

基于嵌入式开发平台的粘度控制器设计

陈 林

(中国船舶重工集团公司第七一〇研究所,湖北 宜昌 443003)

摘 要: 介绍了一种船舶主机燃油粘度试验台中粘度控制器的设计。包括控制机器人界面的设计、控制器的嵌入式开发平台和开发环境、硬件接口设计原理以及软件设计方法等。

关键词: 嵌入式;主机;粘度;控制

中图分类号: TP29

文献标识码: A

文章编号: 1674-7720(2013)15-0098-04

Design of a fuel oil viscosity controller based on embedded development platform

Chen Lin

(710 Research & Development Institute, CSIC, Yichang 443003, China)

Abstract: This thesis explains the design and implementation of a viscosity controller used on a marine main engine fuel oil viscosity test bench. Including the design of the man-machine interface of the controller, The embedded development platform and the development environment of the controller. The design principle of hardware interface and the software design method.

Key words: embedded; main engine; viscosity; control

近代船舶柴油机为降低营运成本,使用低质燃油,由于在常温下其粘度很高,在管路中难以运输,更不能直接喷入气缸进行燃烧。为了保证船舶主机燃烧良好,能正常运行,燃油粘度必须保持在一个合适的范围内^[1]。半实物主机燃油粘度仿真控制系统,把船上的燃油粘度控制系统迁移到了教学实验室中。本文内容为试验台上粘度控制器的设计与实现。

1 粘度控制机器人机界面

粘度控制机器人机界面如图1所示。矩形区域为LCD显示区域。下方8个按键区域为控制状态选择区,分别有粘度控制、温度控制、手动控制和停止控制、主界面、设置模式、报警复位等按键。右边16个按键的区域为数字键操作区域和符号键区域。包括数字键、修改数据、移动光标、清除数据、确定等按键。

2 控制器硬件设计

在本半实物主机燃油粘度控制系统设计中,燃油粘度控制器需要实现操作模式、控制方式、开关量、数据量等状态的显示,也要实现各个变量的修改,也可以进行报警复位的操作。为实现如上功能,除需要嵌入式网络模块外还需有LCD显示、键盘输入、与上位机通信、开



图1 粘度控制器 HMI

关量输出等模块。

2.1 嵌入式网络模块

控制器以成都英创的EM9260嵌入式网络模块为基础开发完成。EM9260是一款面向工业自动化领域的高性价比嵌入式网络模块,其内核CPU为工业级品质的AT91SAM9260,模块已预装正版Window CE5.0实时多任务操作系统,用户可直接使用Microsoft提供的免费软件开发工具eVC(+SP4)或其他更高版本的开发工具,在EM9260上开发自己的应用程序。英创公司针对EM9260提供了完整的接口底层驱动以及丰富的应用程序范例,用户可在此基础上方便、快速地开发出各种工控产品。

2.2 LCD显示部分

EM9260的主要应用领域是那些需要连续工作的工

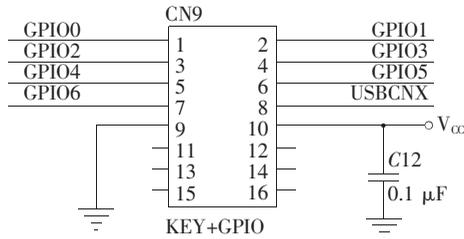


图7 开关量输出接口

3 控制器软件设计

主板安装了操作系统 WindowsCE, WindowsCE 是微软公司嵌入式、移动计算平台的基础, 它是一个开放的、可升级的 32 位嵌入式操作系统, 是基于掌上电脑类的电子设备操作系统, Windows CE 是精简的 Windows 95, 它的图形用户界面相当出色。Windows CE 是有优先级的多任务操作系统, 它允许多重功能、进程在相同时间系统中运行 Windows CE 支持最大的 32 位同步进程。一个进程包括一个或多个线程, 每个线程代表进程的一个独立部分, 一个线程被指定为进程的基本线程, 进程也能创造一个未定数目的额外线程, 额外线程实际数目仅由可利用的系统资源限定^[2]。

软件开发工具为 eVC, eVC 是微软针对嵌入式系统提供的一套功能完善且可免费获取的开发工具, 与 PC 平台通用的 VC 保持了高度的一致性, 同时考虑 C/C++ 的执行效率, 因此 eVC 被作为 EM9260 应用程序开发的首选工具。建立好英创 ARM9 嵌入式模块所需的硬、软件开发环境, 并利用以太网将开发主机和英创 ARM9 嵌入式模块连接成功以后, 用户就可以通过 eVC 开发、调试自己所需的应用程序了。eVC 开发环境和普通的 VC 非常相似, 客户可以通过市面上很常见的大量 VC 书籍资料来进行编程方面的学习。然而, 使用 eVC 开发和 VC 的最大区别之一在于 VC 开发的程序直接运行于普通 PC 机; eVC 开发的应用程序最终要在目标机而非 PC 机上运行调试。这一过程是广大 VC 程序员和刚接触嵌入式系统的工程师所不熟悉的。

3.1 系统主程序软件设计

根据控制器系统设计, 软件开发涉及到 LCD 菜单显示、按键、CAN 通信等操作。控制器主程序流程图如图 8 所示。

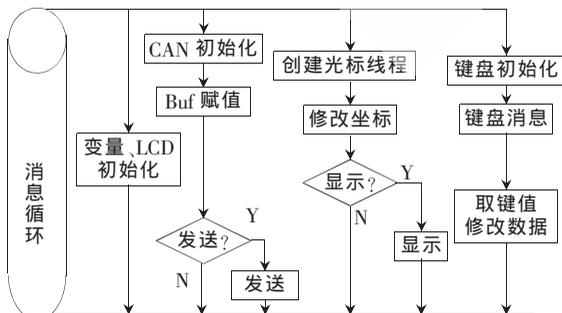


图8 主程序流程图

3.2 矩阵键盘软件设计

矩阵键盘驱动程序每隔 20 ms 扫描一次键盘, 判断是否有按键输入, 如果确认了按键输入, 则产生 WM_KEYDOWN 消息, 紧接着判断按键是否释放, 按键释放后产生 WM_KEYUP 消息, 然后进行下一次扫描, 如此循环。图 9 是按键扫描程序的流程图。



图9 按键扫描程序流程

根据上述流程图, 封装的 KEY_API 类, 该类提供两个方法函数, 如下:

```
//打开键盘扫描
//nRow, nCol 矩阵键盘行列数, 最大为 8x8
BOOL OpenKeyPad(int nRow, int nCol);
//关闭键盘扫描
BOOL CloseKeyPad();
```

通过 KEY_API 类, 在应用程序中, 仅需要调用 OpenKeyPad() 函数, 就能打开键盘扫描程序。当键盘上某个键按下时, 驱动程序将产生 WM_KEYDOWN 和 WM_KEYUP 消息。这两个消息的附加参数 (wParam 和 lParam) 包含的是虚拟键代码和扫描码等信息, 应用程序中可通过 TranslateMessage 函数将 WM_KEYDOWN 和 WM_KEYUP 消息组合转换为一条 WM_CHAR 消息, 该消息的 wParam 附加参数包含了按键字符的 ASCII 码^[3]。

3.3 CAN 通信软件设计

3.3.1 CAN 报文的帧格式

在 CAN2.0B 中存在两种不同的帧格式, 其主要的区别在于标识符的长度, 具有 11 位标识符的帧称为标准帧, 而包括有 29 位标识符的帧称为扩展帧。

3.3.2 启动 CAN 通信接口

CAN 通信接口的驱动程序采用的是 WinCE 下流式驱动程序 (StreamDevice Driver), 并在此驱动程序的基础上为客户封装了一套简单实用的 API 函数。各个函数的定义在 can_api.h 文件下, 在该头文件中对于各个 API 函数均有相应的中文说明。作为流式接口函数通常是和文件系统的 API 函数 (如 CreateFile) 紧密匹配的, 因此在使用英创提供的 CAN 接口的 API 函数时, 首先需要调

用 CreateFile (…) 来获取 CAN 接口设备的句柄 handle, 如使用 CAN1 通信口, 可以用以下函数:

```
m_hCAN=CreateFile(_T("CAN1:"),
    GENERIC_READ|GENERIC_WRITE,
    0,NULL,OPEN_EXISTING,0,NULL);
```

对于 CAN2 通信接口, 只需将其中的名称换成: _T("CAN2:") 即可。在创建 CAN 设备得到有效的 handle 之后, 就可以调用 can_api.h 中定义的相应函数来启动 CAN 设备接口 BOOL CAN_StartChip(HANDLE hDevice); 至此 CAN 通信接口进入工作模式。

3.3.3 设置 CAN 通信接口参数

在进行 CAN 数据通信之前, 需要设置和 CAN 通信相关的一些参数, 包括 CAN 通信的波特率设置以及对接收过滤器的设置。BOOL CAN_SetBaudRate (HANDLE hDevice, BYTE*index) 用于设置 CAN 通信的波特率, 波特率的设置范围为: 10 kb/s~1 Mb/s。具体的定义参见 can_api.h 文件中的注释说明。通过配置接收过滤器, CAN 通信接口可以实现只接收标识符也接收过滤器预设值相一致的报文。接收过滤器由接收码寄存器 ACRn 和接收屏蔽码寄存器 AMRn 来定义, 还可以选择两种不同的过滤器模式, 单过滤器模式或者双过滤器模式。关于 ACR、AMR 中各位的定义, 参见 SJA1000 的数据手册。在英创提供的 API 函数中, 用户可以调用以下函数来实现接收过滤器的设置:

```
BOOL CAN_SetGlobalAcceptanceFilter (HANDLE
hDevice, BYTE*AcceptanceFilter, BYTE size)
```

3.3.4 CAN 通信接口的数据收发

采用的 CAN 通信方案中, CAN 通信的数据收发均采用中断方式, 驱动程序中已自动完成了数据的收发以及内部定义的 CAN 接收缓冲区和发送缓冲区的管理。对于用户开发应用程序来说, 只需要调用英创公司提供的 CAN 通信 API 函数中的收发函数即可。

在进行 CAN 通信应用程序的开发时, 对于 CAN 通信数据接收线程可以采用两种方式: 一种可以采用定时查询, 即定时调用函数 CAN_GetNextReceivedFrame (…) 检测是否接收到 CAN 报文数据; 一种可以利用操作系统的消息机制, 采用事件响应的方式, 一旦硬件接收到数据报文, 底层的驱动接收程序会自动读取报文, 同时发送一个

接收事件。作为应用程序的接收线程在等待到该事件后, 调用 CAN_GetNextReceivedFrame (…) 即可进行 CAN 数据报文的读取。需要注意的是函数 CAN_GetNextReceivedFrame 每执行一次, 只是读取了 1 帧 CAN 数据报文, 如果在应用程序中需要将最新的数据全部读出, 只需反复调用该函数, 直到该函数的返回值为 FALSE^[5]。

CAN 数据报文的发送比较简单, 应用程序直接调用函数 CAN_SendFrame (…) 即可。

在粘度控制器的开发中, 对嵌入式开发平台的熟悉过程至关重要。其中包括开发板接口的使用, 各个模块的调试和实现。对硬件接口规划完成后大量的工作集中在程序的编写。包括扩展键盘的实现中类库的加载、数字量数据位数的调试、can 通信规则的调试、光标的实现过程中多线程方法的学习和使用, 且解决了调试过程中发现的由于电磁干扰导致 LCD 复位黑屏的问题以及 LCD 背光的实现。

经后期在已完成的 3 套粘度控制试验台上使用测试, 该粘度控制器系统稳定、使用简单方便, 满足了预期的控制和通信的功能, 但是由于采用的是成品的嵌入式平台, 开发板资源配置上还可以优化以节约资源成本。
参考文献

- [1] 姜淑琴. 基于单片机的船舶主机燃油黏度控制 [D]. 大连: 大连海事大学, 2010.
- [2] 王洪杰. Windows CE 下多线程多路播放的设计 [D]. 天津: 天津大学, 2006.
- [3] 胡在华. 实时嵌入式系统的研究与应用 [D]. 长沙: 湖南大学, 2002.
- [4] 蔡建平. 关于嵌入式应用开发技术 [J]. 单片机与嵌入式系统应用, 2001 (3): 5-11, 34.
- [5] 张琪君. 基于面向对象的嵌入式系统软件开发方法研究及其应用 [D]. 济南: 山东大学, 2006.

(收稿日期: 2013-03-05)

作者简介:

陈林, 男, 1987 年生, 助理工程师, 硕士, 主要研究方向: 船舶自动化。