

---

# AX2227+CW6639 培训



**Buildwin**

**建荣科技**

Member of AppoTech Group

**建荣集成电路科技（珠海）有限公司**

版本:V1.2

日期:2014-04-14



**Buildwin International (ZhuHai) Limited**

**建荣集成电路科技(珠海)有限公司**

Member of AppoTech Group

---

## 目录

<b>1. 硬件.....</b>	<b>3</b>
1.1 进入升级模式.....	3
1.2 设计注意事项.....	3
<b>2. 软件.....</b>	<b>4</b>
2.1 SDK 目录结构.....	4
2.2 SDK 配置文件及使用规则.....	4
2.3 如何新建自己的 USER 配置.....	5
2.4 修改按键.....	6
2.5 修改显示.....	7
2.6 修改语音资源.....	8
<b>3. 工具使用 .....</b>	<b>8</b>
<b>4. 注意事项 .....</b>	<b>8</b>
4.1 SPIFLASH 容量.....	8
4.2 生产 SPIFLASH .....	8
<b>5. 版本变更说明 .....</b>	<b>9</b>



## 1. 硬件

### 1.1 进入升级模式

#### 1) 通过 USB 连接电脑升级

第一次升级可以不用拉低 P06 也可以下载程序进 SPI FLASH，第二次开始就要拉低 P06 再上电才可以进行升级或者短接 FLASH 第 5、6 脚上电（上电后立即松开），也可以进入升级。

#### 2) 用 TF 卡升级

在 flash 已经下载过程序的前提下，把相应 BIN 文件改名字为 updatefw.bin 再放进卡的根目录，拉低 P06 上电可以升级。

### 1.2 设计注意事项

#### 1) SPI\_Flash 方案 FM 抗干扰注意事项:

(1) FM 不能与 SPI\_Flash 共用数字地，FM 需要放置在模拟地，SPI\_Flash 放置在数字地。隔离地线传导干扰；

(2) SPI\_Flash 部分，可以分别在电源与地线串联磁珠。隔离 SPI 通信时，电源上产生的干扰；

(3) FM 部分，可以在电源上串联磁珠，隔离电源干扰；

(4) SPI\_Flash 靠近主控放置，数据线尽量短；

(5) FM 放置位置应该尽量远离 SPI\_Flash 与 MCU；

(6) FM 天线尽量不要靠近 SPI\_Flash 与 MCU。

#### 2) 蓝牙注意事项:

(1) 可以用 VDDIO 供电，但要接 RC 滤波；

(2) 蓝牙天线底下不要走线，蓝牙模块 PCB 开槽；

(3) MIC 电路和功放电路要远离蓝牙天线，避免蓝牙天线干扰 MIC 和功放；

(4) 蓝牙尽量放置在 PCB 边缘处，同时也要考虑外壳对天线的影响；

(5) 蓝牙天线要尽量宽和近回到供电处，尽量在一面布线，底面一整片地。

#### 3) MIC 注意事项:

(1) 避免靠近蓝牙数据线和 SPI Flash 数据线；

(2) MIC 相关电路不要靠近蓝牙天线。



## 2. 软件

软件是由 OTP 程序和 SPIFlash 程序组成，OTP 程序在 AX2227 芯片内部，一般预先烧写好。SDK 中开发的是 SPIFlash 程序，开发环境是 IAR 720H。

打开 SDK 工程项目，编译后会在“APP\_LLP\Debug\Exe”目录下生成“APP.cod”文件，cod 文件是开发的原始文件，需要通过 MPTool 工具下载到样机中加密，然后导出成 bin 文件，用于量产 SPIFlash。

### 2.1 SDK 目录结构

display	
ledseg	;LED 段码显示
led	;LED 指示灯
ocx.c	;显示控件
makecode	;工具相关
task	
task.c	;公共消息
task_bt.c	;蓝牙流程
task_fm.c	;FM 流程
task_music.c	;音乐播放流程
task_poweroff.c	;关机和唤醒流程
task_record.c	;AUX 流程
task_usbdev.c	;USB 读卡器、USB 声卡流程
user	
user_config	
user_cfgdef.h	;可用配置定义
user_config.h	;默认配置
user_config_2227.h	;Demo 用户配置
user_configCola.h	;Cola 用户配置
user_configLepa.h	;Lepa 用户配置
user_config_s10.h	;S10 用户配置
user_fm	;FM 模块驱动
btapi.c	;蓝牙 api
user_fm.c	;FM api 及 I2C 驱动
user_io.c	;IO 初始化及设备检测
user_key.c	;按键、低电检测相关
user_timer.c	;定时器相关
main.c	;主流程

### 2.2 SDK 配置文件及使用规则

在 user\_cfgdef.h 中，用于定义可供用户选择的选项列表，例如：支持的 FM 型号、显示主题、ADKEY PORT 口等。如果有新添加的选项，可放在此处。



在 `user_config.h` 中，定义了许多功能，各种功能里面又有多种配置，因为配置较多，可能需要多次使用才能熟悉。初次使用时，建议先关注需要的功能。例如，打开 FM 功能时，需要选择 FM 模块，选择 AUX 通道，定义 I2C 接口等。

在 `user_config` 中，会先加载用户配置，然后再补充定义用户未定义的宏；一般情况都是将未定义功能关闭，防止编译报错。所以，开发新案子时，请尽量不要修改 `user_config.h` 的默认配置，可以新建一个 `user_config_xxx.h` 定义需要的功能，然后包含到 `user_config.h` 中，剩下的不需要的功能则可以通过 `user_config.h` 默认关掉。

## 2.3 如何新建自己的 USER 配置

在 SDK 中，我们提供了 4 个型号的样机配置，用户可以找到与自己最相似的配置，复制一份，然后针对自己的板子定义功能，修正一些 IO。

### 2227 Demo 基本功能：

- 通过长按键切模式；
- 6 键 ADKEY；
- 5COM7SEG 段码显示；
- 支持 SD 卡、U 盘音乐播放；
- 支持 AUX、FM、USBDEV；
- 支持蓝牙音箱、蓝牙通话；

### S10 基本功能：

- 上电检测 BT/MP3 模式；
- 3 键 IO KEY；
- 两颗 LED 显示；
- 支持 SD 卡音乐播放；
- 支持 AUX、FM；
- 支持蓝牙音箱、蓝牙通话；

### Cola 基本功能：

- 通过长按键切模式；
- 3 键 IO KEY；
- 一颗 LED 显示；
- 支持 SD 卡、U 盘音乐播放；
- 支持 AUX、FM、USBDEV；
- 支持蓝牙音箱、蓝牙通话；

### Lepa 基本功能：

- 3 键 IO KEY；
- 一颗播放指示灯、一颗低电指示灯；
- 支持蓝牙音箱、蓝牙通话；



### 通用功能:

- 语音播报;
- 低电检测;
- 蓝牙休眠及唤醒;
- 来电报号;
- SD 卡离线升级程序;

## 2.4 修改按键

在 user\_config\_xxx.h 中, 通过宏 USE\_ADKEY 定义按键类型, 若选择 IO 按键, 需要定义 USE\_ADKEY 为 0, 并且定义 3 个按键相关宏; 若选择 AD 按键, 需要定义 USE\_ADKEY 为 1, 并且选择 ADKEY\_PORT, 然后在 user\_key.c 中修改不同电阻对应的 AD 值表。

- IO 按键定义, 以 S10 为例:

```
#define USE_ADKEY                0    //0 为使用 IOKey

#define K_PLAY_INIT()            P0DIR |= BIT(6); P0PU0 |= BIT(6)
#define K_NEXT_INIT()            {P3DIR |= (BIT(0)|BIT(2));\
                                   P3PU0 |= (BIT(0)|BIT(4));\
                                   P3PU0 &= ~(BIT(1)|BIT(5));}

#define K_PREV_INIT()            //在 K_NEXT_INIT 中初始化
#define IS_K_PLAY()               ((P0 & BIT(6)) == 0)
#define IS_K_NEXT()               ((P3 & BIT(0)) == 0)
#define IS_K_PREV()               ((P3 & BIT(2)) == 0)
```

- AD 按键定义, 以 Demo 为例:

```
#define USE_ADKEY                1    //1 为使用 ADKey
#define ADKEY_PORT                ADKEY_P06

#if (USER_SEL == USER_2227)
/*****
* Key          : USER_2227
*****/

#define KEY1_NUM        6                //6 个按键
IAR_CONST u8 tbl_key1_adc[KEY1_NUM+1] = {
    0x3F, 0x8C, 0xA1, 0xB4, 0xCA, 0xF4,    //注意从小到大排列
    0xFF                                //0xff 为无按键
};
IAR_CONST u8 tbl_key1_val[KEY1_NUM+1] = {
    T_KEY_HSF, T_KEY_PREV, T_KEY_NEXT, T_KEY_HSF, T_KEY_VOL_DOWN, T_KEY_VOL_UP,
    NO_KEY
}
```



```
};  
#elif ...
```

## 2.5 修改显示

在 user\_config\_xxx.h 中通过宏 THEME\_SELECT 选择显示主题，目前只支持无显示（LED 指示灯）和 LED 数码管显示。

- 无显示方案，以 S10 的双 LED 为例：

```
/*  
 * Module      : 主题选择  
 */  
#define THEME_SELECT          NO_THEME //无显示方案主题  
  
/*  
 * Module      : 系统细节配置  
 */  
#define LED_BLUE_BLINK        1    //是否使用蓝灯闪烁  
#define LED_GREEN_BLINK      1    //是否使用绿灯闪烁  
  
/*  
 * Module      :LED IO 配置  
 */  
//蓝色 LED  
#define LED_BLUE              5  
#define LED_BLUE_INIT()      P2DIR &= ~BIT(LED_BLUE); P2 &= ~BIT(LED_BLUE)  
#define LED_BLUE_ON()        P2 |= BIT(LED_BLUE)  
#define LED_BLUE_OFF()       P2 &= ~BIT(LED_BLUE)  
#define LED_BLUE_TOG()       P2 ^= BIT(LED_BLUE)  
#define LED_BLUE_IS_ON()     (P2 & BIT(LED_BLUE))  
//绿色 LED  
#define LED_GREEN            5  
#define LED_GREEN_INIT()     P1DIR |= BIT(LED_GREEN); P1DRV1 &= ~BIT(7)  
#define LED_GREEN_ON()       P1DRV1 |= BIT(7)  
#define LED_GREEN_OFF()      P1DRV1 &= ~BIT(7)  
#define LED_GREEN_TOG()      P1DRV1 ^= BIT(7)  
#define LED_GREEN_IS_ON()    (P1DRV1 & BIT(7))
```

- LED 数码管显示方案，以 Demo 的 5COM7SEG 为例：

```
/*  
 * Module      : 主题选择  
 */
```



```
#define THEME_SELECT LEDSEG_THEME_DEFAULT
#define LEDSEG_TYPE_SELECT LEDSEG_5C7S_MHZ //ledseg 屏类型选择
若数码管使用的引脚跟 Demo 不一致，还需要修改 led_5c7s.c
```

## 2.6 修改语音资源

- 1) 把所要添加的音乐文件复制到 MP3RESTOOL 软件所在目录下。（注意：音乐文件的文件名必须为英文名，V092 之前版本蓝牙模式下只支持 8k 8bit 的 WAV 语音资源，V093 可以支持 8k、16k、32k 的（8bit 或 16bit）的语音资源）；
- 2) 打开 MP3RESTOOL 软件，点击“生成语音播报表”，自动生成 mp3res.h 和 mp3res.bin 文件；



- 3) 把 mp3res.bin 拷贝到 APP\_LLPI\Debug\Exe\bin 目录下，把 mp3res.h 文件拷贝到 APP\_LLPI\user\mp3res 目录下；

- 4) 在需要调用语音的地方调用语音播放函数，语音播放函数主要有 3 个：

```
void mp3_res_play(u8 music_name); //播报语音（在非蓝牙模式下）
void mp3_res_play_wait(u8 music_name); //播报语音，并等待播放完毕（在非蓝牙模式下）
void bt_voice_put(u8 msg); //播报语音（在蓝牙模式下）
```

## 3. 工具使用

- 1) MPTool 是下载 cod 文件的工具，详细步骤可参考《MPTool 工具使用说明.PDF》。
- 2) BT\_TOOL 是修改蓝牙名称、蓝牙地址等参数的工具（V093 及之后版本才支持），方便生产相同功能而蓝牙名称不同的产品（无需重新编译程序），详细步骤可参考《BT\_TOOL 工具使用说明.pdf》。
- 3) MP3RESTOOL 是语音资源生成工具，可参考本文档 2.6 章节。

## 4. 注意事项

### 4.1 SPIFlash 容量

Flash 容量  $\geq$  cod 文件大小 + 128K（64K 是系统参数，64K 是蓝牙配对信息）。

最大能支持 16M Byte 的 Flash。

### 4.2 生产 SPIFlash

- 1) MPTOOL 取消“快速”和“复位”选项；





**Buildwin International (ZhuHai) Limited**

**建荣集成电路科技(珠海)有限公司**

Member of AppoTech Group

- 
- 2) 使用 MPTool 下载 cod 文件到首个样机（一定要用自己生产的样机，贴用于生产的、烧好 OTP 的芯片）；
  - 3) 下载完后，点击“导出程序”，将加密后的 SPIFlash 内容导出，保存成 bin 文件；
  - 4) 使用 BIN 文件来进行生产（量产前注意首片测试）；

## 5. 版本变更说明

版本号	变更日期	变更内容简述	更新人	备注
V1.0	2013.12	初稿	境发、旭毅	
V1.1	2014-1-9	添加新功能说明	境发	
V1.2	2014-4-14	更新说明，增加目录	境发	