**SHRD100 DroneID侦测移植设计说明**

|  |  |
| --- | --- |
| 拟 制 |  |
| 审 核 |  |
| 会签 |  |
| 批 准 |  |

**修订记录**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **修订版本** | **日期** | **作者** | **修改描述** | **备注** |
| V1.0 | 2023.07.08 | 谭绍军 | 初始版本 |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[1 引言 1](#_Toc139724201)

[1.1 目的 1](#_Toc139724202)

[1.2 范围 1](#_Toc139724203)

[1.3 缩略语定义 1](#_Toc139724204)

[1.4 参考资料 1](#_Toc139724205)

[2 移植方案概述 1](#_Toc139724206)

[3 移植方案设计说明 1](#_Toc139724207)

[3.1 模块设计 1](#_Toc139724208)

[3.1.1 Droneid侦测功能 1](#_Toc139724209)

[3.2 移植过程 1](#_Toc139724210)

[3.2.1 Droneid的启动加载流程 1](#_Toc139724211)

[3.2.2 Droneid需要设置的GPIO 1](#_Toc139724212)

[3.2.3 Droneid需要配置到的寄存器 1](#_Toc139724213)

[3.2.1 Droneid需要用到的中断 1](#_Toc139724214)

[4 附件 1](#_Toc139724215)

# 引言

## 目的

本文为“SHRD100 DroneID侦测移植设计说明”，主要用于定义软件功能，供项目组开发人员和软件维护人员阅读。

## 范围

本文档只限于塞防科技项目组研发、测试、产品以及项目相关人员作为内部信息对齐使用，未经公司批准以及书面授权不允许任何人以任何形式对本文档复制、传播、改动。

## 缩略语定义

|  |  |  |
| --- | --- | --- |
| **缩略语** | **全称** | **描述** |
|  |  |  |

## 参考资料

|  |  |
| --- | --- |
| **名称** | **版本** |
| SHRD100 Linux嵌入式软件设计说明 | V1.2 |
| SHRD100 GPIO与HAL移植设计说明 | V1.0 |
|  |  |
|  |  |

# 移植方案概述

Drone ID协议解析功能继承于RTOS版本，移植Linux需要依赖GPIO、HAL和AD9361等模块。

Droneid侦测功能保持不变

解析频段：droneid解析覆盖2.4G / 5.2GHz / 5.8GHz。

软件设置AD9361频率，使设备可对下面四个频点（2422M、2452M、5766.5M、5806.5M）进行扫频。

# 移植方案设计说明

## 模块设计

### Droneid侦测功能

AD9361采样率为51.2M，带宽为50M。射频模块初始化配置如下。



软件对四个频点（2422M、2452M、5766.5M、5806.5M）扫频，频点间切换间隔为650ms。AD9361频点切换流程如下。



PL识别到相关峰后，会将19960个点（每个点64bit）的数据存放到DDR0x60000000地址处，嵌入式软件需要丢掉每个点中的高32bit数据，Droneid数据处理流程如下：



## 移植过程

### Droneid的启动加载流程

（参考<https://adasgitlab.autel.com/shrd/shrd100_ps.git> RTOS版本dev\_embed\_5ev分支）



### Droneid需要设置的GPIO

在移植过程中，调试发现启动程序时不能把所有的GPIO都重新初始化，

比如GPIO\_OutputCtrl(MIO\_PS\_ENET\_RESET, IO\_VOL);

GPIO\_OutputCtrl(MIO\_PS\_ENET\_RESET, IO\_VOH); 会重置PS，还有一些与DroneID功能无关其他GPIO可以屏蔽：

MIO\_ETH\_VCC\_1V2\_EN, //会重置网卡，导致ssh连接中断

MIO\_PS\_USB\_RESET ,

MIO\_PS\_USB\_CS,

EMIO\_FAN\_EN, //输出高电平风扇供电

EMIO\_BUZZER\_EN, //输出高电平蜂鸣器供电电源输出

EMIO\_VBRATOR\_EN, //高电平马达电源输出

EMIO\_GPS\_RESET, //GPS\_RESET

EMIO\_GPS\_FORCE\_ON, //GPS\_FORCE\_ON

EMIO\_WIFI\_WAKEUP\_OUT, //

EMIO\_WIFI\_RST, //WIFI\_RST

以下是需要使用到的几个GPIO口：

EMIO\_SWITCH\_5V\_EN, //高电平使能9361 5V输出，开启后配置AD9361才能生效

EMIO\_ADC\_PDN, //低电平给ADC开启供电

EMIO\_SWITCH\_12V\_EN, //高电平使能开关滤波器12V输出

EMIO\_DX\_CTRL\_OUT0,

EMIO\_DX\_CTRL\_OUT1,

### Droneid需要配置到的寄存器

需要配置到寄存器有：

PL\_Version\_date = axi\_read\_data(PL\_AEAG\_VP0); /\*set threshold\*/

PL\_Version\_time = axi\_read\_data(PL\_AEAG\_VP1); /\*set threshold\*/

axi\_write\_data(PL\_FAN\_PWM\_RATE, 0x0001);

axi\_write\_data(PL\_DLY\_EN\_VTC, 0x0000);

axi\_write\_data(PL\_DLY\_D, 150);

axi\_write\_data(PL\_DLY\_LOAD, 0x1FFF);

axi\_write\_data(PL\_DLY\_LOAD, 0x0000);

axi\_write\_data(PL\_DLY\_EN\_VTC, 0x1FFF);

axi\_write\_data(PL\_SL\_SWITCH, 2);

axi\_write(eCFG\_PWR\_GATE\_L, 0x00200000);

axi\_write(eCFG\_PWR\_GATE\_H, 0x00000000);

axi\_write(eWR\_DDR\_ENABLE, 1); // 使能DDR读写

#if NEWBOARD == 0x04 // 这部分需要转换成使用axi\_write\_data，用于IDELAY控制，与上面重复,需要改进

axi\_write(0x80000024, 0x00); // axi\_write\_data(PL\_DLY\_EN\_VTC, 0x00);

axi\_write(0x80000028, 0x00); // axi\_write\_data(PL\_DLY\_LOAD, 0x00);

axi\_write(0x8000002C, 0xA6); // axi\_write\_data(PL\_DLY\_D, 0xA6);

axi\_write(0x80000028, 0x7F); // axi\_write\_data(PL\_DLY\_LOAD, 0x7F);

axi\_write(0x80000024, 0x7F); // axi\_write\_data(PL\_DLY\_EN\_VTC, 0x7F);

#endif

axi\_write\_data(PL\_AD\_START, 1); // 使能侦测状态

Adrf5250Ctrl(eRF5250\_CHANNEL3); // 设置射频通道

SetSwitchFilterGain(1500); // 设置开关滤波器增益

axi\_write\_data( PL\_IRDRATE, IRD\_RATE\_5\_12us ); //设置打击速度间隔, 枪的设置可以删除

### Droneid需要用到的中断

当PL识别到相关峰后，会将19960个点（每个点64bit）的数据存放到DDR0x60000000地址处，并产生READ\_DDR\_BURST\_ID 141中断，

Droneid需要捕获这个中断，然后从DDR映射的虚拟地址读取19960\*8字节数据，取低32位数据存到static uint32\_t bufferAdc[VALID\_ADC\_LEN];

再传给算法进行分析处理。

【注意：需要旁边有无人机发射信号，Tracer才能侦测到信号产生数据并触发中断】

fd **=** open**(**"/dev/PL-burst\_irq\_event"**,** O\_RDONLY**);**

**while(**1**)**

**{**

fd\_set fdset**;**

struct timeval time**;**

FD\_ZERO**(&**fdset**);**

FD\_SET**(**fd**,** **&**fdset**);**

time**.**tv\_sec **=** 0**;**

time**.**tv\_usec **=** 300000**;** // 需要 < 650ms

ret **=** select**(**fd**+**1**,** **&**fdset**,** **NULL,** **NULL,** **&**time**);**

**if** **((**ret **>** 0**)** **&&** **(**FD\_ISSET**(**fd**,** **&**fdset**)** **>** 0**))** **{**

GetFreqRegValue();

memcpy**(**bufferAdc\_64bit**,** **(**uint64\_t **\*)**sourceAddr**,** VALID\_ADC\_LEN **\*** 8**);**

//numDroneID **=** algorithm\_Run1**(**bufferAdc**,** VALID\_ADC\_LEN**,** outList**);**

//UpdataDroneInfo**(**outList**,** numDroneID**);**

**}**

**}**

close**(**fd**);**

从DDR读取到的数据格式如下：

[12340001000d0005] [ffffffffffffffff] [123400030ff60fff] [ffffffffffffffff] [12340005000f0ffb] [ffffffffffffffff] [123400070fff0fff] [ffffffffffffffff]

[123400090ffd0ffc] [ffffffffffffffff] [1234000b00000fff] [ffffffffffffffff] [1234000d0ffb000b] [ffffffffffffffff] [1234000f001e0016] [ffffffffffffffff]

[123400110ffd0012] [ffffffffffffffff] [123400130ffb0ff7] [ffffffffffffffff] [12340015001f0010] [ffffffffffffffff] [123400170ff7000f] [ffffffffffffffff]

[123400190fff000f] [ffffffffffffffff] [1234001b0ff30fff] [ffffffffffffffff] [1234001d001e0017] [ffffffffffffffff] [1234001f0ffb0ff7] [ffffffffffffffff]

[123400210ff60ffd] [ffffffffffffffff] [12340023000d0ff9] [ffffffffffffffff] [123400250ffd0fff] [ffffffffffffffff] [12340027000f000c] [ffffffffffffffff]

[123400290ffb0ffd] [ffffffffffffffff] [1234002b0ffd0fec] [ffffffffffffffff] [1234002d0ff90fff] [ffffffffffffffff] [1234002f0ff70fff] [ffffffffffffffff]

[123400310ffb0005] [ffffffffffffffff] [1234003300090fe7] [ffffffffffffffff] [123400350ff90ffb] [ffffffffffffffff] [123400370fff0016] [ffffffffffffffff]

[1234003900150fff] [ffffffffffffffff] [1234003b0fff0ff9] [ffffffffffffffff] [1234003d0ff50ffa] [ffffffffffffffff] [1234003f0ffb0fff] [ffffffffffffffff]

【此非最终方案，后续会根据FPGA提供方案更新数据读取流程。】

# 附件

无。