

基于 FPGA 的速度信号测量

李 辉, 罗正宇, 徐 娟

(株洲中车时代电气股份有限公司 通信信号事业部, 湖南 株洲 412001)

摘 要: 为了减少 CPU 在进行速度信号测量时的资源占用率以及降低软件复杂度, 设计某型车载设备时使用了 FPGA (现场可编程门阵列, Field Programmable Gate Array) 辅助 CPU 进行速度及相关信息的测量。研究发现该方案既具有很好的精度, 又极大地减轻了此过程的 CPU 的占用率。

关键词: FPGA; 测速; 测距; 相位; 车载设备

中图分类号: U260.5⁺1; TP212.1

文献标识码: A

doi: 10.13890/j.issn.1000-128x.2016.06.019

Velocity Signal Measurement Based on FPGA

LI Hui, LUO Zhengyu, XU Juan

(Signal & Communication Business Unit, Zhuzhou CRRC Times Electric Co., Ltd., Zhuzhou, Hunan 412001, China)

Abstract: In order to reduce CPU resource occupancy rate and the software complexity for velocity measuring, a FPGA(Field Programmable Gate Array) technology to measure velocity and related information was adopted as a supplement of CPU. It was pointed out that this method could increase the measuring resolution and reduce the CPU occupancy rate.

Keywords: FPGA; speed measurement; distance measurement; phase; onboard device

0 引言

轨道交通行业传统的测速测距的方法主要是通过处理光电 / 霍尔 / 测速电机等传感器信号来实现, 来自传感器的信号经过隔离、整形电路后转换为方波信号并被 CPU 所检测, 进而由 CPU 计算出频率、相位等速度相关信息。对于没有专用的高速输入 / 捕获端口的 CPU 而言, 频率信号引发的中断将随着速度上升迅速增加, 这增加了 CPU 的资源占用率, 对其他功能的实现造成了影响。本文介绍某型产品研发过程中利用 FPGA 进行速度及相关信号的测量的方案, 该方案有效地减轻了 CPU 负载, 具有适应性好、便于扩展等特点。

1 测量原理

1.1 速度测量

以轨道交通行业常见的光电速度传感器为例, 频

率和速度的关系为

$$v = 3.6f \frac{2\pi r}{P_n} \quad (1)$$

式中: v 为速度, km/h; f 为频率, Hz; r 为车轮半径, m; π 为圆周率; P_n 为车轮旋转一周传感器输出的脉冲数量。

由式 (1) 可见, 测量速度的关键是测量速度信号的频率 f , 常用的测量频率的方法一般有直接测频法和间接测频法^[2]2 种。

1) 直接测频法

直接测频法就是在确定的周期时间内, 记录被测信号的脉冲个数, 进而计算出频率的方法。如图 1 所示, 设采样周期为 T , C_n 为周期 T 内的累计脉冲数, 则有

$$f = \frac{C_n}{T} \quad (2)$$

在具体实现中, 通常会利用 CPU 定时中断作为“闸门”来开始脉冲累积计数和停止累积计数, 由于采样的

周期起始点和结束点不一定与被测信号同步,测量时会产生 ± 1 个输入信号脉冲的误差。测试结果精度会随着被测信号的频率增加而增加,随频率减少而减小。较大的 T 虽有助于提高精度,但会降低实时性,因此这种方法不适合在低速和极低速(例如速度小于 1 km/h)情况下的测速。

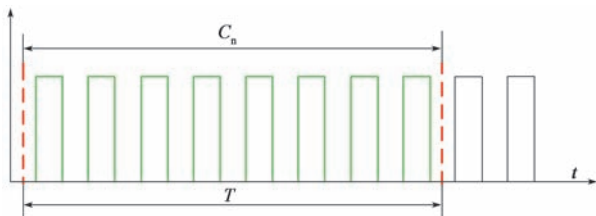


图1 直接测频法周期测量时序图

2) 间接测频法

间接测频法是通过测量被测信号周期内所占用的时间间接计算出频率,常见的有等精度测频法。

等精度测频方法是在直接测频法的基础上发展而来的,主要特点是 T 不是一个固定的值,而是被测信号的整数倍,即与被测信号同步。如图2所示,开始和结束计算 T 均在被测信号的上升沿,因此,理想情况下,精度与被测频率信号无关。这种方法测量的精度很高,一般要求 CPU 具有高速输入/捕获端口以减少中断次数。

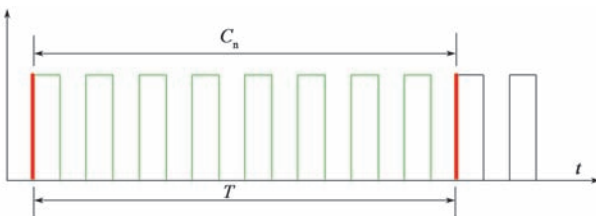


图2 等精度测频法测量时序图

1.2 速度信号相位测量

轨道交通行业中常用的速度传感器2个关联输出通道的信号频率相同,相位的超前或滞后的角度是一个固定值,代表着车轮的旋转方向。

测量相位通常使用过零法、波形变换法、相关分析法以及快速相位检测方法。以较简单的波形变换法为例,该方法将信号转换成方波并进行异或处理,最后利用 CPU 测量方波的宽度,进而得出相位。如图3所示,信号1和信号2被处理成方波信号后,利用异

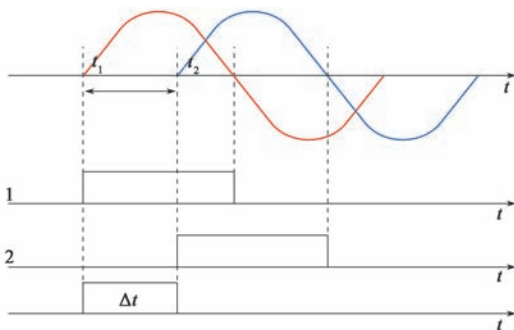


图3 过零法相位测量原理图

或电路求出 t_1 与 t_2 的时间间隔 Δt 。

获得了 Δt 与信号周期 T 后,相位可用以下公式^[3]表示:

$$\phi = 360 \times \frac{\Delta t}{T} \quad (3)$$

1.3 走行距离测量

如果速度传感器旋转一周输出 P_n 个脉冲信号,则每个速度信号脉冲都相当于车辆走行的距离为车轮周长除以 P_n 。因此,在没有轮对空转和滑行的情况下,脉冲的数量即代表车辆走行的距离:

$$S = N \frac{2\pi r}{P_n} \quad (4)$$

式中: S 表示距离, m ; N 表示脉冲个数。

2 设计实现

2.1 设计需求

在某型车载设备开发中,软硬件信息如下:①主处理器为 PowerPC,采用 Linux 操作系统,板载 FPGA 主频为 16 MHz ;②采用具有相位关系的双通道光电速度传感器,传感器每旋转一周固定发出占空比 $1:1$ 的方波脉冲 200 个,向前运行时相位为 90° ,向后为 270° ;③标准车辆轮径 650 mm 。

2.2 FPGA 测量速度设计

根据式(1)及设计需求,可计算得出在运行速度 200 km/h 时,车载设备获得的每路速度信号频率约为 5.441 kHz 。考虑到测量的基准时钟频率(16 MHz)远大于被测信号频率以及需保证在极低速(速度 1 km/h 或者不足 1 km/h)时测速精度,设计采用了等精度测频法进行速度信号的频率测量。

设计中利用 FPGA 的时钟源作为测量基准,通过统计周期 T 内的基准脉冲数量进而得出 T 的时长(见图4)。以 FPGA 的 16 MHz 的时钟源计算,每个基准脉冲时长为 $0.0625 \mu\text{s}$,设速度信号周期 T 内共捕获到 N 个脉冲,则有 $T = N \times 0.0625 \times 10^{-6} \text{ s}$ 。

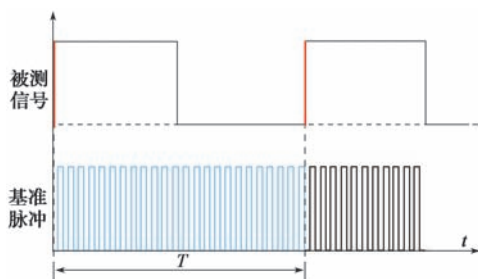


图4 频率测量时序图

为了解决何时开始和结束捕获基准脉冲的问题,设计由 FPGA 在速度信号的上升沿输出计数器值到寄存器供 CPU 读取并重新开始计数,即每个脉冲信号上升沿触发一次 FPGA 的输出和计数器的重置。这样设计另一个好处就是对速度信号占空比误差具有良好的容忍性。

由于测量周期 T 时最多误差 1 个基准脉冲, 即测量的 T 最多有 $0.0625\text{ }\mu\text{s}$ 误差。假设车辆速度为 200 km/h , 则最大负误差情况下计算速度为 199.932 km/h , 最大正误差情况下计算速度为 200.068 km/h , 足以满足 IEEE1474 对车载 ATP 分辨率 $\pm 0.5\text{ km/h}$ 到 $\pm 2\text{ km/h}$ 、准确度 $\pm 3\text{ km/h}$ 的要求。

2.3 FPGA 测量相位设计

根据式 (3), 测量相位需要已知 T 和 Δt , 在测量频率中已得到被测信号的周期 T , Δt 的检测则由 FPGA 在通道 1 的信号上升沿开始基准脉冲计数 (见图 4), 在通道 2 的信号上升沿停止计数并将数据输出到寄存器, 由 CPU 计算得出相位。

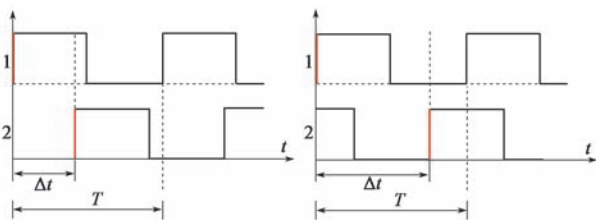


图 5 相位超前和滞后典型波形

因本设计中使用的速度传感器相位差值向前为 90° , 向后为 270° , 设运行方向为向前, 输入 $0.1\sim 6\text{ kHz}$ 速度信号, 并根据测量 T 和 Δt 时最多误差 $0.0625\text{ }\mu\text{s}$ 可得到表 1 数据。由表 1 可见, 相位检测误差随着频率增加而提高, 但在超过极限的 6 kHz 时最大误差仍小于 0.2° , 足以满足本设计中检测车辆运行方向的要求。

表 1 相位理论计算表

频率 / kHz	采集的 T 变化范围 / s	采集的 ΔT 变化范围 / s	计算相位范围 / ($^\circ$)
0.1	$0.999\ 994\times 10^{-2}\sim 1.000\ 01\times 10^{-2}$	$2.499\ 94\times 10^{-3}\sim 2.500\ 06\times 10^{-3}$	$89.997\sim 90.003$
1	$0.999\ 38\times 10^{-3}\sim 1.000\ 06\times 10^{-3}$	$2.499\ 38\times 10^{-4}\sim 2.500\ 63\times 10^{-4}$	$89.972\sim 90.028$
2	$4.999\ 38\times 10^{-4}\sim 5.000\ 63\times 10^{-4}$	$1.249\ 38\times 10^{-4}\sim 1.250\ 63\times 10^{-4}$	$89.944\sim 90.056$
4	$2.499\ 38\times 10^{-4}\sim 2.500\ 63\times 10^{-4}$	$6.243\ 75\times 10^{-5}\sim 6.256\ 25\times 10^{-5}$	$89.888\sim 90.113$
6	$1.666\ 04\times 10^{-4}\sim 1.667\ 29\times 10^{-4}$	$4.160\ 42\times 10^{-5}\sim 4.172\ 92\times 10^{-5}$	$89.831\sim 90.169$

2.4 FPGA 测量走行距离设计

正常情况下, 测量走行距离的准确性取决于是否丢失了速度脉冲, 每丢失一个脉冲即丢失了车轮周长

除以传感器每转脉冲数的距离。至于车轮发生空转和轮滑的情况, 并不在 FPGA 中进行处理, 相关的补偿处理由应用软件进行。因此, FPGA 测量走行距离主要的问题是解决计数溢出和确定什么时候 FPGA 输出数据给 CPU。

由于 CPU 可能在任意时刻访问 FPGA, 为了避免发生丢失数据以及 2 次访问获得同一周期数据等问题, 设计中 FPGA 一边在后台不断累积速度脉冲, 一边利用 CPU 的读信号作为“闸门”, 当 CPU 读 FPGA 时, FPGA 输出自上次 CPU 访问到本次访问之间的速度脉冲个数数值到输出寄存器并重新开始计数, 这样就很好地解决了 CPU 可能在任意时刻访问 FPGA 的问题。

为了防止溢出问题, 设计 FPGA 计数器宽度为 32 位。假设速度信号频率为 6 kHz (此时速度已超过 200 km/h), 2 次访问 FPGA 的间隔时间需要超过 $4\ 294\ 967\ 295/6\ 000$, 即 $715\ 827\text{ s}$ (约 8 天) 才能满足溢出的条件, 显然只存在理论的溢出可能性。

3 结语

本文所列方法已在株洲中车时代电气股份有限公司某型车载设备中应用。该方法具有以下显著特点: 一是对系统处理器的性能要求较低, 与是否采用操作系统或采用哪种操作系统无关, 具有良好的适应性; 二是便于扩展输入信号通道, 且增加输入通道时不影响性能, 具有良好的可扩展性; 三是测量的精度高, 且测量的精度主要受制于 FPGA 时钟源频率和频率稳定度, 与被测频率信号无关, 通过提高 FPGA 的时钟频率和稳定度还可以进一步提高测量精度。

参考文献:

[1] 刘德亮, 王竹林. 基于 FPGA 高精度频率测量仪的设计 [J]. 河北工业科技, 2010, 27 (1): 29-31.
[2] 李红刚. 基于 FPGA 高速等精度频率测量系统设计 [J]. 微计算机信息, 2008, 24 (2): 218-228.
[3] 沈明山. EDA 技术及可编程逻辑器件应用实训 [M]. 北京: 科学出版社, 2004.



作者简介: 李 辉 (1975-), 男, 工程师, 现从事行车安全设备技术研究和产品开发。

欢迎到当地邮局订阅《机车电传动》期刊 (邮发代号: 42-17)