194
4.6.5	压电加速度传感器的结构	195
4.6.6	压电加速度传感器的等效电路	195
4.6.7	压电传感器接放大器的等效电路	196
4.6.8	压电谐振式传感器	197
4.6.9	SAW 传感器的基本原理	198
4.6.10	SAW 压力传感器	198
4.6.11	SAW 热敏传感器	198
4.6.12	SAW 气敏传感器	198
4.7	压电声传感器	199
4.8	半导体传感器	199
4.8.1	半导体温度传感器	199
4.8.2	接触型半导体传感器	199
4.8.3	非接触型半导体温度传感器	201
4.8.4	半导体湿度传感器	201
4.8.5	湿度的定义	201
4.8.6	半导体气敏传感器工作机理	201
4.8.7	电阻型半导体气敏传感器	202
4.8.8	非电阻型半导体气敏器件	203
4.8.9	霍尔传感器	204
4.8.10	磁敏电阻器	205
4.9	电位器式传感器	205
4.10	Z-半导体敏感元件	206
4.10.1	Z-半导体敏感元件	206
4.10.2	温敏 Z-元件的伏安特性	207
第 5 章	无线通信原理	208
5.1	无线信道	208
5.1.1	自由空间无线电波传播	208
5.1.2	移动通信的电波传播特点	209
5.1.3	移动通信电波传播特性	210
5.1.4	短期(快)衰落	211
5.1.5	长期(慢)衰落	214
5.1.6	衰落特性	214
5.1.7	电波传播损耗预测模型	215

5.1.8	Okumura-Hata 模型	215
5.1.9	COST231 Walfisch-Ikegami 模型	218
5.1.10	覆盖设计	220
5.1.11	传播模型的选用及修正	220
5.1.12	基站覆盖预测	221
5.1.13	功率控制	224
5.2	分集技术	224
5.2.1	分集技术的概念	225
5.2.2	分集技术的分类	225
5.2.3	空间分集	225
5.2.4	极化分集	226
5.2.5	时分分集	227
5.2.6	频率分集	227
5.2.7	角度分集	228
5.2.8	信号合并准则	228
5.2.9	最大信噪比准则下的信号合并方法	228
5.2.10	隐分集技术	229
5.2.11	交织编码技术(时间隐分集)	229
5.3	无线通信多址技术	230
5.3.1	多址技术概述	230
5.3.2	频分多址	231
5.3.3	时分多址	231
5.3.4	码分多址	232
5.3.5	扩频技术	235
5.3.6	空分多址	238
5.4	无线通信中的资源分配原理及算法举例	241
5.4.1	蜂窝的概念	241
5.4.2	以 GSM 为例介绍 FCA	242
5.4.3	多信道共用技术	244
5.4.4	微蜂窝与信道动态分配技术	247
5.5	语音编码技术	248
5.5.1	概述	249
5.5.2	脉冲编码调制(PCM)技术	251
5.5.3	自适应差分脉冲编码调制(ADPCM)	257
5.6	天线	258

5.6.1 天线电指标的定义	260
5.6.2 天线的高度和间距	262
5.6.3 ZXPCS 基站天线说明	263
5.6.4 天线电性能技术指标	264
5.6.5 天线环境性能指标	264
综合练习.....	266
附录 A 双工技术、单位换算和集成无线收发器基本测试	287
附录 B 缩略语	294
参考文献.....	296

第 1 章 C 语言编程规范

1.1 排 版

(1) 程序块要采用缩进风格编写,缩进的空格数为 4 个。

说明:对于由开发工具自动生成的代码可以有不一致。

(2) 相对独立的程序块之间、变量说明之后必须加空行。

示例:如下例子不符合规范。

```
if(! valid_ni(ni))
{
    ... // program code
}
repssn_ind = ssn_data[index].repssn_index;
repssn_ni  = ssn_data[index].ni;
```

应如下书写:

```
if(! valid_ni(ni))
{
    ... // program code
}

repssn_ind = ssn_data[index].repssn_index;
repssn_ni  = ssn_data[index].ni;
```

(3) 较长的语句(大于 80 字符)要分成多行书写,长表达式要在低优先级操作符处划分新行,操作符放在新行之首,划分出的新行要进行适当的缩进,使排版整齐、语句可读。

示例：

```
perm_count_msg.head.len = NO7_TO_STAT_PERM_COUNT_LEN
                        + STAT_SIZE_PER_FRAM * sizeof(_UL);

act_task_table[frame_id * STAT_TASK_CHECK_NUMBER + index].occupied
    = stat_poi[index].occupied;

act_task_table[taskno].duration_true_or_false
    = SYS_get_sccp_statistic_state(stat_item);

report_or_not_flag = ((taskno < MAX_ACT_TASK_NUMBER)
                      &&(n7stat_stat_item_valid(stat_item))
                      &&(act_task_table[taskno].result_data != 0));
```

(4) 循环、判断等语句中若有较长的表达式或语句，则要进行适当的划分，长表达式要在低优先级操作符处划分新行，操作符放在新行之首。

示例：

```
if((taskno < max_act_task_number)
   &&(n7stat_stat_item_valid(stat_item)))
{
    ... // program code
}

for(i = 0, j = 0; (i < BufferKeyword[word_index].word_length)
                 &&(j < NewKeyword.word_length); i++, j++)
{
    ... // program code
}

for(i = 0, j = 0;
    (i < first_word_length) &&(j < second_word_length);
    i++, j++)
{
    ... // program code
}
```

(5) 若函数或过程中的参数较长,则要进行适当的划分。

示例:

```
n7stat_str_compare((BYTE *)&stat_object,
                  (BYTE *)&(act_task_table[taskno].stat_object),
                  sizeof(_STAT_OBJECT));

n7stat_flash_act_duration(stat_item,frame_id * STAT_TASK_CHECK_NUMBER
                        + index,stat_object);
```

(6) 不允许把多个短语句写在一行中,即一行只写一条语句。

示例:如下例子不符合规范。

```
rect.length=0; rect.width=0;
```

应如下书写:

```
rect.length=0;
rect.width  =0;
```

(7) if、for、do、while、case、switch、default 等语句自占一行,且 if、for、do、while 等语句的执行语句部分无论多少都要加括号{ }。

示例:如下例子不符合规范。

```
if(pUserCR == NULL)return;
```

应如下书写:

```
if(pUserCR == NULL)
{
    return;
}
```

(8) 对齐只使用空格键,不使用【Tab】键。

说明:为避免使用不同的编辑器阅读程序时,因【Tab】键所设置的空格数目不同而造成程序布局不整齐。不要使用 BC 作为编辑器合版本,因为 BC 会自动将 8 个空格变为一个【Tab】键,因此使用 BC 合入的版本大多会将缩进变乱。

(9) 函数或过程的开始、结构的定义及循环、判断等语句中的代码都要采用缩进风格,case 语句下的情况处理语句也要遵从语句缩进要求。

(10) 程序块的分界符(如 C/C++ 语言的大括号{和})应各独占一行并且位于同一列,同时与引用它们的语句左对齐。在函数体的开始,类的定义,结构的定义,枚举的定义以及 if、for、do、while、switch、case 语句中的程序都要采用如上的缩进方式。

示例:如下例子不符合规范。

```
for(...){
    ... // program code
}

if(...)
{
    ... // program code
}

void example_fun(void)
{
    ... // program code
}
```

应如下书写:

```
for(...)
{
    ... // program code
}

if(...)
{
    ... // program code
}

void example_fun(void)
{
    ... // program code
}
```

(11) 在对两个以上的关键字、变量、常量进行对等操作时,它们之间的操作符之前、之后或者前后要加空格;进行非对等操作时,如果是关系密切的立即操作符(如 $\rightarrow$),后不应加空格。

说明:采用这种松散方式编写代码的目的是使代码更加清晰。由于留空格所产生的清晰性是相对的,所以,在已经非常清晰的语句中没有必要再留空格,如果语句

已足够清晰则括号内侧(即左括号后面和右括号前面)不需要加空格,多重括号间不必加空格,因为在C/C++语言中括号已经是最清晰的标志了。在长语句中,如果需要加的空格非常多,那么应该保持整体清晰,而在局部不加空格。给操作符留空格时不要连续留两个以上空格。

示例:

① 逗号、分号只在后面加空格。

```
int a,b,c;
```

② 比较操作符,赋值操作符“=”、“+=”,算术操作符“+”、“%”,逻辑操作符“&&”、“&”,位域操作符“<<”、“>>”等双目操作符的前后加空格。

```
if(current_time>= MAX_TIME_VALUE)
a = b + c;
a * = 2;
a = b ^ 2;
```

③ “!”、“~”、“++”、“--”、“&”(地址运算符)等单目操作符前后不加空格。

```
*p = 'a';           // 内容操作“*”与内容之间
flag = ! isEmpty;   // 非操作“!”与内容之间
p = &mem;           // 地址操作“&”与内容之间
i++;                // “++”,“--”与内容之间
```

④ “→”、“.”前后不加空格。

```
p→id = pid;        // “→”指针前后不加空格
```

⑤ if、for、while、switch 等与后面的括号间应加空格,使 if 等关键字更为突出、明显。

```
if(a>= b && c>d)
```

(12) 一程序以小于 80 字符为宜,不要写得过长。

1.2 注 释

(1) 一般情况下,源程序有效注释量必须在 20%以上。

说明:注释的原则是有助于对程序的阅读理解,在该加的地方都加上,注释不宜太多,也不能太少,注释语言必须准确、易懂、简洁。