



《射频识别（**RFID**）原理与应用》

赵帅锋

信息科学研究所

计算机与信息技术学院



1. 原理机试验

目的：掌握编码与调制知识，加深对课堂理论学习的理解；

手段：使用示波器，观察解调信号波形；

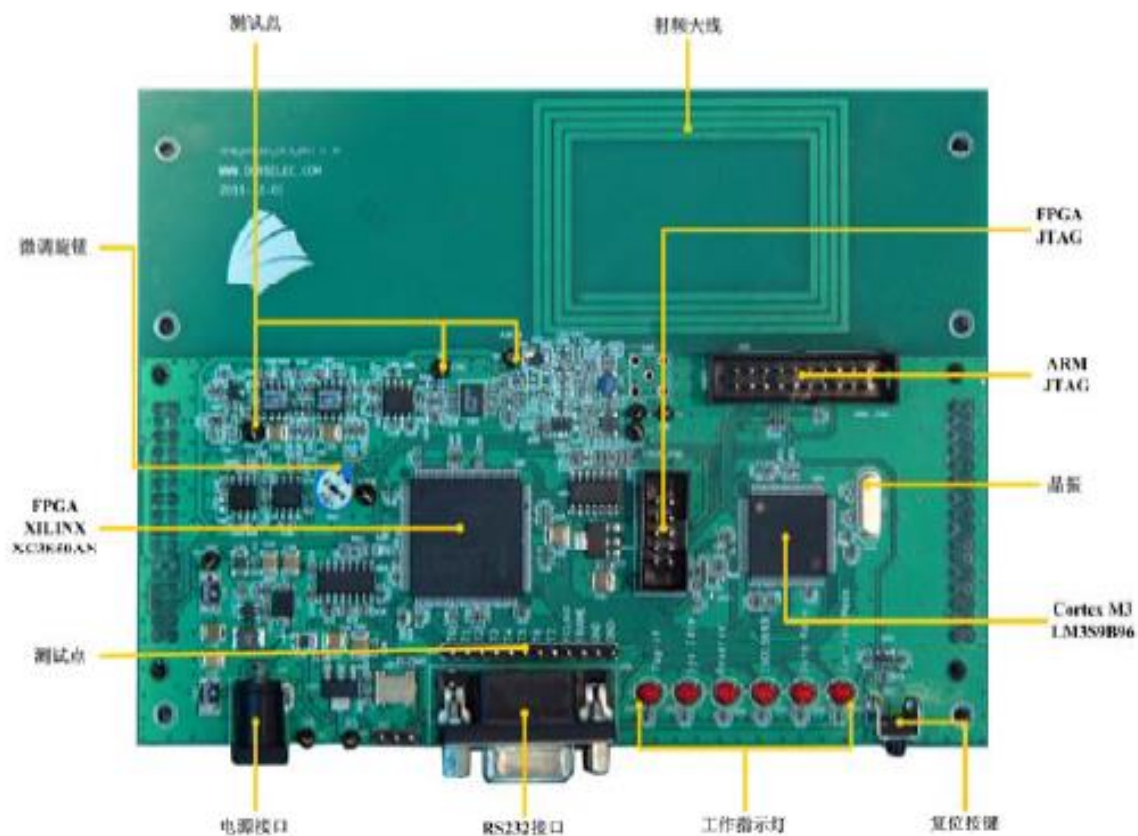
以IS015693协议为例；

实验报告：

记录并结合理论分析每个实验的结果（图形）

实验环境

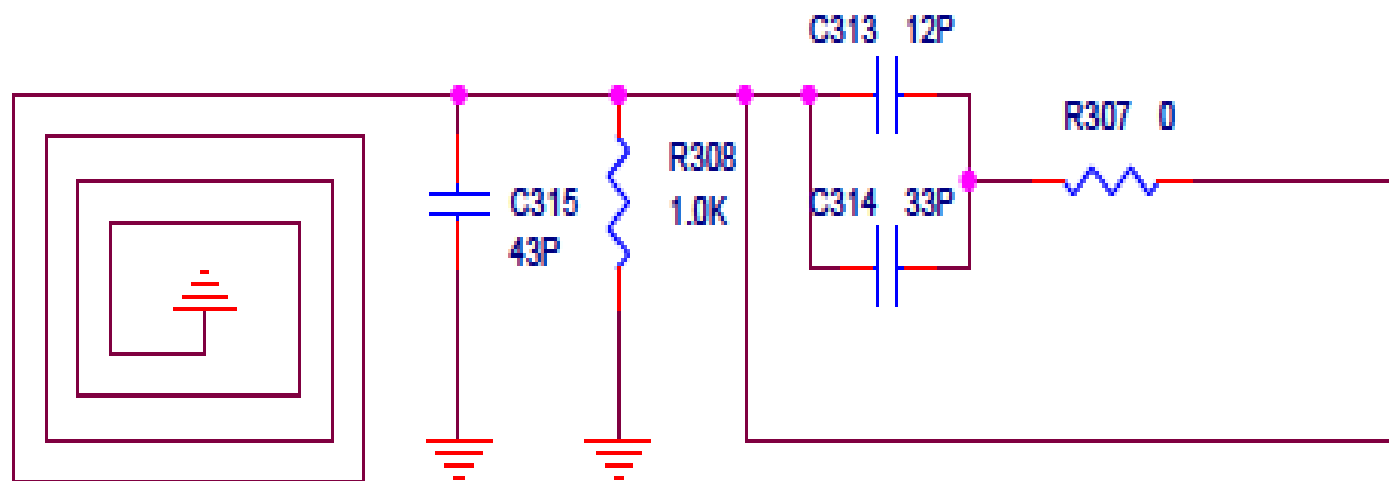
- 左侧：测试点；右侧：实际板子，具有USB接口，以及C315的位置；





测试点

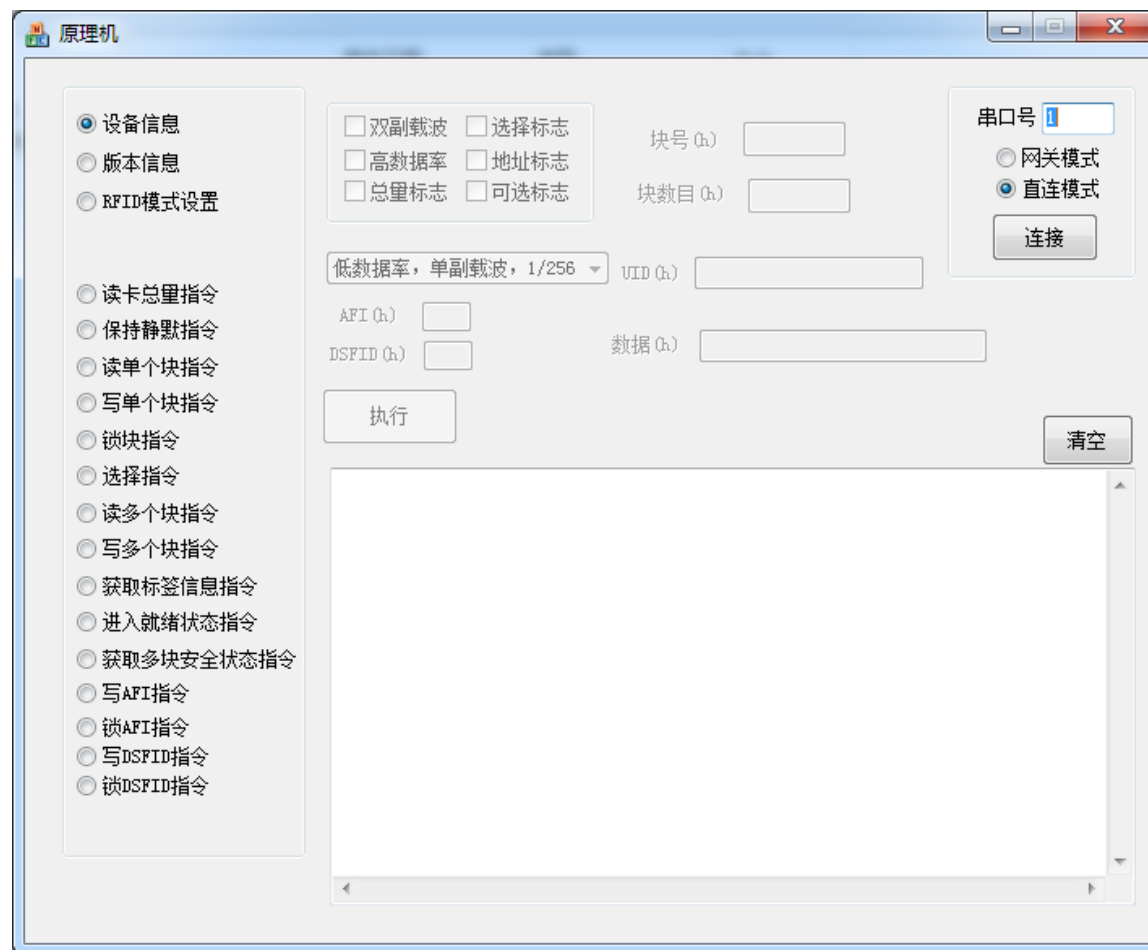
- C513的，天线位置





实验环境

■ 软件: HFPRIN.exe



实验环境

- 示波器 RIGOL DS1052D



OURS-RFID-RP实验指导书2012102

- 存储示波器





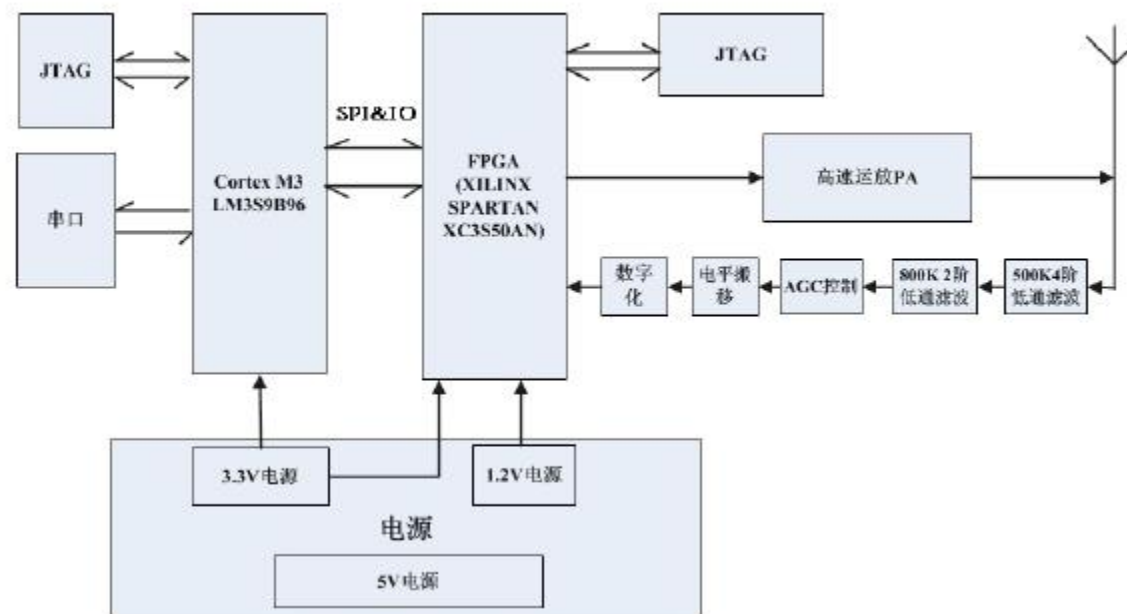
实验指导书

- 指导书：OURS-RIFD-RP实验指导.pdf
 - 实验箱配套;
- 第三章RFID原理机实验
 - 3.1 介绍原理图以及通信协议;
 - 3.2 电路板丝印图;
 - 3.3 编码和调制;
 - 支持编码和调制实验;
 - 3.4 “ISO15693协议标签操作”，运行HFPRIN.exe程序，由HFPRIN.exe向原理机模块发送ISO15693的命令，原理机模块执行ISO15693的命令，并向HFPRIN.exe返回结果；主要用于演示编码和调制功能。实验表明：
 - 使用示波器观察到的波形和理论吻合;
 - 编码和调制技术实现了reader和transponder之间通信;

电路板框图

- MPU: LM3S9B96, 主要作用: 1. 通过RS232口, 和操作显示终端(比如PC机)通信, 2, 从串口接收控制端发来的命令, 发送到FPGA中; 其中一些命令控制FPGA运行, 比如: 运行哪种协议; 另一些命令由FPGA实现组帧、编码、调制后, 发送给VICC, 并接收VICC的结果, 比如15693命令;
- RFID协议的编码、调制等由FPGA实现; MPU和FPGA之间通过SPI总线连接。MPU发送命令, FPGA返回命令执行的结果。

HF 原理机系统框图





通信协议说明

- 串口：
 - 115200bps, 8bits, 1 start/stop bits, 无教研;
- 命令与响应

主机下传指令一般格式如下:

字节 1	字节 2	字节 3	字节 n	字节 n+3
HEAD	LEN	CODE	DATA	CRC
0x55				

从机上传数据（应答）一般格式如下:

字节 1	字节 2	字节 3	字节 n	字节 n+3
HEAD	LEN	CODE	DATA	CRC
0xAA				

- LEN是DATA的长度, 当LEN=0时, 没有DATA和CRC;
- CRC是DATA的模2加校验;



通信协议说明

- 0x02: 设置15693的VCD和VICC的编码与调制模式
 - 与命令保持一致
- 0x10/0x11, 访问FPGA;
- 0x20: inventory
- 0x21: 通过mcu->fpga访问VICC.
 - MCU和FPGA搭建一条通路;

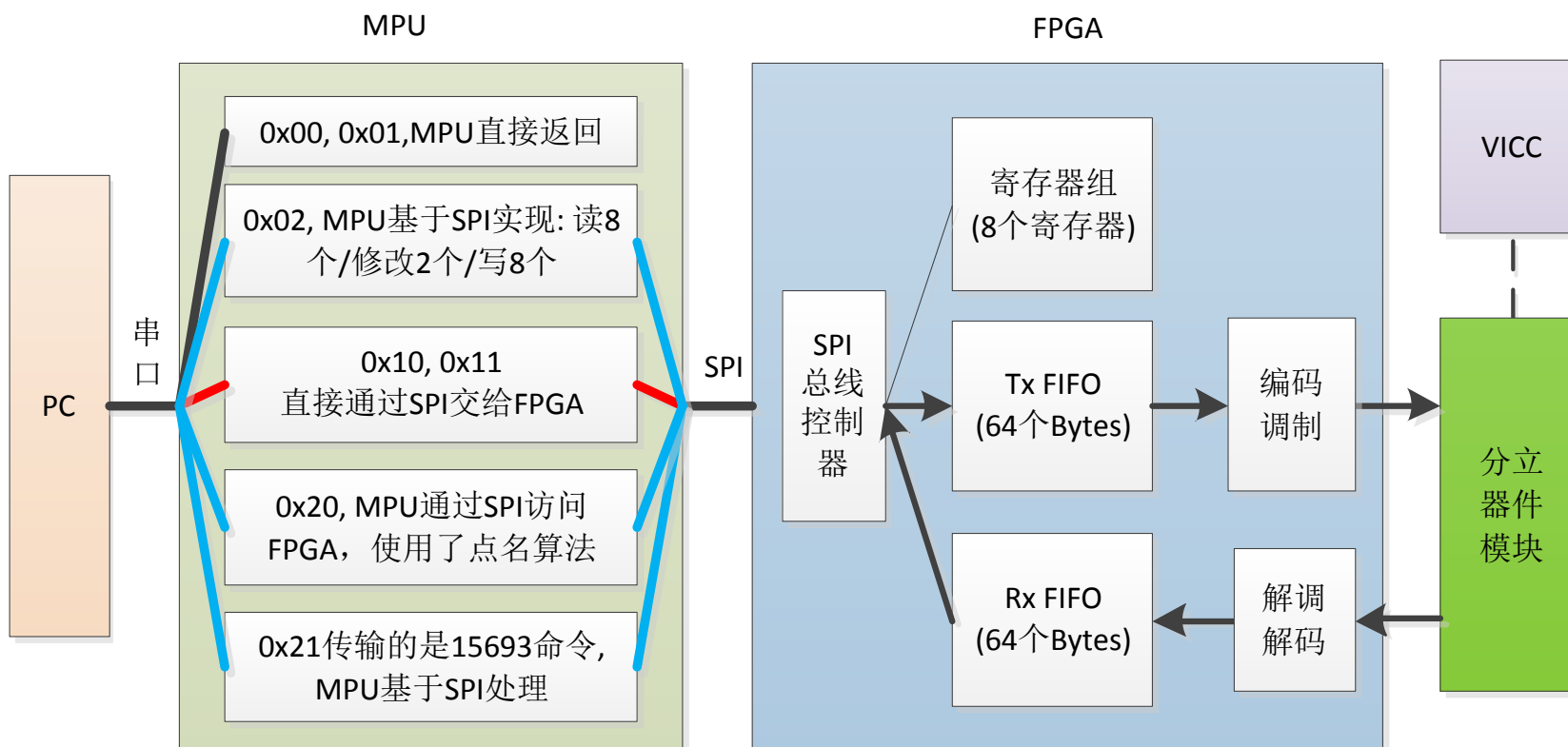
通讯中用到的指令如下:

序号	命令	数据长度		指令解释
		下行	上行	
1	0x00	0	16	设备信息: 固定返回 “www.ourselec.com” 字符串
2	0x01	0	8	版本信息: 返回 “Ver:x.xx” 格式字符串。
3	0x02	1	0	RFID 模式设置字节 (见下例)。
4	0x10	9	0	SPI 接口的写 (透明写), 数据定义详见 “5.2.3”, 注意: 针对 FPGA 内部寄存器写操作要求一次操作固定写 8 字节数据。 针对 FPGA 内部 FIFO 写操作要求预先使用寄存器写设置 FIFO 写操作的数据个数。
5	0x11	2	n	SPI 接口的读 (透明读), 数据定义详见 “5.2.3”, 注意: 针对 FPGA 内部寄存器读操作要求一次操作固定写 8 字节数据。 针对 FPGA 内部 FIFO 读操作要求预先使用寄存器写设置 FIFO 读操作的数据个数。
6	0x20	n	x	读卡总量指令。控制 FPGA 发起一次 VCD→VICC 的请求。
7	0x21	n	x	除读卡总量外其他指令, 控制 FPGA 发起一次 VCD→VICC 请求, 并返回数据。

所谓透明, 是MPU不解释具体含义, 把收到的内容直接传输给FPGA, 提供了PC端对FPGA操作能力。具体数据内容组织由计算机端的软件负责

系统运行框架

■ 软件/运行关系





实验内容及要求

- 合作方式：
 - 2人一组；
- 实验手段：
 - 示波器观察波形；测试点用于触发示波器存储波形，测量点位于C315,用于观察天线端信号的波形。
- 实验内容；
 - 3.3.2/4 载波；
 - 3.3.5 ASK调制；
 - 3.3.8 副载波调制与解调；
- 实验结果（示波器显示结果）拍照；
- 交实验报告；
 - 结合编码和调制理论知识，分析观察结果；



2 ISO/IEC 14443A防碰撞协议

通过pcdsim和piccsim程序，学习ISO/IEC 14443A协议的防碰撞时协议。

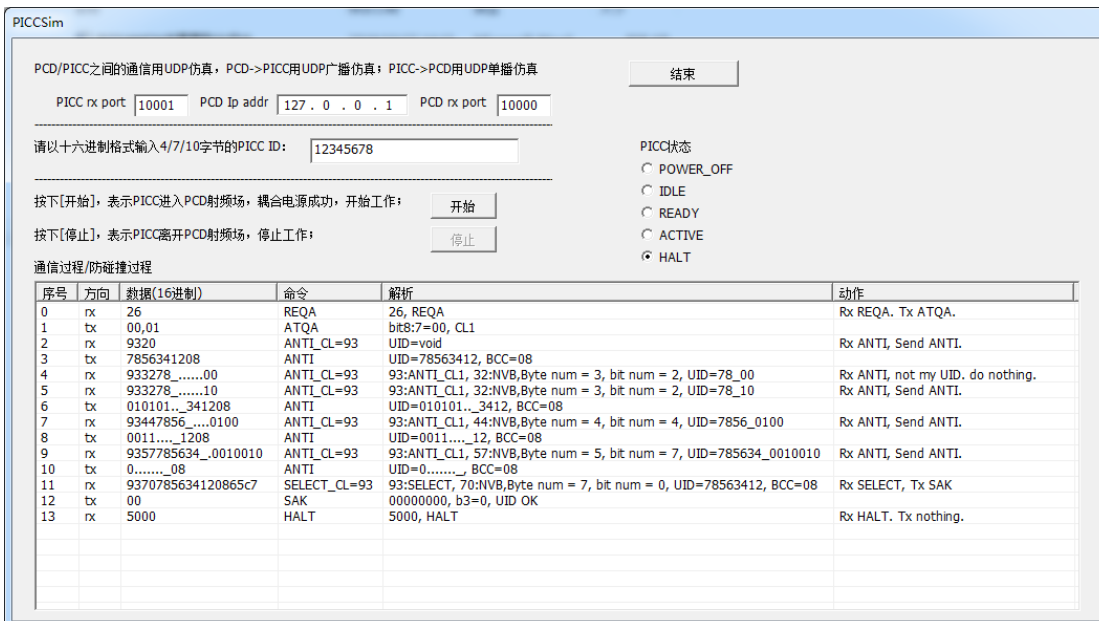
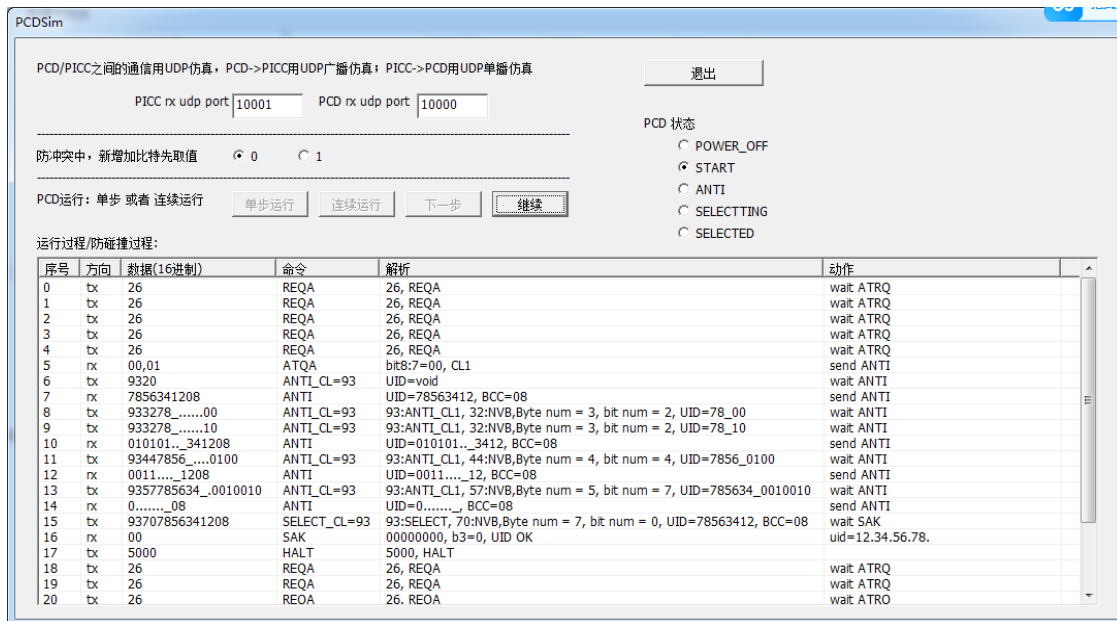
实验报告：

分析三个实验的防碰撞过程



实验环境

- 两个程序：
 - PCDSim, 仿真PCD运行
 - PICCSim, 仿真PICC运行





指导书

- PCDSim和PICCSim使用说明



pcdsim和piccsim使用说

1 概述

RFID 防碰撞协议/算法用于 reader 从当前射频场中存在的多个 transponders 中选择一个进行通信，且不受其他 transponders 的影响；

防碰撞协议/算法由 reader 和 transponders 之间多次通过命令和响应交互实现，这是一个连续的过程，不能延迟和暂时中断（中断意味着防碰撞算法失败），因此配套试验箱没有（也难以）提供防碰撞算法的实验；

为帮助学习和掌握防碰撞算法，开发了 pcdsim 和 piccsim 程序，分别仿真 ISO/IEC 14443A 协议的 PCD 和 PICC，可演示 ISO/IEC14443A 协议防碰撞算法执行过程，以帮助和学习 and 掌握 ISO/IEC14443A 防碰撞算法；



实验内容1

- 使用每个同学的学号作为UID，运行程序，分析防碰撞过程。

以 PICC 的 UID 为 0x12345678，说明防碰撞算法执行过程：

PCDSim	PICCSim
0-3, 发送 REAQ, 等待 ATQA;	
4, 发送 REAQ, 等待 ATQA;	0, 收到 REQA
	1, 发送 ATQA
5, 收到 ATQA	
6, 发送 ANTI, 0x93,20, 无任何 UID 信息;	
	2, 收到 ANTI
	3, 发送 ANTI, 0x78563412, BCC;
7, 收到 ANTI, 0x78563412, BCC;	
8, 仿真发生了碰撞: 随机位置: 10 发送 ANTI: 934278_....,..00, 取 bit9=0	
	4, 收到, 但不响应: 0x56: 0101,0110, 最低两个比特为 10,
9, 发送 ANTI: 934278_....,..10, 取 bit9=1	
	5. 收到, 响应, 发送: 10,1010,3412BCC
10, 收到 0101,01_.3412BCC,...	
...	...

15. 收到所有 uid, 发送 SEL	
	11. 受到 SEL
	12. 发送 SAK
16, 收到 SAK	
17, 发送 HALT	
	13. 收到 HALT
18-发送 REQA	



实验内容2

- 实验7字节和10字节UID，分析防碰撞过程。
 - 7字节构成方式：
 - “201044”+学号
 - 10字节构成方式：
 - “100044”+学号+“1234”
- 实验报告：
 - 每个结果图；
 - 结合14443A协议，分析过程；



3. 防碰撞和访问ISO15693卡

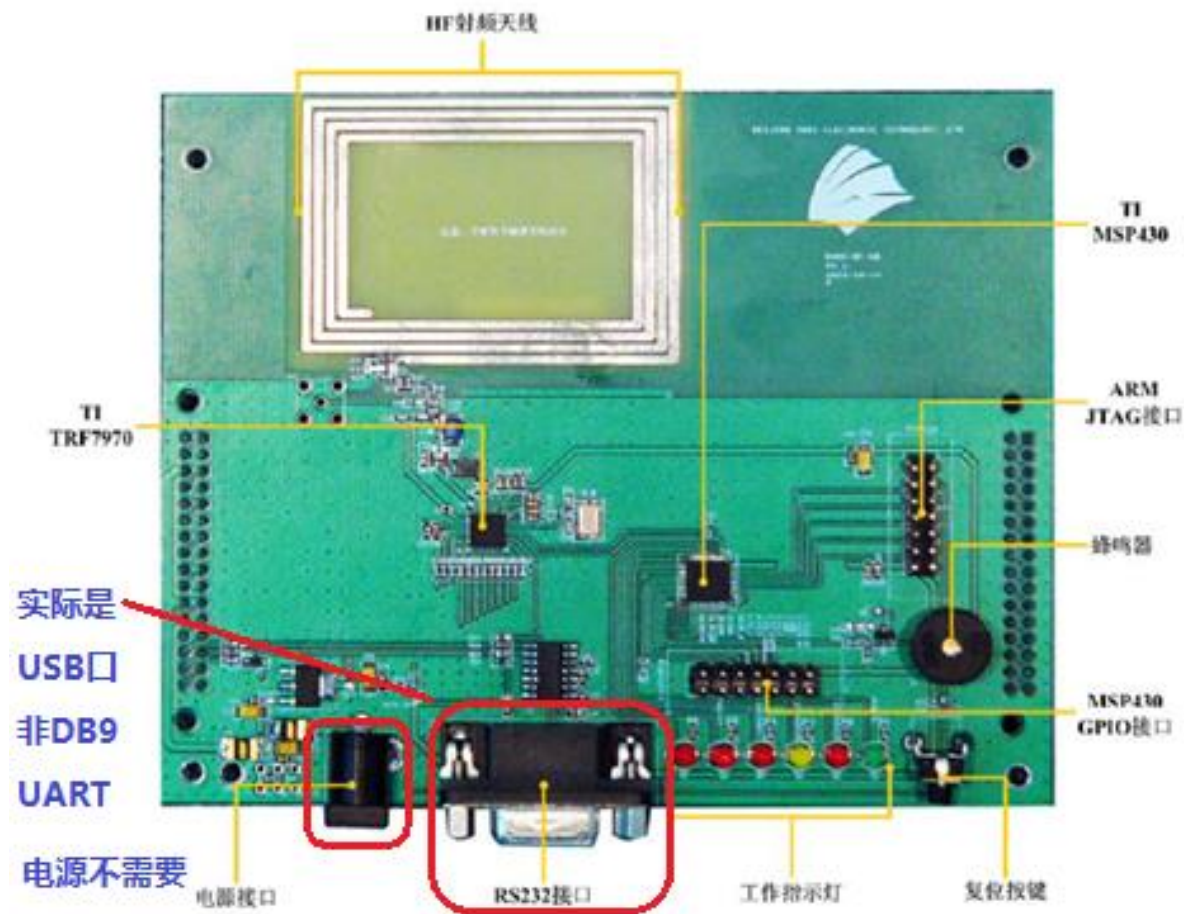
Reader + transponder + computer

实验报告:

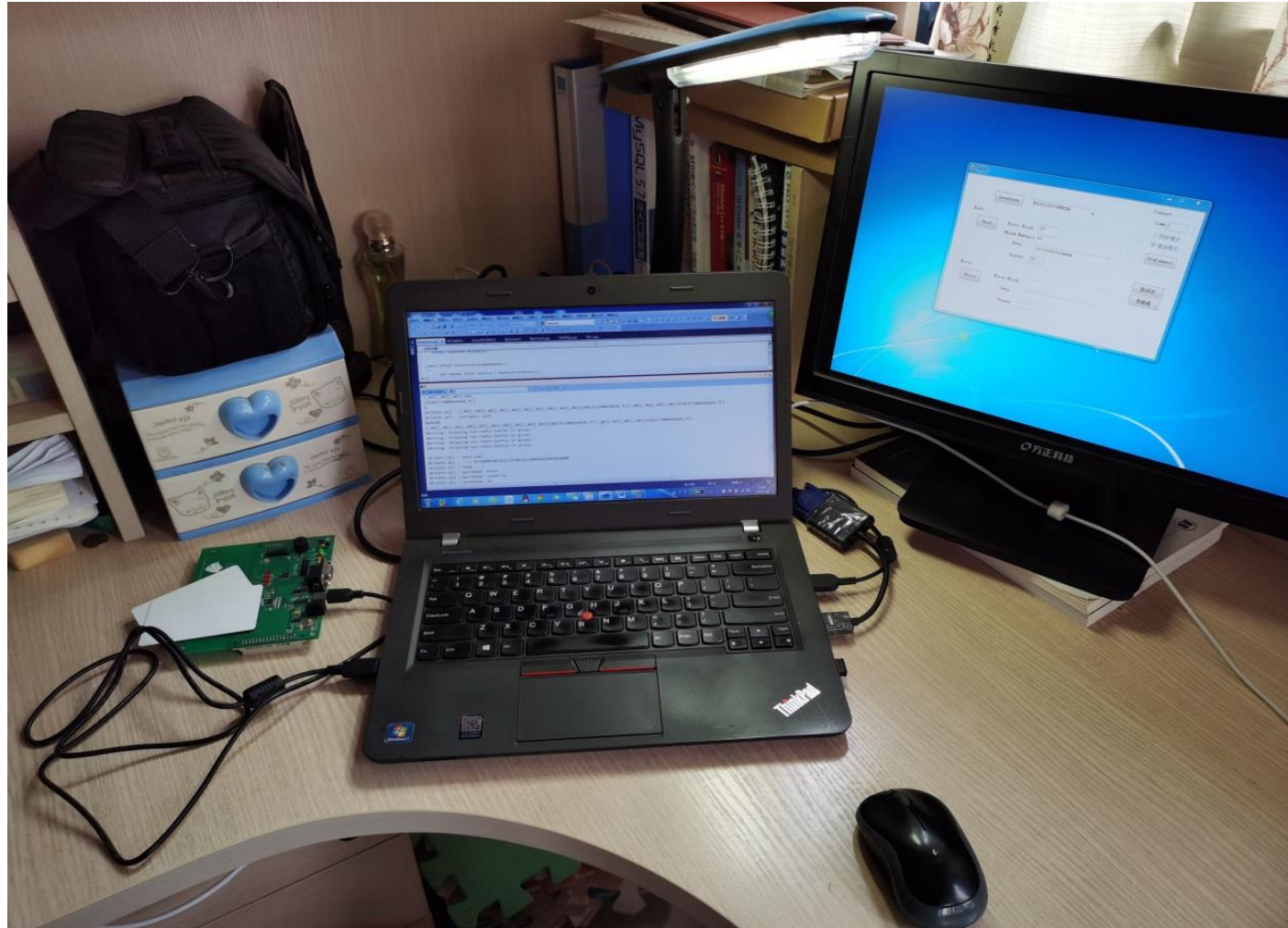
结合15693协议的flag和命令，展示并分析每个实验结果；

实验

- 试验环境: MSP370 + TRF7970

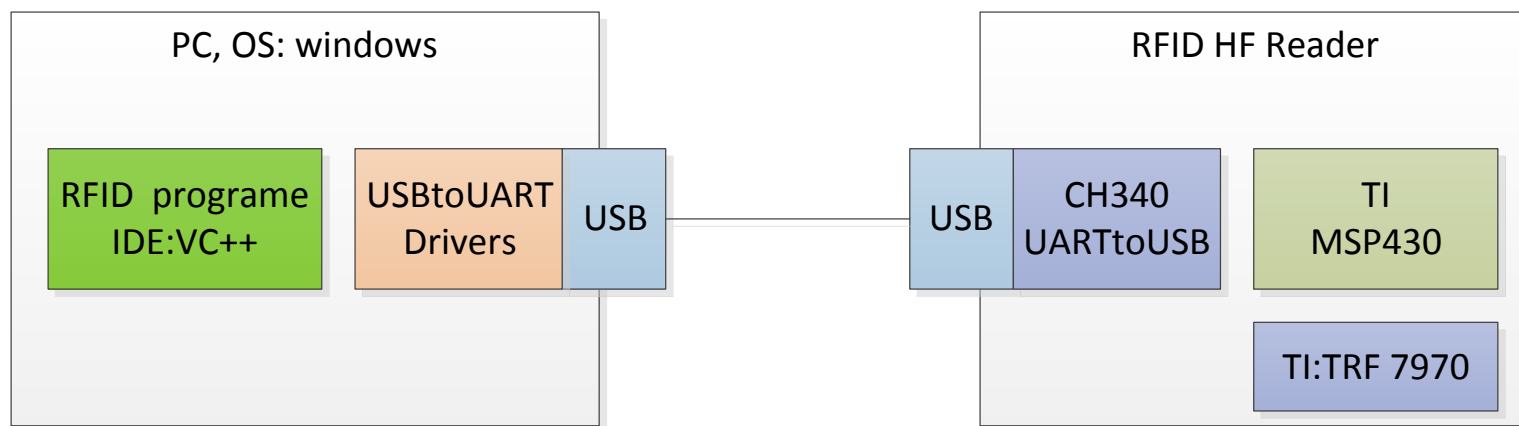


连接





框图



- PC和RFID HF Reader通过USB口连接，但软件接口其实是UART。Reader端使用CH340，CH340和CPU通过UART连接，对外提供USB接口，PC端安装CH340的驱动程序，当Reader连接到USB口上后，PC会发现一个COM端口，软件使用该COM口和Reader进行通信；
- Win7下CH340的drivers由课程提供资料，可以在教学平台中下载，其他版本的windows下的驱动，需要在网络上寻找和安装；



实验简介

■ HF15693(hf15693.dll)

- HF15693生成PC访问reader的接口，该solution是一个DLL project，生成HF15693.dll和HF15693.lib，供HF、HFD和HFExample使用；
- 工作目录为：HF15693，源代码目录为：HF15693/HF15693，两个源码文件为：HF15693.cpp和HF15693.h；研究HF15693.h有助于了解生成的APIs的用法；
- make结束生成dll和lib，分别存于：/HF15693/debug和/HF15693/release下；在使用时，或者把HF15693.dll/lib复制到运行目录下，或者在VS IDE中指明dll/lib的位置；
- dll/lib中提供的一些APIs看不到参数，这是因为其通过全局变量进行传递。这些全局变量定义在HF15693.cpp中，在HF15693.h中通过extern做了说明，保证用户通过#include “HF15693.h”可以直接使用。这些全局变量包括了15693协议中request 命令的flags参数，已经要发送的数据，以及接收到的数据等，详细内容见5.4。



实验简介

■ 主要函数:

- HF15693_API int t15portOpen(int comno);
- HF15693_API int t15portClose();
- HF15693_API int t15setProtocol();
- HF15693_API int t15execute(int cmdno, char *cmdstr, char *reply);
- t115protOpen()用于测试和Reader的连接是否正常;
- t15setProtocol()设置Reader的工作方式, 包括: Reader使用的调制方式 (4选择1, or 256选择1)
 -
- t15execute()实现对transponder的访问; .



实验简介

■ HF.exe

- HF是基于HF15693.dll的程序，用于学习ISO 15693协议的细节；
- 实验指导见：第四章 通信、协议信令部分 4.2 高频部分ISO/IEC 15693；



Command

☒ Inventory

☐ Read Single Block

☐ Write Single Block

☐ Lock Block

☐ Read Multiple Block

☐ Write Multiple Block

☐ Stay Quiet

☐ Select

☐ Reset to Ready

☐ Write AFI

☐ Lock AFI

☐ Write DSFID

☐ Lock DSFID

☐ Get System Info

☐ Get MultiBlock Sec Stat

Request Flags

☐ Sub_carrier

☐ One_slot

☐ HighData_rate

☐ Option

☐ AFI is present

Execute Command

Response Flags

☐ Error

☐ Extension

Error Code

Tag Info

Info Flags

☐ DSFID supported

☐ AFI supported

☐ VICC Memory Size

☐ IC reference

UID

DSFID

AFI

Other fields

Block size in byte

Number of blocks

IC Reference

Set Protocol

Data Coding

1 out of 4

☐ Half Power

☐ Sub_carrier

☐ HighData_rate

Set Protocol

Cmd

Flag

04

Cmd

01

Mask Len

00

CRC16

BC75

Reply

slot

UID

AM RSSI

PM RSSI

0	E00401004B932A00	7	7
1		0	0
2	E00401004B938162	7	6
3		0	0
4		0	0
5	E00401004B9368A5	6	5
6		0	0
7		0	0
8		0	0
9		0	0

PC --> Reader : 040100

PC <-- Reader : [, 40][, 40][6281934B000104E0, 7F][, 40][, 40][, 40]

PC --> Reader : 040100

PC <-- Reader : [002A934B000104E0, 7F][, 40][6281934B000104E0, 7E][, 40]

Connect Comm

串口号

1

☐ 网关模式

☒ 直连模式

DisConnect

Exit

clear



Command

☐ Inventory

☐ Read Single Block

☒ Write Single Block

☐ Lock Block

☐ Read Multiple Block

☐ Write Multiple Block

☐ Stay Quiet

☐ Select

☐ Reset to Ready

☐ Write AFI

☐ Lock AFI

☐ Write DSFID

☐ Lock DSFID

☐ Get System Info

☐ Get MultiBlock Sec Stat

Request Flags

☐ Sub_carrier

☒ Addressed

☐ HighData_rate

☐ Option

☐ Select

Execute Command

Response Flags

☐ Error

☐ Extension

Error Code

UID

E00401004B9368A5

(first) Block

03

Num of block

Data

11111111

DSFID

AFI

02

Tag Info

Info Flags

☐ DSFID supported

☐ AFI supported

☐ VICC Memory Size

☐ IC reference

UID

DSFID

AFI

Other fields

Block size in byte

Number of blocks

IC Reference

Set Protocol

Data Coding

1 out of 4

☐ Half Power

☐ Sub_carrier

☐ HighData_rate

Set Protocol

slot	UID	AM RSSI	PM RSSI
0		0	0
1		0	0
2		0	0
3		0	0
4		0	0
5	E00401004B9368A5	7	7
6		0	0
7		0	0
8		0	0
9		0	0

Cmd

Flag

20

Cmd

21

UID

A568934B000104E0

Block

03

Data

11111111

CRC16

11F4

Reply

Flag

00

Data

PC --> Reader : 040100

PC <-- Reader : [, 40][, 40][, 40][, 40][, 40][A568934B000104E0, 7F][, 40]

PC --> Reader : 2020A568934B000104E001

PC <-- Reader : [0011111111]

PC --> Reader : 2020A568934B000104E002

PC <-- Reader : [0000000000]

PC --> Reader : 2020A568934B000104E003

PC <-- Reader : [0000000000]

PC --> Reader : 2021A568934B000104E00311111111

PC <-- Reader : [00]

Connect Comm

串口号

1

☐ 网关模式

☒ 直连模式

DisConnect

clear

Exit



Command

☐ Inventory

☒ Read Single Block

☐ Write Single Block

☐ Lock Block

☐ Read Multiple Block

☐ Write Multiple Block

☐ Stay Quiet

☐ Select

☐ Reset to Ready

☐ Write AFI

☐ Lock AFI

☐ Write DSFID

☐ Lock DSFID

☐ Get System Info

☐ Get MultiBlock Sec Stat

Request Flags

☐ Sub_carrier

☒ Addressed

☐ HighData_rate

☐ Option

☐ Select

Execute Command

Response Flags

☐ Error

☐ Extension

Error Code

UID

E00401004B9368A5

(first) Block

03

Num of block

Data

00000000

DSFID

AFI

02

Tag Info

Info Flags

☐ DSFID supported

☐ AFI supported

☐ VICC Memory Size

☐ IC reference

UID

DSFID

AFI

Other fields

Block size in byte

Number of blocks

IC Reference

Set Protocol

Data Coding

1 out of 4

☐ Half Power

☐ Sub_carrier

☐ HighData_rate

Set Protocol

slot	UID	AM RSSI	PM RSSI
0		0	0
1		0	0
2		0	0
3		0	0
4		0	0
5	E00401004B9368A5	7	7
6		0	0
7		0	0
8		0	0
9		0	0

Cmd

Flag

20

Cmd

20

UID

A568934B000104E0

Block

03

CRC16

EE36

Reply

Flag

00

Data

00000000

PC --> Reader : 040100

PC <-- Reader : [, 40][, 40][, 40][, 40][, 40][A568934B000104E0, 7F][, 40]

PC --> Reader : 2020A568934B000104E001

PC <-- Reader : [0011111111]

PC --> Reader : 2020A568934B000104E002

PC <-- Reader : [0000000000]

PC --> Reader : 2020A568934B000104E003

PC <-- Reader : [0000000000]

Connect Comm

串口号

1

☐ 网关模式

☒ 直连模式

DisConnect

clear

Exit



Command

☐ Inventory

☐ Read Single Block

☐ Write Single Block

☐ Lock Block

☒ Read Multiple Block

☐ Write Multiple Block

☐ Stay Quiet

☐ Select

☐ Reset to Ready

☐ Write AFI

☐ Lock AFI

☐ Write DSFID

☐ Lock DSFID

☐ Get System Info

☐ Get MultiBlock Sec Stat

Request Flags

☐ Sub_carrier

☒ Addressed

☐ HighData_rate

☐ Option

☐ Select

Execute Command

Response Flags

☐ Error

☐ Extension

Error Code

UID

E00401004B9368A5

(first) Block

02

Num of block

03

Data

000000002222222200000000

DSFID

AFI

02

Tag Info

Info Flags

☐ DSFID supported

☐ AFI supported

☐ VICC Memory Size

☐ IC reference

UID

DSFID

AFI

Other fields

Block size in byte

Number of blocks

IC Reference

Set Protocol

Data Coding

1 out of 4

☐ Half Power

☐ Sub_carrier

☐ HighData_rate

Set Protocol

slot	UID	AM RSSI	PM RSSI
0		0	0
1		0	0
2		0	0
3		0	0
4		0	0
5	E00401004B9368A5	7	7
6		0	0
7		0	0
8		0	0
9		0	0

Cmd

Flag

20

Cmd

23

UID

A568934B000104E0

Block

02

Blocknums

03

CRC16

71D3

Reply

Flag

00

Data

000000002222222200000000

PC --> Reader : 2023A568934B000104E00101

PC <-- Reader : []

PC --> Reader : 2023A568934B000104E00101

PC <-- Reader : [00111111110000000000]

PC --> Reader : 2023A568934B000104E00203

PC <-- Reader : [0000000000222222220000000000000000]

Connect Comm

串口号

1

☐ 网关模式

☒ 直连模式

DisConnect

clear

Exit



4 RFID应用之开发demo: HFD.exe



实验简介

■ HFD.exe

- HFD.exe是HF15693.dll的演示程序(基本协议和DLL)，支持通过reader访问15693 card/transponder，包括了防碰撞（获得卡号），read/write block接口；通过该程序，可以进一步了解ISO/IEC 15693协议，熟悉HF15693的APIs使用方法；
- HFD.exe是一个基于对话框的应用，在MFC的框架结构中：

HFD.h/cpp为应用；

- HFDDlg为对话框
- IDo/MySee/是两个操作的demo，支持增加新的功能。自己带Dlg.

■

- 实验指导见：第六章 综合实训 实验三 ISO15693实例实验；



5 RFID应用之：综合应用



实验简介

■ HFExample

- HFExample是一个基于HF15693.dll的较完整的应用程序，实现了一个职工考勤管理系统。

- 其中：

- 这是一个基于对话框的应用；
- 在PC端使用数据库管理数据，数据库为：access；
- 已经创建了三个用户：

■ 实验指导见：第七章 RFID应用方案及演示 二 基于高频RFID职工考勤管理系统

数据库中现在三个用户，管理员- 用户名：admin，密码：admin，普通用户 - 用户名：a，密码：（空）；用户名：user，密码：user。

- 增加用户：点击“增加”按钮，在弹出的“增加用户”框中输入新用户的用户名，密码（可选），权限，备注（可选），然后点击确定，即可在用户管理栏中看到新添加的用户信息。
- 删除用户：在用户管理栏中选择待删除的用户，点击“删除”按钮。
- 修改用户：在用户管理栏中，选择待修改的用户，点击“修改”按钮，即可修改相应信息，修改完后点击“确定”按钮。



Thanks
