

基于 STM32 的步进电机 S 形加减速控制曲线的快速实现方法

蔡泽凡*

CAI Ze-fan

摘 要

本文介绍了一种步进电机 S 形加减速控制曲线的快速实现方法, 该方法非常适合在微处理器系统中实现。为了验证该方法的有效性, 构建了一个以 32 位微处理器 STM32 为控制器的实验装置, 实验结果表明利用该方法建立的加减速曲线适合对不同负载、不同速度的加减速控制。利用该方法, 可以不用了解步进电机的工作原理, 就可以较满意的控制步进电机的运行。

关键词

步进电机控制 S 形加减速曲线 STM32

doi: 10.3969/j.issn.1672-9528.2014.04.04

由于步进电机具有无需反馈就能对位置和速度进行精确控制的优点, 在工业自动化设备中被广泛第应用极为广泛。步进电机将电脉冲信号转换成角位移, 然后驱动执行机构, 使其按照一定的规律运动, 步进电机的转速和角位移分别由电脉冲的频率和脉冲数决定。为了实现对位置的精确控制, 要求步进电机在运行中必须不丢步和过冲。为了防止丢步和过冲, 在启动时必须采用加速过程, 在停止时必须采用减速过程, 步进电机的运行过程一般如图 1 所示。

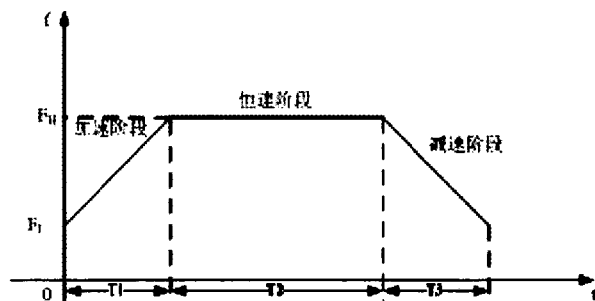


图 1 步进电机运动曲线

图 1 中, 横坐标表示时间, 纵坐标表示驱动脉冲的频率。从图 1 可以看出, 一般来说, 步进电机的运

动曲线一般分为三段: T1 段为加速阶段, T2 段为恒速阶段, T3 段为减速阶段。如果在 T1 及 T3 阶段, 驱动脉冲的频率变化率大于步进电机的响应频率变化率, 步进电机就会失步。总之, 驱动脉冲的频率变化率过高, 会使系统无法精确定位, 严重时甚至产生堵转。一般来说, 加减速曲线最好选择 S 形。

介绍步进电机的数学模型和速度控制的文献已经很多, 有些文献也介绍了如何在微处理器实现步进电机的 S 形加减速曲线, 这对理解步进电机的工作原理和控制方法是很有帮助的, 但都不能够让读者读完以后立即就编写出微处理器的 S 形加减速曲线。本文试图使读者看完本文就能够快速编写出自己的 S 形加减速曲线的代码, 读者可以不用了解步进电机的工作原理, 就可以较满意的控制步进电机的运行。

1 logistic 函数

sigmoid 函数的图形是“S”形的, 它是一个严格的递增函数, 在线性和非线性之间显示出较好的平衡。^[5] 可以利用这个 sigmoid 函数来构建步进电机的加减速曲线。

sigmoid 函数的其中一个例子是 logistic 函数, 简称为 φ 函数, 其定义如式①所示,

$$\varphi(v) = \frac{1}{1 + \exp(-av)} \quad (1)$$

其中, a 是函数的倾斜参数, 控制函数的斜率。

* 顺德职业技术学院 电子与信息工程学院 广东 佛山 528333; 华南理工大学 自动化科学与工程学院 广东 广州 510640

φ 函数的图形如图 2 所示, 当 a 为 0.5、1.0 时, 曲线比较适合作为加速曲线。

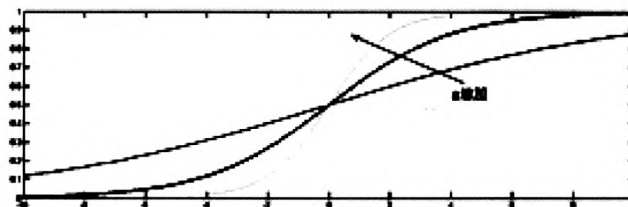


图 2 具有不同倾斜参数 a 的 φ 函数

logistic 函数的对偶函数, 简称为 φ' 函数, 其定义如式②所示,

$$\varphi'(v) = \frac{1}{1 + \exp(av)} \quad (2)$$

其中, a 仍然是函数的倾斜参数, 控制函数的斜率。

φ' 函数的图形如图 3 所示, 当 a 为 0.5、1.0 时, 曲线比较适合作为减速曲线。

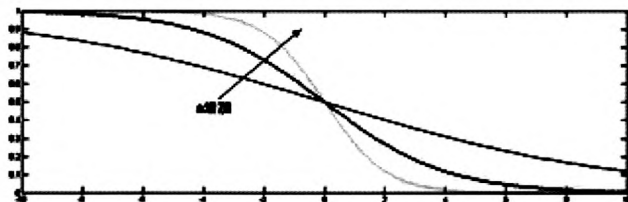


图 3 具有不同倾斜参数 a 的 φ' 函数

2 设计微处理器 S 形加减速曲线的思路

由于步进电机的转速由驱动脉冲的频率决定, 为了突出重点, 本文只考虑驱动脉冲的频率控制, 至于方向信号、脱机信号以及驱动器的硬件实现不在本文的讨论范围之内。

利用微处理器控制步进电机时, S 形加速曲线的构建过程如下所示。

第一, 设置定时器, 把微处理器的定时器设置为计时溢出中断模式, 在中断处理函数中使驱动脉冲的信号翻转。

第二, 确定驱动脉冲最长周期 T_{MAX} (对应最低频率), 该最长周期 T_{MAX} 作为脉冲信号的起始周期。调整定时器的溢出中断的周期, 从而产生不同周期的脉冲信号, 利用该脉冲信号对步进电机进行控制。

第三, 确定驱动脉冲最短周期 T_{MIN} (对应最高频率), 该最短周期 T_{MIN} 作为脉冲信号的恒速周期。

第四, 确定定时器两端的自动重装值。对于 STM32 来说, 定时器的溢出中断周期由时钟预分频数和自动重装值决定, 当前者固定以后, 中断周期仅由

后者决定, 令 $T_{\Delta} = T_{MAX} - T_{MIN}$ 。

第五, 计算某个阶梯对应的自动重装值 T_i 。在式 (1) 中, v 的值从 -10 开始, 以 0.1 作为步进, 到 10 结束, 共有 201 个值, 也就是把加速曲线离散化为 201 点, 对这些点采用零阶保持器, 相当于用 200 个台阶对曲线进行逼近。当 $a=0.5$ 时, φ 的值为 0.0067 ~ 0.9933, 当 a 为其他值时, 的值也都是 0 ~ 1 之间的小数。脉冲数 i (对应台阶 i) 对应的自动重装值 T_i 的值由式③计算。

$$T_i = T_{MAX} - T_{\Delta} \frac{1}{1 + \exp(-a * i)} \quad (3)$$

第六, 对于微处理器来说, STM32 也不例外, 并不适合指数运行, 因此我们可以利用 excel 工具先算出各个 T_i 值。

第七, 把自动重装值 T_i ($i=0, 1, \dots, 200$) 制作成为表格, 并保存在 ROM 中。

第八, 步进电机启动时, 根据负载和脉冲数查表, 得到某个脉冲数对应的自动重装值 T_i 。

第九, 恒速运行。当加速阶段结束后, 最后一个脉冲周期就是恒速运行时的脉冲周期。

减速阶段和加速阶段类似, 可以单独作一个新表, 也可以和加速阶段共用一个表。当步进电机转过的角度很小时, 运动曲线只有两段, 恒速阶段将不存在, 这时只有加速阶段和减速阶段。步进电机典型的三段式运行曲线和二段式运行曲线分别如图 4、图 5 所示。

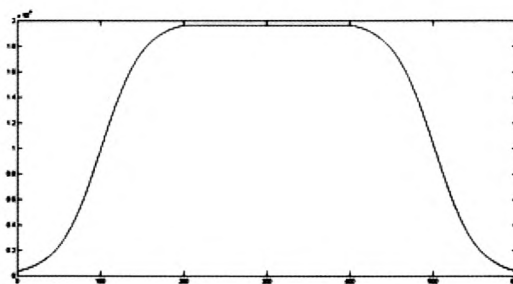


图 4 步进电机三段式运行曲线

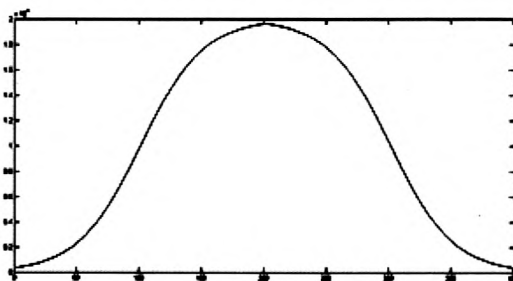


图 5 步进电机二段式运行曲线

3 基于 STM32 的步进电机控制曲线的软件实现

以控制一个步进电机为例,介绍基于 STM32 的实现代码。

3.1 定义步进电机驱动脉冲周期表

以 10 个速度等级、200 个台阶为例,步进电机驱动脉冲周期表如下所示:

```
const u16 TIM_PERIOD_TBL[10][201];
TIM_PERIOD_TBL 的值如下所示:
const u16 TIM_PERIOD_TBL[10][201] =
{
    {19910, 19905, ..., 6590},
    {19410, 19405, ..., 6090},
    ...,
    {15410, 15405, ..., 2090}
};
```

由于篇幅的原因, TIM_PERIOD_TBL 省略了很多值。

3.2 定义步进电机控制结构变量

首先定义步进电机控制结构变量类型 StepMotorTag_Typedef, 其内容如下所示,

```
typedef struct
{
    u16 PulseNumStart;
    u32 PulseNumMiddle;
    u16 PulseNumEnd;
    u8 PulsePeriodClass;
    u8 PulsePeriodTblIndex;
}StepMotorTag_Typedef;
```

再定义一个具有 3 个元素的数组, 如下所示,

```
extern StepMotorTag_Typedef StepMotorTag[3];
```

3.3 编写定时器初始化函数

在本文中,假设用来控制第一个步进电机的驱动脉冲的为定时器 3,其初始化函数比较简单,在此不做赘述。

3.4 编写定时器溢出中断处理函数

定时器溢出中断处理函数如下所示:

```
void TIM3_IRQHandler(void) //TIM3 中断
{
    if (TIM_GetITStatus(TIM3, TIM_IT_Update) != RESET)
    {
        StepMotor1stPWMGenerate();
        TIM_ClearITPendingBit(TIM3, TIM_IT_Update );
    }
}
```

3.5 步进电机设置函数流程图 (见图 6)

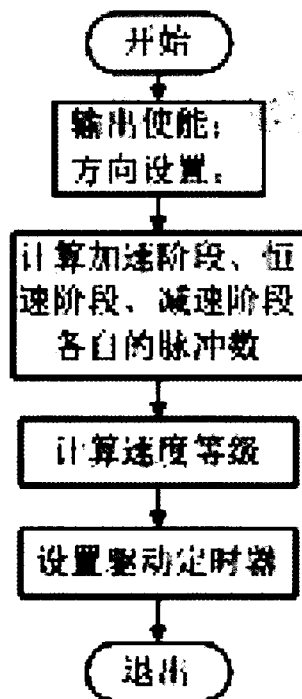


图6 步进电机设置函数流程图

设置的内容为步进电机的方向、使能信号和步进电机控制结构变量。值得一提的是,根据脉冲数,要把步进电机的运行曲线设置成为三段式或者二段式。

3.6 步进电机驱动脉冲管理函数流程图

步进电机驱动脉冲管理函数流程图 7 所示,该函数按照加速、恒速、减速的过程依次进行,恒速时定时器溢出中断的周期保持不变,加速和减速过程的中断周期根据台阶数查表以后动态更新。

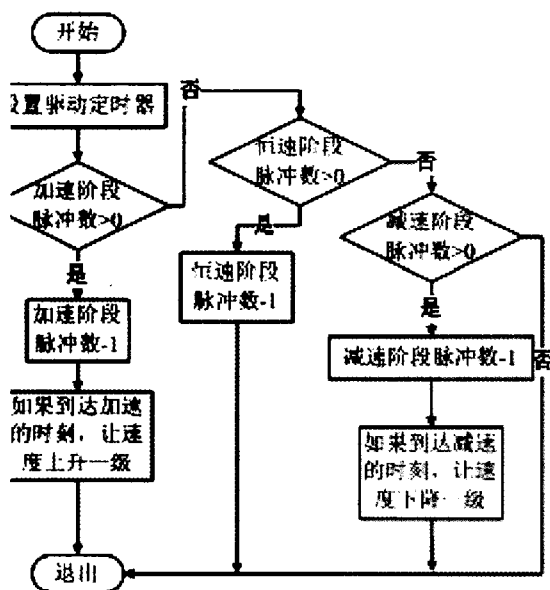


图7 步进电机驱动脉冲管理函数流程图

(下转第 33 页)

性,将该学生学号发送给远程服务器,服务器修改好数据库数据之后便返回,客户端更改相应的显示内容。咨询师可以查看学生心理详情,点击图标之后便跳转到对应 activity。咨询师还可以进行危机干预,选择短信发送及调用系统短信,选择语音推送即跳转到对应 activity。咨询师危机干预伪代码如下:

```
void crisisInter(){
    get all key students data from remote server;
    show as a list;
    put each item of a list three image buttons;
    var cancelBtn,stuDetailBtn,crisisInterBtn;
    cancelBtn.setOnClickListener({
        void click(){
            send the data to remote server;
            show the response from remote server;
            change the icon as tick;
        });
    stuDetailBtn.setOnClickListener({
        void click(){
            dispatch the data to another activity;
        }
    });
    crisisInterBtn.setOnClickListener({
        void click(){
            show the dialogue ;
            var choice1,choice2;
            if(choice1){
                dispatch to another activity;
            }else if(choice2){
                dispatch to another activity;
            }
        }
    });
}
```

(上接第 29 页)

4 结论

实验结果表明,利用本文所介绍的方法,可以对多路步进电机进行平稳的控制,且能够根据负载设置最低速度和最高速度,步进电机也可以进行联动控制。本文虽然以 STM32 作为基础进行介绍,但是本方法完全适合其他微处理器的应用。读者不必懂得步进电机的工作原理,就可以对步进电机实现较优的控制。

参考文献:

- [1] 李汉. 一种步进电机最佳升降速的控制方法[J]. 电气自动化, 2012, 34(5): 32-34.

4 总结

本文实现了学生咨询师使用触屏智能手机通过 Android 来进行心理咨询,使得学生能够更好得进行心理开导减轻压力,也便于形成高校学生心理资料信息化管理体系。

参考文献:

- [1] 曾永乐. 谈大学生心理教育的意义及发展趋势[J]. 赤子, 2012(5): 107.
- [2] 郑洁琼, 陈泽宇, 王敏娟, 等. 3G 网络下移动学习的探索与实践[J]. 开放教育研究, 2012, 18(1): 159-162.
- [3] Journal of Materials Chemistry Group. ANDROID MOBILE APP[J]. Self-Adhesive Materials and Markets Bulletin, 2012, 38(10): 6-7.
- [4] Marcin PAWLAK. Application of mobile devices with the Android system for the induction motors faults diagnosis[J]. Przegląd Elektrotechniczny, 2013, 89(2b): 150-153.

[作者简介] 汪璐萍(1993-), 女, 浙江萧山人, 学生, 主要从事移动应用的研究。

[通讯作者简介] 方硕瑾(1986-), 女, 浙江松阳人, 硕士研究生, 助理研究员。研究方向: 学生工作信息化。(收稿日期: 2014-05-15)

- [2] 高亮, 王华. 8051 单片机对步进电机的控制及步进电机升降速曲线的设计[J]. 测控技术, 2002, 21(11): 64-65, 67.
- [3] 梁广瑞, 莫胜憾. Stm32 步进电机控制器及其在焊缝跟踪系统中的应用[J]. 电焊机, 2013, 43(3): 74-77.
- [4] 王勇, 王伟, 杨文涛. 步进电机升降速曲线控制系统设计及其应用[J]. 控制工程, 2008, 15(5): 576-579.

[作者简介] 蔡泽凡(1979-), 男, 广东陆丰人, 博士。研究方向: 单片机的教学与应用及无线控制系统。(收稿日期: 2014-05-23)