

#### 4.7.1 实现的主要功能:

1、进行安全认证。即对军队配发的 USB 硬盘、优盘, 首先由机要部门对其进行技术标记, 然后与保护器一起配发至各用户单位。保护器接主机, 移动存储设备接保护器, 并在保护器识别该标记后, 方可使用。

2、进行数据审计。保护器记录用户双向数据交换的时间、流量等内容, 以备审查。

3、对信息进行加密。主机信息存储格式不变, USB 设备端存储加密信息, 通过保护器实现加脱密。

#### 4.7.2、设计概要

考虑到功耗、功能等方面因素, 盒式保护器设计为嵌入式系统, 结构如图 4-5

图 4-5

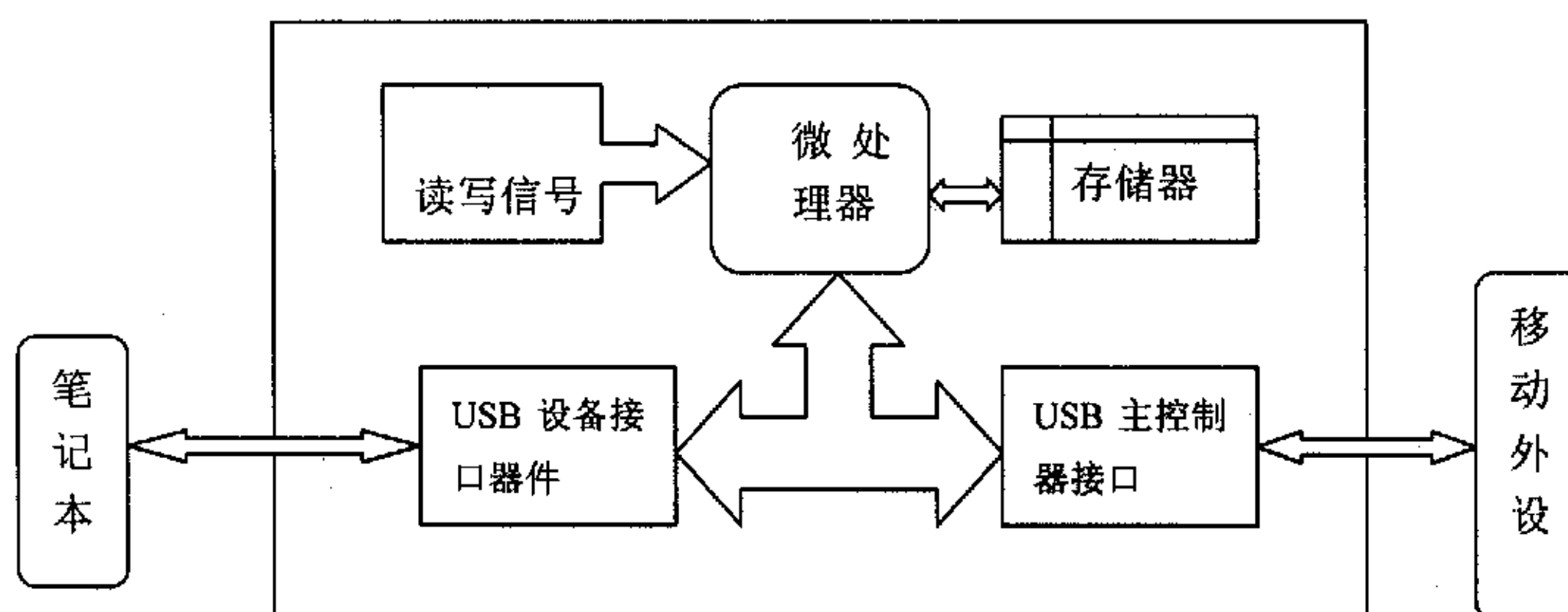


图 4-5 保护器设计框架

硬件上主要是产生一个读写控制信号给微处理器, 由微处理器调用相应的程序来对这一信号作出响应, 从而对微处理器和 Usb 主机的通信作出调整。USB 主机和微处理器的通信是 Usb 主控制器驱动程序的一部分, 因此读写控制, 还是由 Usb 主控制器驱动程序来作出反应。而对于审计和身份认证, 主要在各个功能部件的对应的控制程序中实现。

#### 硬件设计思路

对于盒式保护器, CPU 采用 Philips LPC2210(Arm 7 系列), USB 主控芯片采用 Philips ISP 1161, 部分电路原理图如图 4-7。USB 主控 Philips ISP 1161 支持 2.0 协议标准, 配有 3 端口, 一个端口用于上行, 另两个端口用于下行, 符合我们的设计要求。

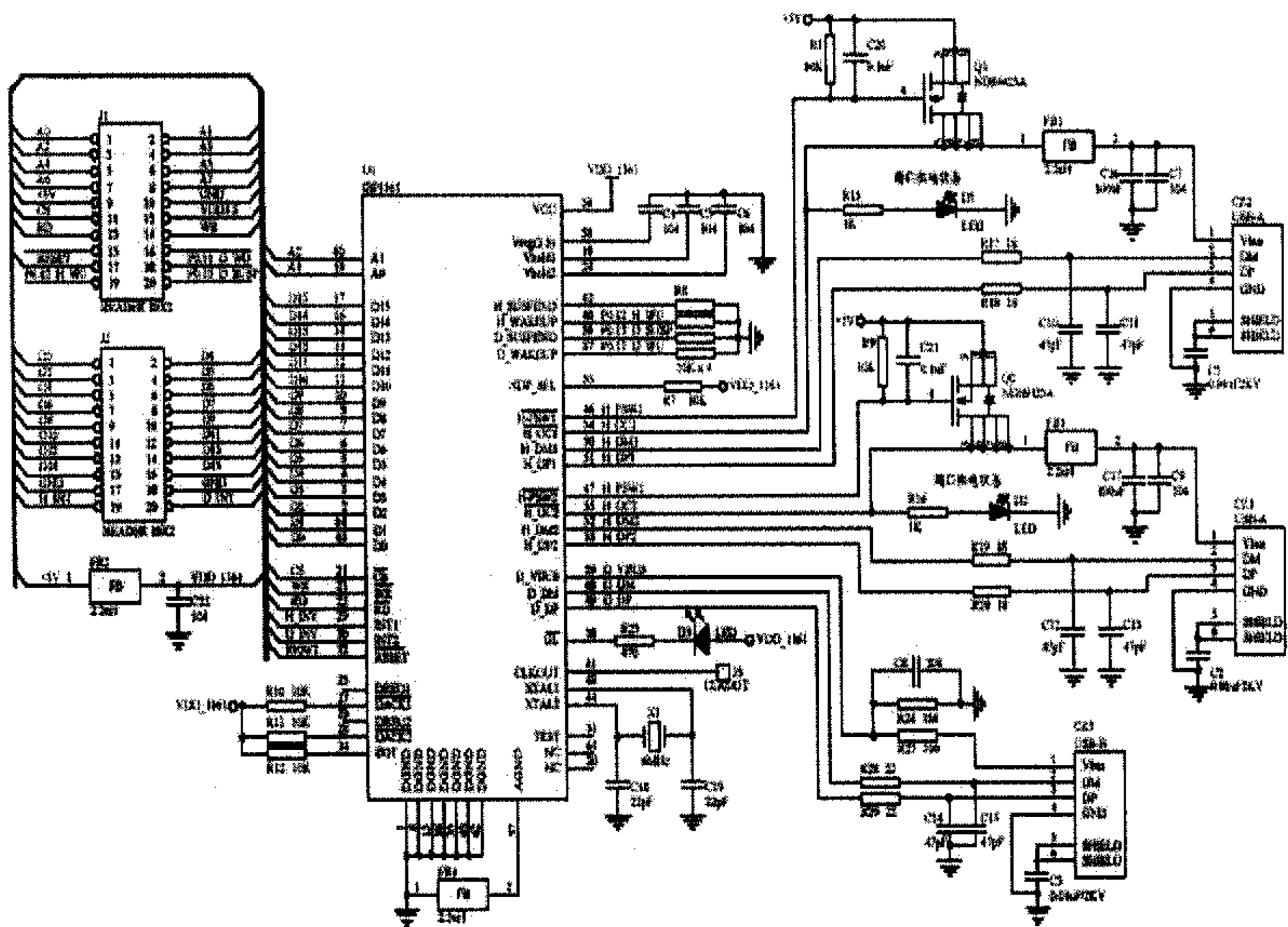


图 4-7 ISP1161 部分电路

### 4.7.3 设备驱动程序开发

按 4.6 节介绍的方法配置环境，选用 VC++ 编译器 Makefile 项目创建开发框架，它生成典型 WDM PnP 驱动程序处理例程，入口和卸载例程 FireInit.c, 即插即用例程 FirePnp.c, 分发例程 FireDispatch.c, 电源管理例程 FirePower.c。下面给出入口和卸载例程 FireInit.c, 即插即用例程 FirePnp.c 主要程序清单。

在入口和卸载例程中主要有 DriverEntry 例程（与 4.2 节介绍的基本相同）、AddFireDevice 例程和 FireDrvUnload 例程。AddFireDevice 例程主要是创建保护器内核设备名和链接名并设置相应字段，程序清单如下：

```
NTSTATUS AddFireDevice(IN PDRIVER_OBJECT DriverObject, IN PDEVICE_OBJECT
PhysicalDeviceObject)
{
    NTSTATUS ntStatus=STATUS_SUCCESS;

    WCHAR KernelDeviceNameBuffer[ ]=L"\\Device\\Phiusb-0";
```