

采用 AVR Flash 微控制器的电动窗户防夹系统

Thierry Corbière, 汽车 MCU 技术市场部经理, Atmel

汽车上可自动关闭的电动窗户或车门设备潜藏着卡死, 挤压以及可能伤人的危险。它们必须能够反向移动以防止马达所施加的力超出正常限制。这种特性意味着必须持续监视速度、电流和玻璃的位置。由于成本和简化的原因, 本文所描述的系统使用普通的带有霍尔效应传感器的刷式马达。基于速度和扭矩导数的检测算法已通过健壮性和容错性的验证。该算法可用于所有带有 A/D 转换器和通过变化引发中断的 I/O 口的 Atmel AVR Flash 微控制器

本文描述的是基本原理, Atmel 网站上的应用笔记有关于实现的详细描述。

现代汽车中的电动设备

目前, 在高端客用汽车中电子组件和系统在成本中已占 20% 以上。增加电子设备的数目可以更好的控制传感器和致动器, 从而增强汽车的舒适性和安全性。可以预测, 大部分的中等或高级汽车将会系统性的装备电动窗户或车门系统。这些设备中的绝大多数是全自动的, 这意味着它们必须附带安全系统以防止伤人或机械故障。已有法规设立了电动系统必须遵守的规则。这一点在车窗的升起和车门的滑动上尤其正确的。这篇应用笔记介绍了如何实现一个防夹算法, 该算法最初是用于电动车窗系统, 但可以轻易地移植到其它可移动设备中。

标准

汽车电动车窗受国际标准的约束, 如美国的 MVSS118 或欧洲的 74/60/EEC。在如何降低对儿童的危险度方面, 这些文档所提出的要求如下:

检测区域: 4 mm 至 200 mm

最大夹物力为 10N

夹住时可以反向

确定偏转角测试: 5N/mm 至 20N/mm

<http://www.BDTIC.com/Tech>

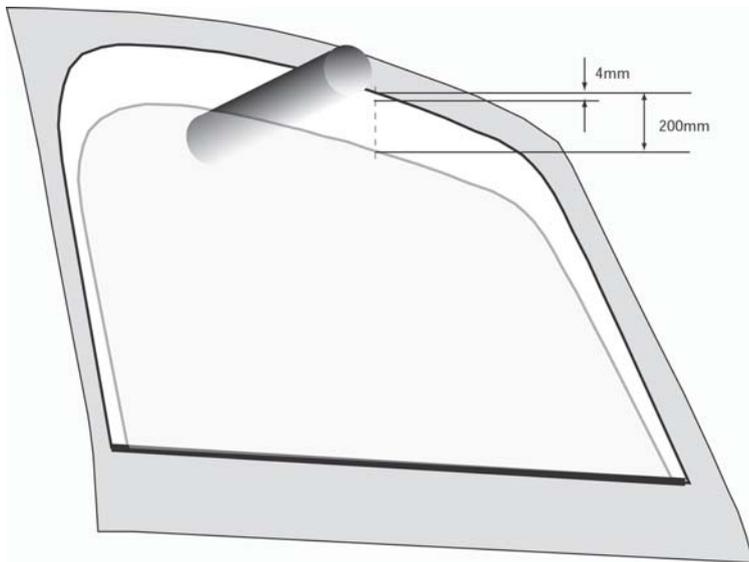


图 1. 遵守 MVSS118 的防夹车窗升起规则

关于硬件

对于确定关键夹物区是否有障碍物进入的不同检测策略有:

无机械接触。在夹物力施加至物体上之前就有反应。因为没有外力施加在物体上, 这是最优的保护方式。它还不依赖于振动、空气动力学变化或变形。但该方法要求有集成的传感器 (红外、超声波, 等等) 以及相关的电路模块和线路, 从而导致附加成本。

有机械接触。所测到的压力传递给系统用于指示有物体被夹住。在这方面，设计者还有两种基本的技术可用：

- 方向测量：力学传感器或接触器集成进车门密封中。这些解决方案成本一直都很高并限制了车窗/车门的样式设计。
- 通过物理监测的无方向测量。这是一种整体成本上最优的解决方案。

防夹算法详述

夹物检测算法一开始就要符合标准 (FMVSS118 & 74/60/EEC) 的要求：

- 检测区域为 4 至 200 毫米
- 最在施加力 100 牛
- 夹住物体时反转方向
- 标准的确认性测试。

必须要自适合的原因：

- 提升系统中的机械部分将会随着时间而变化（老化、局部变形、磨损，等等）
- 电子特性会有很大的变化
- 环境对磨擦力的影响（温度、湿度、结冰等）

系统不应扰动和不正确的夹物检测有反应。对于空气的磨擦、道路的振动、断电等都必须是健壮的。

使用马达的解决方案的物理参数

必须可以通过马达的电流算出施加在玻璃上的力。在速度方面可以持续提供移动部件的位置信息。这些参数都可以用于确定是否遇到物体以及：

- 该物体是否在检测区
- 所施加的力是否超限

本文描述了一种防夹算法，该算法通过测量电机电流和霍尔效应速度指示器来工作。只须很少的改动，就可以将该算法用于象滑动车门或蓬顶这样的系统中。

夹物检测算法

一般情况下，夹物检测算法的运行是通过间接测量车窗提升系统的：

- 电流（扭矩）
- 位置（速度）

与算法相关的应用笔记采用了两种技术，它们是基于：

- 存贮在无冲突内存中的校准扭矩：执行初步学习顺序，将扭矩值存贮在内存中。这种技术很耗费内存，并要求规定校准顺序。
- 速度推导计算：很有意义的一种技术，因为它对内存要求较少，但需要更多的计算，具有两种方法的优点。

在 AVR 上的实现

前几段详细描述过的算法已经实现，并在了一块 AVR ATmega88 开发板上进行了测试。

图 2 描述的是用以实现算法的硬件。它采用了一块标准的 ATmega88 以及模拟链来测量电机电流。硬件带有两个霍尔效应传感器。马达的方向是通过一个两极延迟来控制的，并通过一个场效应管来激活马达的开关。

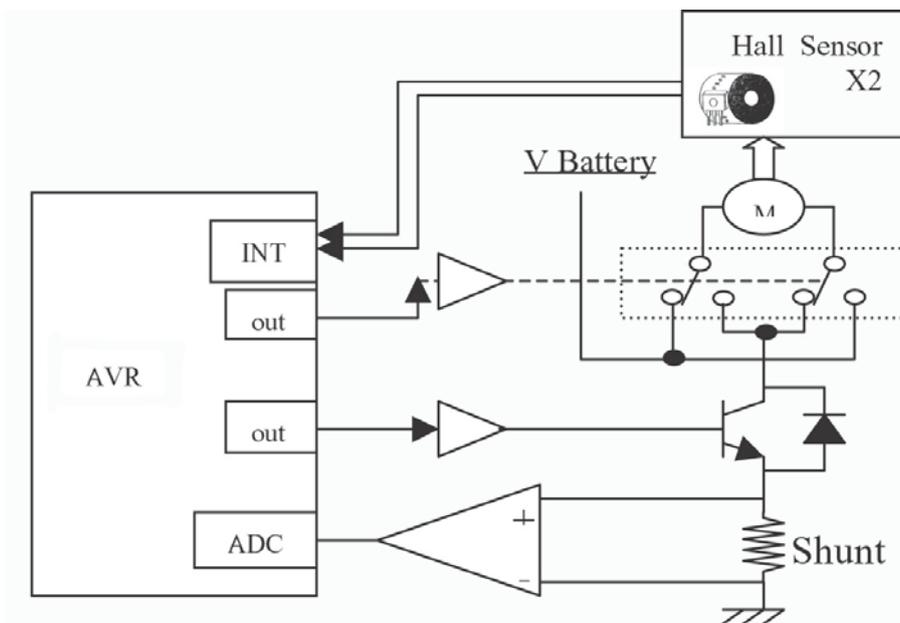


图 2. 通过 ATmega88 控制的带有防夹特征的车窗提升系统

系统内编程 Flash

系统内编程允许任何位于端系统内的 AVR 微控制器进行编程和再编程。通过一个简单的三线 SPI 接口，系统内编程器与 AVR 微控制器进行串行通讯，从而重编程芯片上的所有非易失性内存。

系统内编程无须将芯片从系统中物理上拿出。这样无论是在实验室的开发阶段，还是现场的软件或参数的升级，均可以节约时间和资金。在最终产品阶段将代码上传进 Flash 存储器中时，在多个应用和自定义版本中使用同一个标准的 AVR Flash 微控制器可以简化总量管理。

软件描述

所有代码使用 IAR EWAVR 4.1 以 C 语言方式实现。基本功能（位置管理、初始化、电流管理、车窗操作、防夹监测）的实现需要 2KB Flash。增加扩展功能象校准、阻塞点的检测和存储，可将代码大小扩展至 4KB。

软件代码可在 Atmel 的网站中获得，其结构如下：

初始化脚通过一个霍尔效应传感器（对信号沿的升起和下降敏感）改变所要用的中断。它还初始化用于测量速度和马达电流的时钟和 ACD。

这个函数从 EEPROM 或以初始值方式加载车窗提升参数，以初始化车窗提升器。这些参数包括车窗的大小、传感器值、夹物阈值、夹物区域、已知的最后的位置，等等。如果在位置参数上是一个缺省值，它能够请求一个下降的命令在底端限制值上初始化车窗提升器。

这个函数是将车窗提升参数存进 EEPROM。

这个函数包含了车窗提升状态机。它通过已有的事件参数控制车窗的操作。监视车窗的位置、上升和下降端的限制以及防夹系统的状态。返回车窗提升器的状态（与 get_window_state 函数的值相同）。

这个中断子例和在霍尔传感器沿执行。它计算滚动方向、位置、推导速度和马达电流参考值。通过计算连续的方向变化，它还能够检测出霍尔效应传感器的缺省值（传感器没有连接至一个中断脚上）。

这个函数强制车窗在一个定义好的步骤后停下来。

该函数返回车窗提升状态机的状态（这个返回值用于 window_ctrl 函数中）

建立车窗提升状态：用于强制性操作中（如停止请求…）

计算最后 8 个采样点的均值。用于过滤马达电流。

这个函数监视启动按钮，产生操作命令事件并传递给 window_ctrl 函数。

关于 AVR 微控制器在汽车方面应用的更多的信息可在以下址中找到：

www.atmel.com/products/avr/auto