



名次奖第二名

基于 dsPIC30F4012 的发动机虚拟转矩传感器设计

注册编号： MCHP16bitCDC0174

参赛队员： 洪木南
孟庆然

1 引言

1.1 设计背景

随着排放法规的不断加严，驾驶员对汽车动力性、经济性和可靠性的要求逐步提高，汽车系统构型不断呈现新的特点，控制系统变得更加复杂，更难以控制。发动机的转矩对变速箱控制和新型汽车的控制都具有重要意义^{[1][2]}，但其难以测量，多年来不少学者对其估计方法展开了广泛研究。

发动机的在线转矩估计算法主要分为两种，一种基于发动机曲轴模型^[3]，属于发动机转矩前馈估计，实质是转矩观测器。该方法的估计精度受发动机曲轴模型精度的影响，建模时需要得到大量的参数。另一种是基于发动机曲轴瞬时转速测量^[4]，属于发动机转矩反馈估计，实质是标定出虚拟的发动机转矩传感器。该方法对瞬时转速波动进行傅里叶变换，可取得较高的精度，主要缺点是一般的发动机控制器难以对瞬时转速信号进行实时处理，因而其应用受到限制。然而，随着电子技术的进一步发展，各种新型的微控制器芯片开始具备复杂的信号处理功能，使得基于瞬时转速测量的转矩估计成为可能。

1.2 设计任务

本次设计的主要任务是利用Microchip公司的16位微控制器dsPIC30F4012，采用文献[4]描述的瞬时转速测量转矩估计方法，对发动机转矩进行实时在线估计。具体来说，利用微控制器及其外围芯片，对发动机转速脉冲进行滤波整形，然后采集并送入微控制器，计算出实时转速，由微控制器对其作傅利叶变换，得到对应平均转矩的特征指标，然后利用特征指标和平均转矩的既定关系估计出平均转矩。流程如图1所示。

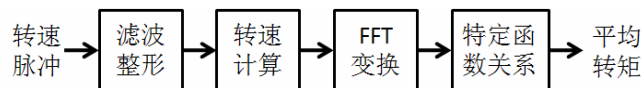


图1 任务流程

1.3 虚拟转矩传感器的载体

Microchip公司的16位数字信号控制器系列将MCU（Micro Control Unit）和DSP（Digital Signal Processing）的功能融合到一个单芯片解决方案中，从而可以为嵌入式应用添加DSP功能。Microchip提供了2个兼容的16位闪存DSC

(Digital Signal Controller) 系列。30MIPS dsPIC30F系列是专为需要工作电压为5V的应用设计的, 40 MIPS dsPIC33F系列适用于需要工作电压为3.3V或需要更多片上存储器的应用。这两个DSC系列均可工作在-40°C~85°C, 满足汽车的一般温度范围。

此外, Microchip的16位控制器具有丰富的开发工具、应用程序库、开发板和参考设计支持, 从而可促进快速便捷地设计高性能嵌入式解决方案。Microchip的所有MCU和DSC开发工具都可以在免费MPLAB集成开发环境(IDE)下协同工作。MPLAB IDE不仅集成了软件, 还集成了Microchip的所有硬件工具以及许多第三方工具。最令人惊喜的是Microchip为其产品提供了三种类型的库, 即外设驱动程序库、数学函数库和DSP算法库。

外设驱动程序库中有超过270种C实用函数, 可在各种模式下设置和操作硬件外设模块。外设驱动程序库所涉及的外设有:

- 10位/12位ADC
- UART、SPI、I²CTM和编解码器接口
- 电机控制PWM和QEI
- 通用定时器
- 输入捕捉和输出比较
- SMPS PWM
- SMPS 10位ADC

数学函数库则可提供单精度和双精度的浮点型ANSI C标准数学函数。这些函数经过优化, 可提供最小的代码大小。此库可用在汇编语言或C语言编写的程序中。数学函数库包含的主要函数如下:

- sin、cos和tan
- asin、acos和atan
- ln、log10、sqrt和power
- ceil、floor、mod和frexp

为了提高执行速度, 丰富的DSP构建模块库已经优化了汇编代码。DSP函数库中的函数可用在汇编语言或C语言编写的程序中。DSP算法库中包含的主要算法如下:

- 级联IIR滤波器

- FIR滤波器和LMS滤波器
- 相关和卷积
- FFT和窗函数
- 矩阵和矢量运算

基于 Microchip dsPIC 系列的上述强大功能，同时考虑到虚拟转矩传感器需要对转速进行采集，需要将计算结果送给其它控制单元，并为了保留一定的二次开发裕量，本次设计选用 28 引脚的 dsPIC30F4012 作为核心控制单元。

2 原理图设计

2.1 设计目标

相应于设计任务，原理图设计需要考虑瞬时转速信号处理与采集，信号的傅利叶变换以及转矩计算，最后还要将计算得到的转矩传递给其它微控制单元。各个功能模块将在下面分别进行详细设计，本设计所采用的软件是 Protel DXP^{[5][6]}。

2.2 转速信号处理

由于一般的转速传感器为电磁传感器，其信号不是标准方波，输入到微控制单元前需要进行滤波和整形，如图 2 所示。

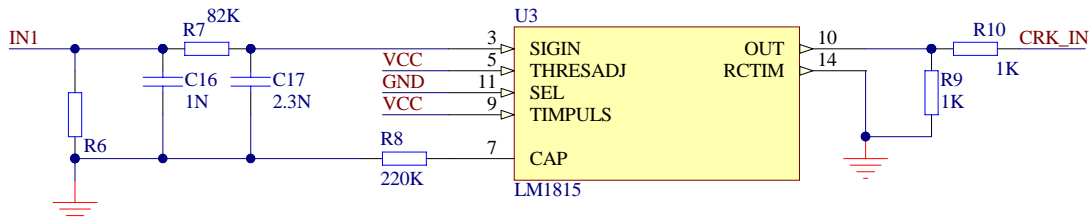


图2 转速信号处理

转速处理芯片采用发动机控制器上广泛使用的 LM1815，输入的转速信号经过阻容滤波网络进入 LM1815，进行整形。处理后的方波输入到 dsPIC30F4012 的输入捕捉端口。dsPIC30F4012 具有 4 个输入捕捉通道，分别为 IC1 和 IC2，以及 IC7 和 IC8，这里采用 IC7。

2.3 转速采集

dsPIC30F4012 在每个下降沿对输入信号进行捕捉，得到输入方波的周期。

以飞轮上的齿数是 60-2 (58 个齿, 除缺齿处外各齿均匀分布, 平均间隔 6 度) 为例, 在正常齿处, 测得其齿周期为 T_1 , 则发动机转速 n 可由式 (1) 计算。

$$n = 60 \times \frac{1}{60T_1} = \frac{1}{T_1} \quad (1)$$

在缺齿处测量得到的齿周期 T_2 约为正常齿的 3 倍, 对采集到的每个齿周期进行判断, 如果当前齿周期大于上一齿周期的两倍, 则可判断缺齿来到。缺齿可作为转速测量的相位参考点。本设计中, 每一发动机循环 (对应 2 转) 做一次转矩计算, 即从缺齿处后开始测量, 第一次经过缺齿时满一转, 第二次经过缺齿后进入中断, 将这两转得到的瞬时转速系列进行傅利叶变换, 计算转矩。而输入捕捉则继续进行, 采集下一循环的瞬时转速, 如此循环。

缺齿处的瞬时转速计算可采用平均法, 假想缺齿处存在虚拟齿, 则缺齿处的齿周期除以 3 可得到正常齿的周期, 如图 3 所示。

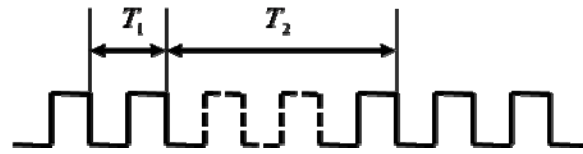


图3 正常齿和虚拟齿

2.4 转矩计算原理

设瞬时转速在发火频率点 (ZCz) 的波动分量为 $N(Z)$, 其中由于气体压力转矩引起的波动幅值为 $N_g(Z)$, 惯性力转矩引起的波动幅值为 $N_j(Z)$ 。对于四缸机或六缸机, 由于气体压力转矩和惯性力转矩反向^[4], 如图 4 所示, 定义 $N_g(Z)$ 的方向为正, $N_j(Z)$ 的方向为负, 于是有:

$$N(Z) = N_g(Z) + N_j(Z) = N_g(Z) - |N_j(Z)| \quad (2)$$

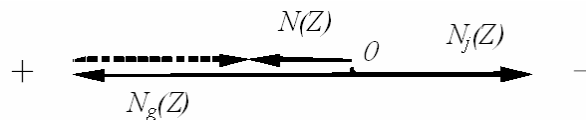


图4 发动机转速波动幅值的符号确定

设往复运动质量为 m_j ，发动机的平均角速度为 ω ，则往复惯性力转矩与平均转速的关系为：

$$t_j \propto m_j \omega^2 \quad (3)$$

由于惯性力转矩的直流分量为零（一个循环的功为零），在发火频率点，惯性力转矩的 Fourier 分量也有上述关系：

$$\tilde{T}_j(Z) \propto m_j \omega^2 \quad (4)$$

由于曲轴的扭振对瞬时转速的影响有限，可视其为刚性，曲轴系统可视为一个积分环节。设曲轴的旋转惯量为 I ，则由 $\tilde{T}_j(Z)$ 引起的瞬时转速波动的幅值为：

$$|N_j(Z)| = \frac{1}{2\pi} \cdot \frac{60}{\omega} \cdot \frac{\tilde{T}_j(Z)}{I} \quad (5)$$

于是惯性力转矩可以表示为：

$$T_j(Z) \propto |N_j(Z)| \omega \propto m_j \omega^2 \quad (6)$$

由上式可得到由惯性力转矩引起的瞬时转速波动幅值与平均转速的关系为：

$$|N_j(Z)| \propto \omega \quad (7)$$

与式（5）类似，由气体压力转矩引起的瞬时转速波动为：

$$N_g(Z) \propto \frac{\tilde{T}_g(Z)}{\omega} \quad (8)$$

设气体压力转矩的直流分量为 \bar{T}_g ，即平均指示转矩。G.Rizzoni^{[7][8]}采用气体压力转矩在发火频率点的幅值作为指标，在平均转矩不变的条件下，得到

$$\bar{T} = \bar{T}_g = k[\tilde{T}_g - \tilde{T}_0] \quad (9)$$

可见平均转矩正比于交流成分 $\tilde{T}_g(Z)$ ，于是有：

$$N_g(Z) \propto \frac{\bar{T}_g}{\omega} \quad (10)$$

结合式 (9)，发动机瞬时转速波动在发火频率点的 Fourier 分量与发动机平均转速及平均转矩之间的关系为：

$$N(Z) = N_g(Z) + N_j(Z) = -k_1\omega + k_2 \frac{\bar{T}_g}{\omega} + k_2 \frac{\tilde{T}_0(Z)}{\omega} \quad (11)$$

上式中的 k_1 和 k_2 均为常数。当平均转速不变时，可用平均有效转矩代替平均指示转矩：

$$N(Z) = -k_1\omega + k_2 \frac{\bar{T}_l + T_m}{\omega} + k_2 \frac{\tilde{T}_0(Z)}{\omega} \quad (12)$$

此处的 T_m 代表平均指示转矩和平均有效转矩的差别。于是不同负荷下，瞬时转速在发火频率点的波动幅值与发动机平均转矩成正比。式 (12) 可改写为：

$$\bar{T}_l = k'_1 N(Z)\omega + k'_2 [\omega^2 - \tilde{T}_0(Z) - T_m] \quad (13)$$

其中 k'_1 和 k'_2 为常数。发动机怠速时 \bar{T}_l 为零，设对应的转速波动幅值为 $N_0(Z)$ ，则有：

$$N_0(Z) = k'_2 [T_0(Z) + T_m - \omega^2] / (k'_1\omega) \quad (14)$$

于是式 (13) 可进一步表示为：

$$\bar{T}_l = k'_1\omega [N(Z) - N_0(Z)] \quad (15)$$

上式给出了估计平均转矩新指标 $N(Z)$ 与发动机平均转矩 \bar{T}_l 和平均转速 ω 之间的关系。

式 (15) 表明，在发动机的平均转速不变时，瞬时转速波动指标 $N(Z)$ 与发动机的平均转矩之间的关系为线性关系。

如果在不同的平均转速下，通过试验标定出各个转速下的线性常数，则可以用来对转矩进行估计。将瞬时转速信号进行傅利叶变换，得到发火频率点的幅

值和相位，可以得到特征指标 $N(Z)$ ，从而根据式（15）获得平均转矩。

2.5 串口通讯

计算得到平均转矩后，将结果发送给其它微控制单元或直接传给上位机，可采用串口通讯或 CAN 通讯。本设计采用串口通讯方式。为了提高通用性，设计中采用了公头和母头两种连接方式。

2.6 电源模块

汽油车的蓄电池电压一般为 12V，而柴油车的蓄电池一般为 24V。为了给 dsPIC30F4012 供电，需要进行电平转换。本设计采用 LM2596S-12.0 将 24V 或 12V 电池电压先统一调整到 12V，电池电压进入 LM2596 之前需要进行滤波和防反接处理。12V 电平接着由 LM2596S-5.0 降到 5V，从而给 dsPIC30F4012 供电。电源上电后，导通一个绿色发光二极管，作为电源指示。

2.7 彩灯模块

为了便于程序调试，在设计中将 dsPIC30F4012 的 6 个数字输入输出口 D1~D6 配置为输出，各驱动一个发光二极管。可以在程序中控制这六个发光二极管的亮和灭，从而对程序的运行状态作出指示。例如，在中断处理程序中将这六个发光二极管点亮，而在主程序中则令其熄灭。如果发光二极管始终保持熄灭状态，则说明程序没有进入中断。只有发光二极管闪烁或发光时，程序才成功进入中断。当输出为低电平时，发光二极管点亮，反之则熄灭。

2.8 备用功能

由于汽车上各控制器之间的通讯多采用 CAN 总线，为二次开发考虑，将 CAN 总线电路也纳入本设计范围。需要说明的是，采用 CAN 通讯时，由于其和编程器接口复用，不能对其进行调试模式下的调试。只能通过跳线转换开关，先将端口用于编程器接口，下载程序；然后改变跳线，将端口用于 CAN 通讯电路。CAN 通讯电路采用 82250 作为 CAN 控制器，并使用共模电感对信号进行滤波。CAN 高和 CAN 低之间需要接 120 欧姆的终端电阻。

2.9 dsPIC30F4012 的晶振和编程接口

外围电路基本设计完成，他们分别占用 dsPIC30F4012 的相应端口资源。同时，本设计还对 dsPIC30F4012 配置了外部晶振，尽管其具有内部晶振，但使用外部晶振可获得更灵活的时钟频率。晶振配置和 dsPIC30F4012 的端口连接如图 5 所示。

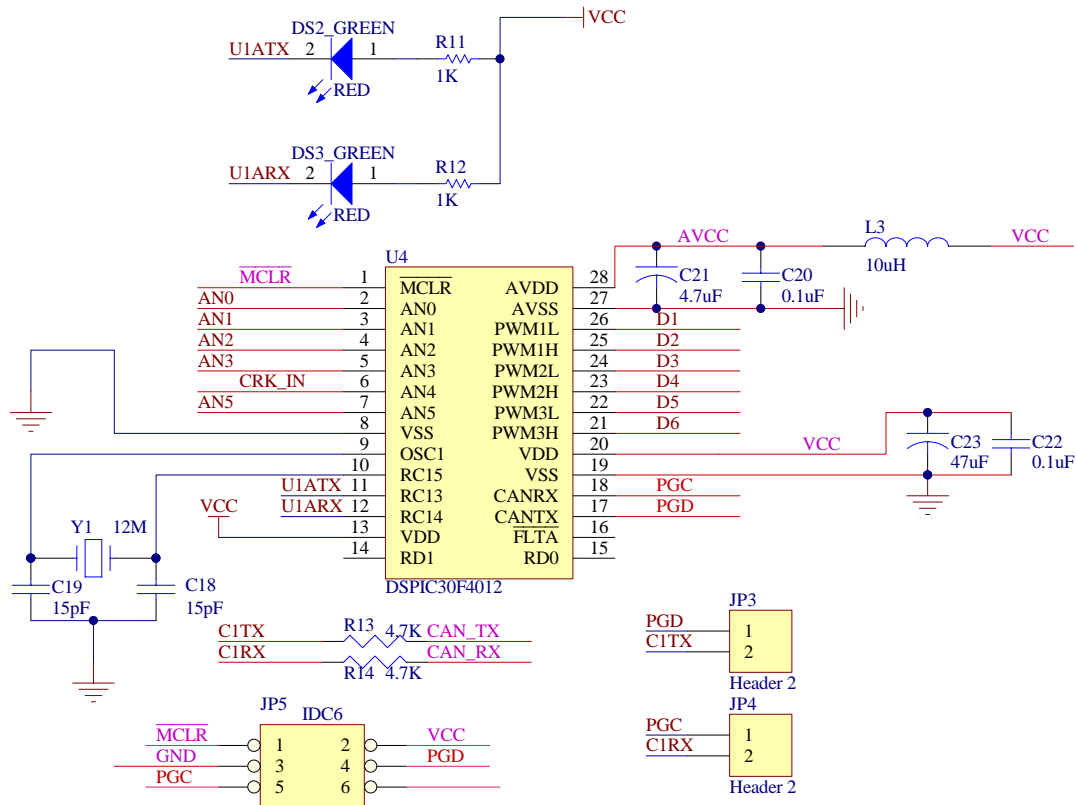


图5 dsPIC30F4012接口配置

3 硬件电路图设计

绘制好原理图后，可以开始设计印刷电路板。第一步，先生成网络表，它是联系原理图和 PCB 板设计的纽带。第二步，在绘制 PCB 板之前，需对 PCB 板作一下规划，包括电路板是采用单面板、双面板还是多面板，电路板的尺寸，电路板与外界的接口形式，各元件采用何种封装形式及其安装位置等。这一步基本确定了电路板设计的框架。第三步，设置环境参数，包括栅格的大小，光标捕捉范围以及所采用的单位等。第四步，装入网络表和元件封装，网络表是自动布线的关键，元件封装是元件的外形，对于每个装入的元件必须有相应的外形封装，

只有正确装入网络表和元件封装才能保证电路板布线的顺利进行。第五步，对元件进行布局，装入网络表后，可以让系统对元件进行自动布局，也可以自己手工布局，只有布局合理，才能进行下一步的布线工作。第六步，布线，同样可以自动布线或者手工布线，或者两者相结合。最后进行 DRC 检验，以确保电路板符合设计的规则。

3.1 PCB 布局

3.1.1 准备工作

根据本次设计方案的复杂程度，选择双面板印刷电路，同时为了保证布线方便，采用的尺寸较大。规划好电路板，生成网络表，载入网络表和元件封装到 PCB 板，即可进行布局。

3.1.2 手工布局

本次设计的电路具有一定的复杂度，采用自动布局结果较差。因此采用手工布局，布局过程中的原则是实现同一功能的元件组合尽可能靠近，同时考虑美观协调。这里电磁兼容性的影响不大，所以不作为主要考虑因素。最后得到的布局效果如图 6 所示。

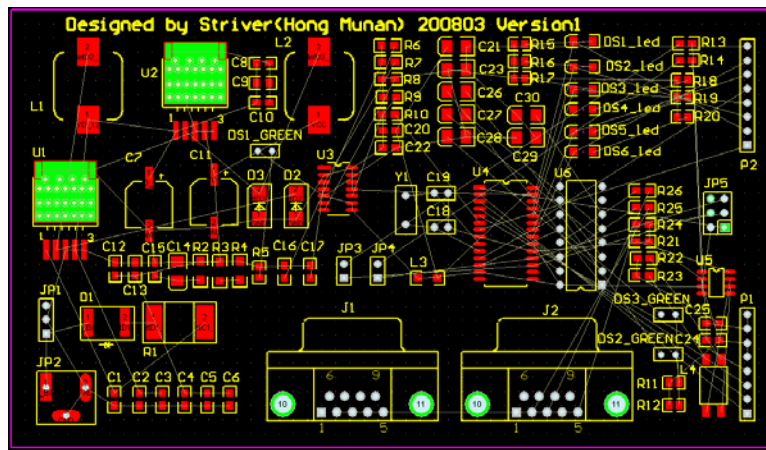


图 6 手工布局效果图

3.2 PCB 布线

完成布局后，可以开始布线。Protel DXP 提供了自动布线功能，但是自动布

线的功能并不十分完善，稍微复杂的电路容易发生死机，因此采用手工布线。手工布线可以调整各线的线宽，以符合不同导线对导电能力的要求，同时还可以增加过孔，避免连线太长或者陷入死回路，比起自动布线具有更大的灵活性。手工布线的原则有三个，一是各元件之间引脚连线尽可能短；二是连线尽量不要从元件的引脚间穿过，否则，容易造成短路；三是连线应尽量整齐美观。布线后的效果如图 7 所示。

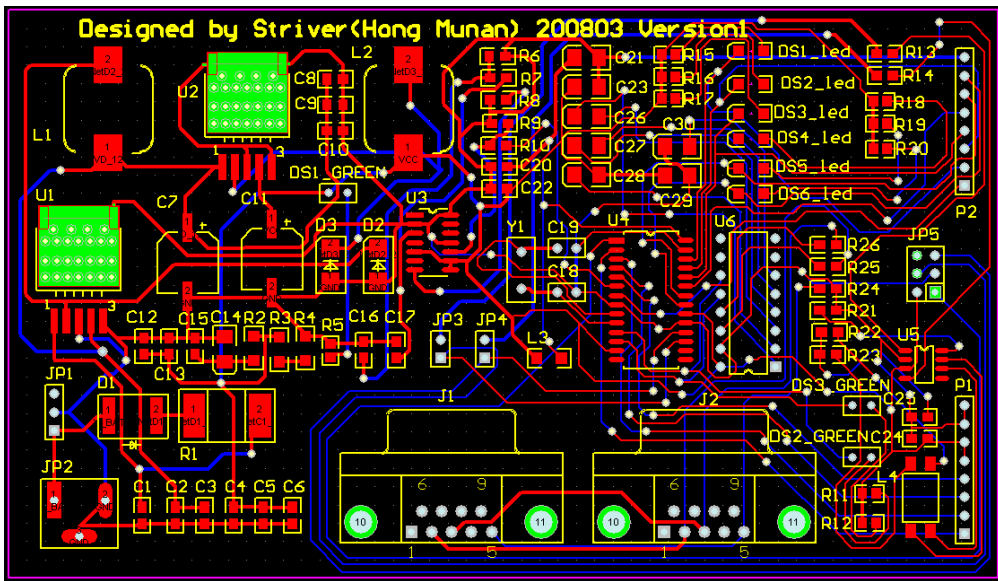


图 7 手工布线效果图

3.3 PCB 制板与调试

完成印刷电路板的设计，即可送往工厂加工，制造电路板。在工厂制板期间，将所用到的元器件清单备齐。将购买到的元器件逐一焊上，调试每一个功能模块，保证每一子功能都能顺利实现。

图 8 是电路板制作完成并焊上元器件的效果图。



图 8 制作并完成焊接的电路板

4 软件设计

4.1 集成编译环境

本设计借助 MPLAB IDE 集成编译环境，使用 C 语言进行编程。MPLAB IDE 不仅集成了软件，还集成了 Microchip 的所有硬件工具以及许多第三方工具。

MPLAB IDE 的主要特性如下：

- 为在 Windows XP、2000 和 Windows NT 系统下运行而设计
- 项目编译和管理
- 灵活的 Watch（观察）窗口
- 鼠标悬停在变量上检查变量
- 使用色彩区分不同功能代码的全功能代码编辑器
- ASM 和 C 语言的源代码调试
- 可搜索的跟踪缓冲区
- 集成版本控制

首先建立新工程，然后选择目标设备，这里采用的是 dsPIC30F4012。在 MPLAB IDE 下可采用图形化界面进行个功能模块的初始化。初始化配置完成后，生成初始化函数代码 `VisualInitialization()`，直接在用户程序中调用该初始化函数即可实现图形界面中配置的初始条件。

4.2 软件编写

4.2.1 头文件和初始化

微控制器通常将特殊功能寄存器定义在头文件中，方便程序员进行软件编写，编程时只需将头文件包含即可。

如前所述，MPLAB IDE 提供了可视化的初始化配置界面，因本设计初始化较为简单，故直接在主函数开头利用 C 代码实现初始化配置。内容包括：

- 变量初始化
- 计时定时器配置（Timer1，用于计算代码执行时间）
- 输入捕捉定时器配置（Timer2）
- 输入捕捉参数配置
- UART 参数配置

需要特别说明的是，代码中调用的一些函数如 `OpenCapture7(.....)` 为 dsPIC 外设函数库中的函数。使用这些函数有助于减少编写代码的工作量。如果调用了这个库中的函数，需要将相应的库文件添加到所建立的工程（Project）。对于 dsPIC30F4012,所需添加的库文件为 `libp30f4012-coff.a`。

4.2.2 输入捕捉

输入捕捉端口设置在 IC7，配置成每个下降沿进行捕捉。当输入脉冲下降沿到来时，进入中断，在中断中读取该脉冲周期，存储在缓冲区。对于 60-2 齿的飞轮，加上虚拟齿，一转可得到 60 个周期，一个发动机工作循环（2 转）得到 120 个周期。当缓冲区存满 120 个周期时，将 FFT 变换需求标志置 1，请求进行 FFT 变换。

4.2.3 FFT 变换

如果借助 DSP 函数库，傅立叶变换（FFT）的实现将变得更加容易。但是调用 FFT 变换函数之前，需要对函数的输入参数做一些必要赋值。其中稍微麻烦的是要先得到进行 FFT 变换所必需旋转因子。FFT 变换的输入数据必须是 2 的整数次方，考虑到计算时间的要求，本设计采用 64 位数据，进行 FFT 变换。如前所述，由于一个发动机循环能采到 120 个周期，即 120 个瞬时转速数据，而进行变化只需要 64 个数据。这里通过抽取法，隔一个数据取一个，则一个发

动机循环有 60 个数据。还缺的 4 个数据，通过令其和最前面的 4 个数据相同，从而得到进行 FFT 变换的 64 个数据。确定需要进行变换的数据个数后，旋转因子可以直接通过式 (16) 计算得到。

$$W_N^n = \left(e^{-j\frac{2\pi}{64}} \right)^n, \quad n = 0, 1, 2, \dots, 63 \quad (16)$$

由于式 (16) 的特殊性，后半部分的旋转因子可由前半部分进行简单的符号变换得到，所以只需提供 $N/2$ 个旋转因子。每个旋转因子的实部和虚部均由 2 个字节进行存储，存储方式为 1.15 小数模式。旋转因子计算还可通过调用 DSP 库的另一个函数 `TwidFactorInit` 得到。完成 FFT 变换的准备工作后，直接调用 DSP 库函数 `FFTComplexIP()` 并输入相应参数即可实现 FFT 变换。

4.2.4 转矩计算

完成 FFT 变换的准备工作，并集成为新的专门为 64 位数据作 FFT 变换的函数。在主函数的死循环中即可直接调用该子函数，完成瞬时转速的傅立叶变换。主函数死循环中判断 FFT 变换需求标志，如果标志被置 1，则从缓存区提取瞬时转速数据，进行 FFT 变换。经过变换得到和转矩相关的特征指标，由于该特征指标和转矩存在线性关系，即可得到平均转矩。其中线性常数和转速有关，需要事先由发动机标定得到。

4.2.5 其它附加功能函数

得到平均转矩后，可通过 UART 将数据结果送到上位机或其它控制单元。

为了方便程序调试，设置 RE0~RE5 为输出端口，控制发光二极管的亮灭。彩灯函数代码实现“两两”移动的功能，即两个发光二极管同时亮，然后向一个方向一步一步移动，如此周而复始。彩灯函数代码段可作为程序运行状况的外在指示进行调用。

4.2.6 主函数

上述已经较为详细地对软件各个子模块进行介绍，主函数的基本流程图如图 9 所示。

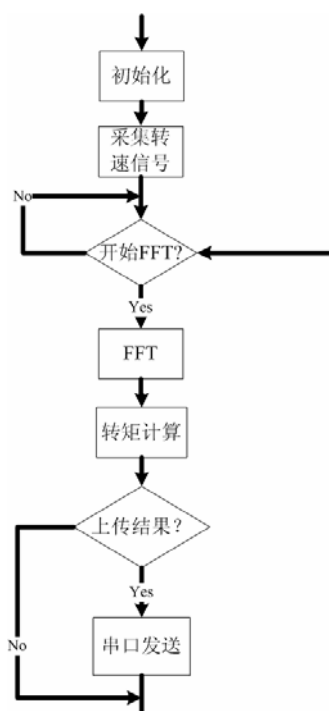


图 9 主函数的基本流程

5 试验验证

经过上述软硬件设计，即可实现本次设计的预期目标，本章将结合试验对其有效性进行验证。

5.1 拟合线性常数的获取

以某 4 缸柴油机为研究对象，在测功机上获取试验数据。各个稳态工况的选取如表 1 所示。

表 1 发动机工况选取

转速	转矩 1	转矩 2	转矩 3	转矩 4	转矩 5
1400rpm	50N.m	100N.m	验证点	200N.m	
1500rpm	50N.m	100N.m	验证点	200N.m	250N.m
1600rpm	50N.m	100N.m	验证点	200N.m	250N.m
1700rpm	50N.m	100N.m	验证点	200N.m	250N.m
1800rpm	50N.m	100N.m	验证点	200N.m	250N.m

转速范围取 1400~1800rpm，间隔为 100rpm；每个转速下的转矩分别取

50N.m~250N.m，间隔为 50N.m，其中 1400rpm 下的转矩最大值只能取到 200N.m。而各个转速下的 150N.m 工况不用于进行线性拟合，是为了留给虚拟转矩传感器进行试验验证。

对表 1 中每个稳态工况点的瞬时转速进行采集，在 Matlab 下进行 FFT 变换求取特征指标。各个转速下的特征指标和平均转矩的关系如图 10 所示。可见，特征指标和平均转矩存在如式 (17) 的近似线性关系。

$$\bar{T}_l = a(n) \times N(Z) + b(n) \tag{17}$$

其中， $a(n)$ 和 $b(n)$ 为线性拟合系数，只与转速有关。

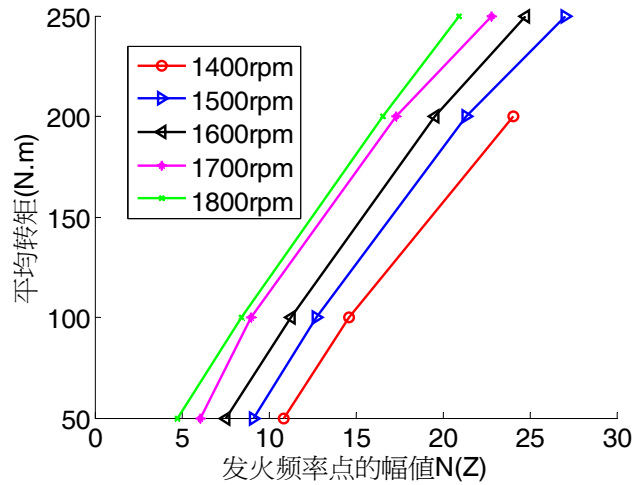


图 10 特征指标和平均转矩的关系

对图 10 所示关系进行线性拟合，即可得到各个转速下的拟合系数，如表 2 所示。

表 2 拟合常数

转速	a(n)	b(n)
1400rpm	11.2099	-68.5054
1500rpm	11.1827	-46.7679
1600rpm	11.6647	-33.8538
1700rpm	11.8656	-13.7888
1800rpm	12.3543	-6.4403

对于平均转速介于 1400~1800rpm 之间的情况，线性系数可通过插值得到。

5.2 试验验证

将上述线性系数存储在虚拟转矩传感器的 Flash 中，从而可对 1400~1800rpm 之间的转矩进行测量。

1400~1800rpm 之间每隔 100rpm 取一个工况点，每个转速下都将测功机转矩控制在 150N.m，利用设计得到的虚拟转矩传感器对瞬时转速进行采集，进而计算转矩，得到的转矩和实际测功机测得的转矩进行比较，误差如表 3 所示。

表 3 结果比较

转速	估计转矩	实测转矩	绝对误差	相对误差
1400rpm	151N.m	150N.m	1N.m	0.67%
1500rpm	142N.m	150N.m	8N.m	5.33%
1600rpm	146N.m	150N.m	4N.m	2.67%
1700rpm	140N.m	150N.m	10N.m	6.67%
1800rpm	147N.m	150N.m	3N.m	2.00%

可见，虚拟转矩传感器可以用来作为发动机的平均转矩测量，且具有 7% 以内的精度。完成 1 次 FFT 变换及转矩计算的时间为 15~20ms，由于发动机在 6000rpm 的最高转速下对应循环周期 20ms，也就是说转矩测量完全可以在一个循环内完成。

6 展望和建议

由于时间和精力限制，所完成的样机存在一些不足，需要进一步改进。例如，串口布局不恰当，电源接口也存在一些问题等等。样机的硬件需要进行几次迭代设计，提高其可靠性；软件代码也可作进一步优化，提高代码效率。

7 致谢

首先感谢 Microchip 公司和电子产品杂志社，给我们提供好产品的同时，又组织了这次嵌入式大奖赛，使我们得到一次锻炼机会。本设计能顺利完成，还要归功于清华大学汽车研究所发动机控制实验室的实验条件，样机的完成全部在该实验室内完成。感谢李建秋副教授的积极支持，感谢所有对本设计提供过帮助的老师 and 同学们。

8 参考文献

- [1] J. Franco, M. A. Franchek, K. Grigoriadis. Real-time Brake Torque Estimation for Internal Combustion Engines. Mechanical Systems and Signal Processing, Volume 22, Issue 2, February 2008, Pages 338-361.
- [2] 童毅. 并联式混合动力系统动态协调控制问题的研究:[博士学位论文]. 北京:清华大学, 2004
- [3] S. J. Citron, J.E. Higgins, L.Y. Chen. Cylinder by Cylinder Engine Pressure and Pressure Torque Waveform , Determination Utilizing Speed Fluctuation[G]//SAE Paper 890486, 1989.
- [4] 李建秋. 电控柴油机分缸独立闭环控制的研究[D]. 北京:清华大学, 2000.
- [5] 朱勇, 吴周桥, 乔宇锋等. Protel DXP 范例入门与提高. 北京:清华大学出版社, 2004.
- [6] 龙腾科技. Protel DXP 循序渐进教程. 北京:科学出版社, 2005.
- [7] Rizzoni. Estimate of Indicated Torque from Crankshaft Speed Fluctuations: A Model for the Dynamic of the IC Engine. IEEE Transaction, Vehicle Technology, Vol. VT-38, NO.3, August 1989.
- [8] Ribbens, G. Rizzoni. Applications of Precise Crankshaft Position Measurements for Engine Testing, Control and Diagnosis. SAE Paper 890885, 1989.
- [9] Microchip, dsPIC语言工具库, 2005.
- [10] Microchip, dsPIC30F 系列参考手册, 高性能数字信号控制器, 2006.
- [11] Microchip, dsPIC30F4011/4012 数据手册, 高性能数字信号控制器, 2007.

附录 虚拟转矩传感器操作说明

1. 连接转速传感器信号, 信号正端接电路板的 R6 或 R7 左端, 信号负端接电路板的地 (P1 的 7 和 8, 或者 P2 的 8);
2. 将串口和上位机相连, 波特率设置为 9600;
3. 给电路板供应 12V 电源;
4. 从上位机发送一个十六位进制数: 0x22, 下位机接收到后即可上传数据;
5. 上位机接收平均转速、特征指标和平均转矩, 每一个发动机循环共接收 64 位, 格式为平均转速高 8 位, 平均转速低 8 位, 特征指标高 8 位, 特征指标低 8 位, 平均转矩高 8 位, 平均转矩低 8 位, 最后 16 位均为 0;
6. 得到的平均转速、特征指标和平均转矩, 均要减去 32767, 才是真值。因为程序中发送数据之前都加了 32767, 以便能够传递负数。