



DEWESoft™
measurement innovation

DEWESoft v7.0

数据采集，数据处理，分析以及存储软件

操作手册

V2.0.0

一般信息

软件版本

本DEWESoft 操作手册对应于软件版本 7.0。

印刷公告

DEWETRON 保持改变技术指标不提前通知的权利。

版权公告

© 2008 DEWESOFT 和 DEWETRON 电子有限公司（elektronische Messgeraete Ges.m.b.H）。保留所有权利。

商标公告

All trademarks are acknowledged to be the property of their owners, with all rights and privileges thereto. No infringement is intended.

Microsoft, Excel and Word are registered trademarks of Microsoft Corporation.

National Instruments is a trademark of National Instruments

Microstar, DAP and iDSC are trademarks of Microstar Labs, Inc.

Data Translation is a trademark of Data Translation, Inc.

Disclaimer

DEWETRON and DEWESOFT makes no claim about the efficacy or accuracy of the information contained herein. Use of this manual is entirely at the user's own risk. Under no circumstances will DEWETRON and DEWESOFT assume any liability caused by the use, proper or improper, of this manual or the information, textual, graphical or otherwise, contained within it.

目录

教程	1
基础教程	2
简单测量	3
通道设置	4
视频设置	5
第一个测量	5
设置更好的显示屏幕	7
分析	7
报告和输出	9
显示设置	10
存储策略	14
用于快速	15
用于慢速	16
触发后快速	18
触发后快速，其他时间慢速	21
后处理	23
测量教程	27
电压和电流	28
通道设置	29
测量	31
电涌触发	33
触发存储	36
电压/电流/数字输出传感器	39
通道设置	40
测量	42
分析	43
温度	45
通道设置	46
PAD通道设置	47
测量	49

振动	50
通道设置	52
数学通道设置	57
视频设置	59
测量	59
分析	61
应变测量	63
试验设置	66
1/4桥设置	67
1/4桥测量	69
全桥设置	72
全桥测量	74
力传感器设置	76
计数器	77
事件计数	78
简单事件计数	79
门事件计数	81
上/下事件计数	82
编码器	83
X1, X2, X4模式	85
带有零位脉冲的编码器	87
脉冲周期和宽度	88
脉冲周期测量	89
脉冲宽度测量	90
占空比测量	91
两个脉冲沿隔离	92
频率 / 超级计数器	94
传感器测量	96
车辆上的计数器	98
频率测量	103
通道设置	103
解码器测量	105
光电探头测量	107
视频采集	111
通道设置p	112

测量	114
视频后期同步	116
CAN 总线采集	118
通道设置	120
测量	122
GPS采集	124
通道设置	124
测量	126
电力模块教程	128
单相电力测量	129
通道设置	131
测量	133
传感器校正	136
动态信号分析仪教程	139
声级计	139
通道设置	140
传声器校准	141
测量	143
扭振	144
旋转振动设置	145
旋转振动测量	146
旋转阶次抽取	147
扭振设置	150
扭振测量	151
扭振阶次抽取	152
人体振动	153
输入通道设置	154
输出通道设置	156
测量	158
阶次跟踪	159
通道设置	159
测量	161

燃烧分析	164
通道设置	164
分析设置	166
测量	173
分析	174
网络数据采集教程	175
设置测量单元和客户端	176
远程控制从动 MU	180
MU 上的通道设置	182
MU 上的显示设置	183
MU 的传输设置	183
测量	184
分析	186
索引	188

数据采集教程

DEWESoft 7.0 同时为您提供基本教程，该教程提供以下程序各部分的操作程序：

基本教程

- 如何在**DEWESoft**软件中进行简单的测量？
- 显示设置
- 储存策略

测量教程

- 电压和电流
- 电压输出传感器
- 温度
- 振动
- 应变传感器
- 计数器
- 频率测量
- 视频采集
- **CAN** 总线测量
- **GPS** 采集

电力教程

- 单相电

动态信号分析教程

- 声级计
- 扭振
- 人体振动
- 阶次跟踪

燃烧分析教程

网络分布式采集教程

1

基本教程

对于第一次使用DEWESOFT的用户来讲，基本操作“必须”熟悉。基本操作让您熟悉如何进行基本测量任务。

	如何进行简单测量	
	测试界面设置	This tutorial explains how to design custom screens .
	后处理	
	存储	

1.1 简单的测量

让我们开始用 **DEWESoft** 来做一个简单的测量。我通常会用到我们最常用的一些硬件：一张声卡，一个网络摄像头。为了能够用这两种硬件进行接下来的演示，我们必须在 **System > Hardware setup > Analog** 标签中选择 **Audio card**，插上网络摄像头，在 **Hardware setup > Video** 标签中选择 **DirectX** 摄像机。虽然我们手上只有这些简单的设备，我们仍然可以用它来演示 **DEWESoft** 软件的强大功能。

以后所有的教程都将列出所需的硬件和软件选项，以符合各教程中的软硬件需求：

所需硬件	声卡，网络摄像头
所需软件	任意license
设置采样率	至少1 kHz (采样率设置需要在 DEWESoft Tuner 中选定，某些数学模块可能需要更高的采样率)

在软件的工具栏做上半部分有两个主要的按钮，分别叫做**采集Acquisition**和**分析Analyse**。这两个工具按钮主要用在采集模式下的采集数据和分析模式下的数据处理两种模式中切换。



当打开 **DEWESoft** 软件时，软件默认处在采集模式下，所以首先来看一下[如何进行通道设置](#)。通道设置完成后，我们将继续熟悉下列章节：

- [视频设置](#)
- [第一次测量](#)
- [设置更好的显示屏幕](#)
- [分析](#) 以及
- [报告和输出](#)

1.1.1 通道设置

点击**通道设置 Ch Setup** 按钮，这时会打开通道设置**Channel setup** 窗口，在窗口中我们能够看到包括两个模拟通道(立体声)的**模拟**

Analog标签以及**视频Video**标签和**数学Math** 标签。

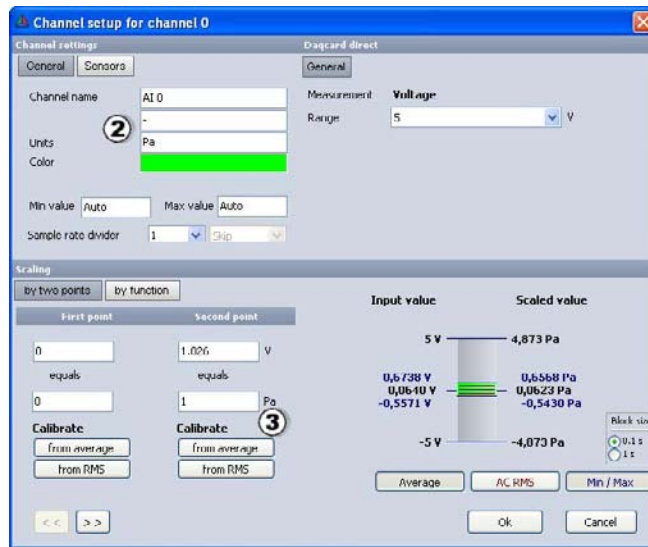


在窗口的上面，我们可以输入文件名和文件存储路径，下面一点则是采样率 **sample rate** ① 的选择。采样率是数据采集的基本速度。我们需要根据采集任务的不同仔细选择采样率。比如我们要测量温度，我们只需要 **100 Hz** (样本每秒) 甚至更低的采样率。如果我们选择的采样率过高，那么可能会浪费太多的硬盘空间，而且并不能获得更多的额外信息。

在本例中，我们需要测量的是声音，我们需要选择的采样率设为 **10 kHz** (10000 个采样值每秒)，这中采样率才能正确采集声音信号。实际上，如果需要更准确了解音频性能，需要设置的采用率要到 **40 kHz** 以上。

接下来我们选择使用的通道。因为我们只有一个麦克风，所以决定只使用其中一个通道。通过点击 **on/off** 按钮来确定。注意每个通道的最右边有一个 **设置 Setup** 按钮。

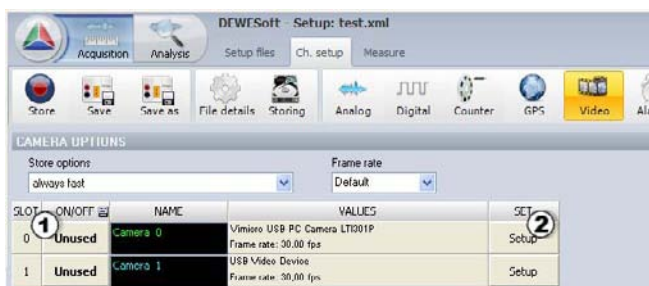
即使我们使用的是很简单的一个麦克风，我还是要强调正确设置参数的重要性。首先我们需要定义测量的单位 **units** ②。麦克风测量的物理量是什么量呢？是声压级，单位为 **帕斯卡 Pa**。因此我们需要把单位设为 **Pa**。



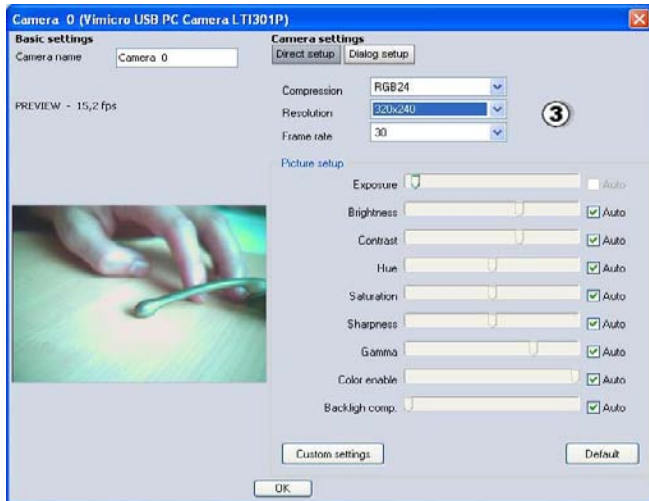
接下来我们做一个简单的“校准”。声音通常用一个正弦波来校准。所以我们简单的模拟一个正弦波，用口哨就可以了。我们可以看到信号的RMS值会不断变大，直到某个比较稳定的级值，而且我们可以使用“**Calibrate from RMS**”按钮③来使之确定等于某个级值，假设为 **1 Pa**。在以后的教程中，我们会真正使用信号发生器来真正按上述方法来校准实际的测量传声器，真正的校准器其产生的声音级值是确定的 **1 Pa** 信号。

1.1.2 视频设置

接下来我们需要设置视频通道。点击选择 **视频 Video** 标签，在这个标签里可以看到已经安装好的网络摄像头。点击 **Unused** ① 按钮，打开摄像头，然后点击右边的 **设置 Setup** ② 按钮。



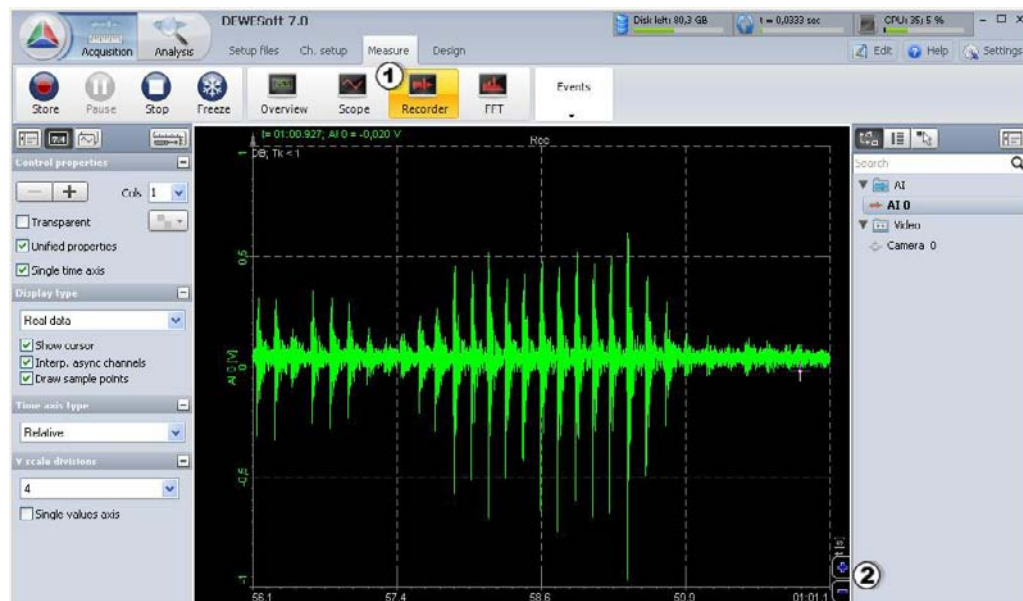
在设置窗口中，我们可以改变摄像头的图像帧速率，分辨率，压缩率以及其它特性③。有关摄像机设置更详细的信息会在 [视频采集教程](#) 中详尽阐述。我们现在只进行简单的设置。



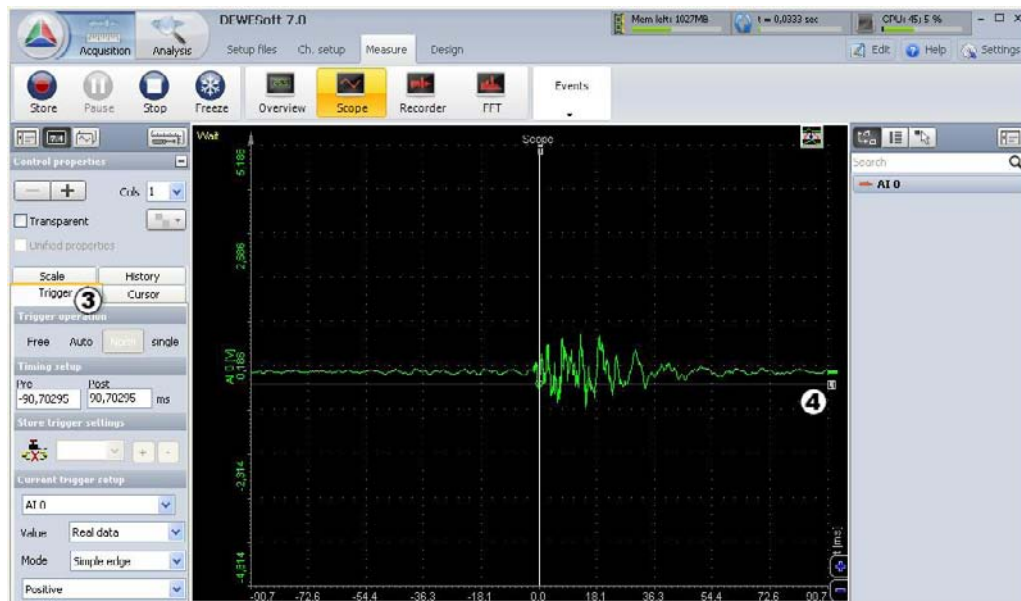
现在我们来进行我们的第一个测量。

1.1.3 第一个测量

点击记录仪**Recorder**按钮①。系统会立即开始采集数据。现在我们来试一下麦克风怎么工作的，我们故意制造一些噪声。我们可以使用窗口右下角的**+**或**-b**按钮②放大或缩小时间轴。



现在我们切换到示波器**scope**窗口在快速情况下观察一下数据。在左边触发**trigger**标签上选择**常规Norm**触发器③，用鼠标拖动级值线 **L1** ④，大概在满量程**10 %** 的地方停止。然后用指头敲击麦克风，模拟产生触发信号。我们就能清楚看到不同的声音会产生不同的形式。



接下来要做到的是切换到**FFT**窗口，进行频率分析。选择合适的分析线数(在本例中，如图所示我选择是**4096 线**⑤)，那么频率分析的分辨率可以达到**1 Hz**⑥以及合适的**坐标**⑥显示，以便能更好的观察谐波。

这时你可以试着唱出你能发出的最好的A调音，(A3 音调频率大概为**220 Hz**，A2 的频率大概为**110 Hz**)，如果你有钢琴或者其他任意乐器，演奏一下，试试看这些乐器调准的情况。还可以看看口哨，歌唱和音乐乐器之间的差异。从这个试验中，我们可以发现确实是声音中的高频部分决定了音乐中的相同音调的音色。



现在我们回到记录仪**recorder**，我们会发现所有的数据仍然还在。我们可以放大，缩小来观察用麦克风记录的整个时间段内的数据。

下一步是储存数据。点击**存储Store** 按钮⑦，存储按钮上面的会出现**红色亮光**。我们也能在右上角看到储存的模拟数据的大小，以及视频文件的大小。

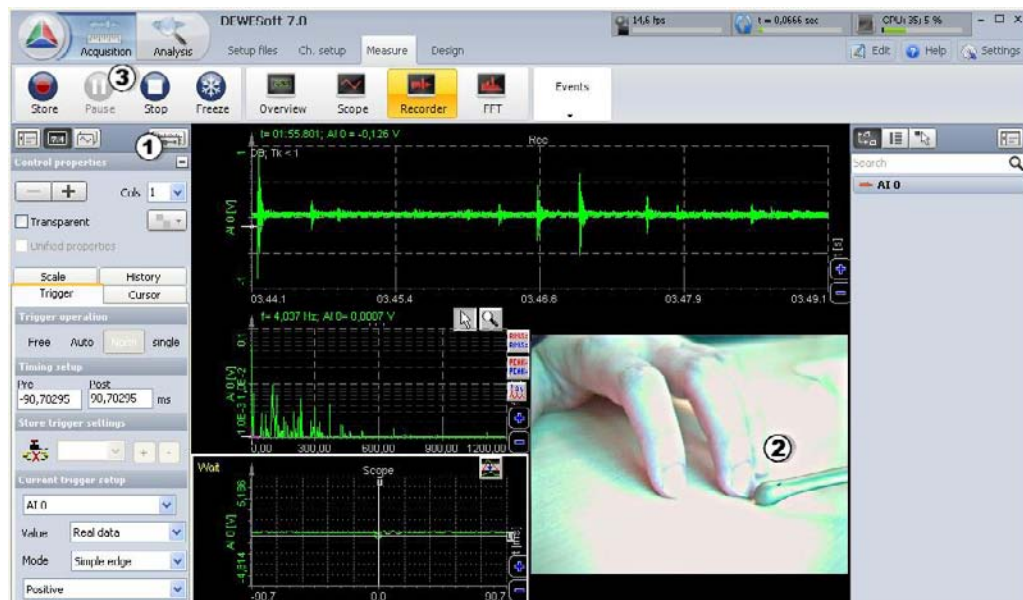
1.1.4 设置更好的显示界面

我们现在来看看什么时候使用每一个仪器。这些虚拟仪器全部都集成在**DEWESoft** 软件中，可以随意设置这些虚拟仪器的组合，创建您喜欢的显示界面。实际上，任何显示界面都可以根据需要随意修改，不过我们还是从概览**Overview**显示界面开始修改，概览**Overview**显示本来的目的就是用来组合各种虚拟仪器。

首先我们需要注意到**分析工具栏Analyse**下面有一个**设计Design**按钮①。当点击设计**Design**按钮时，就会在主工具栏下出现一个虚拟仪器工具栏。我们可以利用该虚拟仪器的工具栏增加任意虚拟仪器。所以，我们先增加一个记录仪**recorder**，示波器**scope** 和**FFT**分析仪。我们可以拖拉，任意改变虚拟仪器的尺寸以适合自身的需要。不要忘了 – 我们还有一个视频信号。所以点击一些摄像机**camera**按钮增加一个视频**video**窗口②。



如果我们想要放大/缩小横轴，或者改变触发电平，必须再点击一下**设计Design**按钮，退出设计模式。此时，虚拟仪器工具栏会消失，然后我们才能完全控制各虚拟仪器的功能。现在您可以联系一下，设置这些常用的虚拟仪器，改变其显示界面。



还记得我们一直在记录数据吗？点击**停止Stop** 按钮③，停止数据存储。

1.1.5 分析

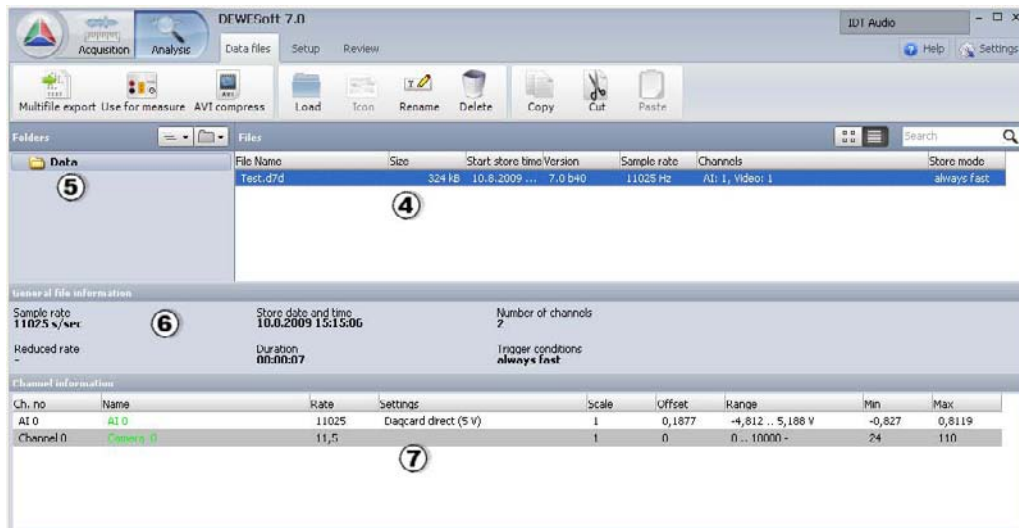
接下来是要进一步深入了解数据，也就是如何分析存储的数据。

起始很简单 – 只需点击**分析Analyse** 按钮。因为我们已经将数据存储于硬盘上，**DEWESoft** 默认最后存储的文件是我们想要打开浏览的文件。如果我们在其它时候点击**分析Analyse** 按钮(比如刚打开**DEWESoft** 软件，或者处在设置窗口)，则会打开文件浏览器④，我们可以看到主文件目录下的各个子文件夹，文件列表以及被选文件详细的文件信息：文件大小，存储时间，版本号，采样率，通道数，存储策略以及视频文件(如果有的话)。不要被这些东西搞糊涂了，只需双击需要打开的文件名即可。

如果还有子目录，双击打开子目录⑤，然后选择子目录下的文件。第一层数据目录可以更改，点击  按钮，这样会打开更高一级的目录，点击文件列表 ，选择 "**设置成默认目录Set by default**" 选项，将选择的文件夹设置成默认的数据存储文件夹。

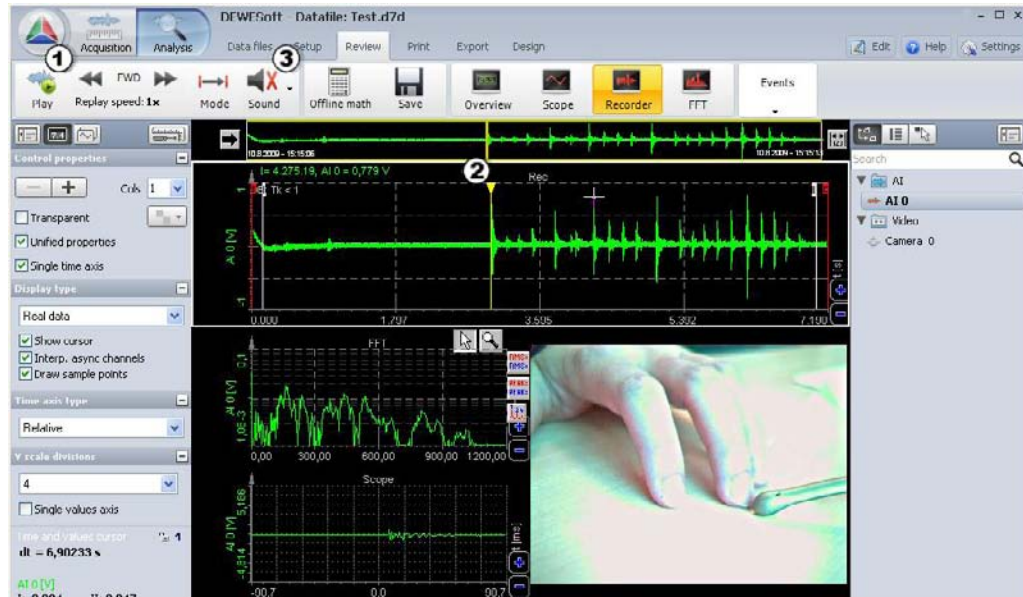
选择某个确定的文件后，软件预先扫描，显示通道的重要信息，事件以及数据文件头。这些信息显示在文件浏览器下面的通用文件信息General file information 窗口中⑥，以及通道信息Channel info列表中。注意，如果采集过程中通道有过载发

生，还会显示最小和最大值，以红色显示。



现在我们可以打开一个文件。我们可以选择任意内置在测量模式内的任意显示界面。有各种各样显示数据文件的方法。在右

上角，有一个回放play按钮①，选择该按钮后，我们能看到一个黄色光标②在数据上移动，FFT分析已经计算出来，示波器scope显示的是当前的数据，视频video文件重新回放。回放开始后，Play按钮变成停止Stop按钮。再点击一次这个按钮将停止stops数据回放。我们设置可以听到我们刚刚存储的声音。紧挨着Play按钮旁边，有一个扬声器loudspeaker按钮③，通常上面有一个红色的叉。点击扬声器按钮，会出现一个通道列表。选择这里唯一一个存储数据的通道，扬声器上面的叉消失。然后在点击Play按钮，此时就会从扬声器中听到我们记录到的所有的信号。再一次点击停止Stop按钮，停止回放，再选择扬声器按钮上的None，关闭音频回放。回放数据时，你可能已经注意到了黄色光标在整个数据内移动。你还可以用鼠标拖动光标，快速在数据文件中移动。



如果素具文件非常长，这时候使用zoom功能就非常有用，快速放大某个特定区域。注意记录仪recorder有另外两个银色光标 – 叫做 I 和 II。通过拖动这两个光标，选择特定的数据区域。

另外一个选择数据的方法是在记录仪的任意位置(此处没有红色事件，或者光标不在此处)点击左键，按住不放，这时候在该区域会出现第一个光标 I，拖动鼠标，然后会出现第二个光标 II。

完成拖动后，当把放置在光标 I 和光标 II 之间时，鼠标会变成zoom光标，点击鼠标左键，此区域会放大zoom in。我们可以重复几次这个步骤，直到完全放大到我们所感兴趣的区域为止。我们还可以拖动 x 轴的左右边界到数据的合适

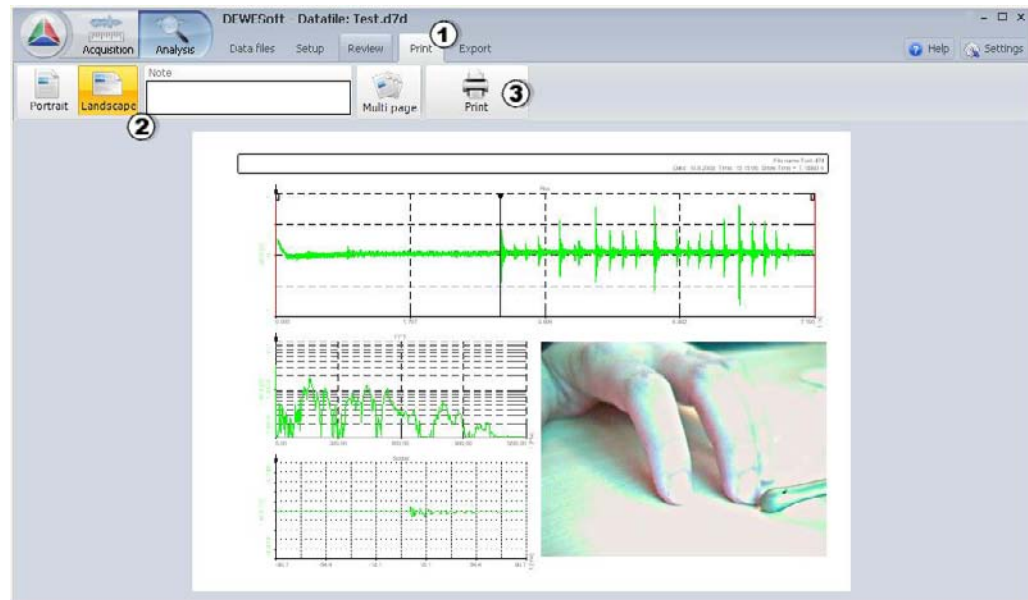


区域。右键点击记录仪recorder将放大选定的数据到整个显示区域。

1.1.6 报告和输出

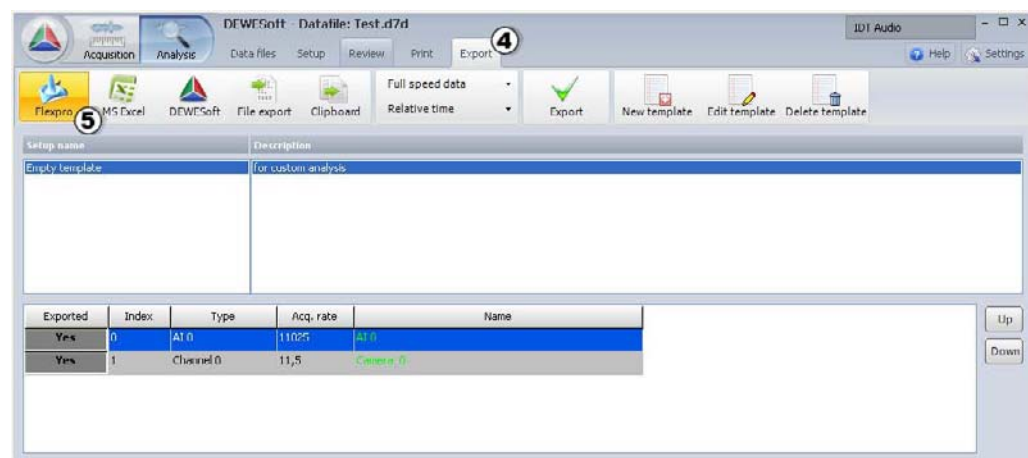
那么我们能对这些数据做什么呢？

首先，我们可以打印当前的页面。只需点击打印Print按钮①，会出现和显示窗口相同的页面(不过显示的是以白色背景"打印友好方式"显示)。这是您可以选择打印机选项②，在打印菜单上点击打印Print按钮，打印数据。



下一步是将数据输出到其他软件中。**DEWESoft** 本身主要是采集软件包，所以如果需要高级分析的话，则需要使用后处理软件。我们可以选择输出Export按钮④，可以将数据输出到其它软件。我们可以选择将数据输出到⑤到MS Excel 或者 Flexpro，设置可以选择script – 这实际上意味着出测量报告。也可以选择文件输出File export，这时我们有更多后处理选项，但是这里我们只用到数据文件。

一个重要的地方就是我们将只输出放大后的数据。因此如果我们电脑上已经安装了Excel，因为Excel不能处理太多的数据，所以最好还是不要选择太多的数据输出到Excel。对于大数据量的数据文件，建议使用Flexpro或者Matlab。



第三个有意思的选项是创建视频文件输出。我们可以通过从菜单中选择Data→ Export screen to AVI。这个选项可以将**DEWESoft**软件屏幕上完全相同的显示生成视频文件。

这是DEWESoft软件功能简单浏览，甚至在我们手头只有简单的硬件情况下就能实现。想象一下如果您拥有专业的设备，那么系统能够做多少工作呢。接下来的测量教程将会教我们如何进行真正的测试。

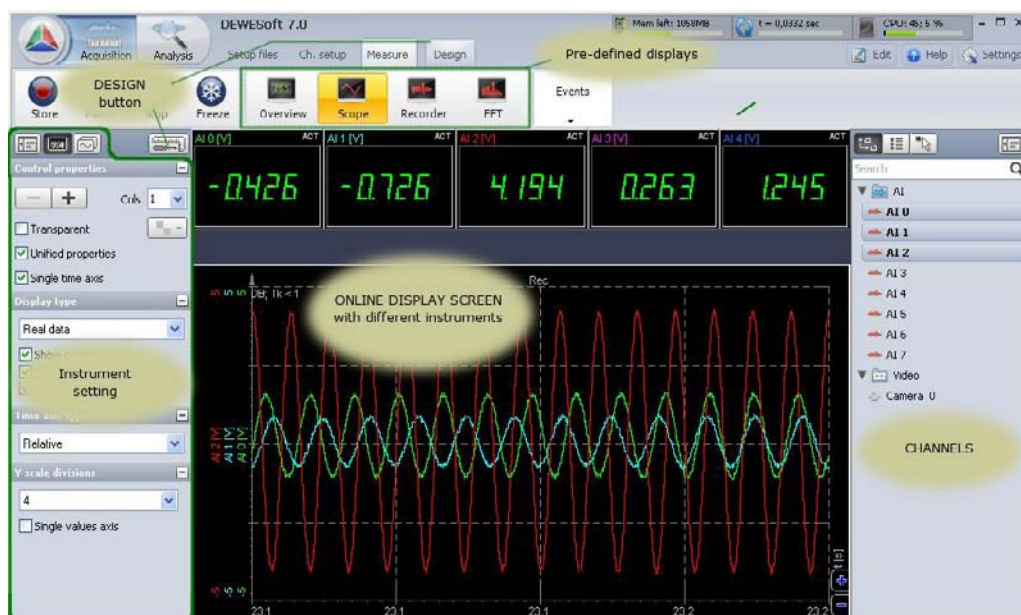
1.2 显示设置

该节教程讲述如何在DEWESoft软件中设计平面显示，以及如何利用这些显示来工作。通常，我们有四种控制显示的方法：

- 显示所有数据的控制(记录仪recorder，垂直记录仪vertical recorder，xy记录仪order，GPS图形)
- 直接显示部分数据或者其它计算的控制(示波器scope，FFT分析仪，倍频程octave，矢量示波器vector scope，谐波FFT分析仪harmonic FFT，表格显示)
- 只显示一个值的控制(数字表digital meter，柱状表bar meter，模拟表analog meter，指示灯 indicator lamp)
- 额外的控制，比如图形，文本或者线条

所有的控制都可以组合在一个显示屏幕上，我们也可以建立几个不同屏幕，用于显示特定的测量。我们还有一些预先定义屏幕显示(比如示波器scope，记录仪recorder，FFT分析仪)，不过我们可以修改这些预定义屏幕满足我们的要求。

现在来看一下，如何使用这些显示，以及如何让这些屏幕显示更加适合我们的要求。概览Overview显示主要目的是让用户建立自己预定义的屏幕显示，同时这种预定义屏幕可以随时加以改变以满足特定需求。



现在我们碰到了一个鸡生蛋还是蛋生鸡的问题。下一章中我们会了解到所有不同的测量显示都已经定义好了，而这一章节又要假设我们已经熟悉创建各种显示的方法。因此，目前我们只采用测量中少数几个通道，然后设置各种显示。在下一章中，我们再看一下这些通道是如何进行采集的。

安装 DEWESoft 软件 时，会 默认在计 算 机上安 装 上

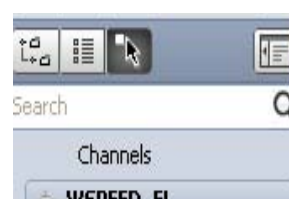
Example_Drive01 数据文件。我们现在打开这个数据文件，这个 文件中包含有许多有趣的通道，非常适合用来演示显示屏幕的创建和 设置。首先，进入到概览 Overview 窗口。这个窗口显得非常拥挤，所 以我们重新建立另外一个显示窗口。从  菜单中选择增加显示Add main display 命令。



这时会出现一个空白的显示，而且处在设计Design模式下。所有以上所述的屏幕控制只能在设计**Design**模式下进行增加等操作，设计模式的选择只需点击**Design**按钮即可。现在我们来看看能从通道中得到什么信息。首先增加一个记录仪recorder，用来记录速度信号。注意记录仪四周有8个可控手柄。我们可以通过鼠标拖拉这些手柄来调整屏幕显示的大小。还能将其拖拉到任何适当的位置。

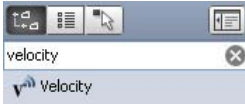


在例子中，测量了所有四个车轮的速度，同时还有GPS测量的速度。当增加一个记录仪时，通常会自动将第一个通道指定给该记录仪。要取消该通道，点击如图最后一个标签，这是所有通道都会显示出来，要取消哪个通道就点击哪个通道

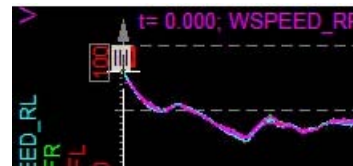


接下来我们可以选择用来显示的通道。注意所有通道已经按类型进行分组。在本例中，有模拟通道，计数器，CAN，数学通道以及GPS通道。因此点击CAN通道组，然后选择 **Bremse3**信息。可以看到所有四个轮速通道。点击一下这些通道，即可将这些通道 增加到显示屏幕控制中。点击选择的通道，显示屏幕控制中自然会取消该通道的显示。每个屏幕显示控制都有一个能够显示的最大通道数的限制。记录仪**Recorder** 最多可同时显示四个通道。这通常是由于y轴的数量限制的。尽管我们只有一个测量值(即 速度)，所有我们可以将所有通道都与同一个单一坐标轴相关，在记录仪中选择**单一坐标轴Single value axis** 选项。在本例 中，记录仪可以同时显示16 通道。

现在我们在通道选择窗口中点击 **Channels**，回到分组显示，选择 **GPS 分组**。在 **GPS 分组**中 选择**速度**通道，我们现在就能看到轮速和 GPS 速度通道之间的比较。



在设计模式下，屏幕显示控制的放大或量度等操作不能执行，因此 我们需要重新点击**Design**按钮，才可以输入最大坐标值，或者自动 定y轴量程。因为整个测量过程中速度都没有超过100 km/h，我们 可以将鼠标移动到y轴的最大值位置，知道鼠标光标变成灰色矩形 框，点击，然后在框内输入确定的值。现在我们设置好了，当然还可以再点击**Design**按钮，继续调整t屏幕显示。



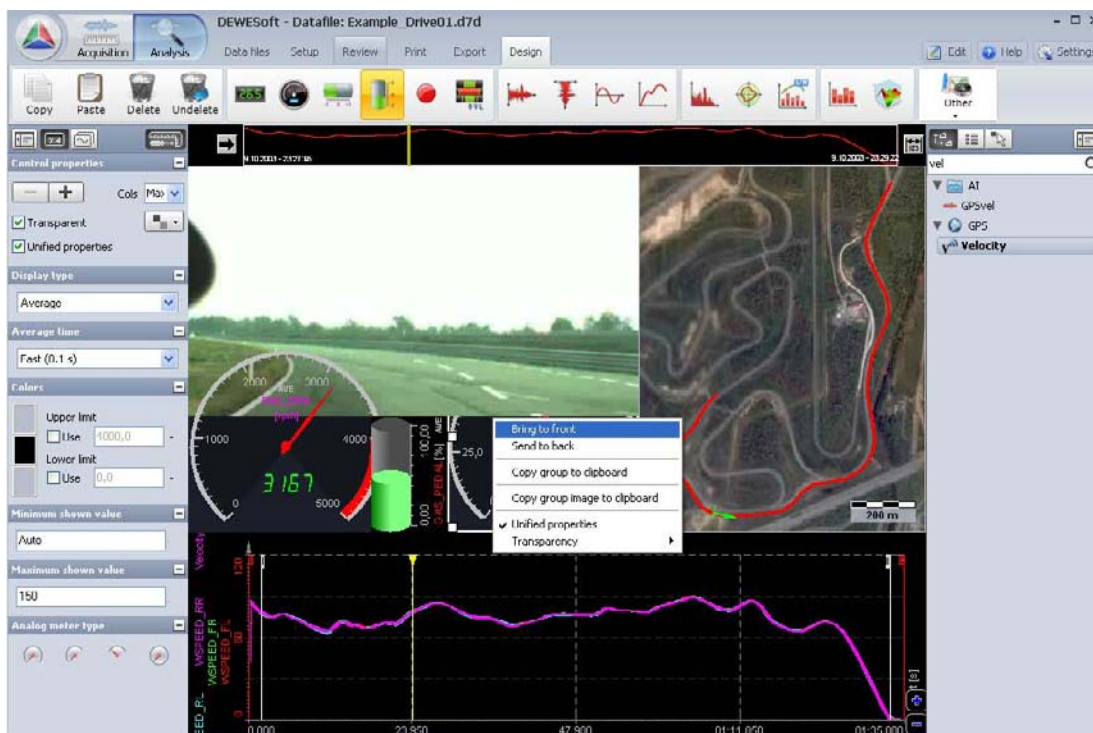
当前情况下最好有一些模拟表盘显示车辆RPM 以及 车辆速度。记录仪recorder通常显示整个 时间段或者所选时间段内的数据，而模拟表盘则只显示当前位置的数据。在测量模式下，模拟表 盘显示的永远都是当前值，而在分析模式下，显示的则是和黄色 光标对应位置的值。所以，我们增加两个模拟表盘，从CAN bus分组中选择**Motor1**下面的**Eng_RPM** 通道，然后选择另外一个模拟表盘，并给分配**GPS速度**通道。我们可以改变其量度值 以适应需要。对于**Engine RPM** (因为这是柴油机)，我们将最大值设为**5000 RPM**。我们还可以选择使用上限**upper limit** (作为报 警)，输入**4000 RPM**作为容差限制。



我们还可以在模拟表盘内加入两个数字表，用数字显示确切的RPM值和速度值。与模拟表盘设置一样，首先增加一个数字表，将其移动到模拟表盘内，然后从通道选择窗口中为其指定一个通道。因为我们已经知道数字表显示的值一定是速度值，所以我们可以关闭数字表标题显示，取消选择**Show caption** 以及 **Show frame**选项。为了能够在数字表背后看到模拟表盘，我们将数字表设置为完全透明。而且，RPM显示还带有一个小数点，小数点我们不敢兴趣，点击**Dec**按钮，减少数字表分辨率。



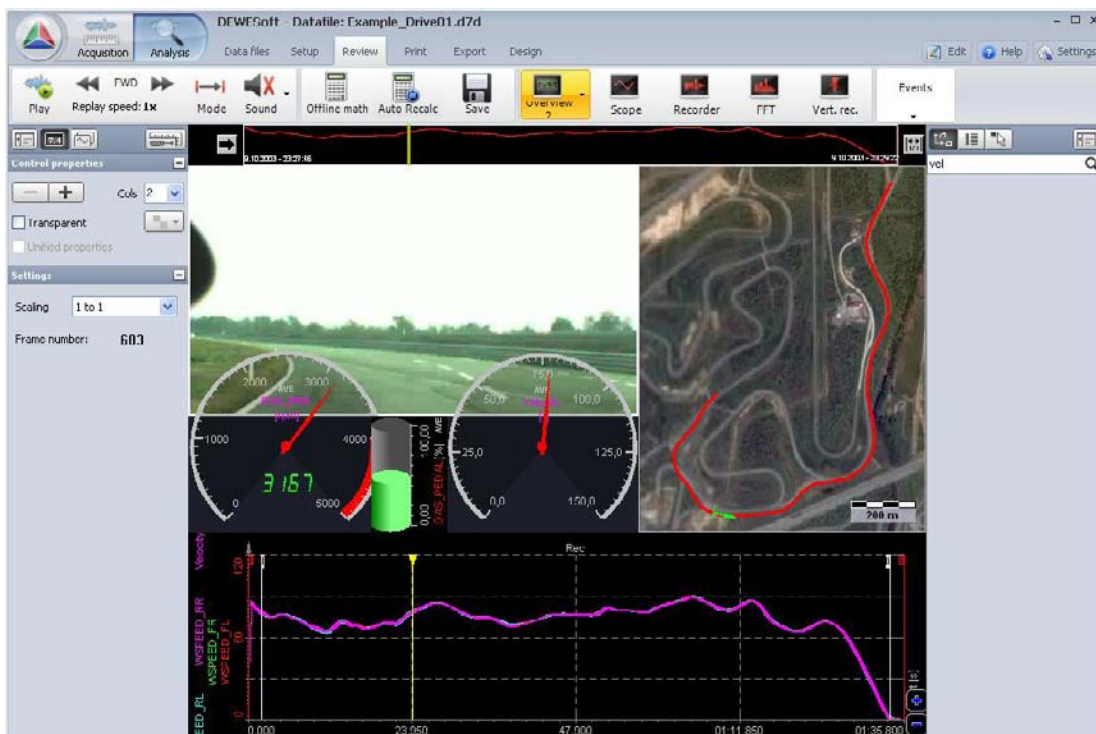
现在我们已经到了最有意思的部分了。因为我们有视频摄像机以及GPS，我们有两个额外的显示屏幕控制，这就是视频摄像机 **video camera** 和 **GPS** 地图 **map**。我们把这两个都加上。如果你在自己的系统上试这个例子，就不会在形式路径后面出现 **GPS** 鸟瞰图，不过我们可以在随后的 **GPS** 教程中学习如何加入图片。然后我们拖动鼠标，控制屏幕显示成自己最喜欢的形式即可。请注意，您可以将屏幕显示 **P** 设置成前景或者背景。因为我们在最后增加的视频屏幕显示的，所以它是作为前景出现的，在模拟表盘的前面。右键点击模拟表盘，选择 **Bring to front**，将表盘设置成前景。当然我们还可以通过设置可视透明度来增加更好的视觉效果，以便视频图片不会遮挡模拟表盘。



接下来我们增加一些诸如外部温度 **outside temperature**，使用的微型 **used satellites** 以及踏板位置 **gas pedal position** 等显示屏幕，完成整个显示屏幕设置工作。显示设置修改可以在采集和 分析两种模式下进行。在采集模式下，这些修改后的参数直接存储在 **setup** 文件中。如果我们在分析模式下，对显示设置做了修改，那么我们需要手动存储这些参数，选择 **Datasave** 标签时，同样我们可以选择 ，将改变后的设置 用于下一次测量。

为了能够真正使用这些修改后的设置，必须关闭 设计 **Design** 模式。现在我们可以记录中进行放大，或者移动黄色光

标选择不同的数据，更新数字表显示，或者回放数据文件。



1.3 存储策略

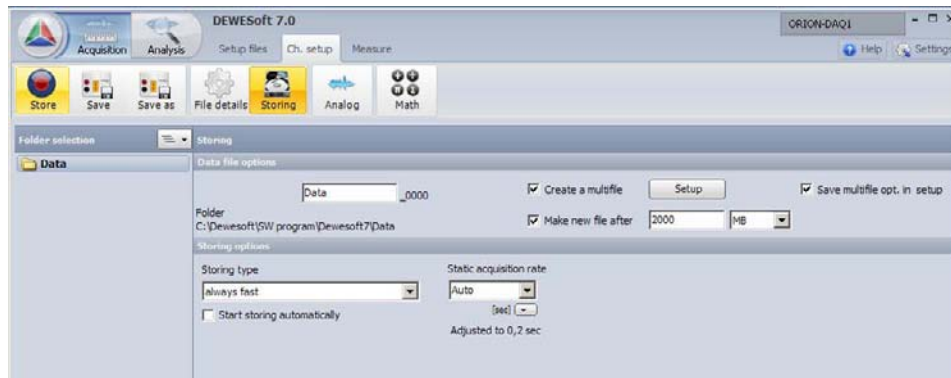
DEWESoft 软件提供了许多种存储数据的方式。所有的设置都在测量设置 屏幕上修

改。首先我们来看一下如何命名要存储的文件。我们可以在文件名域中输入不同名称来独立定义每个测量的文件名。存储文件的默认文件夹可以点击[文件详情File details](#)按钮来改变。



对于需要重复进行的测试，我们可以使用多重文件multifile。多重文件Multifile每次储存开始时自动指定一个新文件名。文件可以是连续命名(比如0001，0002，0003)或者用日期和时间命名。

我们还可以选定"**Make new file after**"复选框，当文件大小达到某个程度时，软件自动创建一个新的文件。创建新文件的评价标准既可以是文件大小也可以是 时间间隔，时间间隔单位可以是秒，分钟或者小时。在该例中，我们选择的是到达某个绝对时间后，软件自动创建新文件。这一点在采集非常长时间的数据时非常有用。如果我们选择每一个小时让人家自动创建新文件，那么软件会自动准确在每个整点时间(01:00, 02:00, 03:00...)创建一个新的文件。时间取自于绝对计算机时间(或者是其他更精确的时间源，如果有的话 – 在硬件设置中定义)。创建新文件时，两个文件之间不会发生任何数据的丢失。

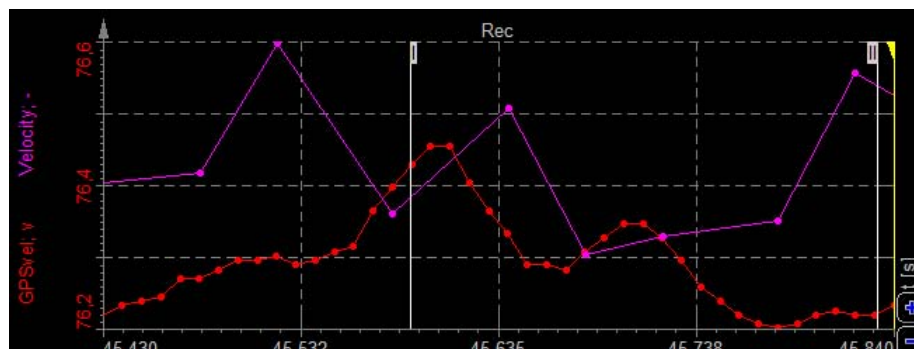


我们提供四种与基本采样率相关的储存策略：

- **永远快速always fast**: 数据将永远以高速存储，存储速度由动态采集速率DYNAMIC ACQUISITION RATE决定
- **永远慢速always slow**: 数据将永远以慢速(降低后的速度)存储，存储速度由静态/降低后的采样速率STATIC/REDUCED RATE决定
- **触发后快速fast on trigger**: 触发后，数据将以动态采样速率DYNAMIC ACQUISITION RATE存储
- **触发后快速，其他时间慢速fast on trigger, slow otherwise**: 触发后，数据将以动态采样速率DYNAMIC ACQUISITION RATE存储，在其它时间则以静态/降低后的采样速率STATIC/REDUCED RATE存储。

接下来我们一个一个来了解它们的存储策略。在了解各个存储策略之前，有必要先解释一下**DEWESoft**软件中两种不同形式的通道：同步通道和不同步通道。

同步通道 (比如模拟通道，计数器通道以及数字通道)是指这些通道的数据是准确按照完全一样的预定义采样速度来同时采集，采样速度由动态采样速率DYNAMIC ACQUISITION RATE定义。不同步通道是指预先不知道采样率的通道(PAD, CAN, GPS...)。这些通道总是按照他们的来源设备的采样速度来进行采集。下图所示是一个典型的例子，包括具有固定采样率的红色模拟通道以及紫色的GPS通道，我们可以看到该通道的采样率不是固定的。



只有同步通道受到动态采集速率DYNAMIC ACQUISITION RATE的影响，因此增加采样率，自然会增加被存储的数据量的大小。不同步通道不受动态采集速率DYNAMIC ACQUISITION RATE的影响，我们只需要注意一下动态采集速率DYNAMIC ACQUISITION RATE要快于不同步设备本身的采样率即可。

比如，动态采集速率设置到100 Hz通常对CAN bus接口来说就足够了，其它数据源比如PAD或者GPS的采样率甚至更低。如果我们只有不同步通道，采样率仅仅用来定义时间标记的精度(分辨率)。

1.3.1 永远快速

如果我们选择的是**永远快速always fast**，数据将在整个过程中都按照采样率进行存储。



我们来试一下。使用的硬件和基本教程使用的相同 – 只有一个声卡麦克风。

要开始存储，在主菜单上点击 Store 按钮。现在数据就按照设置好的采样率全速存储。



这是储存 按钮变成暂停 按钮，如果我们这时点击一下暂停 按钮，设备将暂时停止储存数据，不过数据的采集还是照样进行。这时储存按钮又变成了重新存储 ，点击一下该按钮，数据又开始重新存储。因此，我们得到了两段数据。



如果我们点击 分析 **Analyse**，我们就可以看到这两段数据之间是完全空白的 – 也就是说这里没有存储任何数据。



1.3.2 永远慢速

如果我们想以慢速采集数据，我们可以选择“永远慢速 **always slow**” 储存选项。这时数据会以间隔的形式储存，储存间隔通过静态/降低后的采样速率 **STATIC/REDUCED RATE** 来设定。我们这个例子中，该选项设为 **0,1秒**。这样可以减少大量用于存储

的磁盘空间。

即使储存选项设置为慢速，**DEWESoft** 软件采集数据还是按照全速进行采集的，计算这个时间间隔内的最小值，最大值，平均值以及RMS值，然后只储存这些值。



我们来储存一些数据，看一下这些数据在记录仪`recorder`上是如何显示的。下图是测量模式下全速采集时的样子。



我们在进入到分析`analyse`模式。因为我们的数据不再是全速采集时储存的，而仅仅是每隔**0,1秒**存下的数据，所以我们看到的只是原始信号的一个包络。我们可以在记录仪设置中切换平均值或者RMS值，来分别看看这两个参数的存储情况。



下图中显示的是记录仪记录的信号RMS值。我们可以通过原始信号的平均值，**RMS**值以及最小值`min`/最大值`max`来进行判断。比如我们发现有一个非常大的最大值，但是平均值几乎没有改变，我们就可以判断这个通道存在一个很短的尖峰。



1.3.3 触发后快速

如果数据中包括能够捕捉的事件，我们可以选择触发时快速**fast on trigger**存储。触发事件在软件中定义，然后**DEWESoft** 软件将等待事件的到来，只快速存储真正感兴趣的数据部分。

这个功能可以通过选择“触发后快速**fast on trigger**” 储存选项。选取该选项后，会出现一个新的标签，叫做触发**Trigger** 标 签，在该标签上我们可以设置触发条件和策略策略。

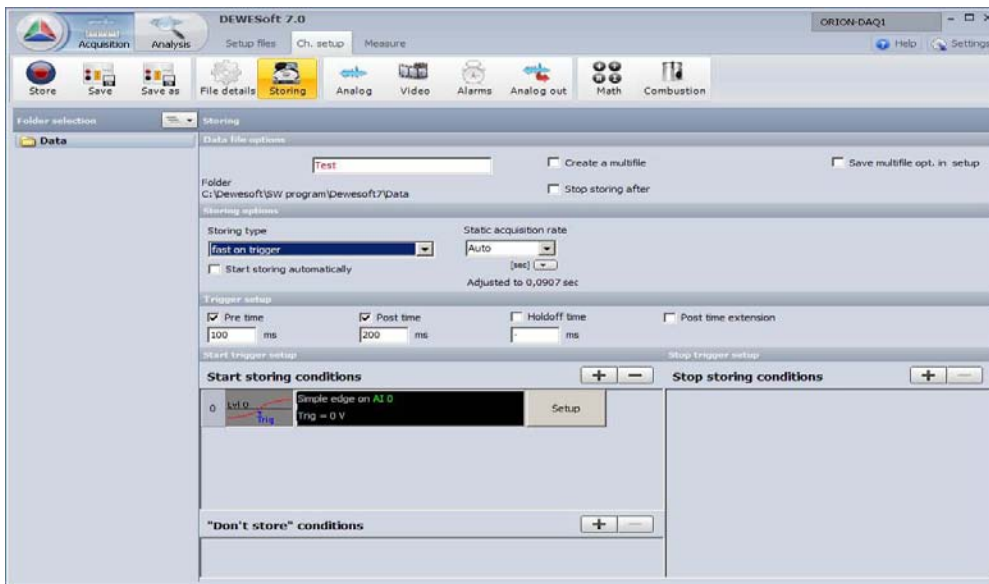
首先，我们先定义预触发时间，后触发时间以及保持时间。

预触发时间Pre time 是指触发事件发生之前需要存储数据的时间。我们定义100 ms的预触发。这表示**DEWESoft** 软件将 在缓冲器中保存100 ms的数据，直到触发事件真正发生，这时候才将这段数据储存到文件中。

后触发时间Post time 是指整个事件结束之后还需存储数据的时间。如果没有定义后触发时间，**DEWESoft** 将一直持续存储 数据，直到我们手动停止，或者停止存储条件发生时才停止储存。因为我们要捕捉的是触发瞬间的数据，我们将后触发时间设为

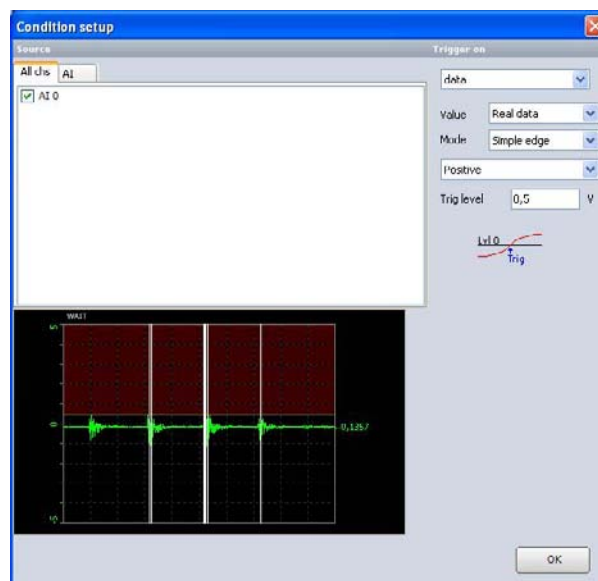
200 ms，所以在整个触发事件过程中，除了触发事件本身的数据之外，我们还捕捉了300 ms的时间前后的数

据。 现在需要我们定义一下开始存储的触发条件。加上一个出发条件，然后点击设置，开始定义。

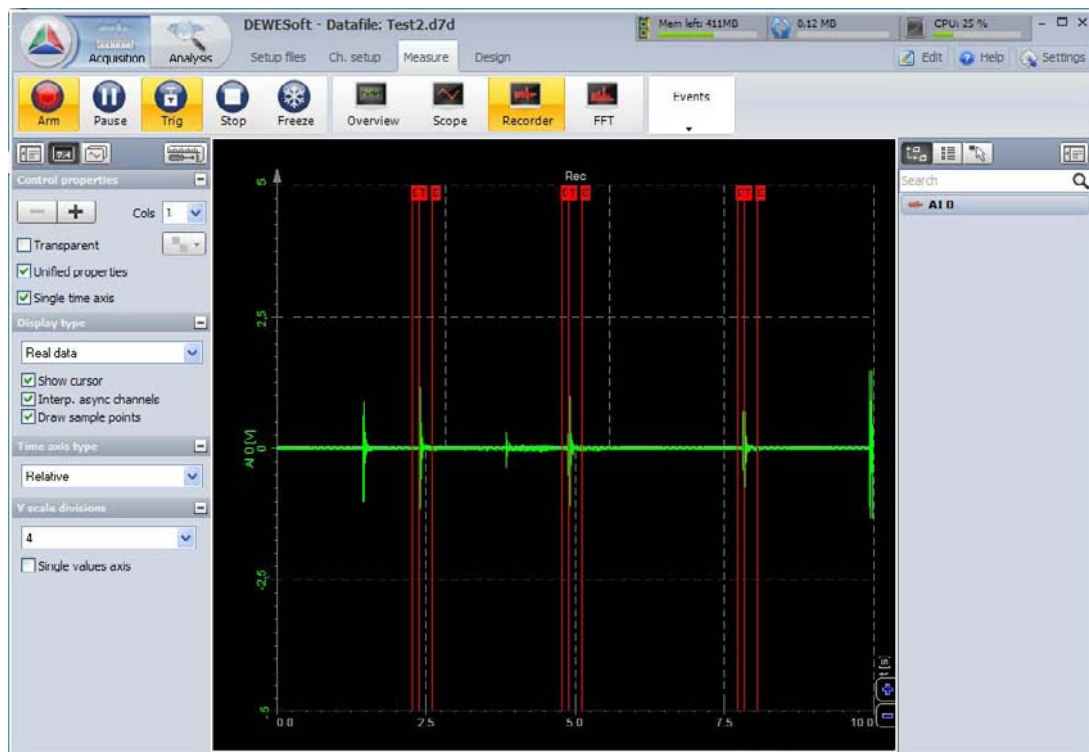


在触发设置窗口，我们通道信号作为触发。我们可以选择几个通道来进行触发，不过现在我们只有一个通道，所以只能选择这个通道。接下来定义出发条件。我们可以通过时间数据time data，时间time 或者FFT频谱来进行触发。时间触发包括 对实际数据的沿触发edge，滤波后的沿触发 *filtered edge*，窗触发window，脉冲宽度触发*pulswidth*... 平均值*average*触发或者 *RMS*值触发。对于我们例子中的这种简单应用，我们只需选择简单的沿触发，触发电平设为1。这表示当信号电平值超过1

V 的时候，就会触发系统开始存储。我们还可以在左边窗口的示波器上实验触发的性能。



现在我们来做一些测量。我只是敲击了一下麦克风，产生一个触发。我们可以从记录仪*recorder*上看到第一次触发信号不够高，因此接下来我敲得更重一点。这下没有问题，我们可以看到触发开始，触发事件整个过程以及储存事件结束。注意一下，储存Store 按钮变成 警告 Arm 按钮，还增加了一个触发 Trg 按钮。这个是手动触发器，在没有事件触发的情况下，可以点击它手动触发。



接下来我们来浏览**review**一下储存的数据。我们可以看到整个过程中只存储了触发的事件，其它时间段内的都是空白的，即没有数据。注意在数据预览窗中，出现了一个名叫“触发模式trigger mode”的新按钮。这个按钮让我们可以快速预览触发事件，而且不用放大数据。点击该按钮，第一个触发事件自动放大。触发模式按钮变成**箭头**按钮，使用该按钮可以在各个事件之间切换。



如果我们点击这两个**箭头**按钮，记录仪会一个一个显示这些触发事件。在数据预览窗口中，我们可以看到现在选择的触发事件高亮显示。要离开触发模式，只需右键点击记录仪，将整个数据缩小到布满整个区域即可。



注意，在存储开始的时候会出现一个小的尖峰。这个尖峰利用触发是无法存储的，但是这个值有时在没有触发的情况下了解该区域的值还是比较有用的。

1.3.4 触发后快速，其它时间慢速

为了使用两种不同的速度采集数据，需要我们使用另外一个不同的存储策略，"触发后快速，其它时间慢速 **fast on trigger, slow otherwise**". 所有的触发设置都和触发后快速 **fast on trigger** 的设置完全一样，但是当我们用这种存储策略采集同一段数据的时候，重新载入，就会发现如下图所示，我们还得到了一些在没有触发事件发生时储存采样速率较低的数据。

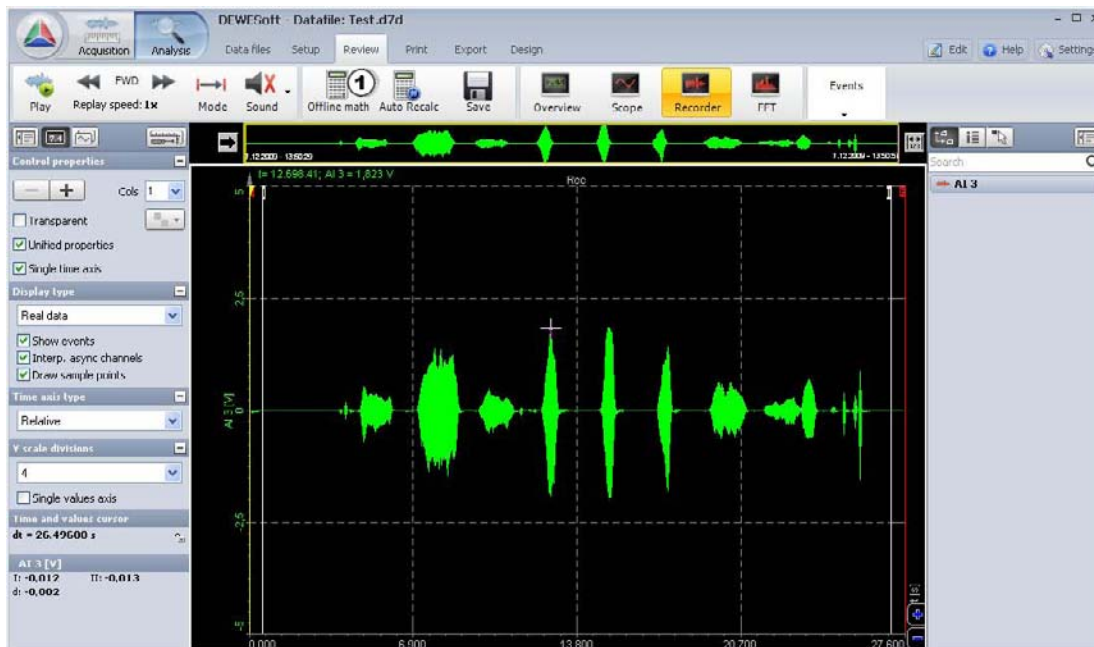


如果这时我们放大数据，我们就可以看到触发发生前存储的采样率降低之后的数据，在这些部分我们只能看到a最大值和最小值，而在触发事件区域内，我们则能看到全速采集的数据。



1.4 后处理

在DEWESoft 7 最大的亮点就是后处理, All t所有的数学功能既可以在采集状态下使用, 也可以针对已经存储的数据进行处理. 让我们采集一些麦克风的模拟信号。



现在点击 **"Offline math"**按钮 ①, 这样就可以打开数学通道的设置面板, 在这里你可以象在采集状态下一样添加数学通道.

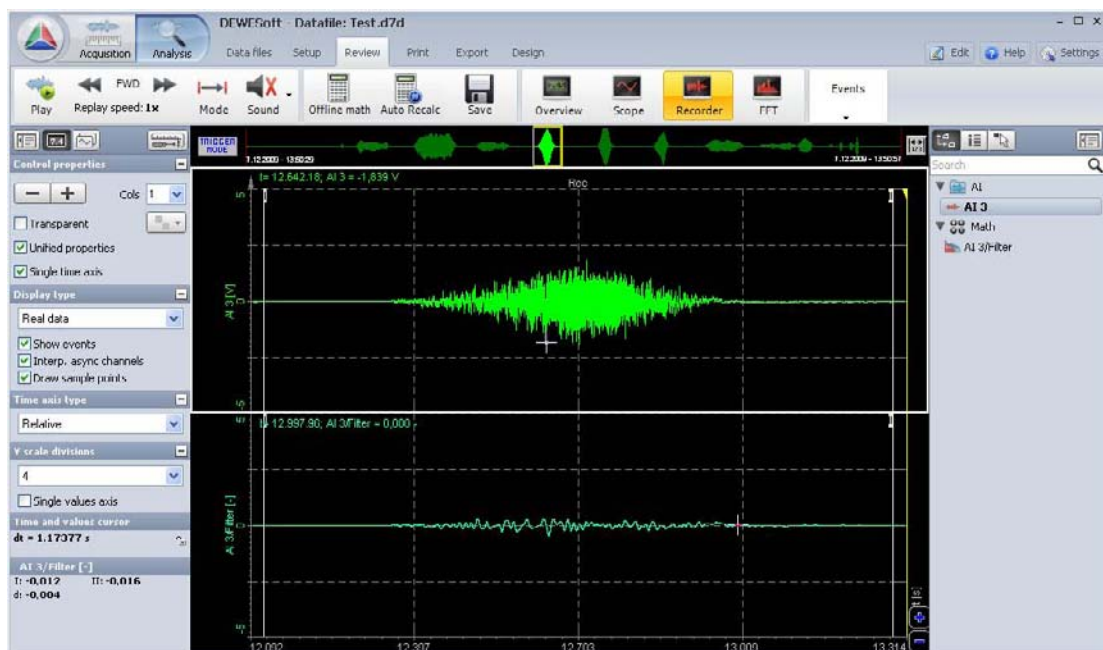


当我们回到 **Review**状态时, **Offline math** 按钮 变成了**Recalculate**, 我们点击 “+” 按钮增加一个记录仪. 在记录仪里增加一个AI3/Filter (刚在数学通道设置好的) 滤波通道时, 我们在记录仪里什么也没看到. 这是因为我们需要手动点击 **Recalculation** 按钮进行运算处理, 这样做的好处是你定义你想要对哪个区域的数据进行处理.



设想一下你有一个巨大的文件. 在多个通道添加了滤波器, 进行运算处理, 然后去喝杯咖啡, 回来看看到底选择哪种滤波效果比较好.

DEWESoft 提供局部放大, 和局部运算功能, 当按下 Recalculate按钮后, 滤波器开始工作, 按钮转换回 **Offline math**.如果结果不够完美, 我们可以返回再重新设定 (点击**Setup** 标签或 **Offline math**) 数学通道.



让我们把截止频率从 100 Hz降到 2 kHz (在设置 **Setup**界面的 IIR 滤波). 做完这些后, 我们看到数学通道的状态从 **Calculated**变 换到 **Offline**, 在线的值Value显示为 "Waiting calculation".



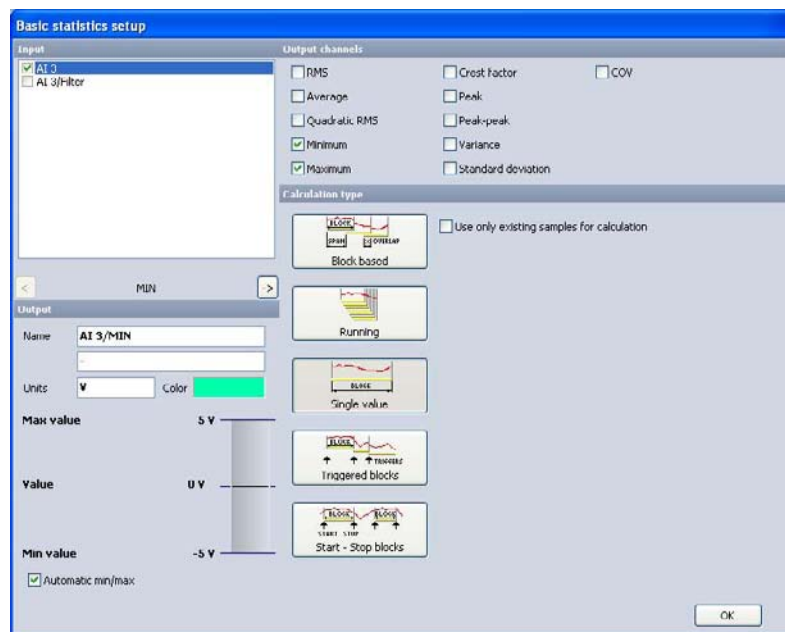
回到预览Review界面点击 **Recalculate**. 现在开始运算并替代前面的运算结果, 当我们对文件的某个部分比较满意的时候, 我们可以把着部分放大。这是我们只会看到这个部分. 按钮 离线数学Offline math 又变为重新运算 **Recalculate**. 点击重新运算 **Recalculate** 后, 我们可以喝杯咖啡, 我们就可以看到一个完美的结果了。



我们可以增加各种类型的数学通道, 不限于公式formulas, 滤波器 filters 和统计 statistics, 也可以是功率 power, 阶次分析order tracking, 扭振torsional vibrations, 燃烧分析combustion, 这些也都可以在线实现

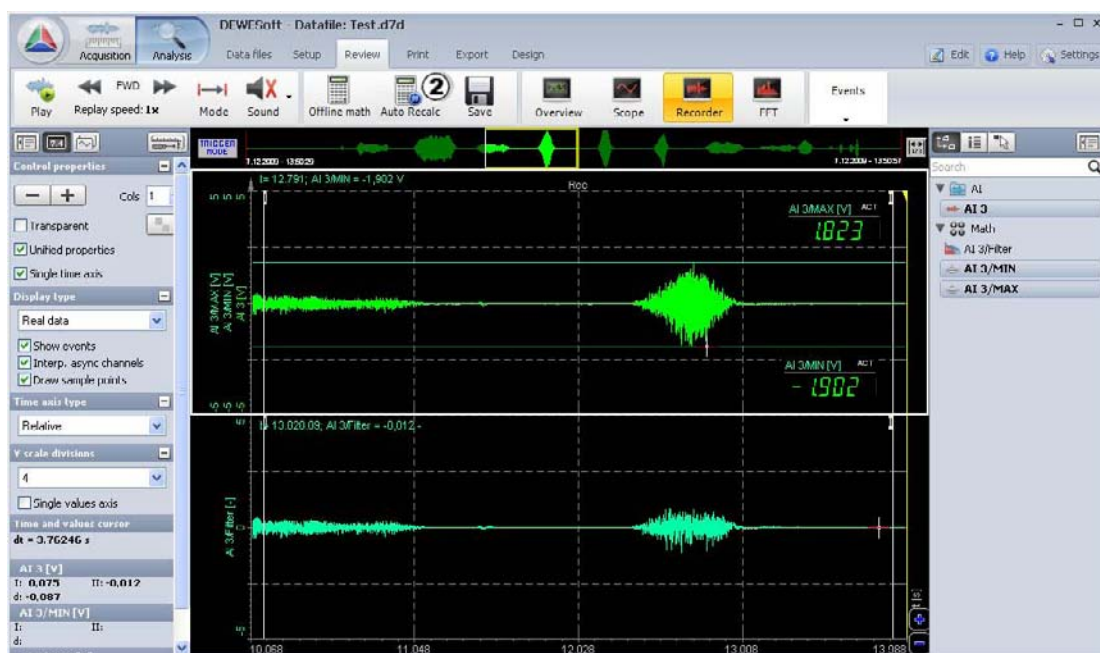
让我们来看离线数学的另一个应用

回到数学通道的设置, 添加一个数学通道, 这次选择统计 **Basic statistic**. 到设置界面选择 **Single value minimum** 和 **maximum**.



现在我们又有了俩个数学通道: **AI3/MIN** and **AI3/MAX**.

回到**Review** 界面, 把数学通道添加到记录仪 recorder. 为了联系显示界面的设置, 我们可以增加一些数码管 meters. 当放大并点击 **Recalculate** 按钮时, 我们可以得到选定区域的最大最小值.



另外一个非常好的选项就是 **Auto recalc**②. 与每次设定完数学通道都需要点击 **Recalculate** 按钮不同的是, **Auto Recalc** 可以自动完成运算.

最后一步是把这些通道都存储到数据文件中, 当我们把整个数据缩放到正常状态并对这些通道进行运算后, 我们可以点击 **Save** 按钮来存储这些数学通道的设置, 跟其他模拟通道一样.

2 测量操作教程

测量操作教程提供了基本技术测量概览。主要是帮助用户了解针对特定的测试任务如何正确选择传感器和放大器。

	电压和电流	电压和电流 -测量电网电压以及消耗/产生的电流
	具有电压输出的传感器	具有电压/电流模拟或者数字输出的传感器 – 我们需要测量传感器输出并且将其转换为实际的测量单位
	温度	利用不同类型的温度传感器，比如热电偶，热电阻，热敏电阻...进行温度测量
	振动	振动测量用来测量指定物体表面的震动情况
	应变传感器	应变片测量：测量原理以及如何测量材料的应变
	计数器	计数器通道用来测量计数，以及进行频率测量
	频率测量	利用计数器，模拟频率通道和PAD模块测量频率，比较其中的差异
	视频采集	视频的采集为分析其他采集的数据提供了额外的重要信息
	CAN bus 测量	“聆听”以及获取汽车和其他车辆的 CAN bus 数据
	GPS 采集	确定地理位置，精确计算运行速度，远程同步多个采集系统

2.1 电压和电流

电压和电流的测量非常普遍。

我们可以将电压测量分成几个不同的部分：高电压电网测量 (测量范围达到千伏 kV)，这种应用需要电压传感器，直接低电压测量 (120/230 V) 以及低电测量(最高到50 V)。高电压转换传感器将千伏级电压信号转换成可测量的电压范围 – 最高可到1kV 以上。Dewetron公司能够提供许多不同的信号调理放大器，信号范围可达1kV以上，所以，如果您购买了Dewetron公司的信号调理放大器，基本上所有的电压测量都没有问题。

电流测量与电压测量类似 – 我们可以利用Rogowsky 电流线圈传感器测量大电流，或者利用电流钳测量小电流，通常电流钳里面用到了一个分流电阻。

Rogowsky 电流线圈传感器可以用来测量交流 AC 电流。它直接测量的是电流的导数，因此必须用到积分电路或者是软件滤波模块。电流钳是根据霍尔效应原理制作的，它的输出电压与电流成正比。所有的理论中都会在输出时产生相位差。

分流电阻在测量小电流时非常有用。它的原理非常简单 – 已知大小的电阻，流过整个测量链的电流是不变的，电压与电流成正比。我们将在软件中校准分流电阻时进一步详细解释这一原理。

在所有的测试中，强烈推荐使用带隔离的信号调理放大器。甚至是被测电压对于未隔离的AD板输入不是很高的时候也是如此，因为这个不高的电压对地来讲可能会有很高的电势，这种高电势可能就会烧坏AD卡。举个例子说明，当用到分流电阻测量电压时，这种分流电阻是一种非常高精度的高电流电阻，我们要测量的是它的电压降。即使我们测量的是仅有几伏的电压输出，与地的电势可能跟高压电网的电压一样高。这样高的电势 – 如果直接接到AD卡上 – 必然会烧坏AD卡。

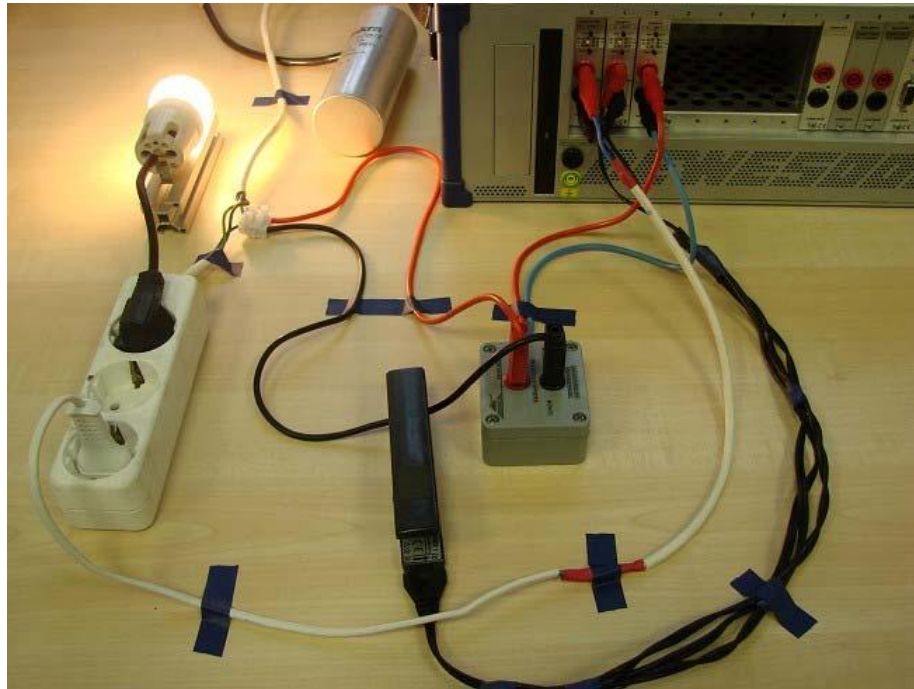
Dewetron 信号调理放大器

我们现在看一个典型的“低”电压和小电流的测量的例子。我们将测量供电线的电压和一支灯泡40 W消耗的电流。在这

种情况下，直接用Dewetron 的电压信号调理放大器测量电压足够了。我们用到了DAQP-DMM 电压信号调理模块，它具有2kV的隔离电压。我们当然还可以在更高带宽的情况下用DAQP-HV模块 (比如高频或者瞬态测量)，或者使用PQL print (4 x 电压和 4 x 电流)。

电流测量有两种方法，对应两种原理。我们将使用电流钳，因为这种电流钳可以打开，安装非常方便。第二种方法是分流电阻测量，这种方式需要切断电线，再焊接上串联的分流电阻。我们必须非常小心，注意回路电流不能超过分流电阻能够承载的最大电流；否则我们只能测量到分流电阻被烧掉时产生的火花了。不过现在让我们把精力集中到电压和电流测量上面来。

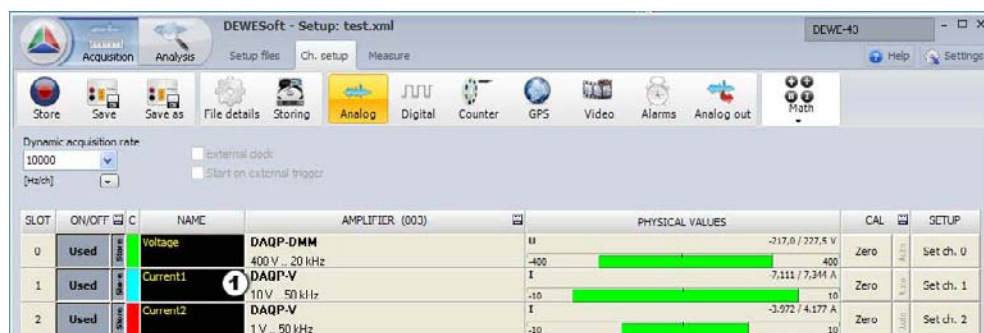
电流转换器输出的是电压信号，所以我们可以用DAQP-V，DAQP-V-A 或者 MDAQ-V 模块直接测量。



2.1.1 通道设置

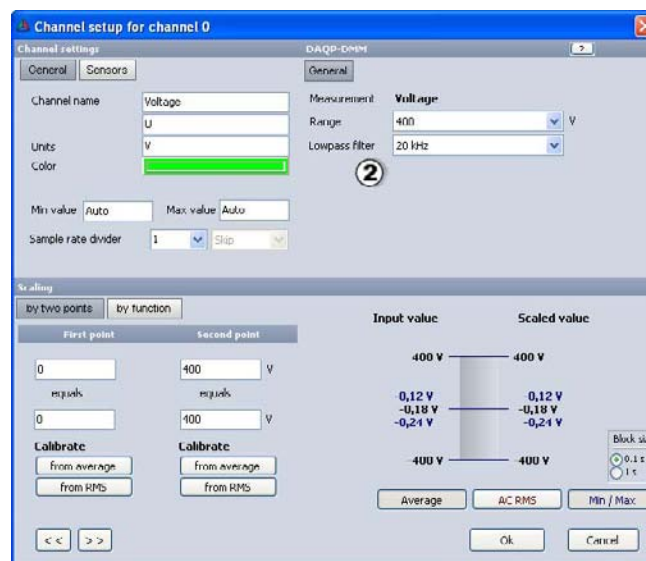
所需硬件	任意 AD 卡, DAQP-DMM 或者 DAQP-HV, DAQP-V 或者 MDAQ-V 模块
所需软件	除了 LT 之外的任意 license
设置采样频率	至少 1 kHz

让我们先看看如何根据上面的表格来设置 **DEWESoft**。我们有三块 Dewetron 可编程模块 – 一块用于高电压测量的 DAQP-DMM 模块 (也可以用新型号 DAQP-HV 模块) 以及两块低电压模块 DAQP-V①。



因为的 DAQP-DMM 模块的输入电压最大为 +1000 V，我们需要测量的电压可以直接接入模块，无需任何附加的传感器。我们将电压量程设为 400 V，为什么这么设呢，因为 $230 \text{ V}_{\text{rms}} \sqrt{2} = 325 \text{ V peak}$ 。

我们还需要特别注意设置低通滤波器 **lowpass filter** ②。如果低通滤波器设置得比采样率要低，滤波器将切掉本来处在测量频率范围内的信号。有时候需要这样设置，但是通常情况下错误设置过低的低通滤波器，测量结果将变得无效。



现在我们来校准电流。下图显示的是电流钳传感器。



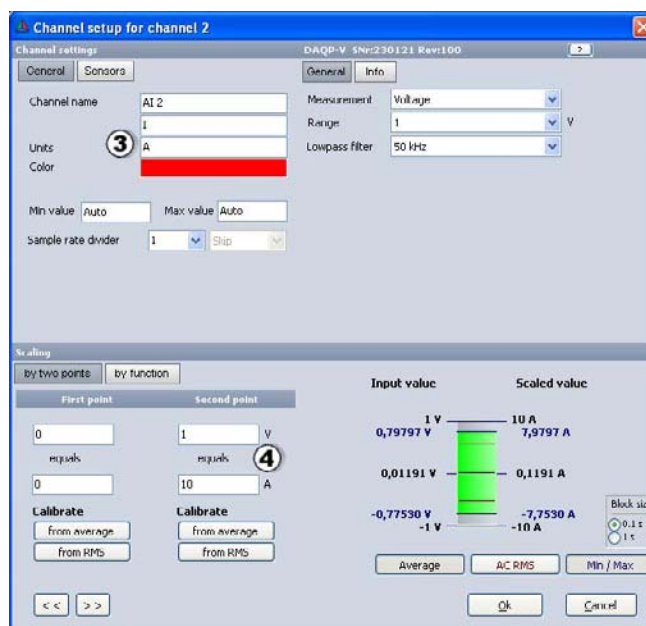
因为我们只有一个白炽灯泡，**10 A 量程**足够了。非常重要的一点是测量量程是根据可能的最大信号来选择的。如果选择过大的量程，电流钳的准确度将会降低，直接影响正确读取信号的精确度。在我们的试验中，传感器的灵敏度是**1mV/mA**，这表示输入**1 mA**的信号，输出是**1mV**。在通道设置 **channel setup** 中，我们测量的是电流**I**，输入电流单位**unit A** ③，比例因子 **scaling factor** 不变，还是**1**。因此这种设置非常简单。现在来看看如何设置分流电阻的量度。

通常我们可以非常简单的确定分流电阻的量度 – 接下来我们会简单回顾一下，不过首先看看如何设置分流电阻量度的理论知识。我们的分流电阻是 **0.1 Ohm**。因为分流电阻是串联在电路中，所以整个电路中的电流都是一样的。我们需要测量分流电阻上面的电压降。从欧姆定律，我们可以通过下式计算出这个电压降：

$$U = R \cdot I = 0.1 I$$

所以，我们可以得到**1 A** 电流输出 **0.1 V** 电压信号，**10 A** 电流输出 **1 V** 电压信号。

现在进入到比例因子 **scaling factor**一栏中。输入测量单位**A** ③，然后到**第二点second point**。输入 **1 V** 等于**10 A**④。这就设置完了。



我们还有一个额外的选项 需要用来定义分流电阻类型。对于能够输出电流信号的传感器，有时候电流是唯一可能传递机制，而真正的测量值则是不同的（类似于压力），这种情况下，我们需要组合应用需**combine scaling factors**量度设定，首先将真正的测量值（压力）转变成电流，然后再将电流转变成电压。

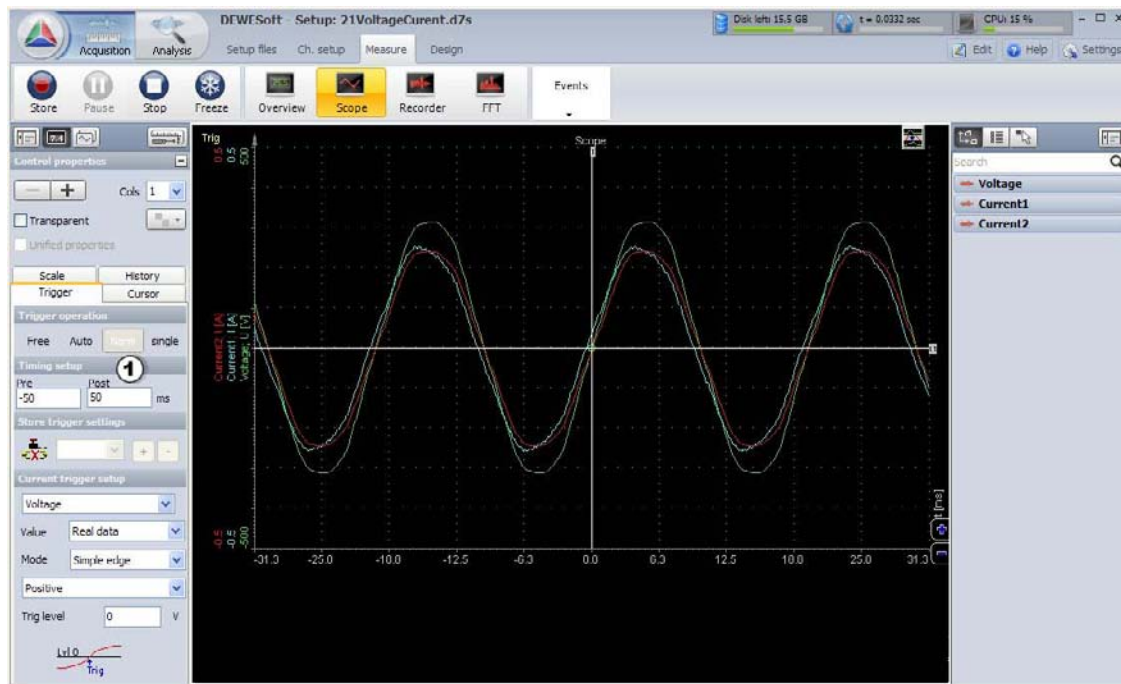
因此可以简单的将分流电阻作为信号调理放大器的一部分来进行设置。在信号调理放大器⑤部分，我们可以定义分流电阻类型**shunt type**。标准分流电阻1 (Shunt 1)是精密的50 Ohm 电阻，用来测量4÷20 mA或者0÷20 mA的电流回路。分流电阻2 (Shunt 2) 是0.1 Ohm电阻，用来测量大于5 Amps的电流。

现在我们在下一章节教程→[电压型/电流型/数字输出型传感器](#) **Sensors with voltage/current/digital outputs** 中来看看如何使用分流电阻。

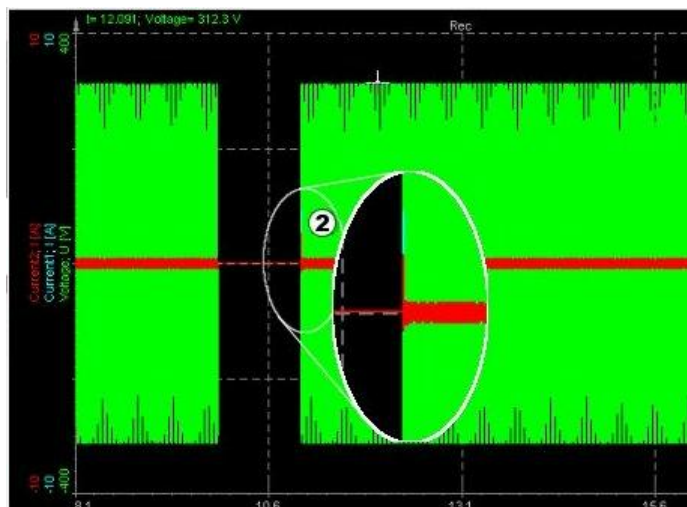
2.1.2 测量

现在我们来做一个真正的测量。我们三个通道 - 一个是高电压通道，另外两个是电流通道。观察波形的最好办法是利用示波器scope。当我们第一次打开示波器时，通常几乎看不到任何东西，这是因为示波器是在自由运行Free run模式下。我们必须通过触发Trigger→普通触发Norm trigger功能，定义触发源和触发电平来“保持住hold”当前测量。对当前的测量，保持默认触发条件不变即可 – 即触发源选择电压通道，触发电平为0。现在我们可以看到用分流电阻测量电流和用电流钳测量电流之间的差别。白炽灯的电阻负载中流动的电流值最终确切显示为电压曲线。这在示波器上通过分流电阻测量得到的红色曲线可以清楚观察到，而从电流钳测量得到的蓝色曲线则由于相位角变化产生了失真。

从电流钳得到的失真信号将随后得到错误的电功率。继续阅读随后有关描述功率模块的章节，您将发现我们可以利用定义传感器传递曲线的方法来进行补偿。

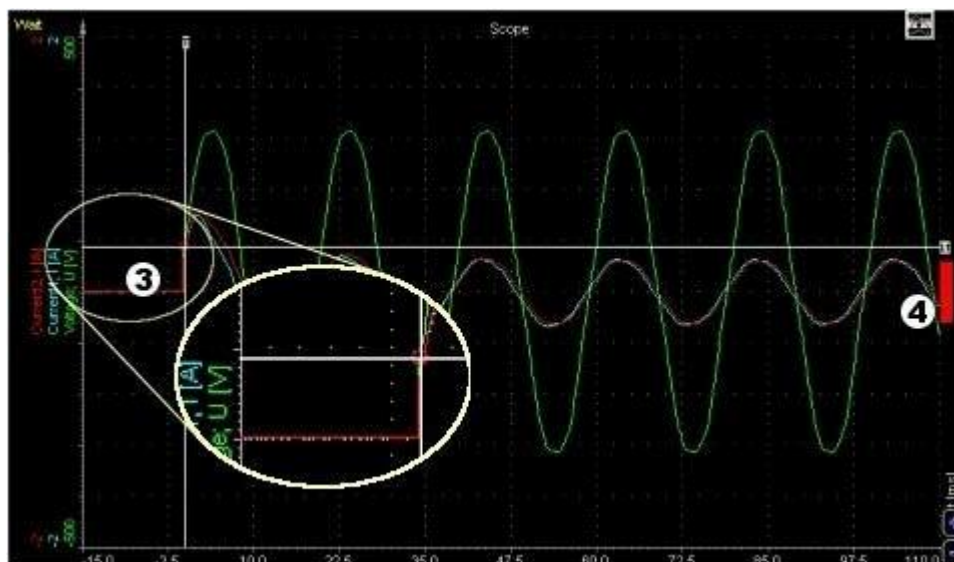


现在看一下记录仪recorder。如果我们拔掉灯泡电源，随即马上插回去，我们可以在记录仪中(红色曲线)发现在一段短暂的时间内电流值远高于正常的电流消耗②。这是因为在低温下白炽灯的电阻比高温下的电阻要大的缘故。



我们来试一下如何更好的捕捉住这一事件。回到示波器scope界面，再回到触发设置**trigger setup**。我们不能真正通过电压来触发，因为电压电平大小从启动到正常运行整个过程中都不会发生变化。所以我们改为电流触发。首先定义触发源为电流，定义超过即可触发的触发电平(我们的例子中为**0.36 A**③)。当示波器scope未触发时，在窗口的右侧有一个红色的棒显示当前信号的电流值，因此我们可以进一步根据正常的信号值优化触发电平大小。我们还可以设定**自动触发Auto trigger**模式。如果经过**几秒钟**还没有触发采集信号，屏幕数据将显示没有触发。

不过让我们回到**普通触发模式Norm mode** – 当设置好另外一个触发条件时，示波器scope会显示当前电流大小，直到新的触发事件发生，示波器才被触发开始保持住信号波形。



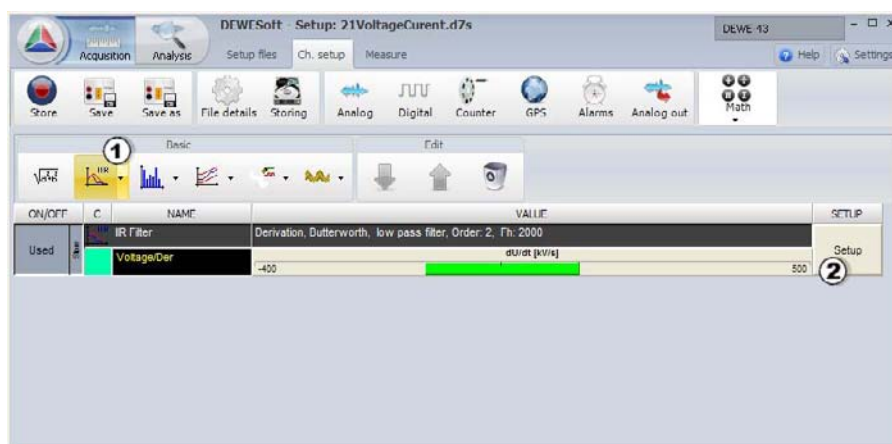
但是如果我们想通过触发来采集电涌，该怎么办呢？电涌和正常信号不同，电涌信号的降低最多也就是几个百分比，因此非常不明显，也不可能通过定义某一个电平值来采集所有的电涌事件。

2.1.3 电涌触发Glitch triggering

电涌触发最好的方式是通过触发源求导数。电涌在导数信号(dU/dt)中看起来更加直观，因为导数信号在正弦波的差异表现最灵敏。

对于纯粹的220 V, 50 Hz 正弦波来讲，基本的求导公式是最大值 $315 \cdot 2 \pi \cdot 50 = 100 \text{ kV/s}$ 。电涌量值变得更大 - 从300 kV/s到更高。因此更加容易用它来作为触发。

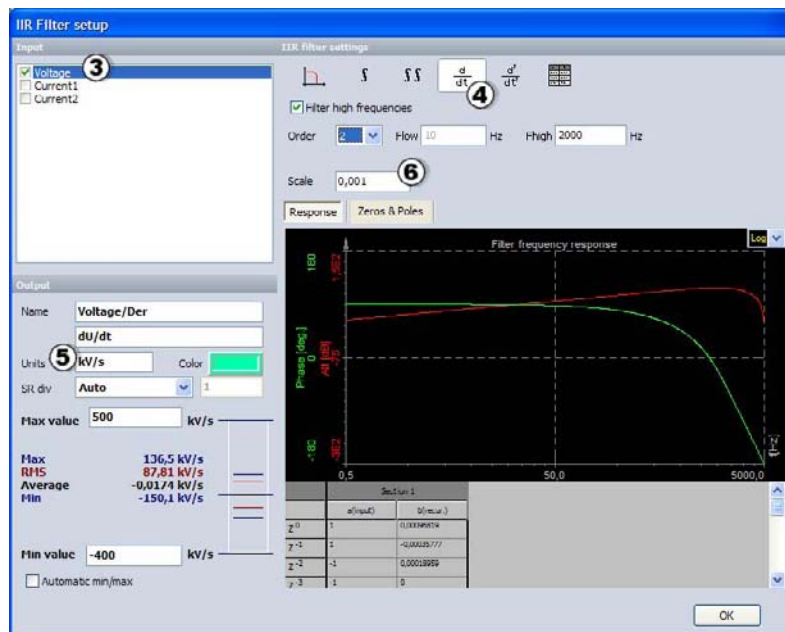
回到设置 setup 项，选择数学 Math 标签，定义 IIR 滤波器。如果从图标上看不到 IIR 滤波器图标，可能是你以前使用过 FIR 或者 FFT 滤波器。这是可以从下拉菜单中选取 IIR 滤波器。



现在按下 **Setup** 按钮②来定义滤波器。选择电压 **Voltage** ③作为输入 *input*，然后选择 **d/dt** 图标④，计算导数，然后定义

单位 **units** 为 **kV/s**，再定义好量度因子 **scaling factor**。通常，如果输入单位是 **V**，输出单位应该是 **V/s**。因为这个量度单位非常大，所以我们最好把量度单位设置为 **kV/s** ⑤，同时将量度因子定义为 **0.001** ⑥。

高频滤波器 **high frequency filter** 的设置不是强制的，但是它可以用来平滑信号。如果没有使用滤波器，电流的求导计算和先前的例子有些不一样。通常会增加更多噪声，因此我们需要增加一个低通 **low pass** 滤波器来使输出信号更加平滑。



如果此时我们再来在示波器 **scope** 中观察该信号的结果，原先是一个纯粹的正弦波的信号，现在看起来已经不是标准的正弦信号了。纯粹的正弦波经过求导后，正常情况下仍然是正弦波，只不过相位相差 90° 。但是在这个地方不是这样的。



现在，让我们用滤波后的数据重新用示波器显示。不知道大家是否注意到第一页的图，图中显示一个用在电机上的大电容器。事情就是这么巧合，我们可以用它来在这里产生电压电涌 **voltage glitches**。当电容器充电后，它从电网中得到了许多能量，因此在电容器两端会有一个电压降。我只是把把电容器连接到平常的插座里，然后给插座接上电，瞬间即可给它充电。

如果您需要自己来做这个实验，必须注意两个非常重要的方面：第一，将电容器连接到电源的时间要非常短，否则电容器就会烧坏。第二，永远记住在金属上放电。因为电容器瞬时放电电能非常大。在自己身上放电可不是什么好主意。

警告

当充电完成后，触摸电容器的两极，签完不能触摸电压线!

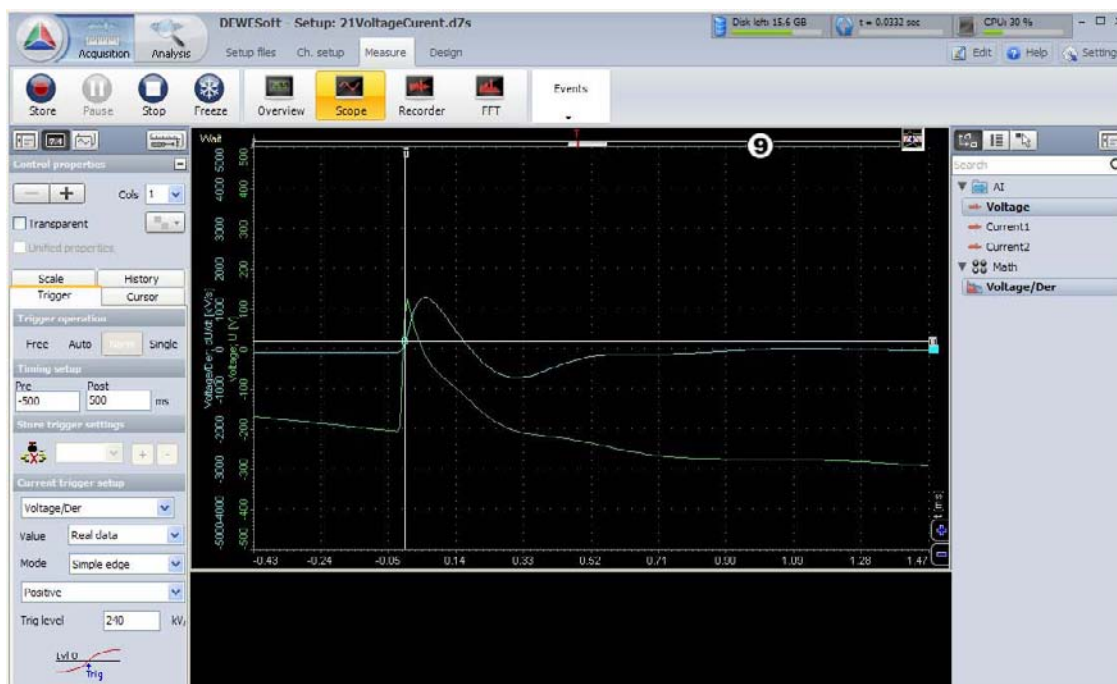


那么，我们先设置 **setup** 示波器 **scope** – 触发 **trigger** 参数改成 **Voltage/Der** ⑦通道，触发电平 **trigger level** 设置为 **240 kV/s** ⑧。现在我们创建几个电涌，此时我们可以清楚看见示波器如何捕捉这些电涌事件的。



接下来需要按逻辑处理的事就是放大 **zoom in** 窗口，用更高的分辨率来看看电涌的情况。如果我们按放大按钮，通常就会丢失当前事件。不过我们有一种专门的示波器放大模式，在这种模式下放大时，不会丢失任何事件。在示波器的右上角，有一个放大 **zoom** 按钮⑨，该按钮可以弹出一个额外的放大窗口。

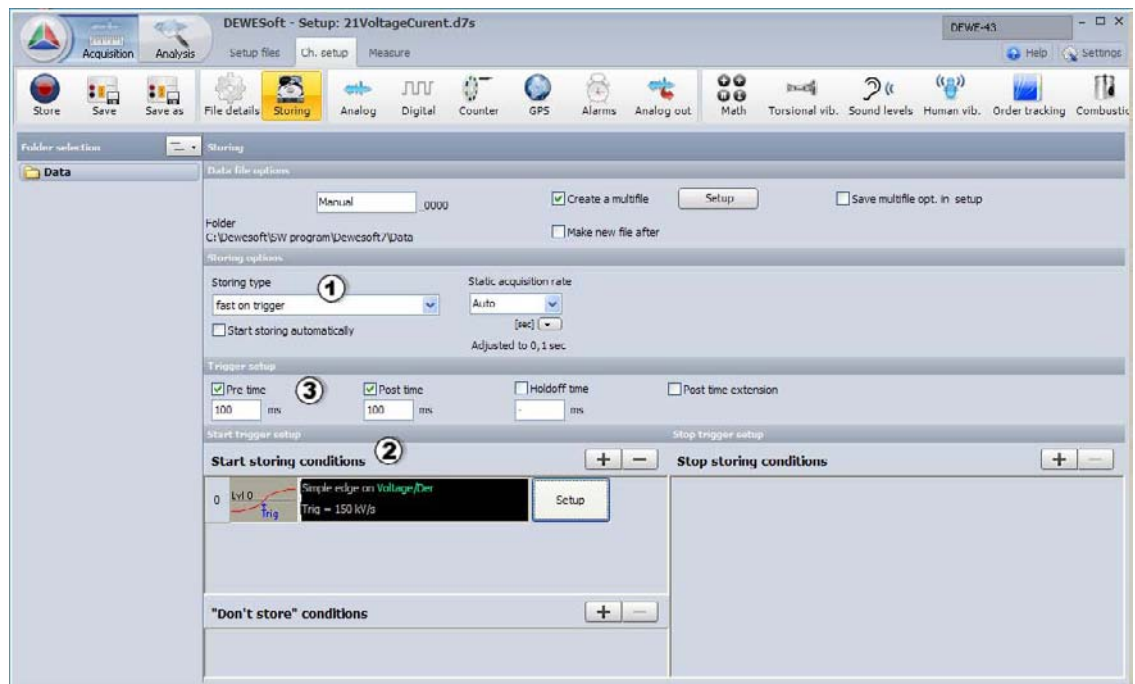
在窗口的上面，我们能在事件中看到当前位置的棒状显示，还有放大`zoom in`/缩小`out`按钮，所以我们可以对特定区域进行放大。还可以在示波器窗口上往左，往右移动光标，来观察触发位置附近的信号情况。



现在让我们来看看如何存储这些数据。Now let's see how to *store* these data.

2.1.4 触发存储

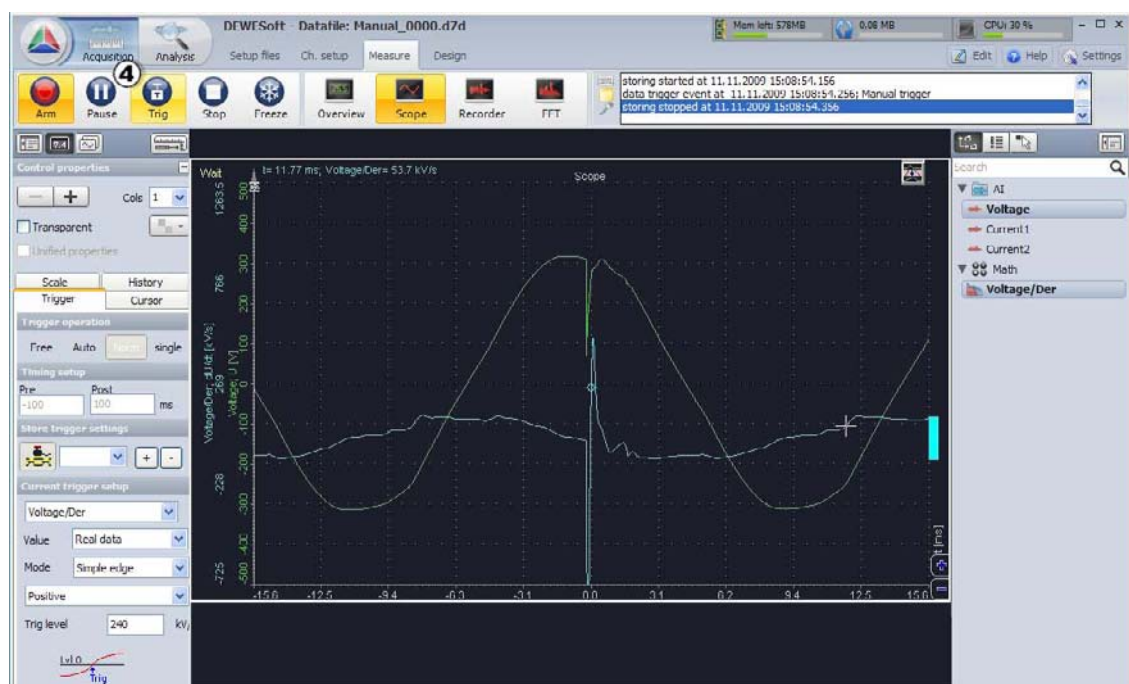
对于电涌的监测，需要较高的采样率①。这样高的采样率意味着磁盘存储空间的浪费。如果我们永远以快速存储方式存储数据，我们可能会最终得到几个G的数据，几种有用的数据可能只有几秒钟。为了解决这个问题，最好是当事件发生的时候才进行存储。要做到这一点，我们应该选择“触发后快速存储Fast on trigger”②的触发存储方式。我们需要定义开始存储条件start storing condition③，这个条件就是触发的电平level trigger，当电压求导后的大小大于150 kV/s后触发采集。我们还需要定义预触发时间pre time和后存储时间post time④。预触发时间是指在触发开始前的一段存储数据的时间，而后存储时间则是事件结束后还需要存储数据的时间。如果没有定义后存储时间，数据将会持续存储下去，直到手动停止，或者是达到了停止存储条件stop storing condition（在我们的例子中，没有定义任何停止存储条件）。对于电涌监测，确实有必要采集事件发生前和事件结束后的一段时间内的数据。



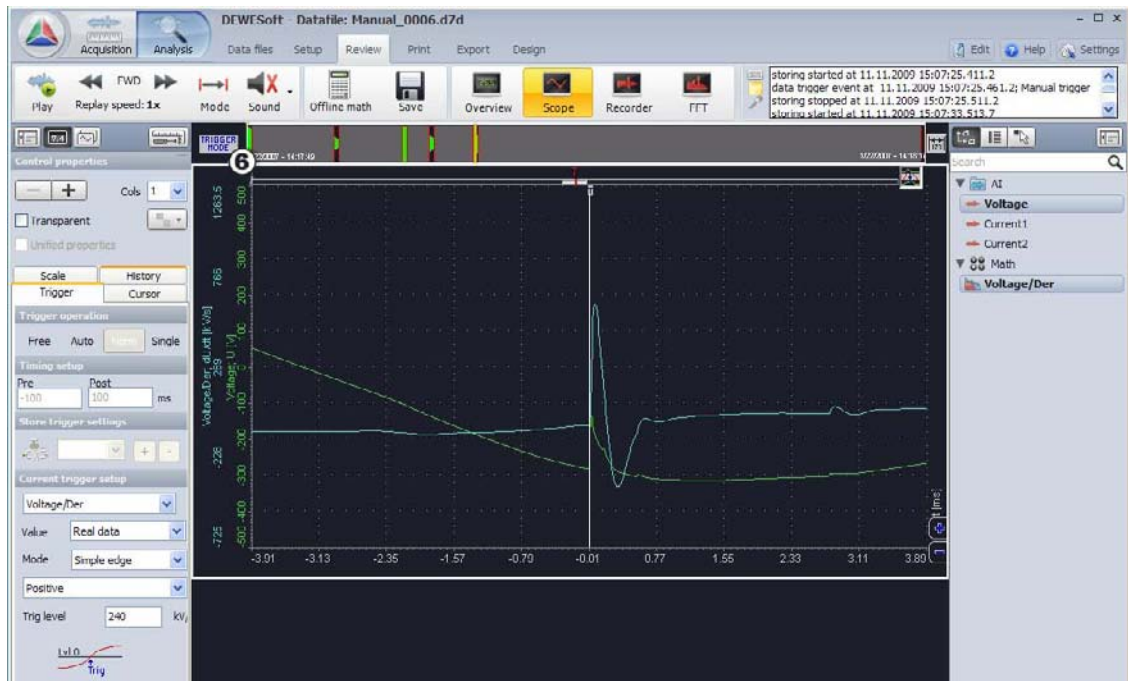
这是完全手动设置触发的情况。我们还可以通过一个比较简单的方法来设置触发条件，在示波器scope的触发部分trigger section点击锁定触发按钮**Lock trigger** 。这种方式直接采用当前的预触发时间pre time和后存储时间post time，触发源和触发电平。

两种方法实际上都设置了触发标准。现在来试试存储数据。当按下存储**Store**按钮后，会在菜单栏上出现一个额外的触发**Trig** ⑤按钮，出现这个按钮，就表明现在是在等待触发，使用的是触发方式存储。当然，我们也可以点击该按钮，进行手动触发。

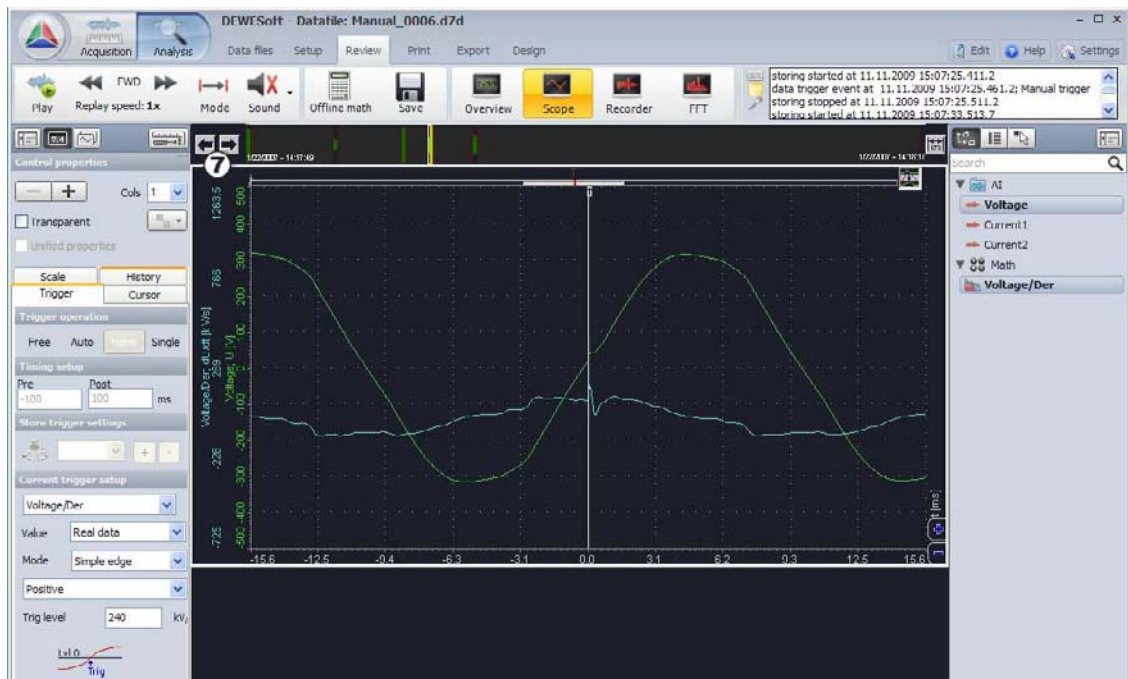
现在让我们来做一些触发。触发数据显示在示波器上，触发**Trig**按钮会**闪动**。



打开文件后，第一个触发立即显示在示波器上。然而，完整的显示区域都已选定（比如在记录仪recorder中）。现在点击触发模式按钮**Trigger mode** ⑥，以触发-触发的方式显示数据。



这时，该按钮变成左右箭头图形⑦。点击该按钮，可以按照一个一个触发事件来浏览数据。还可以使用示波器放大缩小按钮来观察触发事件的特定区域数据。下例中所示比较好的显示了电压求导触发如何记录电压的失真。



2.2 电压/电流/数字 输出传感器

在先前的教程中，我们熟悉了电压和电流的测量方法。不过，通常我们有的传感器有电压或者电流输出，而且需要我们测量电压值，然后将其转换成实际的工程单位。让我们来看一个简单的例子 – 气体质量流量传感器。这种传感器输出 $4\div 20\text{ mA}$ 电流， $0\div 5\text{ V}$ 电压信号，同时还有 RS232 接口。



我们连接它的所有三个接口到我们的系统上。蓝色和黑色线是电压输出，连接到 DAQP-V-A 模块。电压模块上的9针接头有一个非常明显的优点，就是他可以为传感器提供供电电压。如果传感器没有超过模块的电流消耗指标，传感器可以直接从电压模块进行供电。红色和黑色线是电流输出，连接到第二块 DAQP-V模块上，但是在模块前面我们能够看到一个分流电阻。另外一根粗的灰色电缆是 RS232 接头，连接到系统的 RS232 端口。在 DEWESoft 软件中，编写了一个软件插件 **plugin**，用来读取 **RS-232** 的数据。



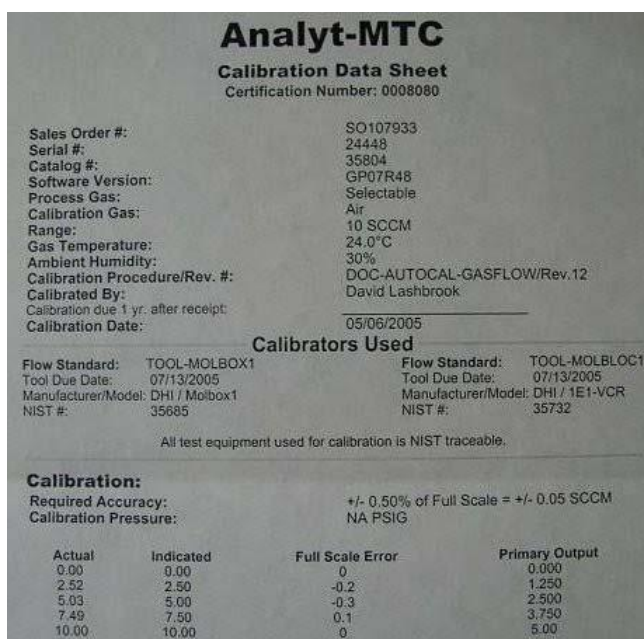
2.2.1 通道设置

所需硬件	任意 AD 卡，DAQP-V 以及任意输出电流信号的传感器
所需软件	除 LT 版本以外的任意 license
设置采样率	至少 1 kHz

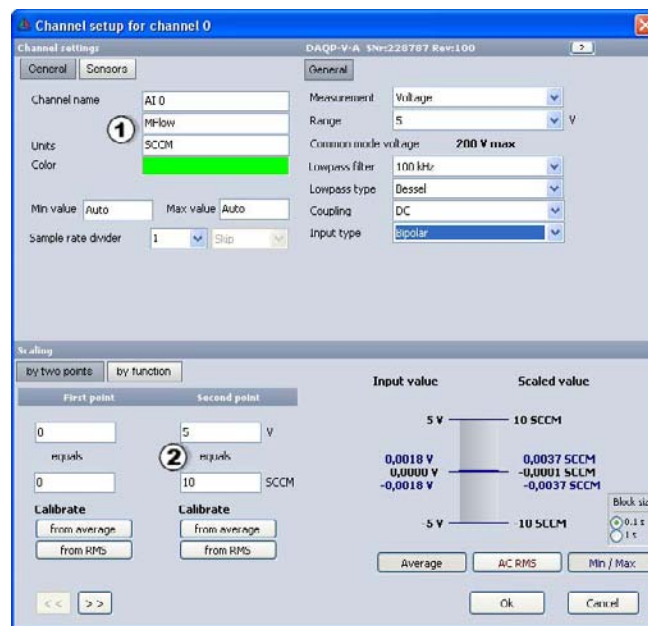
我们只需对两个通道进行简单的通道设置 **channel setup** –第一个模块设置为电压输入，第二个模块设置为电流输入。



首先，我们来量化电压输入的大小。从传感器指标可以看到，传感器的最大量程是 **10 SCCM** (10 标准立方厘米每分钟)。因此我们可以得到，对于电压输出，**0 SCCM** 输出**0 V** 电压，**10 SCCM** 输出**5 V** 电压；对于电流输 **4 mA** 对应**0 SCCM**，**20 mA**相当于**10 SCCM**。

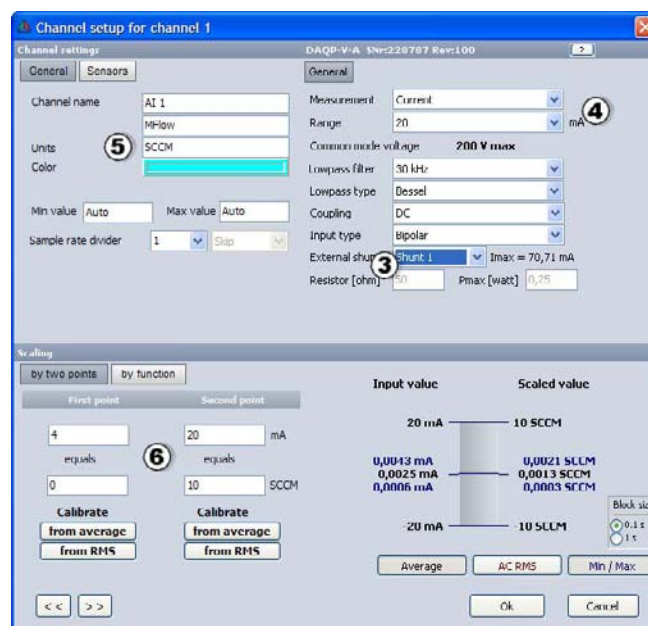


在电压测量中，设置时定义测量单位**units** ①为 **SCCM**，然后设置 **scaling**。 **5 V**相当于**10 SCCM**②。就这么多了。



对于电流输入，我们需要首先设置分流电阻类型。调理模块不能自动识别分流电阻，因此我们需要告诉调理模块分流电阻的类型。在这个例子中，我们用的是标准分流电阻 **Shunt 1**，其电阻值为 **50 Ohm** (3)，我们也可以使用定制的分流电阻。我们只需要知道确切的分流电阻值就可以。请注意模块的输入类型从 **V** 变成 **mA** (4)。

然后输入测量的单位 (**SCCM**) (5)，确定校准 **scaling**。利用 **两点校准法** 比较方便，在这里输入电流限制值。 **4 mA** 相当于 **0 SCCM**， **20 mA** 相当于 **10 SCCM** (6)。



2.2.2 测量

现在我们可以做一些实际测量了。进行测试的时候，我只是用了一个管来给传感器鼓风，然后监测结果。绿色曲线是电压输入，红色曲线是电流输入，蓝色曲线是 RS232 的输入。

因为这个测量过程非常慢，我们只存储了不带触发的结果数据。通常来讲，用记录仪，几个数字表以及一个柱形图就足够显示结果数据了。保存一些数据，然后观察数据曲线之间的差异。



2.2.3 分析

当我们重新载入数据文件，然后放大某个特定数据部分，我们可以发现所有的曲线不是完全准确重叠。主要存在三个明显差异。首先我们可以发现幅值上存在差异。这种差异是因为传感器已经相当老旧，而且校准时间也过期了，因此电压输出就不那么一致了。

第二，我们还可以发现 **RS232** 数据存在延迟现象。这种现象比较普遍，这是因为数据是通过一个速度较慢的数字接口传递的。不过，通过修改软件插件 (在 **DEWESoft** 软件内该设备的驱动) 可以补偿这种延迟。当然，利用 **RS-232** 接口传输数据还是比模拟接口慢的多。

第三，我们可以发现电压和电流输出不能输出负的质量流，即使传感器支持负的质量流输出也不行。



因此，用哪种输出最好呢，电压还是数字接口？我们没办法确定哪个最好，因为输出的好坏取决于传感器本身。首先我们需要知道传感器的构造。如果传感器首先是模拟类型的，电压输出是传感器的主输出，从逻辑上讲，我们最好是使用模拟的电压输出。使用模拟输出的局限性在于信号会引入噪音，会存在长电缆的问题，随着时间的推移，电压输入需要不断进行校准。

如果传感器的数字接口是默认的主输出，而且内部有 DA 转换器，将数字信号转换成模拟输出，这种情况下无疑只能使用这个模拟接口，因为我们其他选择了。当然，使用数字接口需要作一些额外的工作，比如编写驱动程序，只要这个数字接口不是标准的即可（比如 CAN bus 接口）。

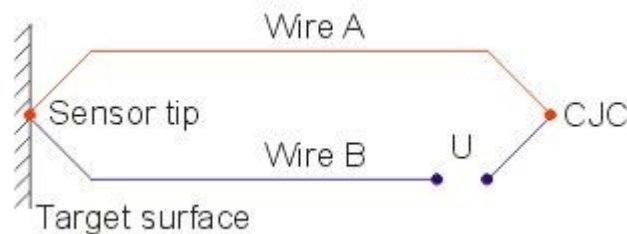
2.3 温度

温度的测量是最常见的测试之一。有许多不同的传感器。

最常见的一种是热电偶 (K, J型), 热电阻(PTC, TC), 热敏电阻(RTD)以及其它类型的传感器。首先我们来看看两种最常见的温度传感器类型。

热电偶

首先就是热电偶, 热电偶就是将两根不同材料的金属线的一端连接在一起, 因为不同材料的温度特性不一样, 我们可以在两根金属线的开口端测量到一个细小的电压 (通常是毫伏级的)。我们需要将两种不同材料的金属线即在热端连接, 同时还在冷端连接, 在冷热两端产生的电压几乎和这两端的温度差成正比。所以, 冷端的温度必须是已知的。出于这个目的, 有必要测量冷端的温度 (就像Dewetron 的温度模块一样)。这种方式叫做**CJC** 或者 冷端补偿**cold junction compensation**。电压值 其实和温度之间也不是线性关系, 因此需要在硬件和软件上进行线性化。这种"家庭作坊"式的传感器测量的精度 $accuracy$ of通常不会超过1 摄氏度, 使用热电偶传感器, 我们可以进一步提高测量的精度。



热电偶可测量的范围非常宽。根据不同的热电偶类型, 可以测量的温度可达1800 摄氏度。当前有很多种类型, 最常用的是 K 型热电偶, 这种类型的热电偶测量范围从 -200 直到 1200 摄氏度, E 型热电偶最适合用于低温测量, N 型热电偶具有和 K 型热电偶类似的特性, 不过它更稳定, 在高温下更不容易氧化。

B, R 以及 S 型热电偶都含有铂金, 因此更贵一些, 不过它们的高温稳定性更好, 所以通常也更多的应用在这些场合。

T 型热电偶的两种金属分别为铜和康铜。因为这两种金属都没有磁性, 所以这种热电偶广泛使用在发电机组等具有磁场的地方。

RTD热电阻

RTD热电阻 表示电阻温度传感器。这种传感器的原理是传感器的电阻随着温度变化而变化。它们比热电偶更贵, 而且最高测量温度只能到 600 摄氏度 (某些型号可到800摄氏度), 不过热电阻更准确。最为人熟知的型号RTD是 Pt100。型号名称一般就表明了热电阻的基本材料是铂金, 额定电阻值在0 摄氏度时是100 Ohms。精度 $accuracy$ 比热电偶更高 – 低于0.3 摄氏度。

现在我们来看看在DEWESoft软件中如何进行温度测量。在下一页, 我们先借用一下一杯咖啡。

2.3.1 通道设置

所需硬件	任意 AD卡, DAQN-THERM 或者 PAD-TH8-P模块
所需软件	除LT版本以外的任意 license
设置采样率	至少 1 kHz

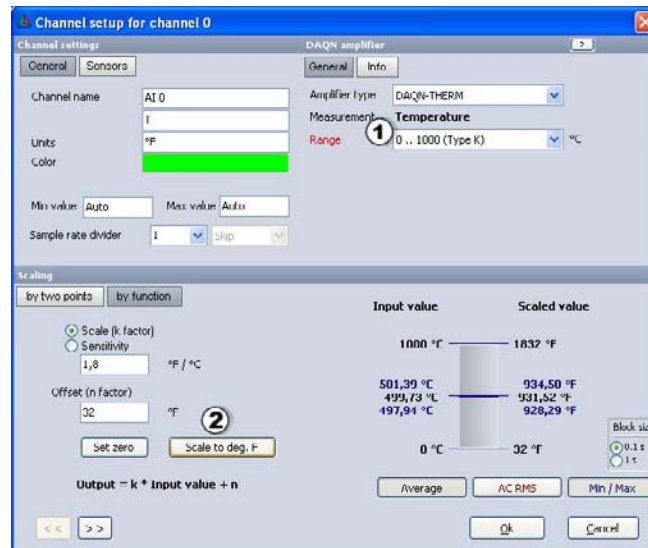
下图显示的是温度测量的典型配置。我们能够看到两块 DAQN-THERM 模块, 还有一块 PAD-TH8-P 模块。EPAD 版本也能使用。

DAQN-THERM 模块是热电偶专用模块。我们连接了两个 K 型热电偶。PAD-TH8-P 能够连接(跟外接的连接模块有关)热电偶和热电阻RTD。对于热电阻RTD, 我们通常使用 DAQN-RTD 模块。



我们准备了一杯咖啡, 咖啡温度很高, 与环境温度相差很大, 让我们来看看通道对此的响应。现在, 测量液体的温度就是我们需要做的工作, 其实可以说更大程度上是测量液体表面的温度。The mounting of the **s**传感器安装可以是将末端捆扎, 拧紧, 焊接上或者是其他办法, 只要这种方法能让两根金属线末端充分接触。如果, 金属线两端存在电势的话, 通常会出现一些问题, 不过因为 DAQN-THERM 以及 PAD-TH8-P 都是隔离的, 所以不会存在这种问题。DAQN-THERM 的隔离电压是 1 kV, 而 PAD-TH8-P 的隔离电压是 350 V。

为了能够使用 DAQN-THERM模块, 我们必须首先进入到System → Hardware setup 菜单下, 模拟Analog标签中激活 DAQN 模块。然后进入通道设置channel setup, 打开通道设置Channel setup 窗口, 在该窗口中你能发现已经出现了 DAQN-THERM模块。在我们的例子中(见图), 我们能找到分配在地址 0 和 1 的两个模块。从列表中选择 DAQN-THERM-3 模块①。这时可以看到正确的测量范围, 从 0 到 1000 摄氏度, 测量范围同时还标注在模块上面。因为软件会根据我们的选择对传感器进行了线性化, 所以我们必须正确设定模块; 否则度量值scaling会出错。对于美国用户, 我们可以进入到System → General setup 菜单下, 来转换 摄氏度deg. C 和华氏度deg. F, 这时会出现一个额外的按钮②, 叫做 Scaling by function。按下该按钮, 实现自动单位转换(从摄氏度到华氏度, 首先乘以1.8, 再加上 32)。



因为我们系统中有两个模块，所以我们需要同时设定两个模块，然后才会出现下面图示的通道设置channel setup。

SLUI	UN/UNIT	C	NAME	AMPLIFIER (U02)	PHYSICAL VALUES	CAL	SETUP
0	Used	AI 0	DAQN THERM	0 ... 1000 °C (Type K)	931.5 °F	Zero	Set ch. 0
1	Used	AI 1	DAQN-THERM	0 ... 1000 °C (Type K)	499.8 °C	Zero	Set ch. 1

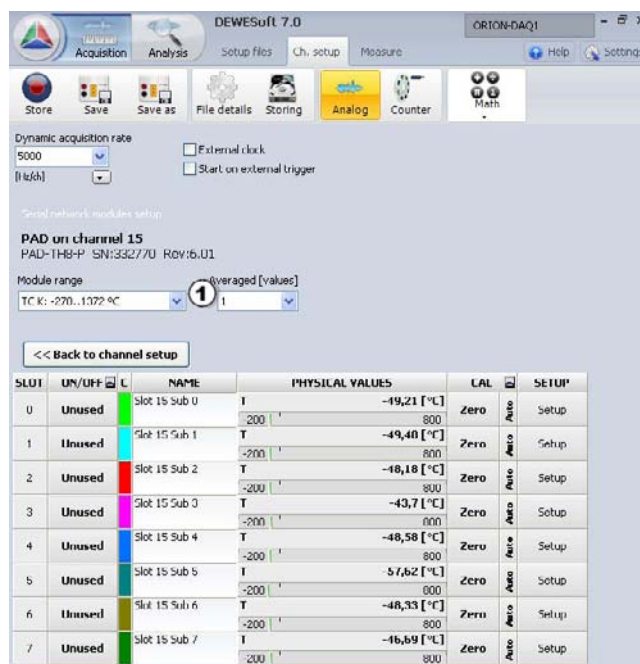
我们在系统中安装了两个温度传感器，所以我们必须同时选择这两个模块，设定为**Used**。

2.3.2 PAD 通道设置

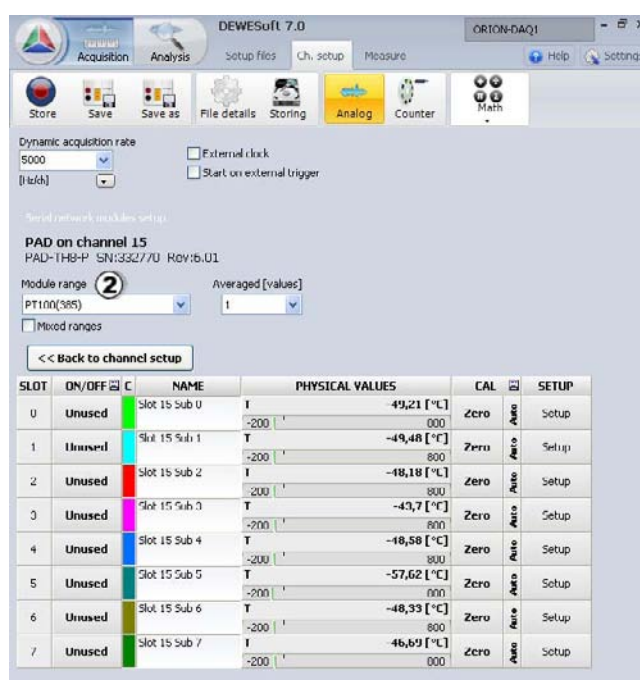
另一种测量温度的方式是通过 PAD-TH8-P 模块。这种模块直接通过串型接口传输数据，因此速度比较慢，但是用这种方法，每个插槽我们可以传输**8**个通道的数据。DEWETRON 提供不同的外接模块，用来连接 J, K 和 T 型热电偶以及 RTD 传感器。PAD-TH8-P 模块自动识别外接模块，因此测量范围也已经设置好了。唯一需要做的事情是设置我们需要使用**Used**的通道。我们还可以定义平均 *averaging*，这样可以让曲线更加平滑，不过这又会降低响应时间。

平均次数 **number of averages** 降低了噪音，同时降低了采样速度。如果是 **1** 次平均①，那么采样速率就是 **6 Hz**，如果是 **4** 次平均，那么采样速率就是 **1.5 Hz**。另外一方面，如果我们连接了更多 PAD 模块，那么读出频率不会超过 **1 Hz**，因此如果采集更多 PAD 模块的数据，注意设置平均次数不要超过 **4** 次。

如果我们连接的是热电偶类型的外接模块，PAD 模块会自动识别外接模块的类型，**测量范围**也会自动调整到实际的范围。



如果我们连接的是热电阻 RTD 外接模块，那么我们需要告诉系统连接到模块上的传感器类型②。模块能够支持的传感器类型有 Pt100, Pt200, Pt500, Pt1000, Pt2000 和 Ni120。而且在同一个模块上不能混合使用不同类型的传感器。

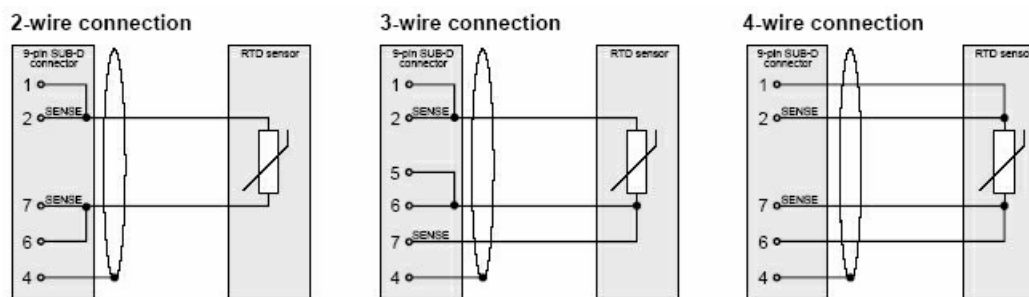


同时还要注意，使用Pt100或者类似的传感器是基于简单的电阻测量方法来测量温度的。请注意金属线接头的电阻会影响测量的结果。电阻随着温度的变化，接线长度的变化以及接头直径的变化而变化。

4 线连接方式可以完全去除由于接头电阻引起的测量误差。

使用2线连接方式，接头电阻的影响不能完全清除。尤其是在PAD-CB8-RTD 和温度传感器之间使用长的，直径很小的电缆的时候，会造成测量结果的极大失真。

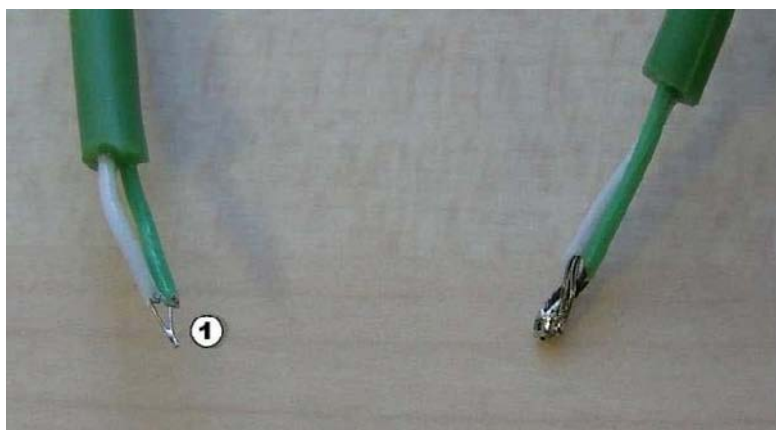
如果3线连接方式中的三根线具有相同的直径和长度，这种连接方式也可以完全补偿接头的电阻影响。在这种连接方式下，假设所有三根线的接头电阻一样，还是比较安全的。因此，只需测量一根电缆的电阻，就可以用来消除接头电阻的影响。



2.3.3 测量

现在，我们来做实际的测量。咖啡以及冷却下来了，现在开始比较好。我们有两个传感器 – 一支的接头比较大，另外一支接头比较小，两根线焊接在一起①。这时候有一点比较重要 – 我们现在在图上看到的都是很成熟的传感器。实际上，两个接头不能焊接在一起，而应该按压在一起，因为焊接的材料是第三种材料，它会改变整个传感器的特性，仅有两种原始材料互相接触连接，这样测量的结果最准确。

不过在例子中，这样做已经足够了 – 我们可以很好区分大接头和小接头传感器之间差别。小接头传感器可以增加传感器的测量带宽，减少传感器反应时间。现在我们看看传感器是如何工作的。



我们把两支传感器同时放在咖啡里，然后观察测量结果。我们已经在通道设置中同时选择了这两个通道，只需进入到记录仪 *recorder*。



在记录仪 *recorder* 中选择所有通道，我们可以在所有通道使用同样的坐标轴(单一值坐标)。这时我们可以改变y坐标的量程

恰当的显示我们的测量量程。

蓝色曲线是细接头传感器测量的结果，而绿色曲线是粗接头传感器测量的结果。将两支传感器同时放入咖啡杯中的瞬间，传感器的响应确实不同。细接头(蓝色)传感器测量的温度几乎是瞬时上升的，而绿色曲线则花了几秒钟时间才上升到正确的温度显示。换句话说，就是细接头(蓝色)传感器对环境温度的变化更加灵敏。



温度测量是一种较慢的过程，所以我们可以使用采样率除法器 **sample rate divider**。我们需要使用较高的采样率的同时还需要同时采集较低速率的温度，这时使用采样率除法器就非常有用，它可以大大减少 需要使用的硬盘空间。这种方式对于使用 DAQN-THERM 模块来测量温度的应用非常重要，因为来自于 PAD 模块的数据是不同步的，而且已经按照采样率存储下来了。

2.4 振动

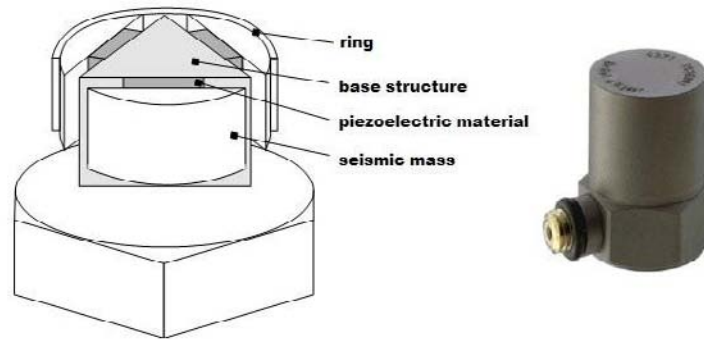
振动测量是测量物体表面振动最有效的办法。我们可以用很多不同类型的传感器来进行振动测量。不过，最常用的传感器是压电式传感器和 IEPE 传感器 (带有内置信号调理电路)。还有一些其它比较典型的传感器，比如低频 (如压阻式传感器，变电容传感器或者 MEMS 传感器) 以及非接触式传感器 (涡流式电流传感器和激光传感器)。

压电式传感器

压电式振动传感器的工作原理是一块压电晶体放置在传感器封装的底部和一个质量块之间，当传感器随着物体表面运动时，质量块压缩压电晶体，压电晶体就会产生一个非常小的电压。要将这么小的一个电量通过电缆传送出去，需要丰富的知识以及昂贵的电缆，因此，这种类型的传感器最近越来越多逐渐被 IEPE 型带内置放大器的传感器所取代。

不过，还是仍然有一些应用需要使用这一类型的传感器。这些应用包括极高加速度和高温 的情况。这种类型的传感器幅值测量范围高达几千个 g 。传感器的形式有单轴的，也有三轴向的。

需要使用 Dewetron DAQP-CHARGE-A 模块配合这类传感器使用。



IEPE (内置电子放大器压电式 Integrated Electronic PiezoElectric) 传感器

IEPE 传感器需要一个 4 或 8 mA 的恒流源供电，输出标准的 5 V 信号，这样通过长电缆传递信号就变得容易多了。而且，这类传感器的放大器制造更容易了，因此更便宜。不过，其幅值测量范围比较有限。我们几乎很难找到测量范围超过 100g 的此类传感器。这种类型的传感器也有单轴向和三轴向的两种。相对来讲，其尺寸比较好控制 – 我们可以在市场上找到 10 mm 的立方体三轴向传感器，重量小于 5 grams。

需要使用 MDAQ-ACC, DAQP-ACC, DAPQ-ACC-A 和 DAQP-CHARGE-A 等模块和这类传感器配合使用。



静态加速度传感器

所有上述传感器都有一些共同的局限性：就是他们不能用来测量静态加速度。通常他们能够测量的低频只能在 0.3 Hz 到 10 Hz 左右，不同的传感器有所不同。对于静态或者非常低频信号的测量，我们需要用到其他类型的传感器。最近最流行的传感器是 MEMS 传感器，这类传感器实际上是一个微型晶片，这种晶片也有类似的机械结构(一个悬臂梁或者是质量块)，这种机械结构随着振动加速度的变化，晶片的电特性 (通常是电容) 也发生改变。仅仅在不久前，这类传感器还属于非常特殊的传感器，用来测量地震或者非常缓慢的运动。不过，最近由于气囊技术的发展，测量静态加速度对此类传感器需求，使得需要更便宜的此类传感器的需求大增。因此，对于这种应用，出现了单晶传感器解决方案。最近，这类传感器又用在了非常便宜的陀螺仪系统中，所以我们能在市场上找到具有相当不错的带宽，可以高达几 kHz，具有相当低的噪声的此类传感器(不过，其尺寸还是比具有相同测量范围 IEPE 传感器要大)。



选择正确的加速度传感器

那么，选择正确的传感器，关键要看哪些因素呢？

低频响应 对于振动测量，通常需要传感器的低频响应截止频率比在测设备上我们感兴趣测量的频率要低。因此，如果需要测量具有**50 Hz** 低频的旋转机械，我们最好选择低频截止频率为**5 Hz**的传感器。不过，有时候最好选择不具有太低低频截止频率的传感器，这是因为低频越低，传感器需要从冲击或则过载中恢复的时间就越长。同时，外面的放大器也需要满足传感器的低频指标。所以，如果放大器至少具有两种量程，在测量时就更加灵活，这种放大器就比较好。

典型的低频振动测量的例子是造纸厂压辊的振动测量。这种振动的频率大概在 **1÷5 Hz**，因此需要具有**0.3 Hz** 或更低的低频响应的振动传感器。对于这种应用，最好使用电荷或者IEPE 类型的传感器。

如果我们需要测量的频率直到 **DC**，即**0Hz**，那么我们就需要一些特殊的传感器技术，比如MEMS传感器。

带宽(高频范围) 带宽的选择同样取决于应用的情况。如果我们对高频信号不感兴趣，或者我们已经知道在高频范围内没有有用的信息，那么我们可以选择带宽不那么大的传感器。某些应用，比如轴承条件监测需要带宽高达**20 kHz** 以上。这种应用情况，我们需要用到 IEPE 或者电荷型传感器。同时还需要注意到，安装的情况会影响到传感器的带宽。请阅读下面的段落了解有关信息。

幅值范围

这些传感器类型中，电荷型传感器的幅值范围是最大的(可以高达**100 000 g** 或更高)，不过IEPE 类型的传感器也不差(高达**1000 g**)。MEMS类型的传感器通常具有有限的幅值范围 (可达**几百个g**)。对于普通目的的测量，最好使用IEPE类型的传感器；而对于高级值的振动测量，最好使用电荷压电晶体传感器。有时候(比如地震应用)，需要用到灵敏度非常高的传感器 (**2 g** 或者更低量程)。

最大抗冲击范围

这些传感器类型中，电荷型传感器对于冲击最不灵敏。它们通常能够承受高达**100 000 g** 的冲击，不过IEPE通常也能承受不超过**5 000** 到**10 000 g**的冲击。MEMS 传感器对冲击最敏感。

噪声 残留噪声实际上确定了传感器能够测量到的最低振动幅值。这也是为什么我们需要在传感器最优化的测量范围内使用该传感器，因为传感器的量程越高，其引入的噪声也越大。

温度范围

所有内置放大电路的传感器，在使用温度范围方面都具有高温局限性，最多可达**130 摄氏度**。电荷型传感器的使用温度范围更高 – 甚至可达**500 摄氏度**。不过，请注意在高温应用时，需要选用高温电缆。

重量 在某些应用中，比如模态测试，由于质量负载效应的影响，传感器的重量是一个非常重要的参数。结构上增加的传感器重量会影响结构的动态特性，因此理想传感器对结构没有任何质量影响。

在其他一些应用领域，比如预测性的动态监测，环境非常恶劣，选择的传感器就必须非常坚固，能够承受各种可能的损害，就像我的一个老朋友经常对我说的那样，你必须在泥土里面还能找到它。

传感器安装

有许多振动传感器本身带有安装孔 (通常是螺纹孔), 有的可以用安装夹安装。 有时候将传感器与结构隔离显得非常重要。 因为IEPE放大器通常不具有电涌隔离, 所以我们需要测量前做好隔离的工作。 当然, 有些传感器本身的封装就可以直接和连接头的地隔离开。

注: 上面所提的数字只是为了便于理解。其实可能有一些特殊的传感器, 它们的指标会超过上面给出的数字。

传感器安装 传感器的安装方式各种各样。安装方式不一样, 传感器的性能, 尤其是带宽会受到影响。 很明显, 最好是在

被测试的试件表面钻孔, 然后用螺栓将传感器紧紧拧在测试试件结构表面。这种安装方式几乎不会影响传感

器本身的动态特性。但是很明显, 有时候用户听说要在自己最新型的试验样机的机翼上钻孔, 会非常愤怒。他们无法接受破坏试件。另外一种不会太大影响传感器带宽的安装方式是在传感器按照表面涂上一层薄薄的双面胶或者蜂蜡 (但是这样在另外一方面又降低了可用的温度范围)。

在机械故障诊断应用中, 比较广泛应用的一种安装方式是磁铁安装。这种安装方式对传感器带宽的影响小一些, 但是要求结构表面必须是磁性的 (不是铝金属或者塑料)。对于可以用安装夹的传感器来讲, 我们可以把安装的前面用胶粘贴到结构表面, 然后把传感器加上即可。

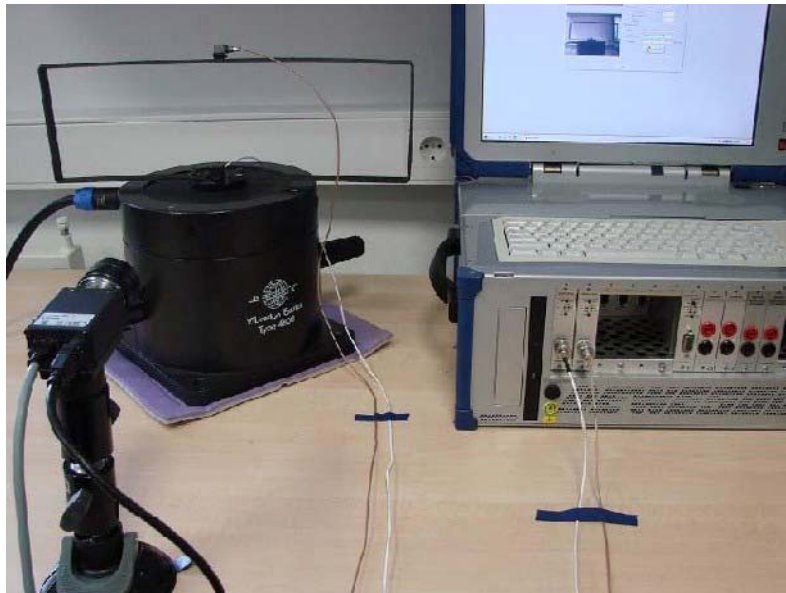
还有一种 "快速但肮脏" 安装方法, 就是用手按住装在安装杆的传感器, 让它贴紧结构表面。这种方式在测量难以触及的地方的振动时, 比较有用, 但是传感器带宽会降低到 $1 \div 2$ kHz。

2.4.1 通道设置

所需硬件	建议使用 24 位 AD 卡(Orion 1624), MDAQ-ACC, DAQP-ACC-A 或者 DAQP-CHARGE
所需软件	PROF 或者 DSA 版本
设置采样率	至少 5 kHz

现在让我们在DEWESoft.软件中来做一些实际的振动测量。因为振动烈度难以想象, 而且我们有很多有关加速度, 振动速度, 振动位移之间的差异问题, 我想实际显示振动。

我已经准备了一个激振器, 激振器上安装了一个塑料方形框架, 塑料框架的固有频率很低。同时, 我用了一台高速摄影机来记录塑料框架的运动情况, 这样可以真正直观看到用加速度计测量到的振动情况。



首先必须强调的是，测量振动时，选择AD卡一定要是带有抗混叠滤波器的卡，比如Orion 1624。否则我们永远无法确认我们所做的测量是否正确。通常来讲，高频(大约 20 kHz)范围内的加速度幅值会非常高。如果AD卡没有抗混叠滤波，而我们又用较低的采样频率来采集数据，那么高频的数据会映像到低频范围。尤其是在做诸如 模态分析等实验时，这是最重要的评价因素。

下图现实的是模拟通道设置。这里有两块 DAQP-ACC-A模块，用来测量振动。让我们首先来看一下如何确定测量的量度。



传感器设置

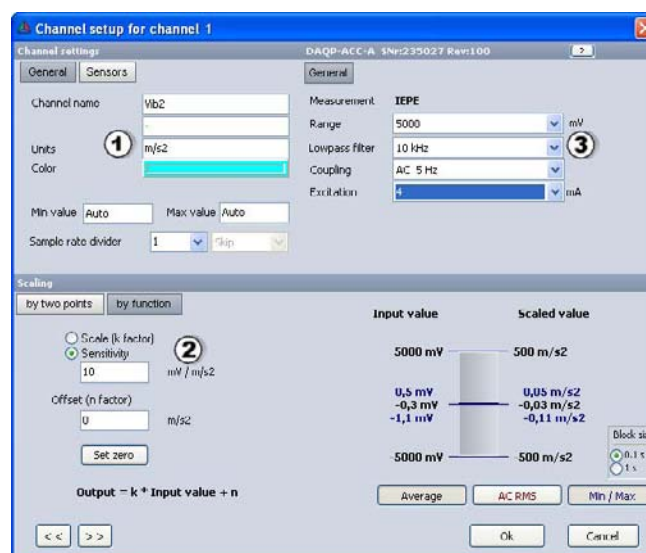
在软件中，我们有三种设置传感器的方法 – 第一种方法是直接从校准数据表上将校准数据输入进去，第二种方法是用校准器对传感器进行校准，第三种方法则是利用 TEDS 技术直接读出校准数据。

1. 从校准数据表中的数据直接输入。我们首先看一下传感器校准表格。通常 IEPE 型振动传感器的灵敏度用两种方式表示， $\text{mV}/\text{m/s}^2$ 或者 mV/g (或者两者头提供)，而压电式(电荷式)传感器的灵敏度则表示为 pC/g 。下图现实的是三轴向的振动传感器的校准数据表。其中的参考灵敏度是关键性的参数，这就是我们需要输入到 **DEWESoft** 设置中的灵敏度值。

	X	Y	Z	axis
Reference Sensitivity ¹⁾ at 159.2 Hz ($\omega = 1000 \text{ s}^{-1}$), 20 ms ² RMS, 4 mA supply current and 23 °C:	9.863	9.988	10.14	mV/ms ²
	9.872	9.995	9.941	mV/g
Frequency Range	0.2-5.5k	0.25-3.0k	0.25-3.0k	Hz
Amplitude ($\pm 10\%$):	1.5-3.0k	1.5-3.0k	1.5-3.0k	Hz
Phase ($\pm 5^\circ$):				
Mounted Resonance Frequency:	18	9	9	kHz
Transverse Sensitivity: Maximum (at 50 Hz, 100 ms ²):	< 5	< 5	< 5	%
Transverse Resonance Frequency:	9	9	9	kHz
Calculated values for TEDS ²⁾ :				
F_{res}	18.7	9.52	9.56	kHz
Q	8.88	1.98	1.67	
Amp. Corr:	-2.5	-2.4	-2.5	%/dec
F_{mp}	0.010	0.010	0.010	Hz
F_{sp}	3.00	3.00	3.00	kHz
Measuring Range:	$\pm 500 \text{ ms}^{-2}$ peak ($\pm 50 \text{ g}$ peak)			
Polarity of the electrical signals are positive for an acceleration in the direction of the arrows on the drawing.				

首先，通常我们要先输入“测量单位”。在这个应用中，我们使用 m/s^2 作单位①。所以最好是进入到 **scaling** → “by function” 标签，选取 **灵敏度Sensitivity**，然后在灵敏度文本框中输入 **9.863 mV / m/s²** ②。

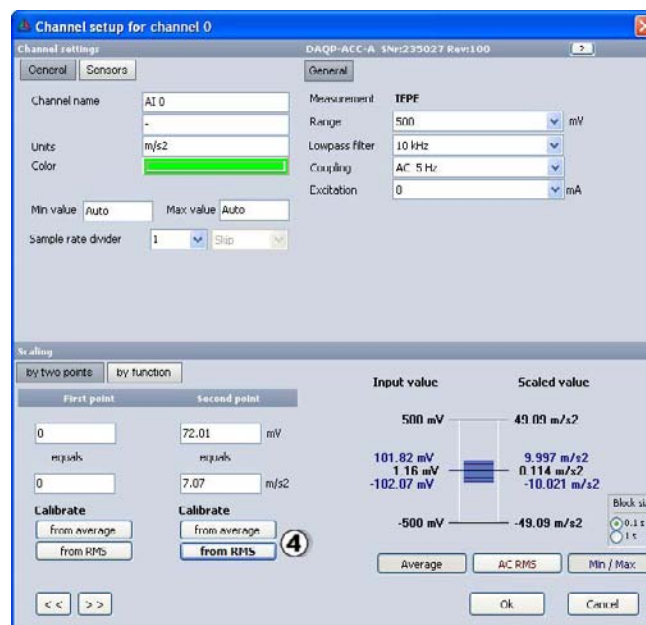
接下来我们需要调整放大器 **量程** 以便最好的配合当前的加速度值，还有滤波器是配合适当的采样率。如果使用了 Orion 1624，**滤波器** 可以设置成最高值 (因为这种 AD 卡本身就提供斜率非常高的抗混叠滤波器)，除非我们是有目的性的设置一个适当的低频范围③。



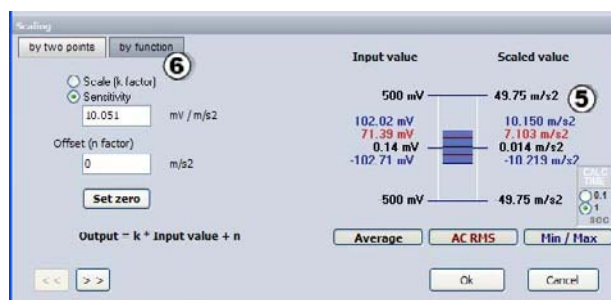
2. 第二种方法是进行现场校准。我是用的是比较老的一个加速度校准器，他能输出 10 m/s^2 峰值的加速度 (7.07 m/s^2 RMS)。把传感器安装到校准器上，调整加速度级值以适合传感器的质量。



然后我们在两点校准two points scaling 中输入加速度级值 7.07 m/s^2 ，然后点击"from RMS"校准④。当前测量的电压级值就会自动写入第二点的文本框中，单位为mV。



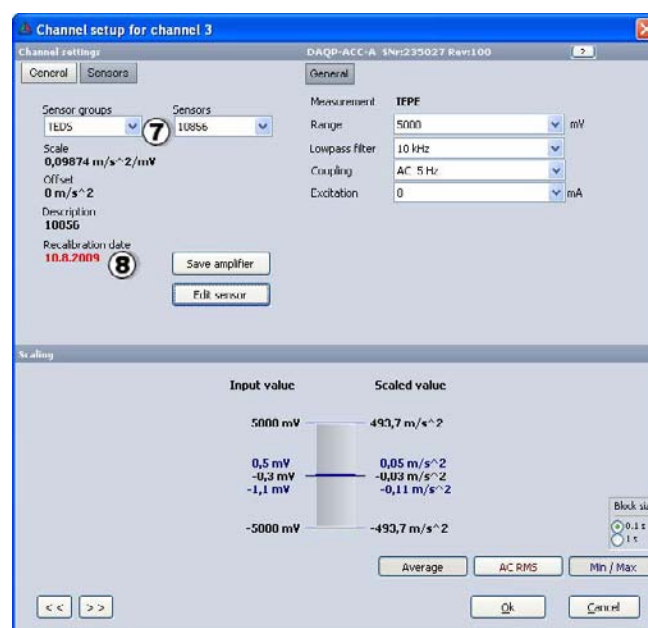
然后我们就能看到校准成功与否。在数据预览窗口，我们可以发现峰值加速度大概为 10 m/s^2 ，而RMS值大约为 7.07 ⑤。建议在数据预览窗口中将校准时间Calc time 选项改为一秒。我们还可以进入到Scaling → "by function" ⑥中，和校准数据表上的值进行比较。



3. 第三种方法，这是一种非常新的传感器设置方法，就是使用电子数据表 - **TEDS**。Dewetron的很多模块 (MDAQ-ACC用于IEPE传感器) 都支持**TEDS** 接口。必须在**hardware setup** 窗口中模拟标签中打开 **TEDS**支持。

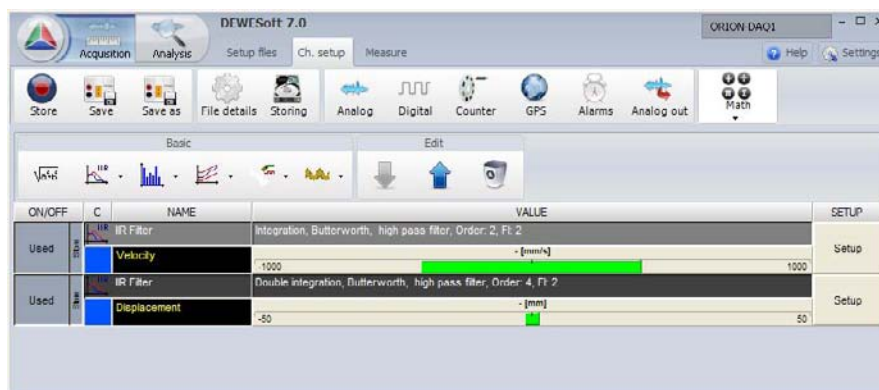
如果我们手上有 TEDS传感器，那么校准设置就比较简单。接入传感器，运行**DEWESoft**软件，软件会立即发现接入的传感器。对于MDAQ-ACC模块，**TEDS**接口只能在放大器设置在IEPE模式下才能使用 (**TEDS**接口不能工作在电压模式下)。如果我们稍后设置(在第一次扫描完成之后) 或者是**DEWESoft** 软件已经运行在**设置setup s**窗口的情况下接入**TEDS**传感器，那么我们就需要重新扫描**TEDS** 传感器。要重新扫描**TEDS**传感器，只需在基本**设置**屏幕窗口中点击**放大器 AMPLIFIER**列的抬头，然后选择**"Rescan TEDS"** 选项即可。

当传感器正确识别后，软件会从传感器中直接读出*scaling factors*, *sensor serial number* ⑦ 以及*recalibration date*⑧ 等数据。这时候，我们来看一下设置屏幕窗口，你就会发现无需再输入灵敏度，因为它已经从传感器读出来了。这种传感器的原理比较简单，而且非常直观，不容易出错。



2.4.2 数学通道设置

接下来第二步是计算振动速度和位移。振动速度和唯一可以通过**数学通道**运算得到，实际上积分本身就是一种滤波器。所以，我们在数学通道设置中增加两个**IIR filters** - 第一个应该是单积分 (用来计算振动速度)，第二个是双积分 (用来计算位移)。

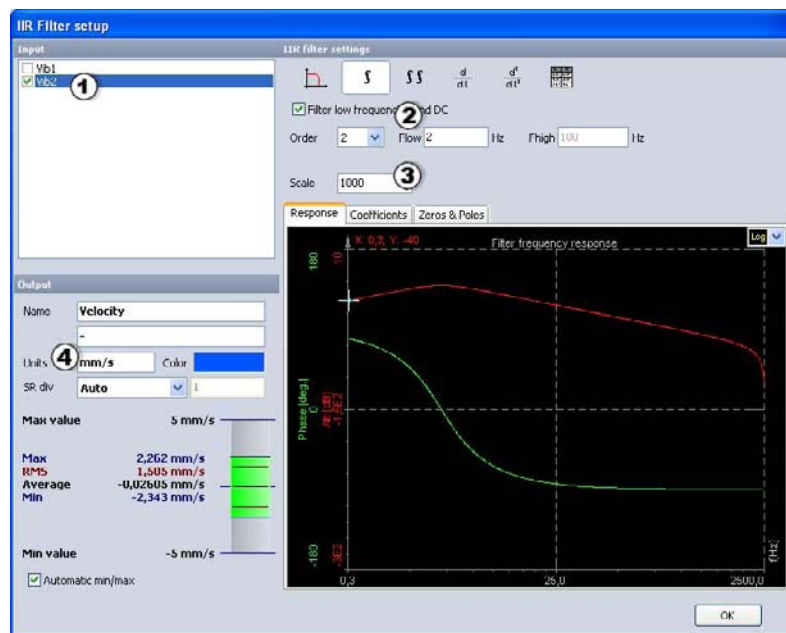


我们先来设置第一个滤波器。首先我们需要选择输入通道。因为第二个通道是上面一个加速度计，取消选择 **Vib1**，然后选择 **Vib2**①。通常比较容易犯的错误就是忘记选择正确的输入通道，因此我建议首选做这一步工作。

然后再选择积分器。因为 **DC** 偏置很容易导致测量和计算的错误，所以我们需要设置**高通滤波器high pass filter**，截断DC偏置。对于**单积分**来讲，滤波的**阶次**至少需要设置为**2**②(如果滤波器阶次为**1**，那么结果中会残存静态偏置，如果没有阶次，那么它就会产生漂移)。

下一步输入**单位units**。我们是从加速度积分得到速度，加速度的单位是 m/s^2 ，输出单位应该是 m/s 。因为这个单位过大，所以我们将它缩小**1000**③倍，也就是用**mm/s**④作单位。

正如我们在预览窗口中看到的，激励器的振动非常大。通常，一般的机械上(在普通机械上，振动是一种由于运动产生的副作用)，速度的振动值不会超过**10-15 mm/s**。



因为我们打算用摄像机来观察结构的运动情况，所以振动的位移同样是我们感兴趣的参数。我们接下来设置另外一个数学通道，选择Vib2 ⑤，然后选择**双积分**。因为双积分实际上是一个二阶的滤波器，所以我们需要给它设置**高通滤波器**⑥，至少是三阶或者三阶以上。通常来讲，由于振动产生的位移，肉眼无法看见，只有几个毫米，不过我们在这个测量中的位移比较大，所以我把位移**单位**设定为**mm** ⑦。从**量值**上讲相当于放大了**1000** ⑧倍。现在，我们就能从 预览窗口中看到其峰峰值大概能到**15 mm** ⑨，而且这个量值的位移，肉眼已经可以分辨了，所以我们可以确认量值单位等的设定是正确的。



现则需要设置一下摄像机，然后就可以看见运动的情况了。

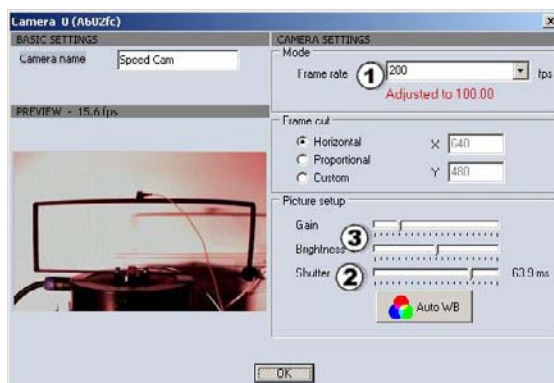
2.4.3 视频设置

我手上有一台 Basler 摄像机连接到系统上。首先我们要在 DEWESoft 设置中的 Video 标签上选择 "Used"，然后点击该标签上的 **Setup** 按钮，打开视频设置窗口，进行摄影机设置。



将 **frame rate** 设置成 120 ①，这种采样率足够让我们清晰分辨振动情况。设置好 **frame rate** 后，需要设定的最重要的参数是摄像机的快门速度 **shutter speed** ②，因为快门速度确定了摄像机获取图片的时间。如果我们需要获取快速运动，快门速度需要调的更快，但是光线必须更强。

接下来根据当前的光线的条件，设置好增益 **Gain** 和亮度 **Brightness** ③。



现在让我们采集一段数据。

2.4.4 测量

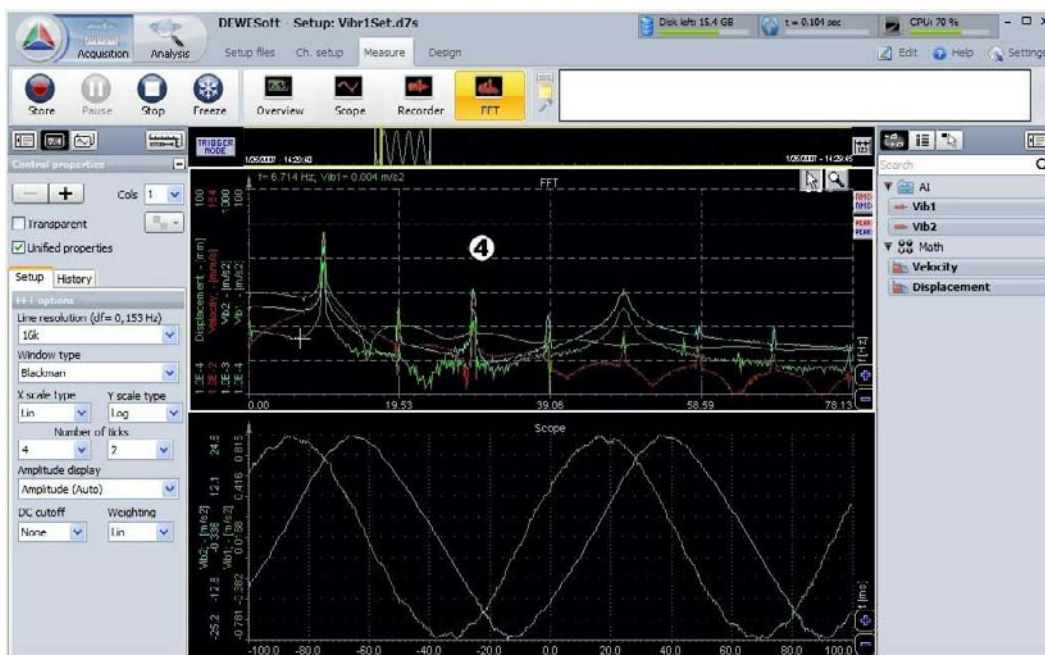
首先，我们先在同一个显示界面上增加记录仪 **recorder**，示波器 **scope** 和摄像机图像 **camera picture** ④。在记录仪

recorder ①上我们可以清晰的看到加速度，速度以及位移三者之间的相位互相相差 **90 deg**。这种结果和理论完全符合 – 如果对正弦函数进行积分，那么就得到余弦函数。

我们用示波器 **scope** ②来观察激振器(绿色曲线)的基本及速度和框架上面的加速度。因为固有平率与基础的运动之间相差 **90°**，所以我们利用示波器来设定能够达到最大响应的恰当的信号发生器的频率。



对于振动测量来讲，最常用的分析仪是FFT ④。FFT 分析仪将随机的波形信号分解成一系列的正弦波，因此我们可以清楚看到振动的结果 (在旋转机械中，我们可以清楚看到不平衡，未对准，齿轮错误以及其它故障)，利用模态分析测试，我们可以在FFT中非常容易看到固有频率，我们还能把它用于特殊的形式(3D FFT)，从而用在升速和降速试验中的阶次跟踪分析中。



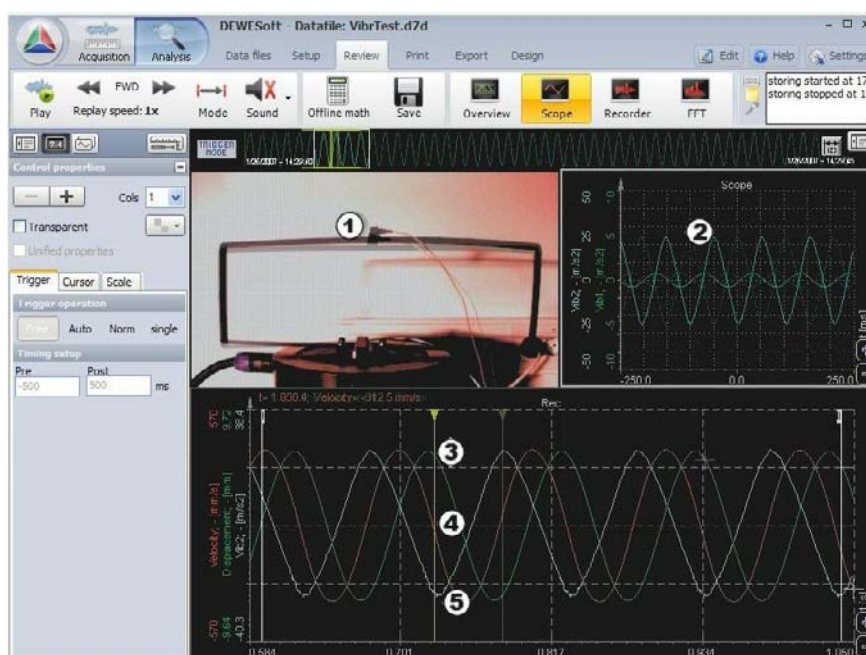
在本试验中，我们唯一感兴趣的东西是看看激励源的非线性(正弦波形误差)，因为激励器的输出必须是一个纯粹的正弦波。

2.4.5 分析

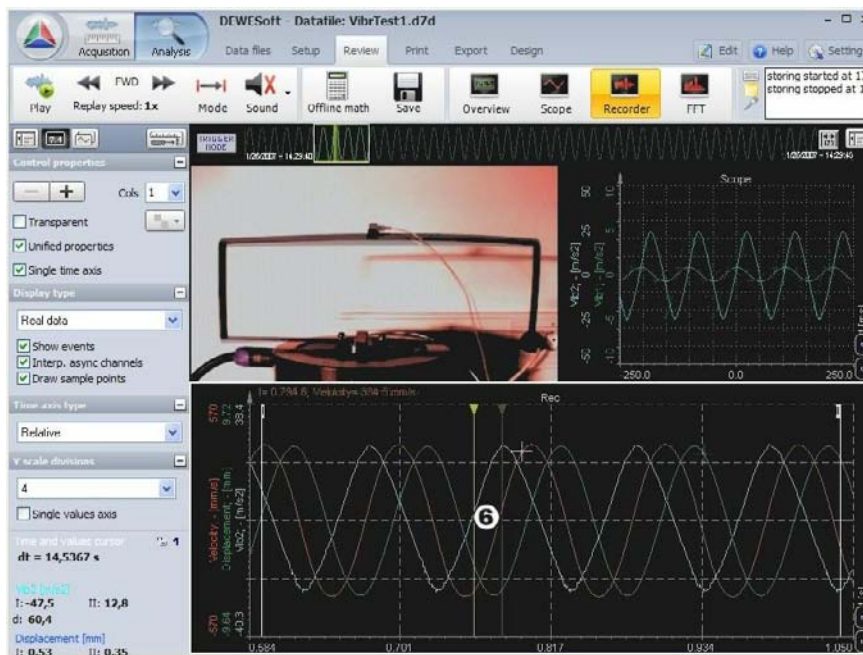
在分析模式中，我们可以从数据中看到更多内在的东西。我讲几个图形重叠在一起，这样方便观察框体上梁的振动情况。下面的第一个图显示的是上梁中心点①的位移。

在示波器scope窗口中，在视频右边我们非常清楚看到Vib1 和 Vib2两者时间相位相差 90° ②。因此，这就是我们需要的该结构的固有频率。

下图中的记录仪recorder 中，我们可以详细分析加速度，速度和位移。上梁中心位置的位移（绿色曲线）③可以从视频①中清楚地看到。速度（红色曲线）是0④ - 这一点非常清楚，因为上梁中心点的最高点是一个转折点，当该点往上运动时，越往上其速度变得越低，到最高点，速度变为0。同样是这个点，它的加速度（蓝色曲线）是最大的，不过方向是负的，即反向的⑤。我们可以把加速度看作是速度的变化速率。我们可以从速度曲线上看到中心点速度变化的速率在最高点时达到最大；也就是说上梁中心点运动到最高点时，加速度达到最大。



接下来我们看一下运动的下一个最重要的点 - 中间点。我们能够确认这个点是中间点，是因为在该点位移为0⑥。中间点的速度此时最大，但是是反向⑥的。这个比较清楚 - 上梁到达中间点时，速度也到最大，然后开始慢慢降下来。在这点的加速度值是0 - w当某个结构静止时，或者是速度恒定时，其加速度为0⑥。这一点可以通过观察蓝色加速度曲线可以确认。



第三个重要的点是最低点。在这里我把最高点的图形重叠在一起作为参考背景⑦。当位移达到最低点时⑧，速度值为0⑧，然后开始增加，加速度值则到达最大值，但是方向是反向的⑧ - 也就是说，速度在这一点的变化率是最大的。

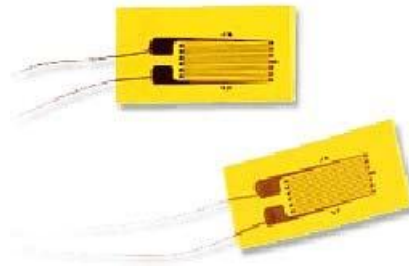


在这里我们进行了一个简单的实验，以便更好的理解振动测量。实际上，真正的振动测量跟这个比起来，看起来会相差很远，不过即使例子再简单，我们使用的仍然是同样的理论基础。

2.5 应变测量

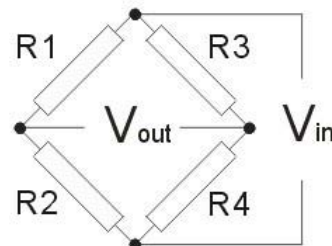
应变片测量是一个非常好的反应测量应力理论的例子，利用应变片，还可以测量材料的应变。应变片基本上都是线圈式电阻，它的基本原理是金属线的线电阻与金属丝长度成正比，而与金属线的截面面积成反比。通常应变片材料使用的都是康铜(铜-镍合金)，这种合金在一个很宽的温度范围里面都具有恒定的电阻。

应变使应变片的电阻发生变化，这种变化非常小，因此我们需要一个质量很好的放大器，深刻理解测量原理，才能够测量这样小的差异。



惠斯通电桥

测量应变片使用最广泛的理论是惠斯通电桥 *Wheatstone bridge*。这种电桥包括四个电阻。在电桥的输入端，我们要输入一个所谓的激励电压，然后测量输出端的电压。如果桥路是平衡的，输出电压肯定是0。



现在有很多种安装应变片的方法。安装方法主要取决于不同的应用。要了解如何使用软件的知识，电桥中各个值的含义，让我们首先来看一下桥路的基本方程。

$$V_{out} / V_{in} = \frac{R1 \times R4 - R2 \times R3}{(R1 + R2) \times (R3 + R4)}$$

电桥中最常用的配置是1/4桥。在1/4桥中，只有一个电阻是应变片，其它三个都是固定的电阻，它们具有和应变片电阻值相同的电阻(这种情况下电桥是平衡的)。因此，在上述方程中， $R4=R+dR$ ，其它电阻则相等，即 $R1=R2=R3=R$ 。把这些代入上述公式中。

$$V_{out} / V_{in} = \frac{R \times (R + dR) - R \times R}{(R + R) \times (R + R + dR)} = \frac{R \times dR}{2 \times R \times (2 \times R + dR)} = \frac{dR}{4 \times R + 2 \times dR} = \frac{dR / R}{4 + 2 \times dR / R}$$

分母中 $2 \times dR/R$ 部分非常小 (拉长1%，误差只有1%)，所以可以将上式简化为：

$$V_{out} / V_{in} = \frac{dR / R}{4}$$

应变片的基本参数就是所谓的应变因子 k 。这个应变因子说明的是随着应变的变化，电阻变化会有产生多大。

$$k = \frac{(dR/R)}{(dL/L)} = \frac{dR/R}{e} \quad \rightarrow \quad \text{因此, } dR/R = k \cdot e$$

如果将上式插入方程中, 得到:

$$V_{out}/V_{in} = \frac{1}{4} \times k \times e$$

如果我们实际中使用各种不同的桥路配置, 那么需要我们在 **DEWESoft** 中输入一个桥路配置因子。这就是所谓的桥路因子。那么公式最后变成:

$$V_{out}/V_{in} = \frac{1}{4} \times B_F \times k \times e$$

对于所有不同的桥路配置, 所有桥路因子都列在下表中。

从应变片测量得到的值就是应变:

$$e = dL/L$$

应变的单位通常表示为 **um/m**, 用毫米表示被拉长的长度和用米为单位表示的样本长度进行比较。因此如果测量结果是 **2000** 的话, 能够说明什么问题呢? 首先, 我们可以用百分比来表示这个值。以 **um/m** 为单位的应变除以 **1000000** 就是被拉长的百分比。因此, **2000** 个微应变表示样本被拉长了 **0.2%**。

我们还可以从这些值里发现它们是否仍然处在材料的弹性范围内。对于钢来讲, 这个弹性限制大约为 **2%** (与不同类型的钢有关), 因此, 如果应变超过 **20000 um/m**, 样本将会超过它的弹性范围, 最终会被永久性的拉伸。我们接下来将进一步看到在全桥的例子中, 如何实现从力到应变的计算。

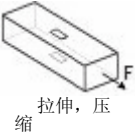
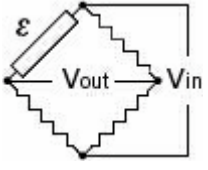
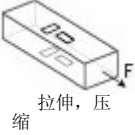
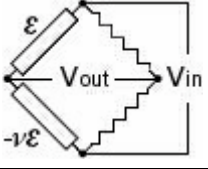
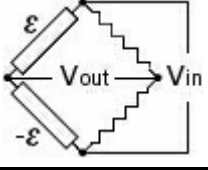
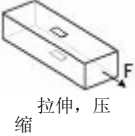
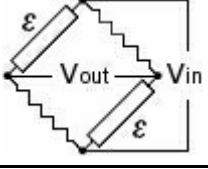
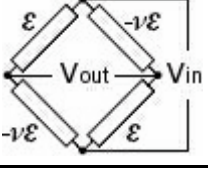
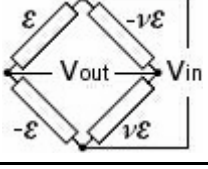
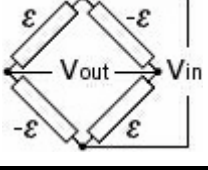
桥路配置

对于基本的测量, 可以有多种桥路配置。首先我们需要知道材料的特定影响 – 也就是说, 当材料被拉伸时, 材料本身在一个 (两个) 方向 (通常来讲) 会变细。横向应变与拉伸应变之比叫做泊松比 **ν**。当所有应变片都交

叉 **90°** 垂直安装时, 知道横向应变比就非常重要, 然后将其代入公式中。

对于钢来讲, **ν** 泊松比大概在 **0.27** 到 **0.31** 之间 (通常取 **0.3**), 对于铝则为 **0.33**。

下表中所列的是应变片的几种基本配置。通常我们将应变片的电桥分为1/4桥，半桥和全桥三种电路。

测量类型	TYPE	桥路模式	方程 V_{out}/V_{in}	桥路因子	线性	说明
 拉伸, 压缩	1/4		$\frac{k \times \epsilon}{4 + 2 \times k \times \epsilon}$	1	否	用一个应变片测量拉伸和压缩 – 基本配置
 拉伸, 压缩	半桥		$\frac{k \times \epsilon \times (1 + \nu)}{4 + 2 \times k \times \epsilon \times (1 - \nu)}$	$(1 + \nu)$	否	一个应变片在主轴方向, 另外一个应变片在垂直横向上 – 通常用于温度补偿
弯曲	半桥		$\frac{k \times \epsilon}{2}$	2	是	两个应变片分别分布在材料的上下两端 – 通常用于测量弯曲
 拉伸, 压缩	半桥		$\frac{k \times \epsilon}{2 + k \times \epsilon}$	2	否	两个应变片具有同样的应变 – 通常用来抵消弯曲
拉伸, 压缩	全桥		$\frac{k \times \epsilon \times (1 + \nu)}{2 + k \times \epsilon \times (1 - \nu)}$	$2 \times (1 + \nu)$	否	两对应变片, 一对安装在主轴上, 另外一对在垂直的横轴上 – 温度补偿和弯曲抵消
弯曲	全桥		$\frac{k \times \epsilon \times (1 + \nu)}{2}$	$2 \times (1 + \nu)$	是	两对应变片, 一对安装在主轴上, 另外一对在垂直的横轴上 – 温度补偿和弯曲抵消
弯曲, 扭转	全桥		$k \times \epsilon$	4	是	两对应变片, 全部都在主轴上, 只是另外一对在反面 – 通常用于测量弯曲

传感器安装

应变片的安装过程包括清洁材料表面, 粘贴 应变片, 连接电缆线 以及保护应变片。安装的详细过程不在本手册的讨论范围之内; 要了解详细的安装过程, 请参考相关的网页或者应变片生产厂家的手册。如上图所示, 因为不同的应变片配置需要采取不同的连接方式, 所以把应变片和放大器连接的时候需要非常注意。

作为一个特殊的说明, 我想讲一个有关应变片连接的小故事, 然后再结束这一段内容。我相信大家都熟悉墨菲法则。墨菲法则说的是“凡事只要有可能出错, 那就一定会出错”。这句话非常著名, 但是很多人不知道这句话出自于应变片测量。该法则的“发明者”是艾迪·墨菲上校, 他有一次在爱德华空军基地用应变片系统来进行g-force测试, 在该测试中, 他们打算测试(在人体上)

人体能够承受的最大加速度 (顺便说一下, 人体能够承受的最大加速度为 **40 g**)g force。

然而, 第一次的测量结果为**0**, 原因很简单, 因为他们连接电缆的方式让应变片正好相互抵消了。艾迪.墨菲上校指责他的助手造成这种失误, 是他的助手用错误的方式连接应变片。故事的另外一部分(甚至是测量过程中最有趣的方面) 在于墨菲拒绝检查整个系统, 即使在测试之前有人提醒他检查一下系统。

所以这个故事的意思在于, 整个测试应该包括: 连接 - 接电缆 - 验证 - 测量。如果艾迪.墨菲能够遵从这个过程, 墨菲法则就不会(至少不会在这件偶然事件中)被发明出来了。

2.5.1 实验设置

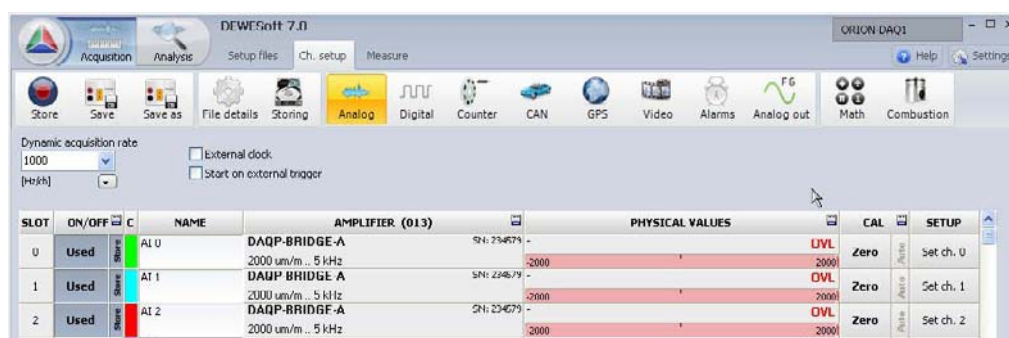
所需硬件	至少 16 位AD卡, DAQP-BRIDGE或者MDAQ-STRAIN 模块
所需软件	SE, PROF或者DSA 版本
设置采样率	至少 1 kHz

我准备了三种常用类型的应变片配置。第一我们有一个音叉, 音叉上面粘贴了一个应变片, 第二我们还有一个自己做的力传感器, 力传感器是用全桥应变片连接的, 最后我们还有一个全桥的未安装负载的重量传感器。



从放大器这部分来讲, 我们现有三块带隔离输入的 DAQP-BRIDGE-A模块。当然, 我们也可以用同样的过程使用MDAQ-STRAIN 模块。

DAQP-BRIDGE-B模块的带宽更宽, 量程更多(量程可以到**0.1 mV/V**), 不过它不带隔离。



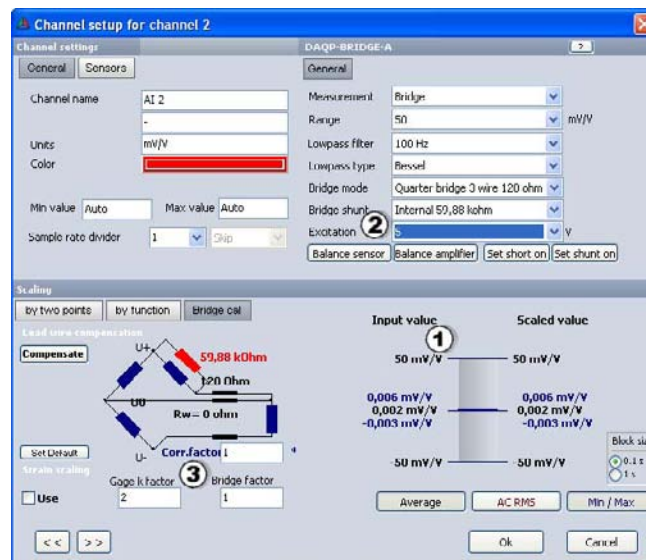
2.5.2 1/4桥设置

首先我们来看看使用DAQP-BRIDGE-A模块在软件中如何设置1/4桥。我们现在有一个应变片粘贴到钢叉上。从应变片随机提供的技术指标表格上可以看到应变片的电阻是**120 Ohms**，应变因子**k**是**2.07**。当我们选择应变片时，我们有两种选择，**120**和**350 Ohms**两种。**120 Ohms**的应变片消耗的能量小一些，所以升温加热的效果低一些，而**350 Ohms**具有更大的信号，因此在长电缆时工作更好。

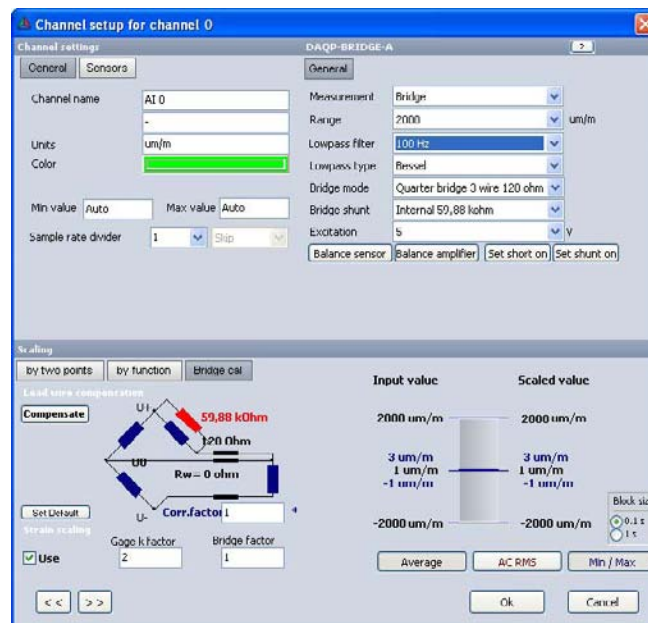


根据技术指标信息，我们可以设置 1/4桥应变片为**120 Ohms**输入类型。这表示4个桥路中只有一个是真正的应变片，而其他三个则是内置的精密电阻。现在我们把输入量度设为**mV/V** ①。

我们可以自由选择激励电压②。激励电压越大，信号也会越大，因此噪声会降低，但是又会导致应变片更多发热，增加更多电源消耗。所以，通常建议在长电缆的情况下选择大一点的激励电压。



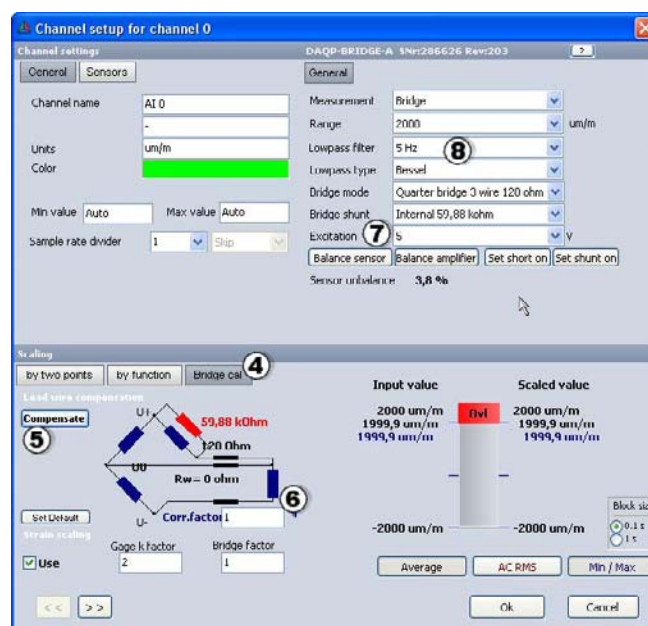
下一步就是根据技术指标输入 **k factor** 以及**bridge factor** ③。因为这是一个简单的1/4桥，所以我们输入桥路因子为**1**。



我们的音叉上面的应变片连接方式是三线连接。三线连接方式的优点在于可以抵消电缆线电阻的影响。尤其是在使用长电缆的情况下，长电缆会大大降低应变片的灵敏度，而三线电缆连接方式再加上分流校准，这样我们就能够测量这个电阻，消除这种误差。

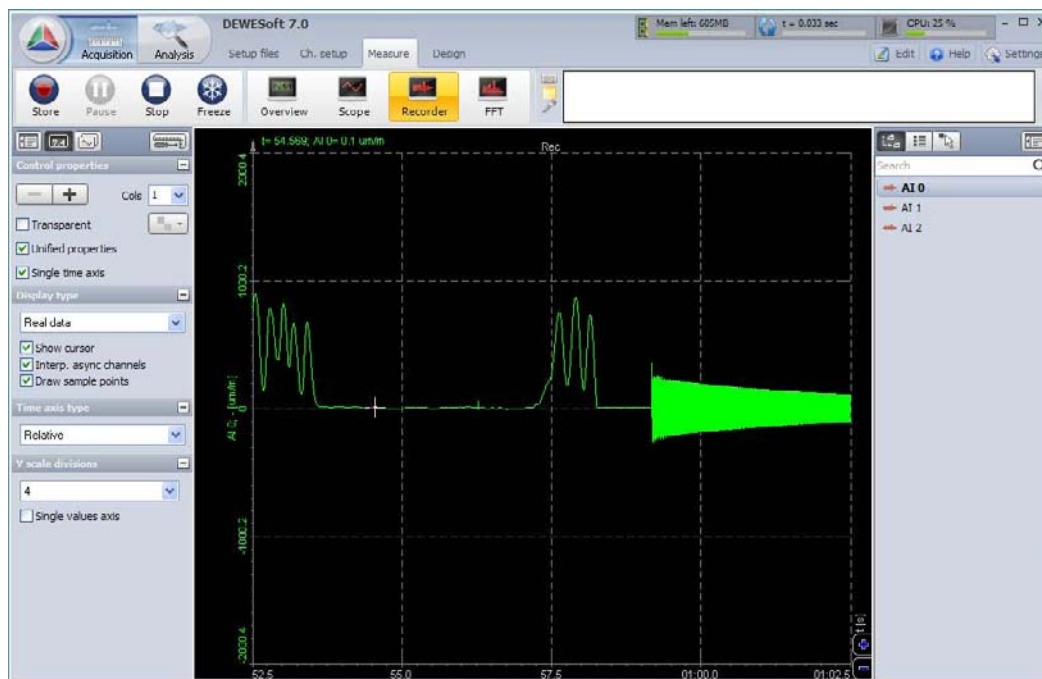
因此，在本例中，我们可以使用分流校准测量并校正电阻误差。我们需要选择 **Bridge cal** 标签④，然后点击 **Calibrate** 按钮⑤。分流校准需要花费一点时间，因为校准需要设置几个配置参数，然后测回来一些结果。测量完成后，测量得到的电缆线电阻值显示在图上， **correction factor** ⑥显示在文本框中，可能的话， **激励电压**⑦有可能会往上调，直到满足应变片所需的灵敏度。

最后一步，我们将 **Lowpass** 滤波器设置为 **5 kHz** ⑧，这样让我们能够看到音叉的固有频率，音叉的固有频率大概在 **440 Hz** (A4 音调)。



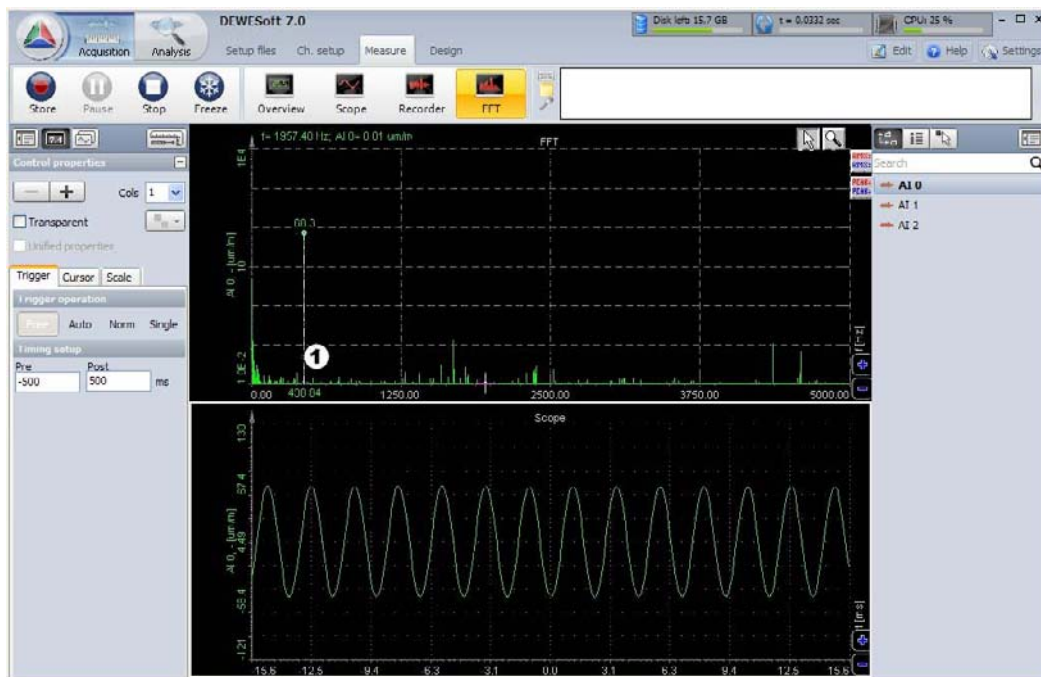
2.5.3 1/4桥测量

现在我们来做一些测量。先看一下记录仪 *recorder*。如果我们给音叉施加一个静态力，我们可以把它看做是信号的偏置改变。我们还可以敲击音叉，因为这样可以让音叉发出响声，其响声能够反映出音叉的固有频率（音叉的通常用法）。我们可以看到这是一个高频的振动，其幅值逐渐减少，原因是空气摩擦和音叉内部的摩擦。



当我们用 FFT 分析仪分析时（我们必须把Y轴改成对数坐标 *logarithmic scale*，这样我们能够更好的观察频率幅值，我们可以发现大约在 **440 Hz** 的地方有一个明显的峰值。我们在FFT分析仪的峰值处点击一下光标。从光标观察，现实的频率是 **438.8 Hz** ①。这个值不是精确度的 **440Hz**，这是因为一方面经过多年使用，音叉的频率不可能还是原先的频率，另一方面是因为FFT分析的线分辨率导致的。

FFT的线分辨率取决于采样率和FFT采样的分析线数。因此如果我们想在FFT分析中取得比较快的响应，那么我们需要选择更少的分析线数，但是频率分辨率会降低。如果我们想得到更精确的频率，我们可以选择更高的分析线数。这些操作在参考指导 *reference guide* 有详尽描述，简单的规律是：如果我们获取1秒钟的数据用来计算FFT，那么FFT的结果只有 **1 Hz** 的线分辨率。如果用2秒钟的数据，线分辨率就能达到 **0.5 Hz**。

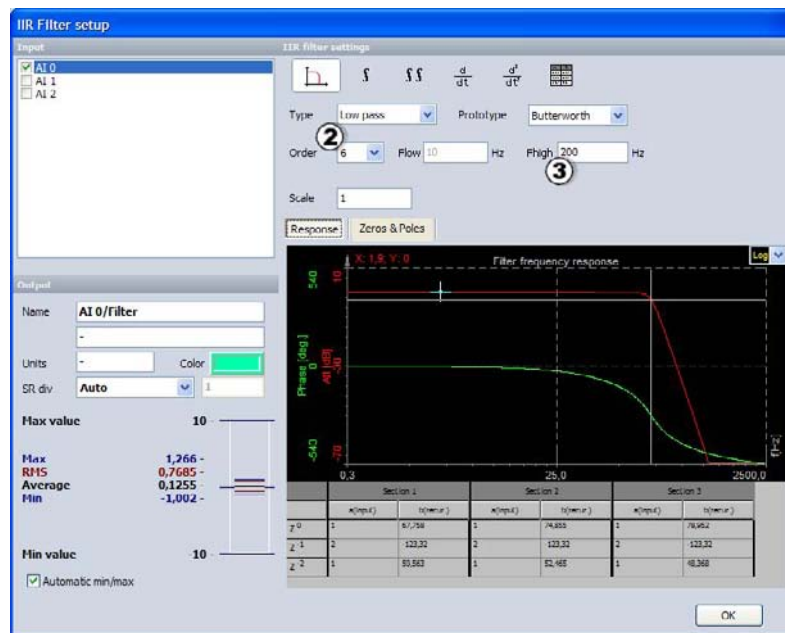


这是一个非常好的例子来展示如何在 **DEWESoft** 软件中使用 **filters** 。非常明显，我们有一部分偏移形式的信号 (静态负载)，还有一部分在 **440 Hz** 频率下变化的动态形式的信号。

如果我们想从原始波形中抽出这两部分的信号，我们需要设置两个滤波器 — 一个是低通滤波器，另外一个高通滤波器。因此，我们需要在 **Math** 通道中加入两个滤波器。



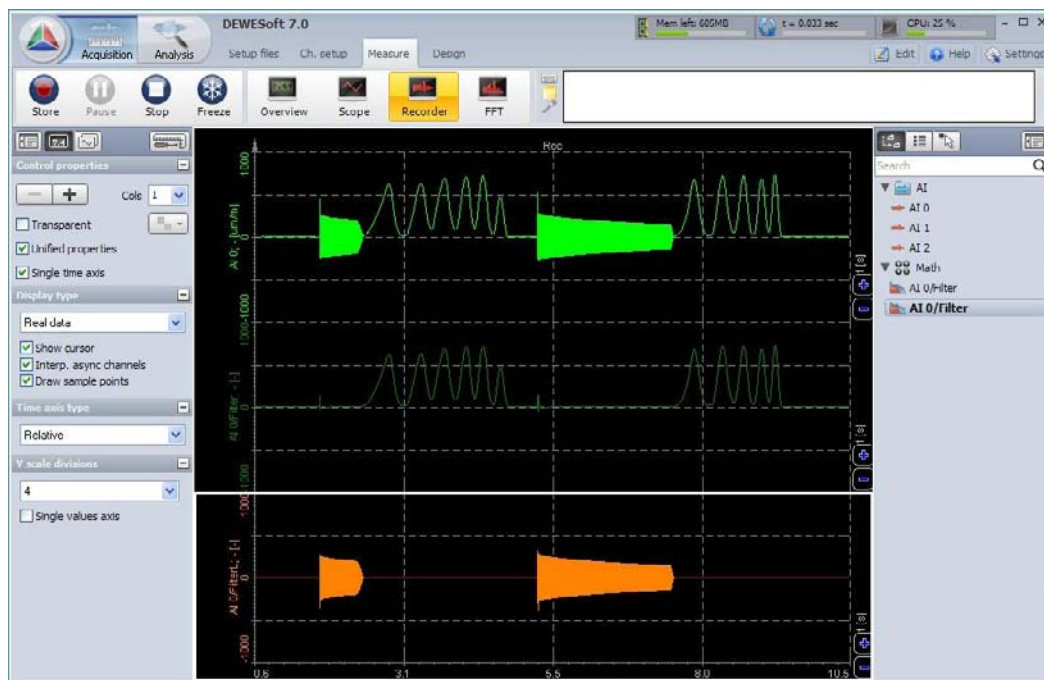
1. 在本例中我们首先选择设置(AI 0)。然后为该通道设置Low pass, 6th②阶次滤波器, 设置FHigh截止频率为 200 Hz③。因此, 这将让频率低于200 Hz以下的信号通过该滤波器, 同时切掉频率高于200 Hz以上的信号。



2. 第二个滤波器设置成高通high pass, 6th 阶次, 和FLow 滤波器相同的截止频率④。



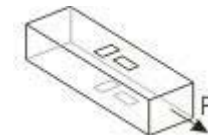
如果我们在记录仪recorder上显示这两个滤波后的信号, 我们可以看到信号被完全分解成静态负载信号和动态振动信号。我们可以使用此技术切除掉信号中不需要的部分, 或者抽取某信号中确定的某个频率成分。



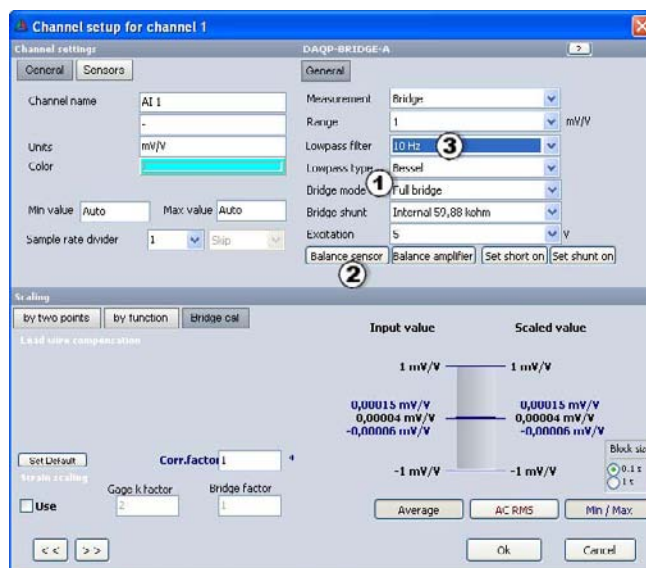
到现在为止，我们都是使用的IIR滤波器，在此处我们是想得到更高的计算速度和截止速率，如果我们不希望产生任何相位差，我们也可以使用FIR滤波器。可以在用户手册中滤波器比较一节中找到更多有关滤波器的详细信息。

2.5.4 全桥设置

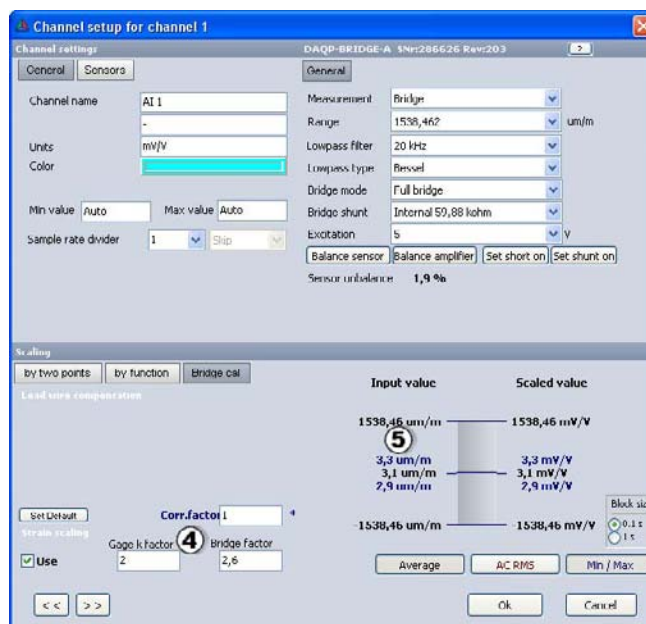
对于全桥传感器，我们使用的是“自己做”力传感器。传感器有四个(两组)xy应变片，每组应变片中的一个应变片和主轴平行，另外一个则在垂直的横轴向上。另外一组以同样方式粘贴在另一面，这种传感器具有温度补偿，消除弯曲的作用。



接下来我们对这种配置按步骤进行校准。首先我们必须将输入类型input type改为全桥Full bridge①，再点击进行桥路调零bridge zero ②。如果调零不成功，那么需要选择尽可能大的量程，调零，然后选择需要的量程，在进行调零。然后设置低通滤波器Lowpass filter ③。因为我们或多或少需要在一个值非常低的量程内做静态测试，使用非常低的低通滤波器可以大大提高结果的质量。在我们的例子中，设置成10 Hz ③。



接下来我们设置直接桥路量度 **Direct bridge scaling**，将应变片 **k factor** 设置为 2，**bridge factor** 设置为 2.6④。**k factor** 的值是从应变片校准数据表中读出来，而 **bridge factor** 的值则来自于桥路配置表。现在的输入值是应变值，单位为 **um/m** ⑤。



现在我们需要做最后的定量 **final scaling**，因为我们需要测量的是力，单位为 **N**。现在需要做一些计算。我们需要使用下来方程：

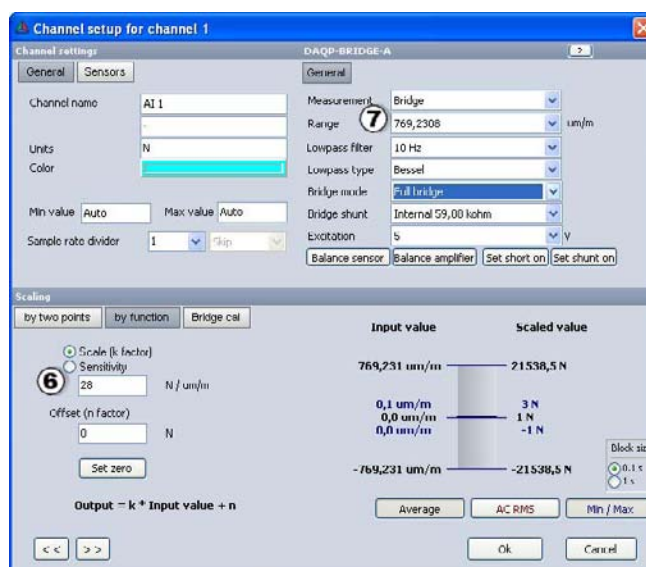
$$\text{应力 Stress} = \text{力 Force} / \text{面积 Area} \quad \text{以及}$$

$$\text{应力 Stress} = E \cdot \text{应变 Strain}$$

因此得到 **Force = Area · E · Strain**。从弹性模量理论可知(杨氏模量)，钢的弹性模量是 **210000 N/mm²**，传感器交叉截面面积是 **139 mm²**，那么我们得到：

$$\text{Force} = 139 \text{ mm}^2 \cdot 210000 \text{ N/mm}^2 \cdot \text{strain} / 1E6 = 28 \cdot \text{strain} \quad \text{⑥}$$

其中要乘以一个因子 **1E6**，是因为我们测量得到的应变单位是 **um/m**，而我们需要将所有的其他量的单位都对上这个量值单位。



现在我们得到了真正单位为N的实际数据。最后要做的事情是根据输入⑦选择测量量程。在本例中，最低量程就可以满足我们的测量需要了。

我们也可以选择量程的单位为 $\mu\text{m/m}$ (放大器)，然后通过点击测量量程右边的下拉菜单，选择 **Show scaled ranges** 变成实际测量单位。

短路Short 以及 **分流按钮Shunt button** 的作用是检测应变片。Short 按钮用于在输入端短路输入端的信号针，测量桥路偏移。Shunt (分流电阻同样还用在了分流电阻校准过程) 可以用来观察桥路是否响应 – 据此判断连接是否正常。

2.5.5 全桥测量

手册的编写非常烦人。所以到现在为止，我们的手册差不多有将近200多页了，所以我们需要短暂休息一下，做点特别的事情。

这个力传感器就是一个绝佳的机会。我增加了一个摄像头 来记录实时的测量过程，我们可以在这件屋子里进行一场竞赛，看看谁能够把这个力传感器拉的更开，最终评出谁是最强壮的家伙。因为该力传感器的测量量程是30kN，也就是说将近3吨，所以当然不可能有人能有这么大的力量。

不过既然我们用传感器测量力，我们可以非常容易判定结果。

现在首先轮到我们这里最强壮的家伙 (希望如此)，他是我们的 CAM/CNC工程师，拉出了570 N①的力，相当于57 公斤。



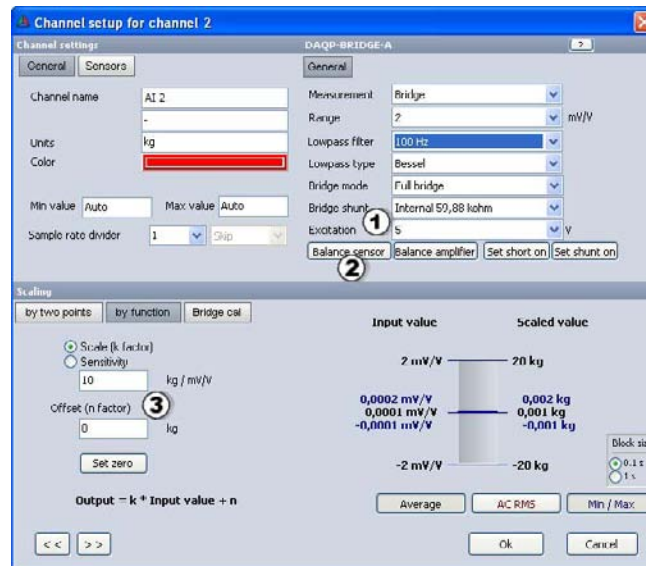
但是最大的问题在于：如何管理这些应用？正如我们从图片中看到的，这需要付出很多努力，需要更加详细的讨论，虽然这些工作对我们的结果影响不是那么大...



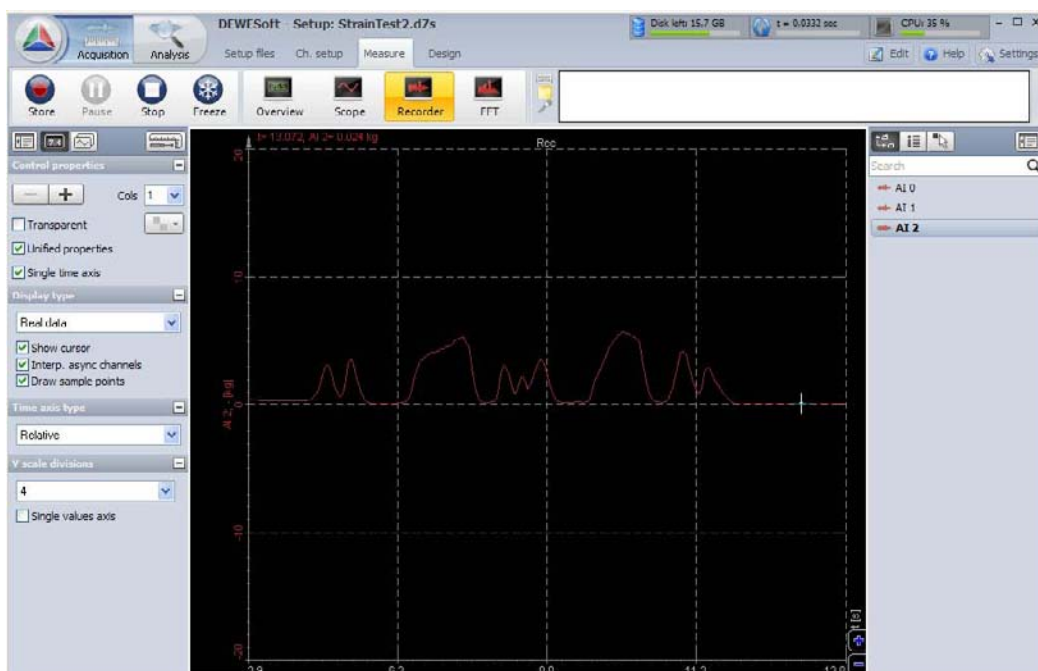
2.5.6 重量传感器设置

重量传感器的设置是所有传感器设置中最简单的。重量传感器使用的是全桥电路 *Full bridge*，因此我们只需设置正确的输入类型 **Input type**，设置恰当的激励 **Excitation** 电压 ①，然后在不加任何负载的情况下进行 **桥路调零** **Bridge zero** ②。

下一步就是输入传感器的 **scale factor**。重量传感器是用公斤为单位来校准的，我们可以从传感器的技术指标表中找到传感器的灵敏度，其输出为 **1 mV/V 每10 kg**。我们在 **scale factor** ③ 文本框中输入该值，然后我们的所有设置就结束了。



实际上从测量过程来看，我们真正能从力传感器上看到的東西不太多。我們當然可以在房間也搞一個“誰是最重的傢伙”的競賽，但是這個做起來就沒那麼有意思了 – 因為我們知道我們的應用程序確實能夠做到這一點。



2.6 计数器

计数器通道 主要用于进行计数以及频率测量。典型应用包括:

- 计数模式
 - 事件计数
 - 简单计数
 - 门事件
 - 向上/向下计数
 - **X1, X2, X4** 模式
 - 零脉冲
 - 解码器计数
 - 周期测量
 - 脉冲宽度测量
 - 占空比 **Duty Cycle**
- 计时模式
 - 周期和脉冲宽度
 - 两个脉冲沿分离
 - 频率/超级计数器
- 传感器模式

不同的AD卡能够提供的计数器不一样。目前最先进的计数器是在Orion 1624 和Orion 1616 AD卡上的计数器。这两种AD上的计数器有一种特殊模式,叫做超级计数器,用它可以进行精确的频率测量,而且可以直接支持m当前最常用的传感器。所有其他类型的计数器只能提供这些应用的一部分功能。

基本上,计数器的输入必须是很干净的数字信号,大多是AD卡默认支持5 V电平信号(某些卡提供更高的信号输入,甚至是可变触发电平)。如果我们信号更低或者更高,那么我们就必须在计数器前端使用调理器。

最后我们到计数器最常用的地方看看其实际的应用:

- 汽车上的计数器

2.6.1 事件计数

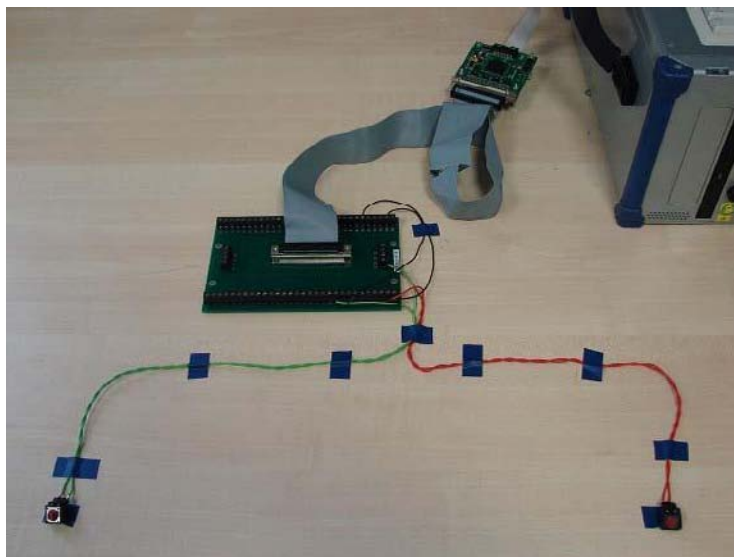
事件计数是计数器应用中最简单的应用:

所需硬件	Orion, NI (带有计数器的AD卡), NI-MX, DAP, Data translation AD卡
所需软件	任意版本
设置采样率	最少1 kHz

有两种事件计数的特殊模式:

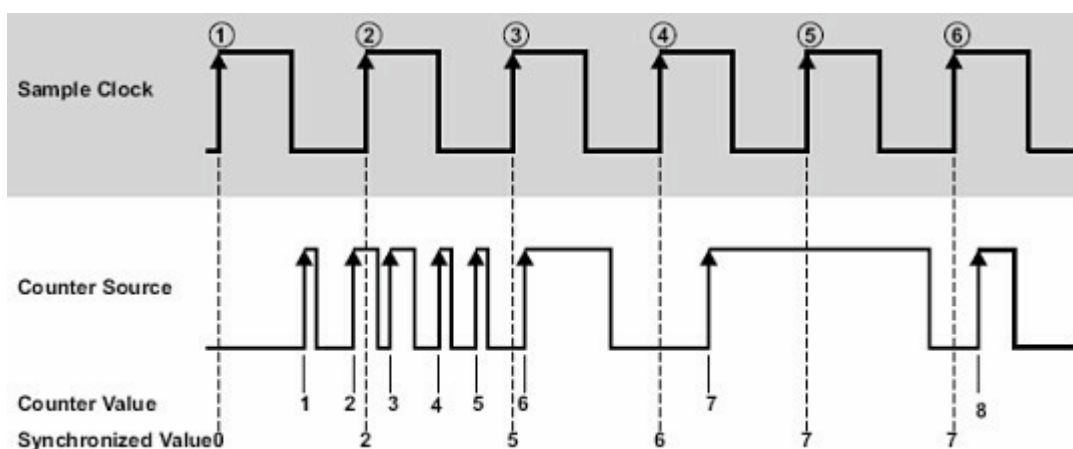
1. 门事件计数,事件只在当门信号是高电平的时候计数 (只有Orion 计数器能做)
2. 向上/向下计数,当门信号为高电平时,事件向上计数;当门信号为低电平时,则向下计数 (Orion, NI MX, NI E 系列以及DT 卡)。

我们设计了以下一个简单的测试配置，包括两个按压开关，连接到Orion AD卡的扩展计数器上。

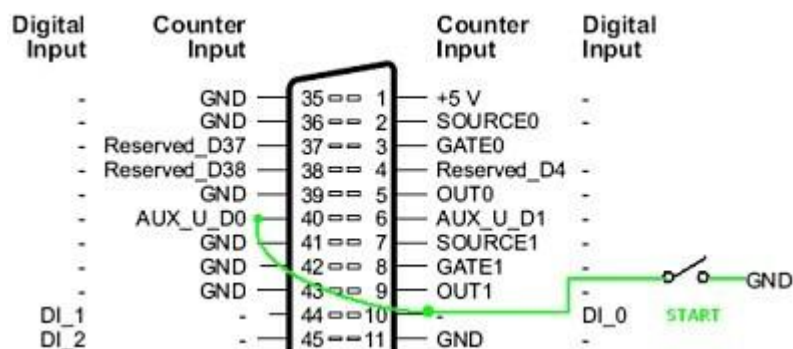


2.6.1.1 简单事件计数

简单事件计数模式就是在信号的下降沿和上升沿对其进行计数。请注意，在该手册中描述的计数器的功能可能在其他AD卡上不能实现。



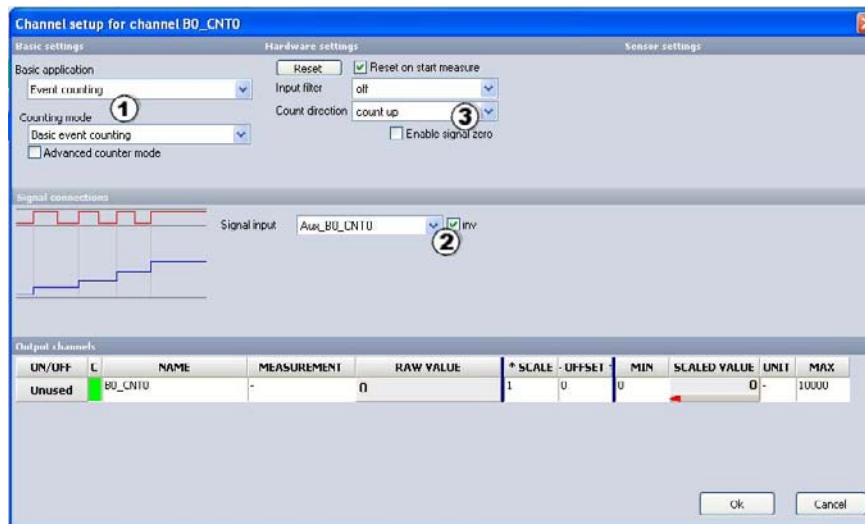
在我们的例子中，我们只是在计数器输入和接地之间增加了一个开关。因为所有输入都存在高电平pull-ups，接通开关(短路)就会得到 数字0，开关打开时，得到数字1。为了参考方便，我们将同样的信号接入到DI_0。



然后我们进入到计数器counter标签，选择CNT0通道。

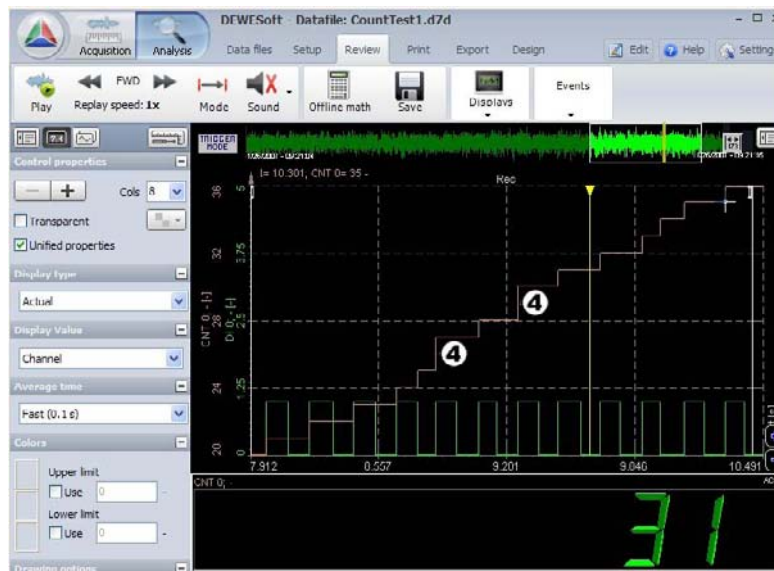


点击设置按钮，设置计数器。在基本应用Basic application域中"事件计数Event counting"以及计数模式Counting mode域中"基本事件计数Basic event counting"①两个选项都已经默认选择好了。接下来选择信号输入Signal input。通常信号输入为Source0，不过我们已经把信号连接到Aux_B0_CNT0上，我们需要选择这个信号②作为输入。在正常状态下(按压式开关没有按下)，电压电平是高的，所以我们最好是选择inv复选框③，将信号转换为相反的信号。这样做有两种效果：首先电平值发生改变，如果开关按钮没有按下，那么电压电平值变成低电平，计数器必定会计数下降沿。我们可以选择计数方向Count direction为向上计数count up或者向下计数down③。



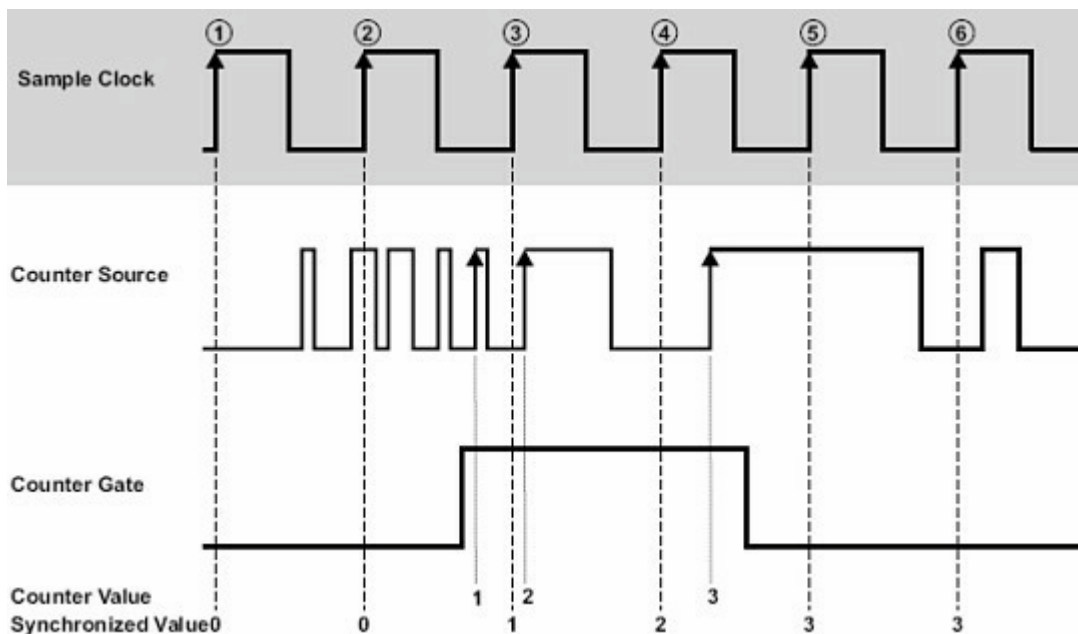
滤波器filter设置也非常重要，以防止双计数情况发生。我们需要将滤波器设置得比我们想要测试的事件稍微反应快一些，或者用另外一种逻辑来讲，需要将滤波器设置得比我们想要测试的信号中的短时脉冲信号的频率稍低。考虑到我们的手动开关以及80 MHz的时基，我们可以确信信号中肯定有一些短时脉冲。不过我们还是来看看我们的测量。

绿色曲线显示的是来自开关的数字信号，黄色曲线显示的是计数值，而数字表使用数字显示开关的次数。每当电压值从低电平变到高电平，计数器的值就增加一次(在我们的例子中，因为信号数值已经转换过来了，实际的电压变换是从高电平变成低电平)。我们可以发现在某些点一次开关量被重复计数，即计数两次④。这是因为计数器可以发现信号中的每一个短时脉冲(甚至低于20纳秒的短时脉冲)。因此我们需要使用滤波器将这些短时脉冲信号过滤掉。我们可以使用最快的滤波器，5毫秒，这是因为我们按开关的速度确实很慢。

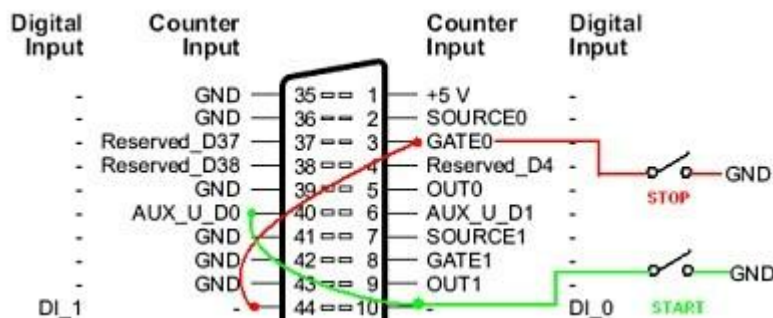


2.6.1.2 门事件计数

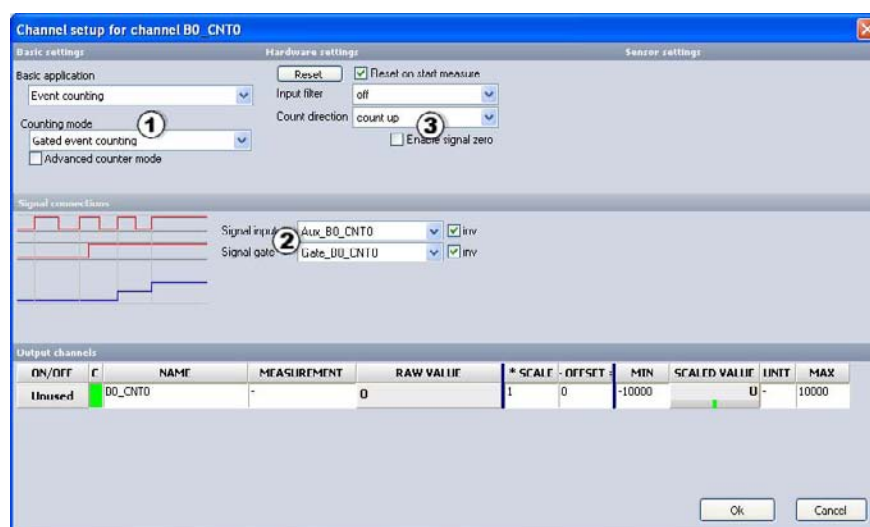
门事件计数模式指的是计数器只在门信号为高电平的时候计数。这种门事件计数方法只能用ORION计数器才能实现。我们的教程继续从前面“简单事件计数”一章接着讲。



计数器信号与简单事件计数信号完全一样，但是我们增加了一个额外的信号 – 即门信号。因此我们把第二个开关连接到 GATE 0 上，在连接到 DI_1 上，从 DI_1 上我们可以观察这个信号值(当然真正测量的时候没有必要这么做，在这里我们只是为了演示更加直观)。



接下来我们对计数器通道进行设置。我们首先要选择门事件计数Gated event counting模式①，将信号输入源signal input，设置为Aux_B0_CNT0通道，将信号门gate signal设置为Gate_B0_CNT0通道②。此后，计数器仅仅在当门信号的电平值为高电平时，才对信号从低到高转换时计数，我们同样选择将门信号进行高低转换，也就是说仅当按钮按下时才进行计数。

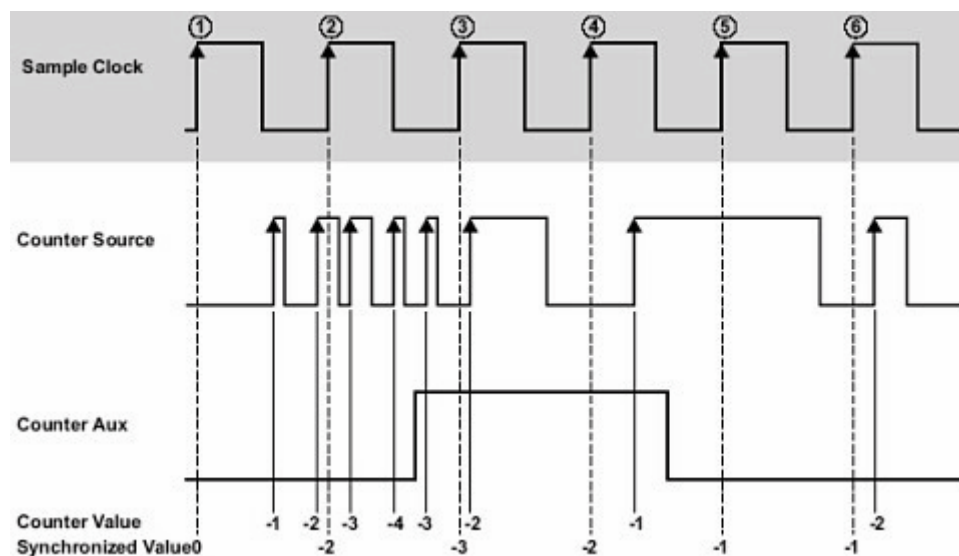


在下面所示的例子中，我们可以看到计数器的工作方式。黄色信号(计数器)仅仅在当门信号(蓝色)为高电平④的时候才对绿色信号从低电平转为高电平时进行计数。

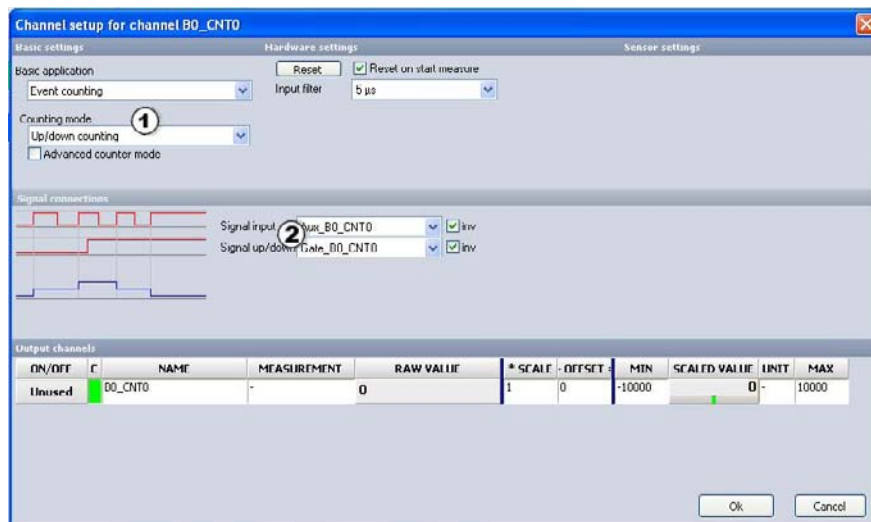


2.6.1.3 上/下计数

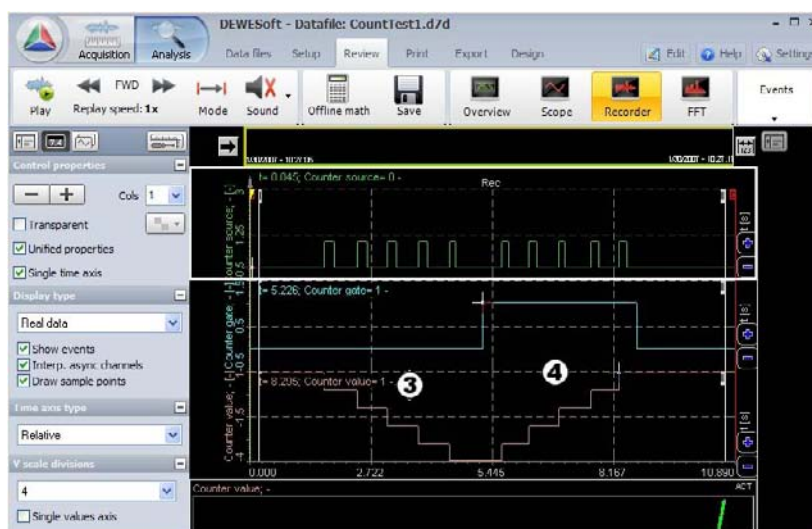
上/下计数模式指的是当门信号的信号电平为高电平时向上计数，当门信号电平为低电平时向下计数。解码器的运行方式与此类似，实际上可以使用这种模式进行X1 解码器测量，但是我们需要在前面加一点电子部件。



开关等的链接方式和前面的连接方式一样 – 即还是进行门事件计数。然后在设置中选择上/下Up/Down计数①方式，在选择输入input和上/下up/down信号②。



在测量屏幕上，我们可以看到当门信号(蓝色信号)为低电平时，计数器反向计数③，当门信号为高电平时，计数器正向计数④。

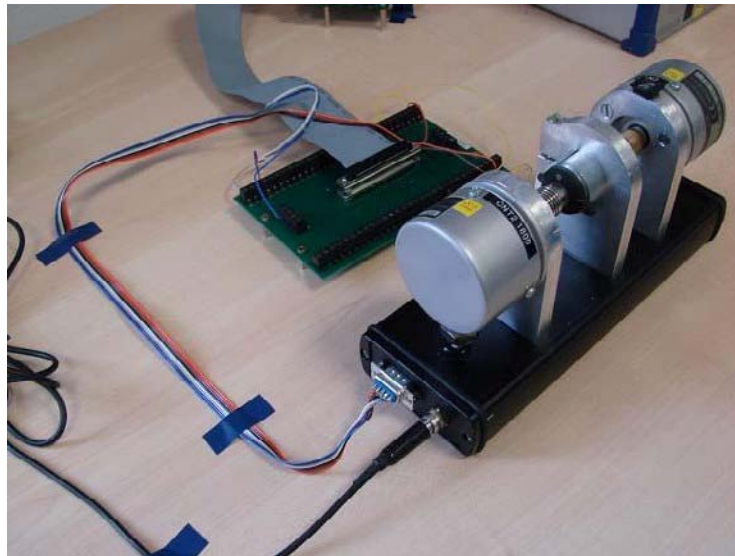


2.6.2 解码器

解码器是一种码盘(或者是一个带有黑白标记的条形码)，上面都带有条码。常用的解码器带有两个具有 90° 相位差的条码，这两个条码可以用来确定条码的运动方向，除此之外，还有一个零位置脉冲标记 – 它每转产生一个脉冲，可以用来确定解码器的绝对位置。

这两个信号，信号A和信号B用来确定转动或者是运动的方向。当信号A的输入领先于信号B时，计数器的值就一直增加，反之，信号B领先于信号A时，计数器的值就一直减少。

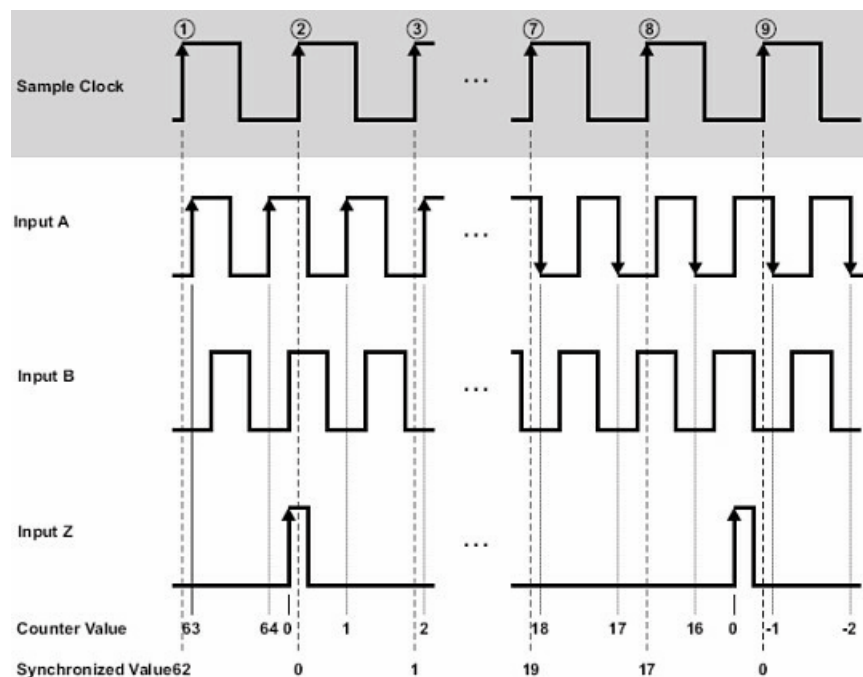




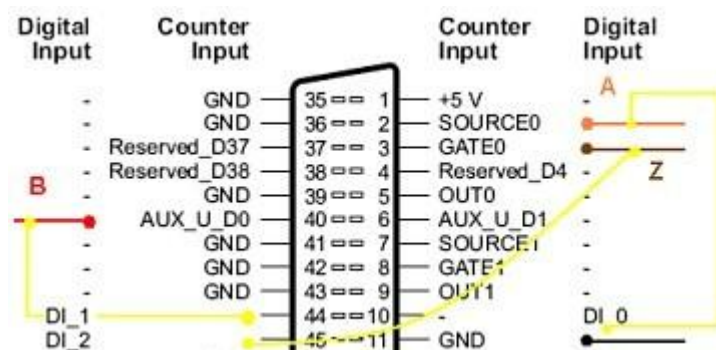
有几种解码器的形式 – 第一种是只测量输入信号A的上升沿 – 这种测量模式叫做**X1**模式。第二种 – 叫做**X2**模式，它只测量输入信号A的上升沿和下降沿，而**X4**模式则同时测量输入信号A和信号B的上升沿和下降沿。如果我们需要用线性的条形码测量慢速运动，用X2和X4模式极其有用，因为这两种模式会乘以一个2或者4，实际上大大增强了测量的分辨率。

所需硬件	只能用ORION卡，NI-MX 只支持某几个模式
所需软件	任意版本
设置采样率	至少 1 kHz

假如我们需要进行快速动态测量(比如扭振测量)，如果这时候使用x2和x4模式的话，则有可能引入更大的误差，因为这两种模式假设其占空比gap ratio为精确的0.5，而解码器的电子开关在暗和亮之间同样用这个转换速度进行开关动作。我们会在下一章节中，用脉冲周期和脉冲宽度测量来评价这种误差的大小。

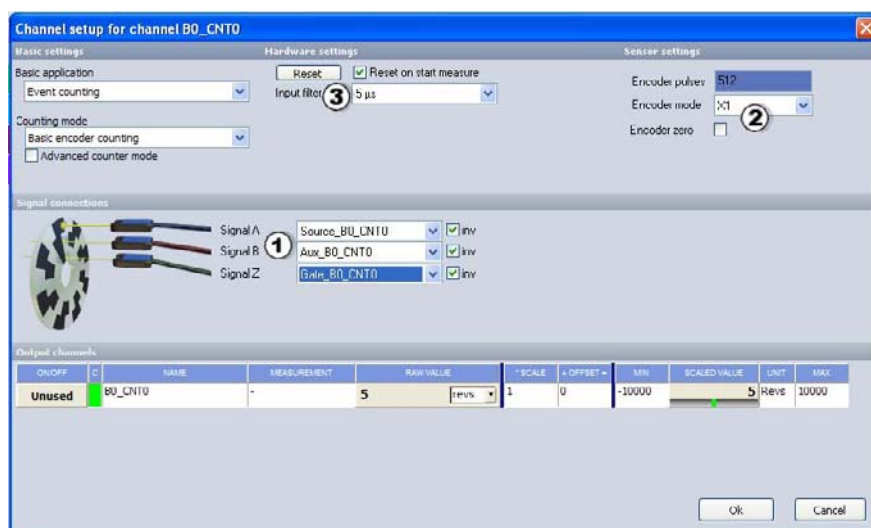


我们将解码器的信号A连接到SOURCE0，信号B连接到AUX_CNT0，零位置脉冲信号zero pulse连接到GATE0。另外，只是为了演示的需要，我们还把这些信号分别连接到DI_0，DI_1和DI_2上面，不过真正进行测量的时候无需这么做。

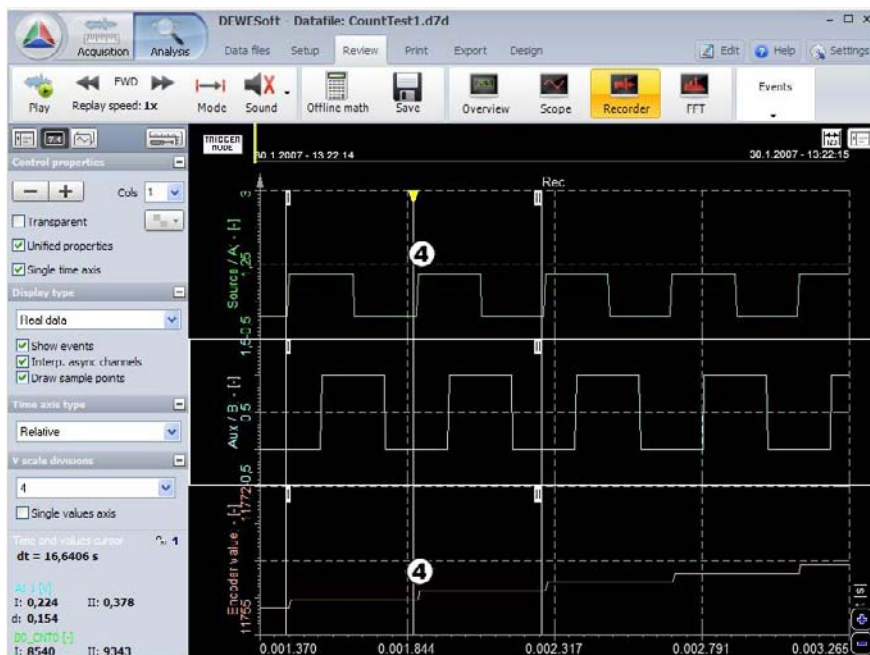


2.6.2.1 X1, X2, X4 模式

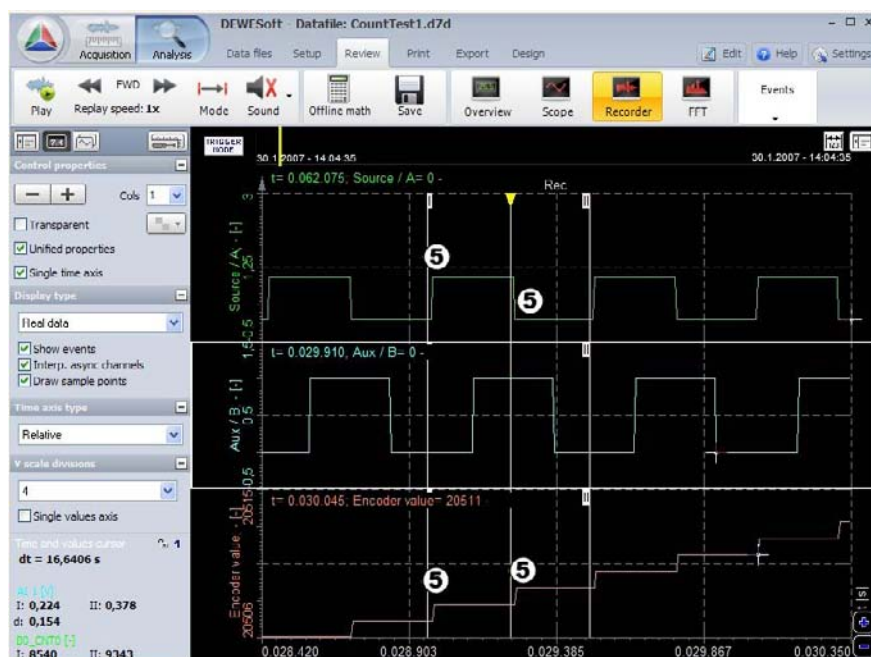
通过选择事件计数Event counting – 基本解码器计数Basic encoder counting来选择简单的解码器模式。首先设置解码器。分别将Source_B0_CNT0分配给signal A，Aux_CNT0分配给signal B，Gate_CNT0分配给signal Z①。将解码器模式Encoder mode设置成X1②，输入滤波器Input filter设置成和测量的最高频率③相匹配。在这里还可以输入量值变换因子scaling factor – 如果我们的解码器是每转512个脉冲，我们输入单位为转revs，然后输入512个计数值相当于1转rev。



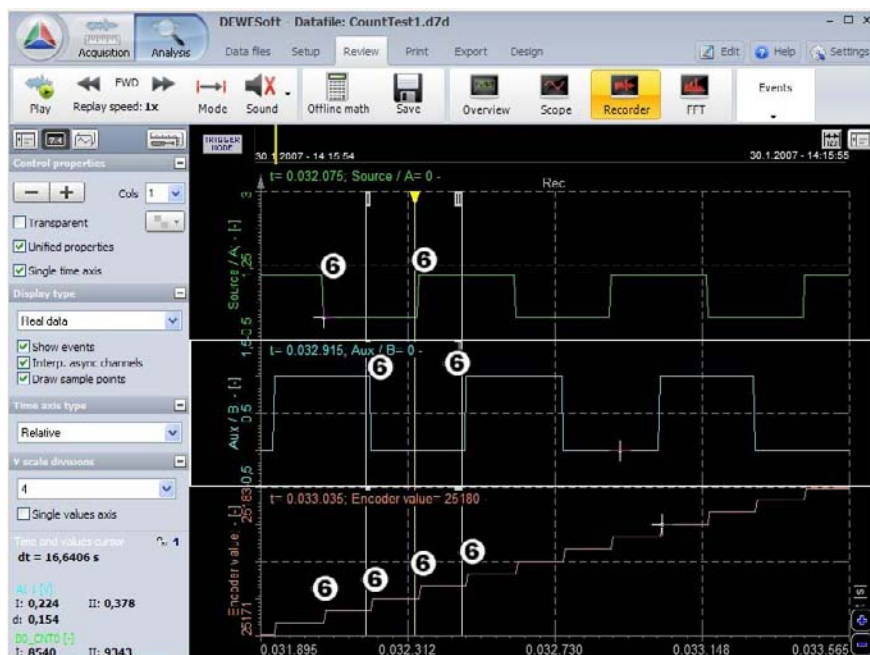
现在我们来实际测量。这个计数器的输出是这样的，当信号A领先于信号B时，向上计数，而当信号A落后于信号B时，则向下计数。信号A的上升沿用来计数④。



如果在设置过程中选择X2模式，计数器将对信号A的上升沿和下降沿同时计数⑤，因此其技术分辨率实际上增加了两倍。因此此时量值变换scaling也需要改变。不能再用每转512个脉冲，而应该是每转1024个脉冲。



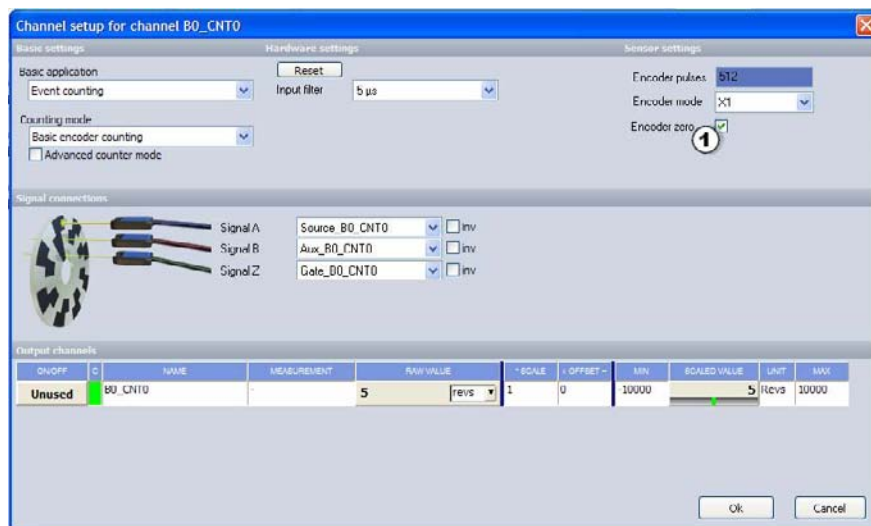
X4 模式，计数器将对信号A和信号B的上升沿和下降沿同时计数⑥。因此其技术分辨率实际上增加了4倍，因此此时量值变换scaling应该是每转2048个脉冲。



请注意阅读另外一章“Working with sensors”，进一步了解选择计数器传感器的其他方式。

2.6.2.2 带有零位置脉冲的解码器

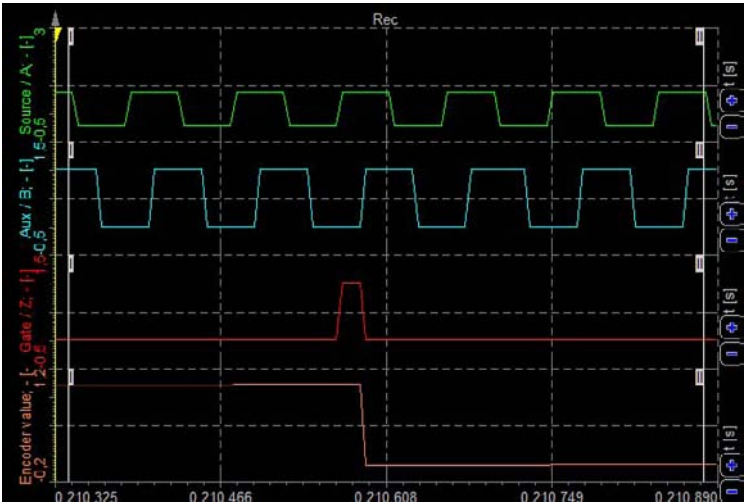
零位置脉冲通常用来重置一个测量(识别到Z脉冲后)。对以前的设置位移需要更改的地方是选取Encoder zero 复选框①。当零位置脉冲通过时，计数器值会重新置零。我们当然还需要设置用于内部计算的每转脉冲数(该例中为512个脉冲)。



下图所示为测量运行的情况。红色曲线是零脉冲信号，橙色曲线是解码器输出信号。当检测到零位置脉冲信号时，计数器的值重新设置为0②。



下图所示为记录仪中某部分的放大图。该图清楚显示当监测到零位置脉冲信号时，解码器计数值重新设置为0，然后重新计数信号A的上升沿。

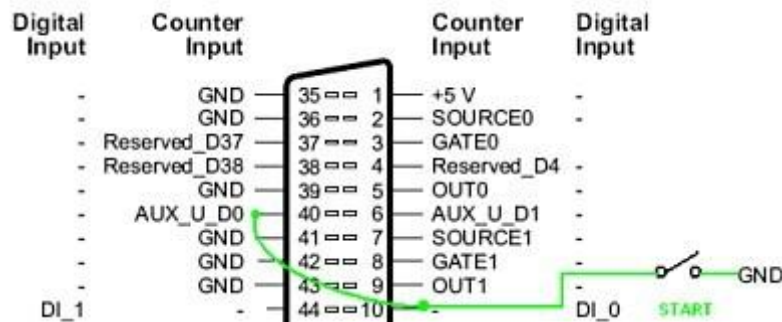


2.6.3 脉冲周期和脉冲宽度

脉冲周期和脉冲宽度的测量是两个比较相似的功能。脉冲周期的测量是指测量两个连续的低电平到高电平转换之间所花费的时间，而脉冲宽度测量则是指信号电平为高电平时的整个时间。

所需硬件	Orion 卡
所需软件	任意版本
设置采样率	至少 1 kHz

对于这个测量，我们将采用有两个开关按钮完全一致的设置。硬件连接很简单 – 只需将一个开关连接到AUX-CNT0，并同时把它连接到DI_0，用于监视其信号。



然后进入到通道设置channel setup屏幕，选择一个计数器，然后进入该通道的通道设置channel setup窗口。



我们还可以将脉冲周期和脉冲宽度测量和占空比测量结合在一起。

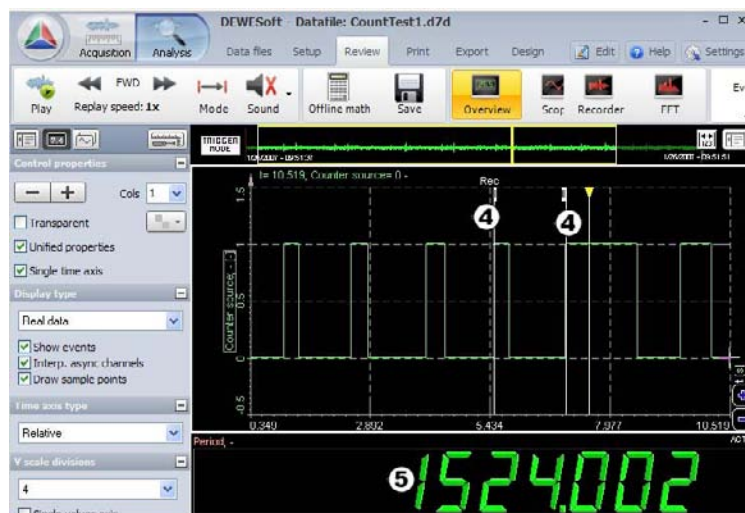
2.6.3.1 脉冲周期测量

首先我们要从Basic application选择Waveform timing，从Timing mode中选择Period, frequency①。将信号输入Signal input设置成Aux_CNT0通道，并且选取inv复选框，转换信号的正负，因此当按下开关时其信号电平变成高电平，而开关打开时电平变低；我们还可以设置信号输入滤波器Input filter，以隔离短时脉冲信号②。

这时会出现另外一个需要设置的域，我们用这个域来告诉当没有新的值计算出来的时候，软件应该如何动作。软件只在信号值从低电平转变成高电平时才计算出一个新的值。因此大部分时间里是不计算出新的值。如果我们选择的是当没有新的值计算出来no new value available时重复最后一个值repeat zero value的话，相同的值会被加入，直到出现一个新的低电平到高电平的转换。相反的，我们如果选择的是当没有新的值计算出来no new value available时输出最后一个值output zero value的话，我们只能在新数据点得到尖峰，而其他时间段内的数据都成为0③。

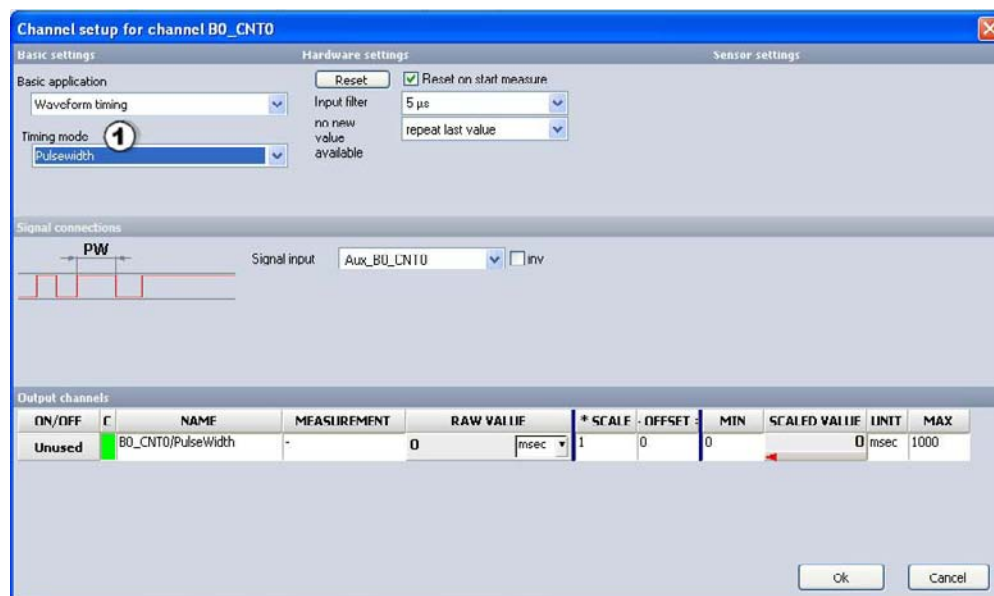


下面所示的例子中，显示的是如何进行测量的。记录仪上的两个白色光标显示的两个脉冲之间的时间差(时间差显示在记录仪左侧设置窗口屏幕上④)。这个读数是1.529秒。当然计数器计算出来的值更准确，它显示的值是1524.002秒⑤。因为我们的计数器时钟为80 MHz，所以我们的分辨率是在毫秒以下。除了周期测量之外，计数器同样能够计算频率，很简单，频率=1/周期。只需在设置窗口中设置好周期的单位以及在通道中设置频率单位即可。

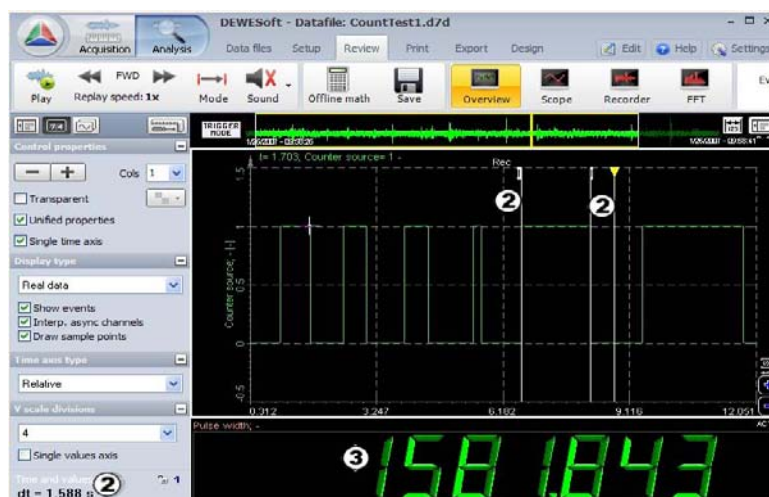


2.6.3.2 脉冲宽度测量

脉冲宽度测量的设置和前面所描述的脉冲周期的设置相同。位移的差别是在Timing mode中选择Pulsewidth①。



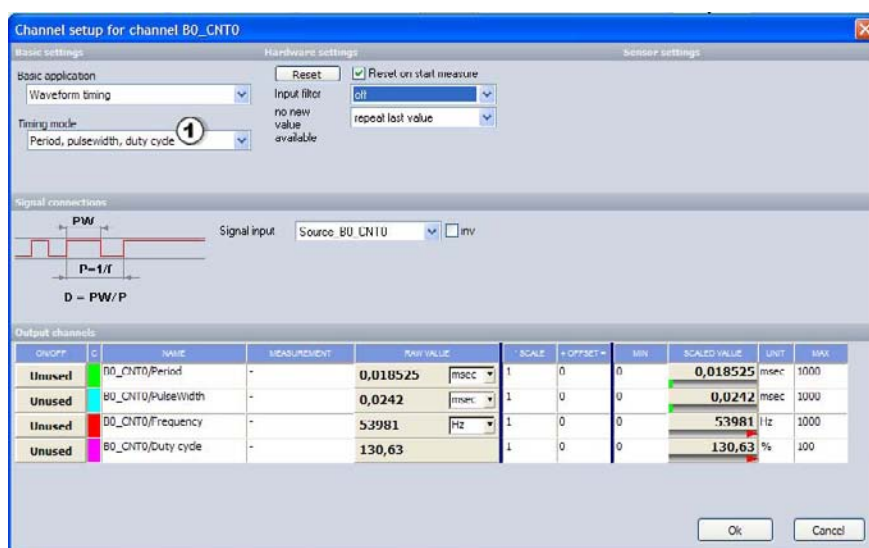
现在的读数计算则是按照从高电平到低电平转换时进行。记录仪光标的读数是 1.588 秒②，额计数器计算值为 1581.843 毫秒③。我们还可以使用超级计数器来测量每个计数器所处低电平和高电平时间，或者把脉冲周期和脉冲宽度测量和信号占空比测量结合在一起。



2.6.3.3 占空比测量

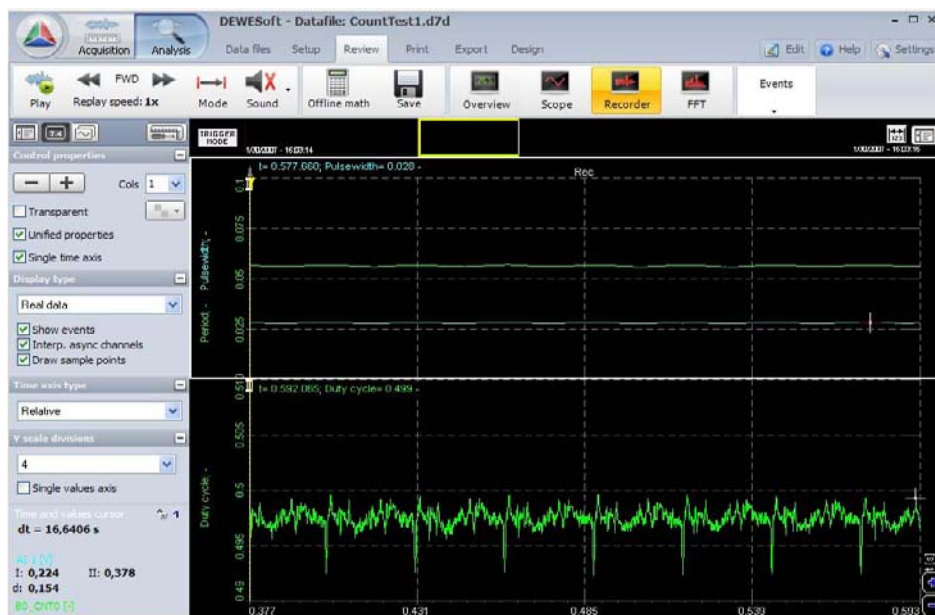
占空比指的是信号的高电平(或者低电平)所占时间和整个脉冲周期的比, 占空比测量就是测量占空比的整个过程。前面我们在解码器教程中讲过, 是否使用X1, X2模式还是X4模式, 取决于解码器和解码器电子电路的质量。现在我们还是继续使用在解码器教程中使用的相同解码器, 然后测量器质量。

对于此类测量, 我们需要将其设置成"Period, pulsewidth, duty cycle"模式。如果这种模式不能使用的话, 那就表明计数器不支持这种功能。要么它是一个超级计数器对(只有ORION卡具有这种功能), 要么它不是ORION卡。



接下来我们可以直接选择"Period, pulsewidth, duty cycle"模式, 以及频率输出通道。对于占空比测量, 计数器会自动进行设置。

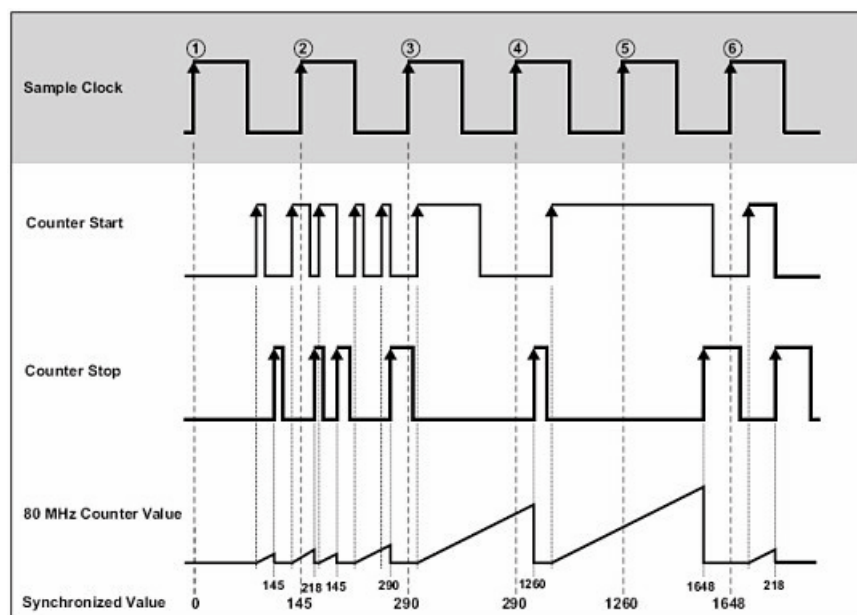
现在来看看占空比测量。图的上半部分所示为信号的周期和脉冲宽度, 下半部分所示为解码器几个转速的占空比。我们可以非常清楚地看到其中有几个点, 它们明显比其他数据的误差更大。在此处的值大概是0.49, 也就是说解码器X2模式大概误差在1%左右。



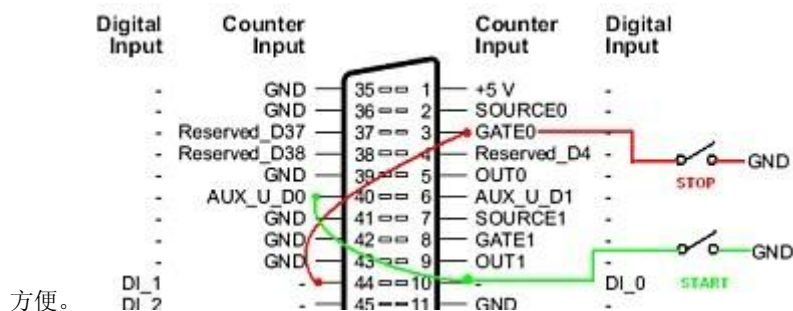
2.6.4 两个脉冲沿隔离

所需硬件	只能使用 Orion 卡
所需软件	任意版本
设置采样率	至少 1 kHz

两个脉冲沿隔离是ORION计数器特有的一个功能。它测量的是相邻两个脉冲信号的上升沿之间的时间差。这种应用对于需要高精度测量两个事件的间隔时间时非常有用。典型的应用是测量物体通过两个相邻光电传感器时的移动速度。

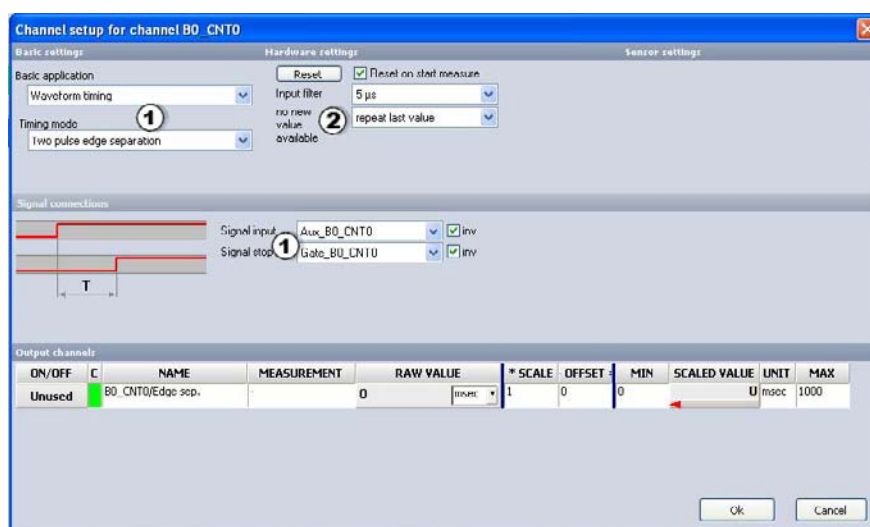


我们需要测量两个信号 – 其中一个作为开始信号，另外一个作为停止信号。在我们的例中，开始信号连接到AUX_CNT0通道上，停止信号连接到GATE0通道上。同时，开始信号还连接到DIO通道，停止信号还连接到DI1通道上，这样做只是为了演示的



方便。

要在DEWESoft软件中设置这种测量模式，必须打开一个计数器，然后进入设置窗口。在Timing mode域中选择Two pulse edge separation模式，在Signal input (开始t)中选择Aux_B0_CNT0通道，在Signal stop中选择Gate_B0_CNT0通道①。两个信号都要进行正负转换，因为我们的布线方式决定了按钮没有按下时其电平值是高电平。在我们的例子中，实际的触发条件实际上是输入信号的下降沿。选择没有新的值计算出来no new value available时重复最后一个值repeat last value(和脉冲周期测量一样)②。



现在开始进行测量。首先按下第一个开关按钮，然后按下第二个开关按钮。从光标读数，我们得到1.738秒③，计数器则更准确，其读数为1733.838毫秒④。



接下来，我们利用这个值用来测量速度值，如果我们从两个光电探头作为触发信号。如果我们确切知道两个光电探头之间的距离，那么速度值就可以利用公式 $v = s/t$ 计算出来，这里s是一个常数，t是两个脉冲沿分离测量得到的时间结果。利用数学通道即

可算出速度值。

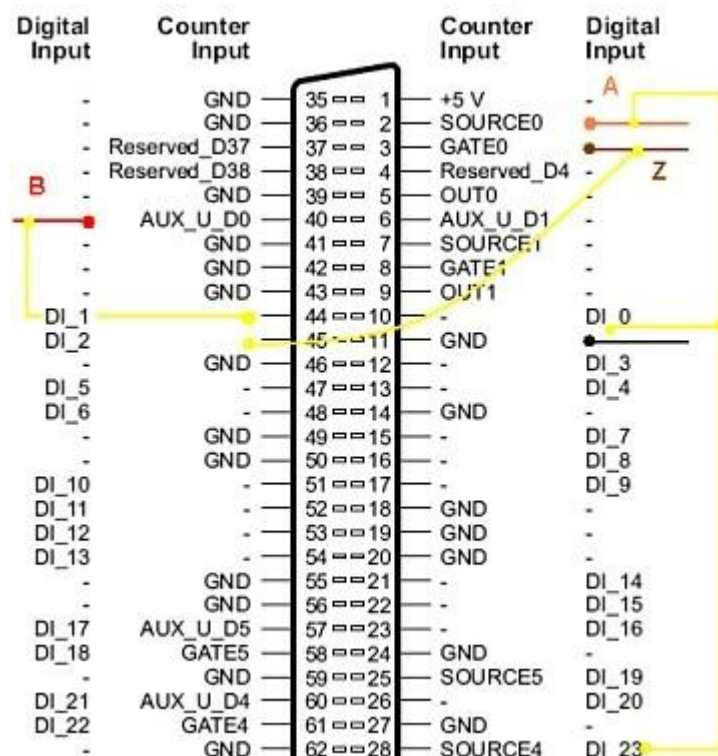
2.6.5 频率 / 超级计数器

所需硬件	只能使用 Orion 卡
所需软件	任意版本
设置采样率	至少 1 kHz

最后我们来讲一讲相对于传统的计数器，我们的超级计数器有哪些优点。

传统计数器的问题在于计数器的值仅仅在采样间隔内发生闭锁。因此我们只能得到每个采样样本的离散值。但是由于ORION卡的第二个计数器可以精确测量脉冲处在两个采样点之间的确切位置，所以我们能够借此计算出两个东西：在两个采样点之间的准确计数器插值位置，以及脉冲的准确频率。

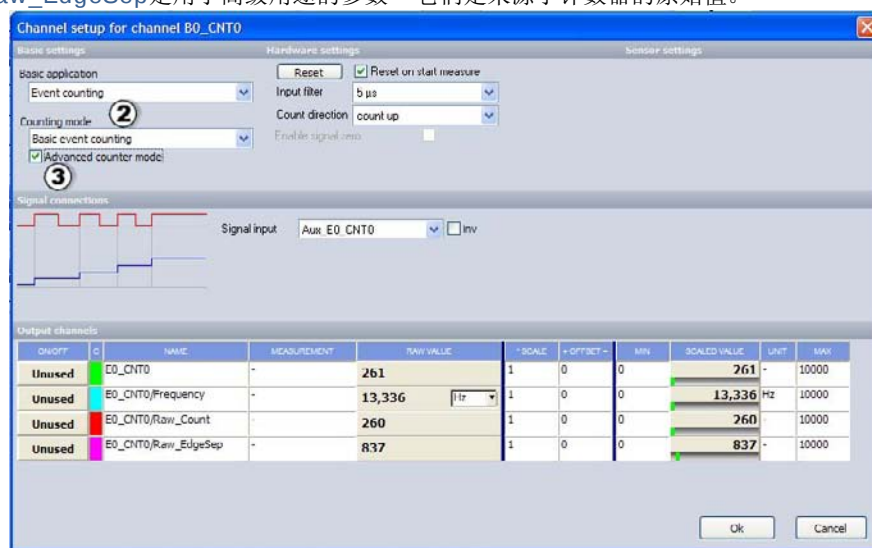
因此我们建议首先理解一下这种计数器的工作原理，然后才能更好的了解这种工作模式。硬件的设置如下：我们将信号(可以是来自于解码器的信号)连接到SOURCE0通道。我同时还把它连接到COUNTER4输入，这样可以用它来和普通的计数器进行比较。



接下来我们来设置通道。对于超级计数器以及频率测量，我们都需要用到两个计数器。另外一个限制是计数器通道必须是偶数通道(CNT0, CNT2 或者CNT4)。这些通道都在计数器名称旁边做了标记①。计数器对是两个相邻的计数器，比如CNT0和CNT1，必须同时成对使用。



计数器0设置成简单的事件计数器模式Events Counter mode，唯一要做的事是检查是否选定"高级计数器模式Advanced counter mode"。选定后，计数器会自动设置好，我们就拥有了精确的计数和频率通道，作为计算输出通道。其中Raw_Count和Raw_EdgeSep是用于高级用途的参数 – 它们是来源于计数器的原始值。



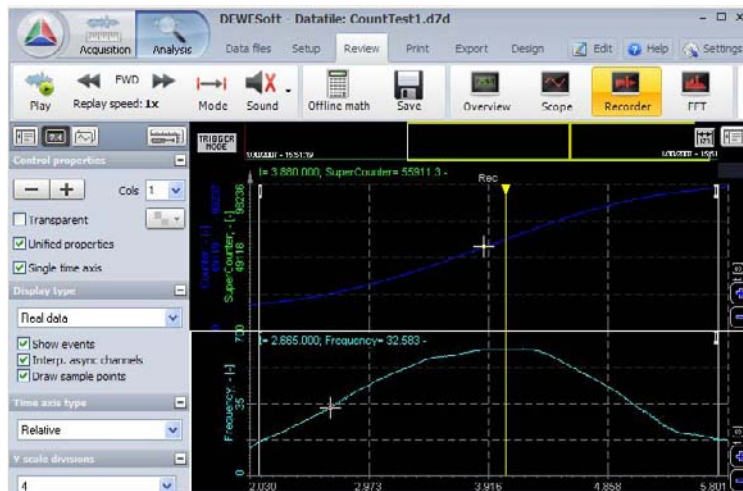
我们把Source0信号显示成数字线，在第二张图中，蓝色曲线是普通计数器的计数曲线(原始计数器值)，这种计数器每采集到一个样本数字增加一位，红色曲线是超级计数器的计数曲线②，这种计数器在各个计数之间进行插值，最重要的是，在样本间也进行插值。

如果您是按照这个教程自己进行试验，建议将信号频率设置成从采样率的一半一直到高于采样率的50%以上。这样就能看到普通计数器在一个样本中首先保持不变，然后突然跳一个数，然后在保持不变，再往上跳一个数。简而言之，结果非常差。但是如果用到超级计数器，计数器的值将和输入信号完美对齐，如下例所示。

还有频率的测量同样非常完美。



下图所示的数据文件显示的是转动机械测试过程中的一个完整升速和降速实验过程，在此处的超级计数器显示出完美的测量结果。所以我们强烈建议在高级DSA测量中，比如阶次跟踪，扭振测量，旋转振动分析中采用这种测量方法。



我们也可以使用其他AD板卡，DAPQ模块或者PAD-CNT模块来做同样的测量，但是结果会更差。请参考[频率测量教程](#)，比较不同测试方法之间的区别。

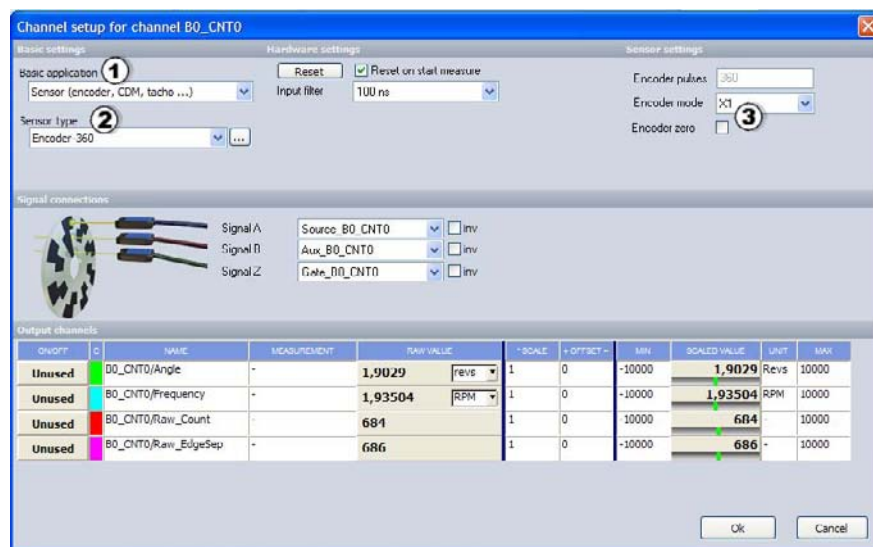
2.6.6 传感器模式测量

所需硬件	只能使用Orion卡
所需软件	任意版本
设置采样率	至少 1 kHz

超级计数器模式还能用一种特殊的计数器模式，叫做“传感器Sensor”模式(从[基础应用Basic application](#)下拉菜单中选择①)。这种模式允许我们直接使用在[计数器传感器编辑器Counter sensor editor](#)中定义好的数字速度/位置传感器。我们可以选择旋转编码器，线性条形码编码器，CDM传感器(带有0位置参考的角度传感器)，带有一个或两个缺齿的齿轮以及光电探头。

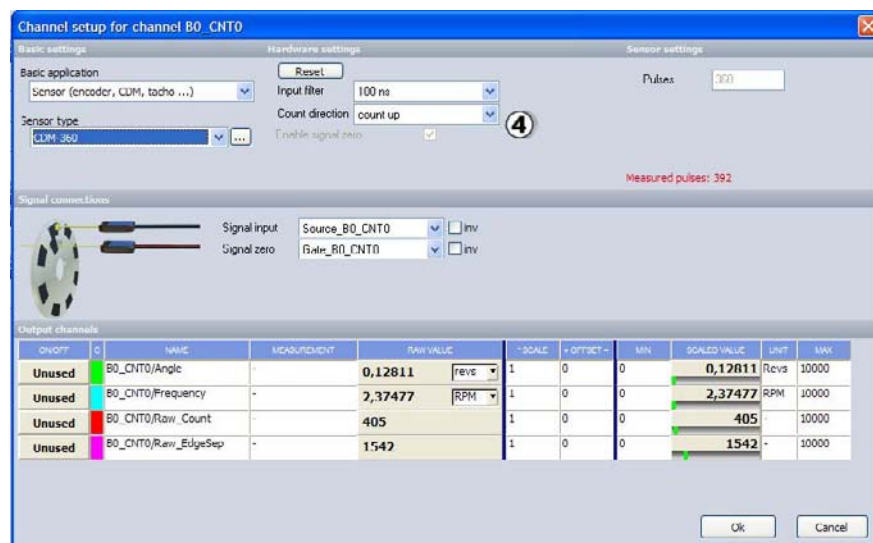
唯一需要做的事情是从[传感器类型Sensor type](#)下拉列表②中选择合适的传感器。如果传感器还没有定义好，在下拉列表旁边有一个由三个椭圆小圆点组成的图标，点击该图标，打开计数器传感器编辑器，在此处编辑定义新的传感器类型。传感器将永远运行在超级计数器模式，显示准确的频率和角度。

使用传感器模式的好处是量值转换全部是自动完成的，因此我们不用担心任何事情。还有许多选项可以加以选择。对于解码器，我们能够选择[解码器模式Encoder mode \(X1, X2 或者 X4\)](#)③，还可以决定是否选择[Encoder Zero](#)。我们还可以决定计数的方向([Count direction](#))。



如果用到了解码器的零位，就会有一个额外的信息高速我们在两个0位之间有多少个脉冲数，从这个脉冲数中可以告诉用户大致的设置或者可能存在的链接错误(如下图所示)。

对于CDM传感器，唯一特殊的设置是计数方向。CDMA传感器默认状态下只能往上计数，但是我们可以用inv选项来转换计数方向。



光电探头和齿轮传感器没有特殊的设置，不过这两种传感器仅仅依赖于传感器在传感器定义时的设置。

2.6.7 车辆上的计数器

所需硬件	Orion, NI MX 卡, 只用在基本测量
所需软件	任意版本
设置采样率	至少1 kHz

我们现在来看一下车辆上常用的两种应用：方向盘测量和轮速传感器。

方向盘传感器

在我们的例子中，会用到一个带有积分式解码器传感器的测量方向盘，用来测量被测车辆的方向盘角度位置和角速度。我们使用的积分式解码器分辨率率为每转1800 脉冲。

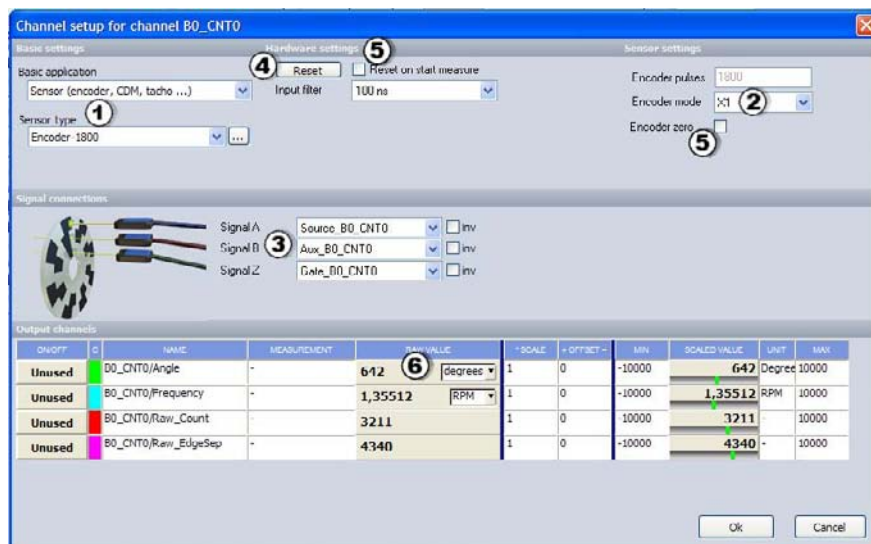


通道设置

我们首先来设置方向盘传感器。

对于计数器模式Counter mode，我们从基础应用Basic application下拉列表中选择将其设置成"Sensor (encoder, ...)"，用来解码积分式编码器传感器的信号，从传感器类型Sensor type①下拉列表中选择合适的传感器(Encoder)。在Encoder mode域中，可以设定角度测量的分辨率。我们的例子中，选择的是"X1" ②，计数器每转输出1800 个脉冲。如果选择模式，则每转输出3600 个脉冲；如果选择模式，则每转输出7200 个脉冲。建议使用输入滤波器Input filter来避免由于信号(本例中 为100ns)不稳定或者尖峰造成的测量误差。积分式编码器传感器通常输出三个信号。在这个应用中只用到了Signal A和Signal B ③。

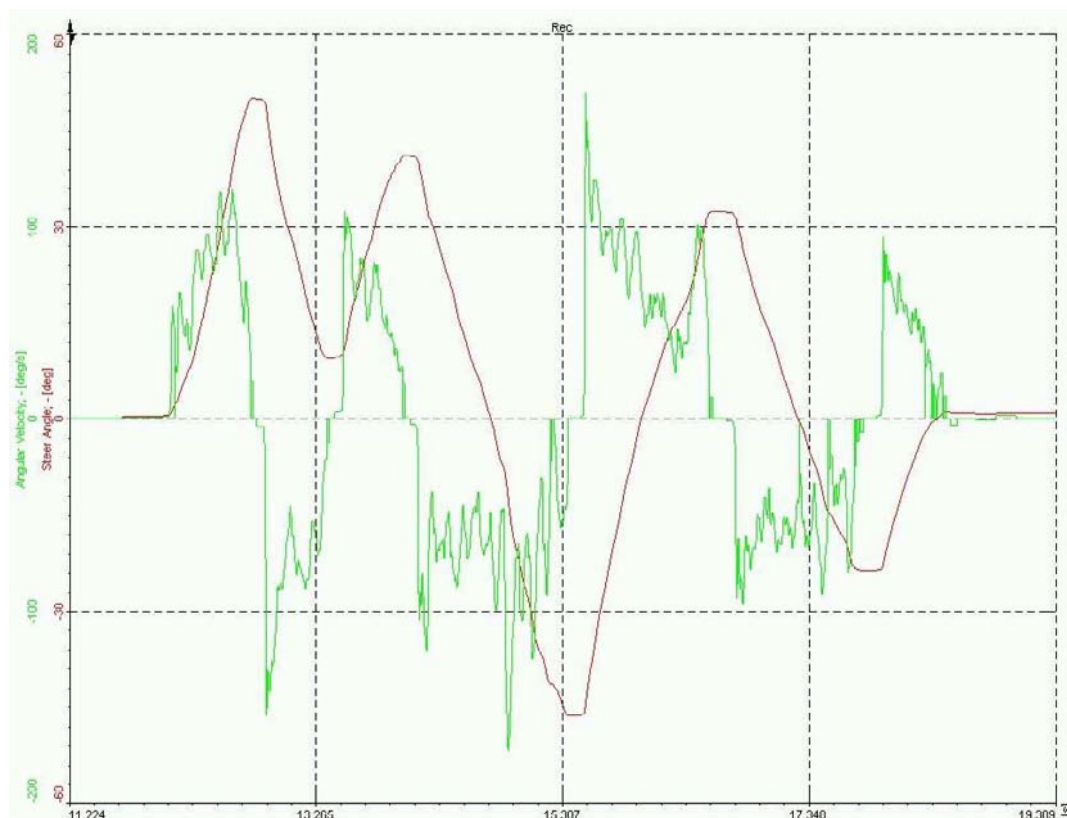
其中重要的一点是不需要选择Encoder zero选项。Signal Z连接到计数器输入上面。否则在频率通道中会出现跳动的情况，并且手动定义的零点会丢失。最好的方法是从列表中简单选取传感器，然后所有的量值转换全部由软件自动完成。必须取消选取Encoder zero选项，而且也不能选择Reset on start measure选项，保证不丢失初始零点校正。



第一个输出的是角度，我们可以选择“度degrees”为单位(当然也可以使用转速revs或者计数counts)，第二个输出的是频率，单位已经设置为RPM或者Hz。

零点定义

等到所有通道设置都完成后，方向盘的某个特定角度必须设置成零位置点。在大多数应用中，车辆直行情况下的方向盘转角设置成零位置点。可以通过在通道设置窗口中点击“Reset”按钮来定义零位置点④。进行这个动作之后，方向盘在该位置的角度设置成零位置点。最好是在一条非常平直的测试道路上来定义零位置点，这样可以避免出现方向盘角度误差。非常重要的一点是，不能选择“Reset on start measure”⑤选项，因为如果选择了该选项，在每次测量时，计数器值重新设置为0，定义的零点位置就丢失了。下图所示为方向盘测试的一个结果。

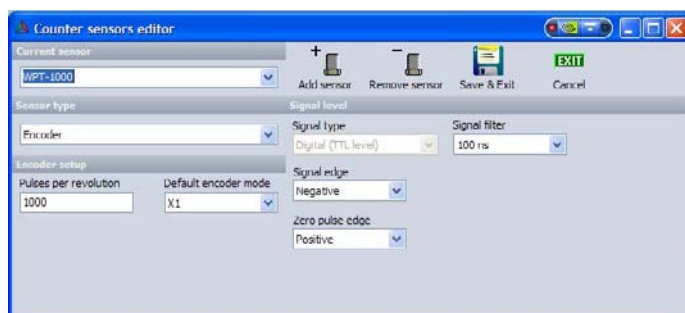


轮速传感器

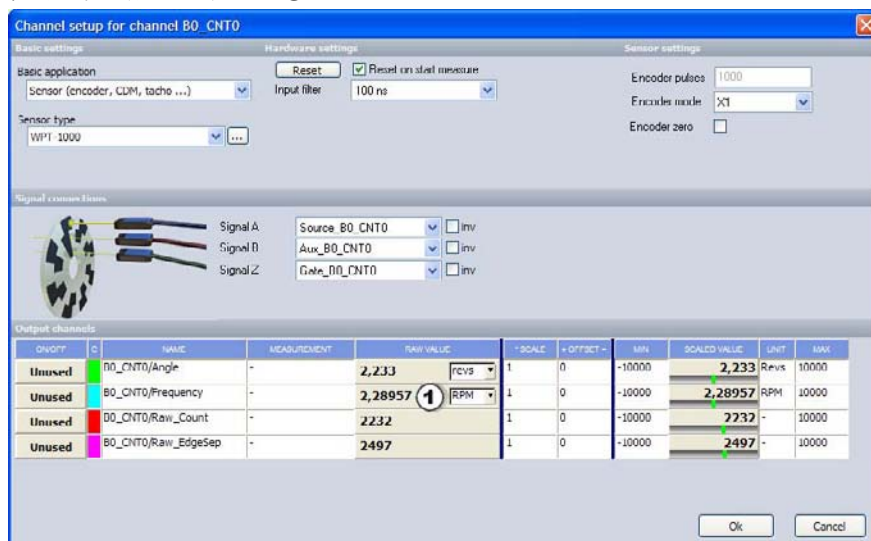
轮速传感器通常用在汽车高精度轮速和旋转测量中。我们还可以测量汽车行驶位置和速度，但是这种应用需要我们清楚轮胎动态直径的尺寸。不过，轮胎的直径又和轮胎的运行状态有关系，比如路面状态，驾驶情况，转弯能力，轮胎压力以及因素有关。因此对于精确位置测量，不建议使用动态轮胎直径的方法，我们推荐的是GPS解决方案，比如VGPS，或者是雷达/光电式传感器解决方案。



我们又需要对轮速传感器进行通道设置`channel setup`。这种传感器在DEWESoft软件中没有作为标准传感器定义好，因此需要我们在计数器传感器编辑器Counter sensor editor中重新创建。



传感器创建完成后，只能在传感器编辑器中加以选择。角度输出`angle output`可以设置成转`revs`，度`degrees`，计数`count`。频率`frequency`单位可以设置成Hz或者RPM①。

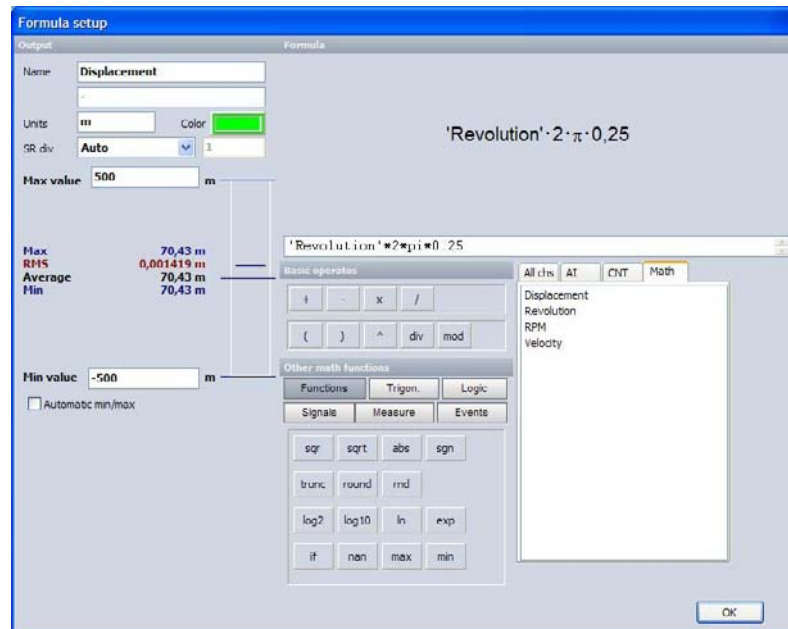


计数器设置完成后，接下来进行行驶位移`traveled distance`和行驶速度`velocity`的数学通道的设置。第一个数学通道用来计

算位移。这些是用轮胎动态直径来计算的。

位移计算的公式是：

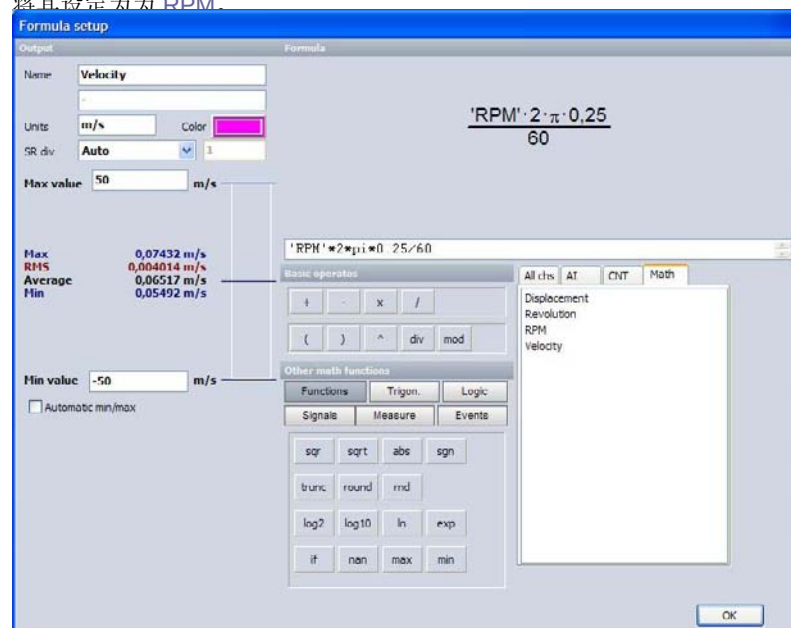
'B0_CNT0/Angle' $\cdot 2 \pi \cdot 0.25$ [m] 'B0_CNT0/Angle' 来自于计数器的角度通道输出。在本例中轮胎的动态直径是 **0.25 m**。



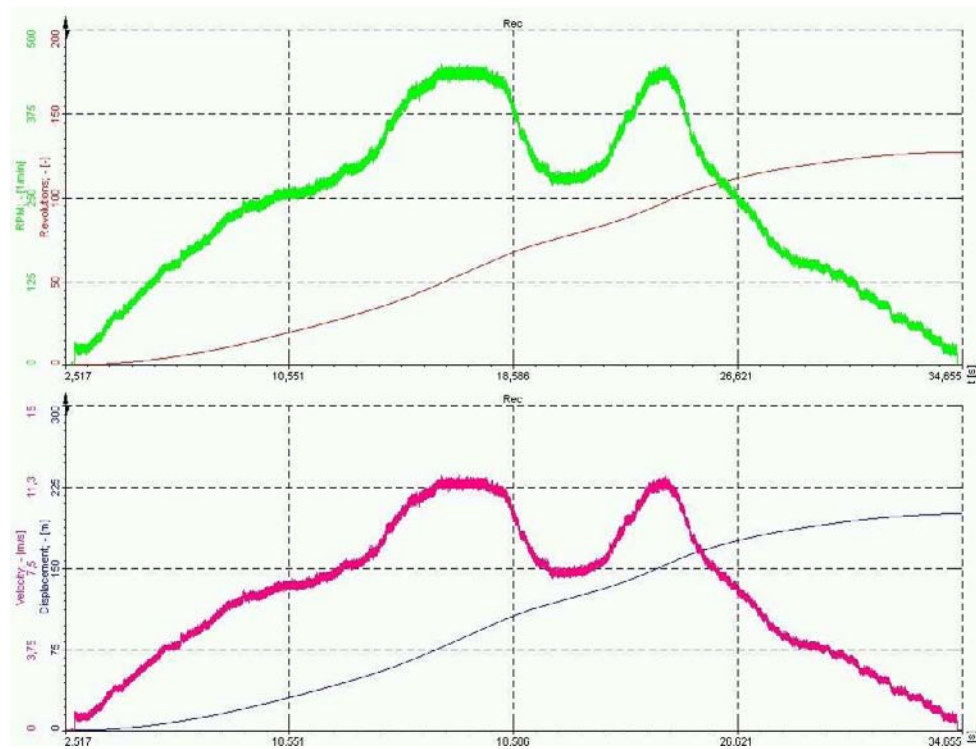
第二个数学通道用来计算车辆行驶速度。同上面一样的轮胎动态直径也用在这里的计算中。

车辆行驶速度的计算公式如下：

'B0_CNT0/Frequency' $\cdot 2 \pi \cdot 0.25/60$ [m/s] ... 'B0_CNT0/Frequency' 来自于计数器的频率通道输出，加入将其设定为为 RPM。



所有通道 – 不管是测量的通道还是计算的通道 – 都可以实时显示，无需任何硬件或者额外的布线，下图所示为轮胎传感器的测量结果。



2.7 频率测量

DEWESoft 支持三种测量频率的模式：基于计数器的模式，用 DAQP-FREQ-A模块的模拟模式以及使用PAD-CNT2的串行模式。

在前面的章节中，讨论频率和超级计数器测量的时候，我们已经了解了如何设置基于计数器的测量。现在我们还继续采用这种方式进行测量，来了解一下这几种模式之间的具体差异。现在我们使用同样的连接方式进行连接。

我们使用相同的电机接上解码器，我们同时把相同的信号送入DAQP-FREQ-A模块和PAD-CNT2模块。因此，基本上讲，这只是个TTL 脉冲信号。



我们可以确定如下事实：

- 当我们需要最精密的测量时，而且输入信号已经调理好了，我们可以采用计数器的方法(比如阶次跟踪，扭振测量)
- 当我们需要接入变化的输入触发信号电平，而且输出的延迟对测量来讲不是一个大问题的话，我们客户以采用模拟FREQ-A模块测量方法
- 当我们测量的频率信号变化非常缓慢，而且带宽，输出延迟根本不重要的话，我们可以采用PAD模块

所需硬件	Orion卡，DAQP-FREQ-A 模块或者PAD-CNT2模块
所需软件	SE, PROF, DSA
设置采用频率	至少 1 kHz

接下来让我们看看如何进行测量设置，以及详细阐述我们是如何得出上述结论的。

2.7.1 通道设置

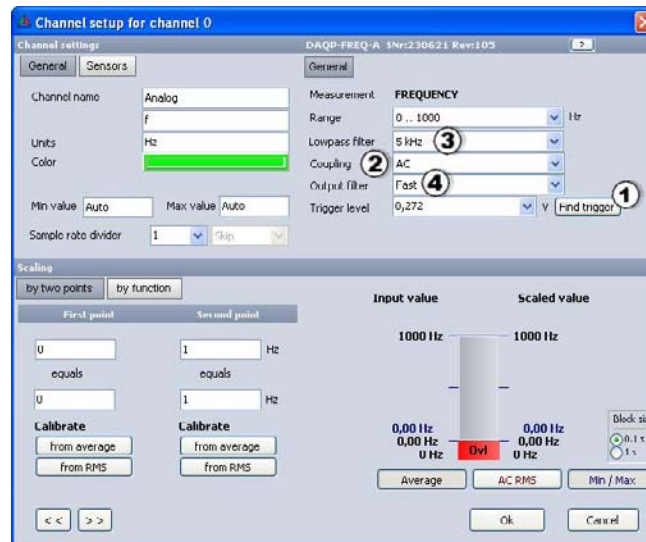
首先我们需要设置模块。在**模拟Analog**标签中，我们这里有两种模块 – 一个是模拟DAQP-FREQ-A模块，另外一个为PAD模块。

SLOT	ON/OFF	C	NAME	AMPLIFIER (001)	PHYSICAL VALUES	CAL	SETUP
0	Used	Analog	DAQP FREQ A	SN: 226/19	0V	Zero	Set ch. 0
1	1 chnl	AI 1	PAD-CNT	Frequency	0	NA	Set PAD 1

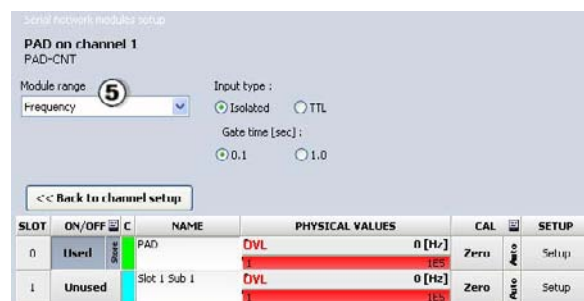
首先我们设置FREQ-A 模块。DAQP-FREQ-A模块的最大优势是可以输入变化的触发电平信号，这个电平信号可以是任意触发

级值。另外，它还有一个寻找触发值 **Find trigger** 按钮①，该按钮会自动测量输入信号的最大值和最小值，然后将触发电平调整到两者的中间值。我们还可以使用 **AC Input Coupling**②，切除信号的 **DC** 直流成分，因此当 **RPM** 发生变化时，补偿任意偏置。

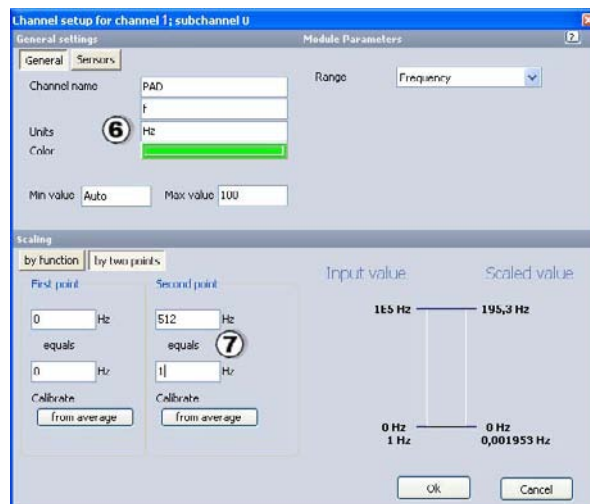
触发电平设置好后，我们接入一些输入信号，我们就可以采用这些设置才进行测试了。我们需要设置输出单位 **Output unit** 和 解码器脉冲 **Encoder pulses**。在我们的例子中，我们输入 **512 个脉冲每转**③。这时 **量程Range** 输入域内的数字变成曲轴的真实转动频率。低通 **Lowpass** 滤波器④实际是一个数字滤波器，它可以帮助阻止双触发(比如电涌)。接下来我们选择输出滤波器 **Output filter**⑤，它用来确定带宽以及输出频率的延迟。



下一步点击 **PAD** 模块的设置。基本的输入框跟我们以前见到的 **PAD** 设置有所不同。我们需要改变 **模块量程Module range** (这个域在这个地方看起来更像模块功能)⑥，将计数器变成频率，然后在第一个通道上点击设置 **setup**，这些设置在下面的测量中会用到。



PAD 模块的通道设置 **channel setup** 相当简单 – 我们只需输入测量单位⑦，调整 **Scaling** 因子即可 – 本例中，**512 Hz** 的输入信号等于曲轴只转了一圈，即 **1 Hz** 转动速度⑧。

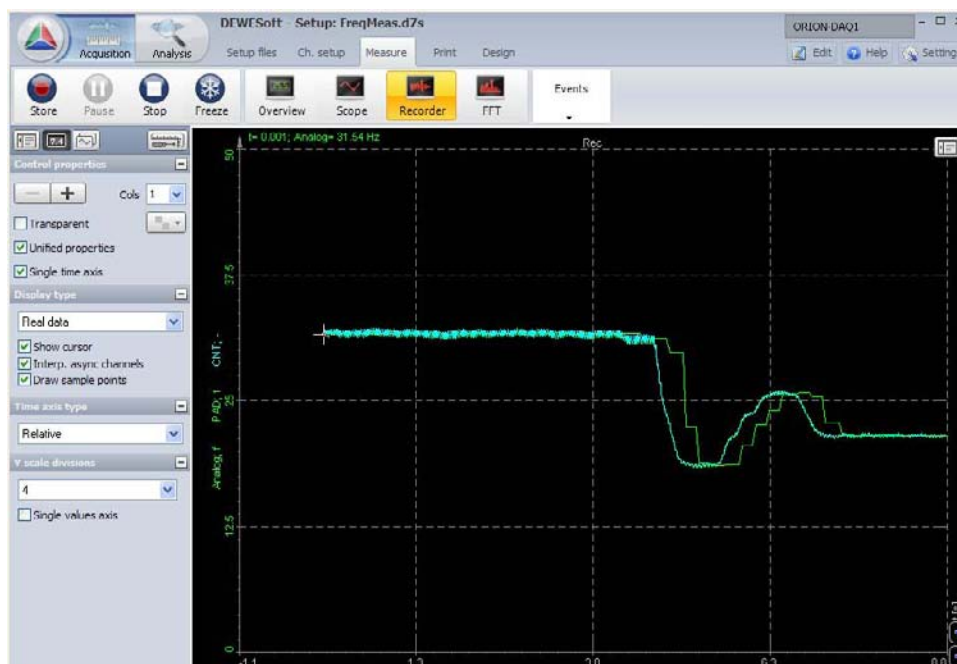


接下来我们来设置ORION 计数器对。第一个通道通过程序控制将其设为事件计数Event counting，第二计数器则设为频率Frequency。因为我们在前面的章节中已经详细讲过了，所以在这里我就不多讲了。如果您还是不知道如何设置计数器通道，可以参考计数器Counter教程中有关频率/超级计数器Frequency / super-counter的章节。

ON/OFF	C	NAME	VALUE	SETUP
B0_CNT0		Event counting mode		
Used		CNT 0	31	Events
B0_CNT1		Waveform timing mode		
Used		CNT 1	51,7 kHz	Frequency

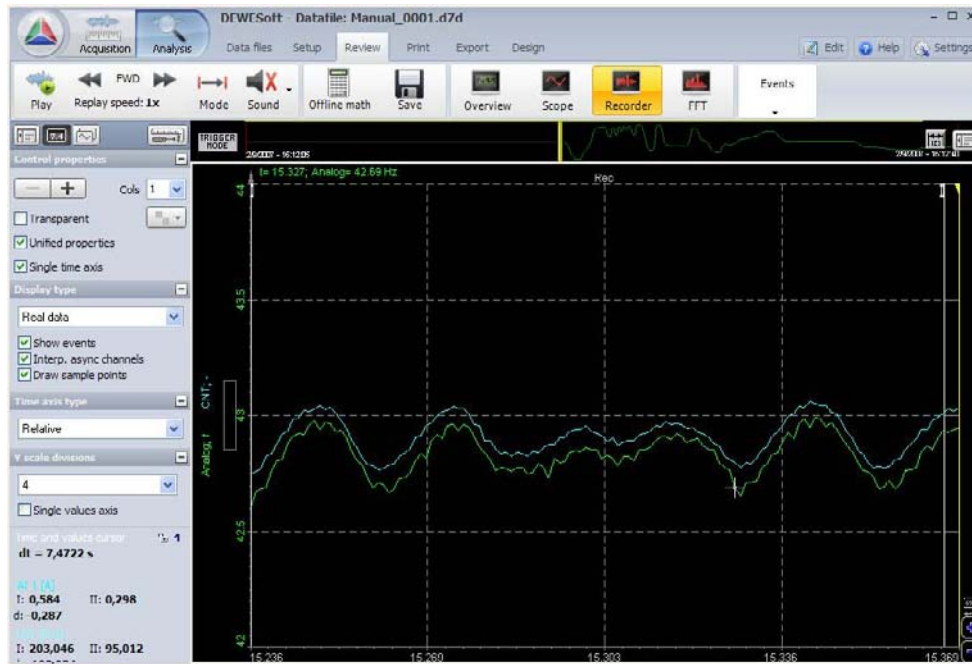
2.7.2 解码器测量

所以我们现在开始用解码器来进行测量。将记录仪设置成在一个显示图形中显示所有三个通道的值。我们首先就能观察到PAD模块(绿色曲线)延迟时间很长，每隔一秒才更新一次测量值。而其它两条曲线(模拟和计数器)对齐情况非常好。



如果我们存储测量，然后分析数据，在记录仪中放大，我们可以看到模拟曲线(也显示为绿色)和计数器曲线(蓝色)仅仅在放大倍数很高的情况下才会出现微小的时间和幅值差距。我们可以相信计数器测量更加准确，因为它仅仅依赖于基本时钟的精度。模

拟值是错误的，是因为电路内部误差，以及AD卡的测量误差造成的。它同样存在延迟，但是在本例中使用解码器时延迟最小。

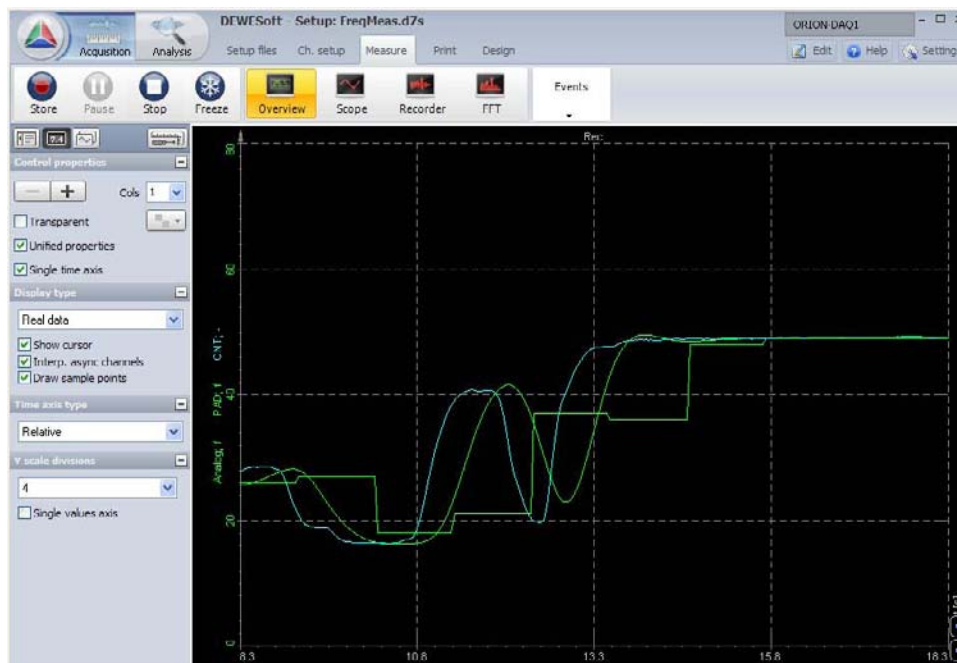


要成为最好的频率测量技术代表(只是想确认是否还有人读这些东西)，我们的竞争对手面临的第二个挑战是从每转一个脉冲的条件下来进行频率测量。

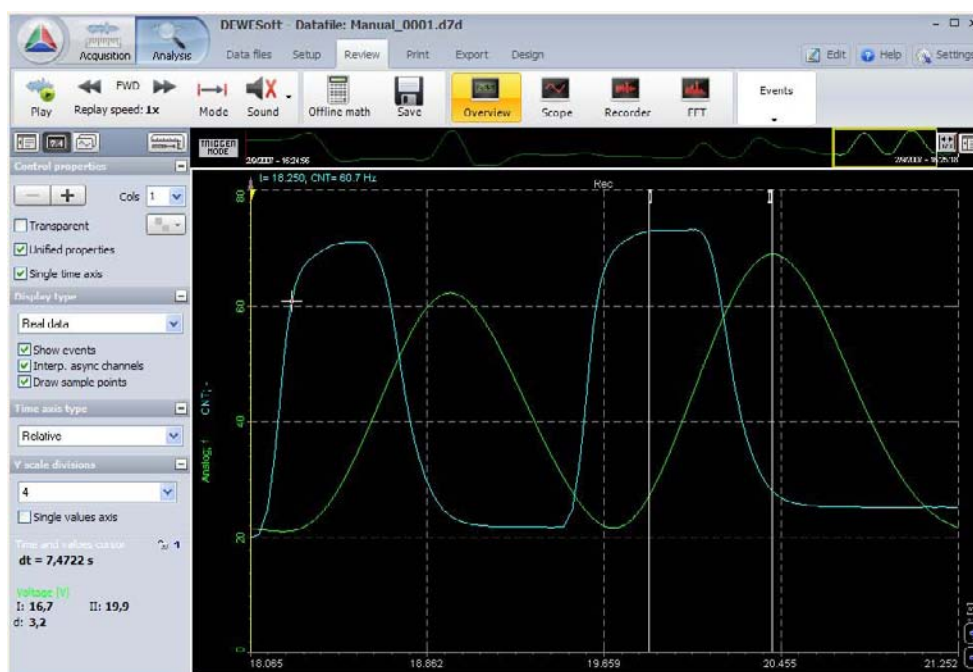
2.7.3 光电探头测量

因此，用解码器或者光电探头(每转一个脉冲)测量是否更容易一些呢？有的人可能会说测量光电信号更容易一些，但是这种说法并不正确。解码器测量需要获取更多的信息量，因此与采用每转一个脉冲的光电探头测量相比，其造成的测量延迟相对要小的多。

所以我改变了一下设置，所有三个通道的传感器都设置为光电探头。这时我们在来看一下测量，马上就可以发现PAD曲线(方形 绿色曲线)以及模拟曲线(平滑的绿色曲线)与计数器测量曲线相比，他们产生的延迟要大得多。



如果我们在分析模式中观察，我们可以发现这个延迟大概要到半秒钟，并且信号仍然很平滑，并没有反映出真实的信号情况。因此我们可以得出结论，测量需要反应快速时，最好只使用计数器测量方法。



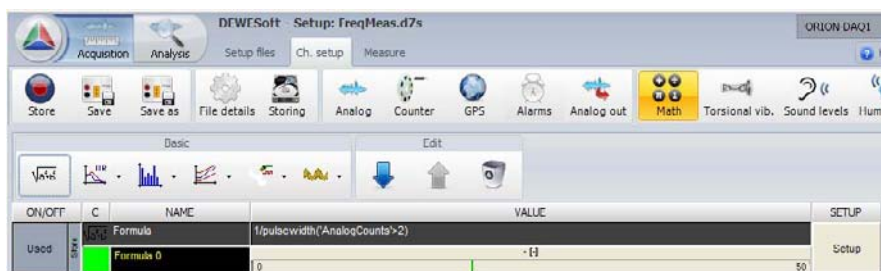
不过，和平常一样，当提高测量结果时，有一点点小窍门。对于一些使用光电探头的应用情况，DAQF-FREQ-A模块真的是非常有用，因为它可以切除信号中的直流DC部分，而且具有可变触发电平功能。

简单了解一下DAQP-FREQ-A模块的用户说明书，可以发现模块的第8根针是一个调理后的触发信号输出针。因此当发现触发信号后，这根针将会以模拟信号传递一个电压信号出去。我把这个电压信号接到另外一个电压输入通道上(因此我们在这个测量中有两个模拟通道)，这样的话，我们就得到了一个转动开始时的一个参考信号。这点在阶次跟踪分析或者其他动态信号分析等应用中非常有用。

另外，我们还可以使用数学通道来测量频率。频率是周期的倒数。因此我们可以利用下面的数学公式：

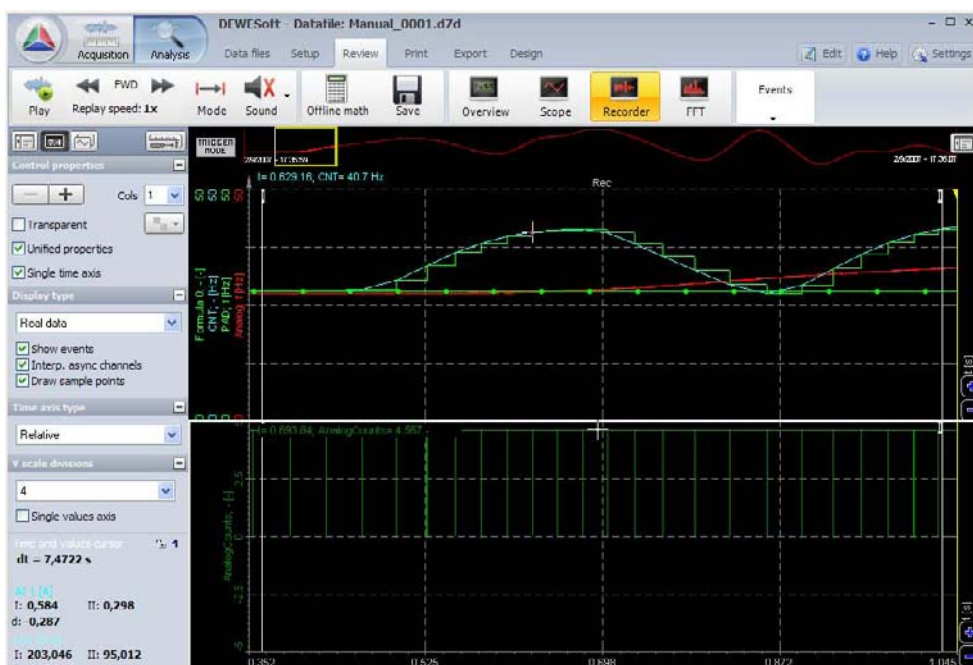
$$1/\text{pulsewidth}(\text{'AnalogCounts'}>2)$$

从公式中可以知道它是通过测量"AnalogCounts"输入信号的脉冲宽度来计算频率的。脉冲宽度函数需要真正的数字值(即0或者1)，因此我们需要通过一个逻辑函数运算来算出这样一个信号的数字值。



因此，我们看一下红色曲线的结果(这个红色曲线是模拟测量的频率)，曲线显示它具有巨大的测量延迟，而步进绿色曲线，它是数学信号，几乎没有延迟。不过它在两个转速之间会出现很陡的阶梯，这是因为在两个脉冲之间没有任何新的信息用来计算频率。

下图所示为利用模拟信号测量的脉冲。这些脉冲可以很好的用于相位角的高级分析。脉冲测量同样也是一个最困难的部分：如果我们的采样率非常低的话，那么我们很可能错过这些脉冲信号，永远不可能得到好的结果。脉冲的宽度取决于输入脉冲的宽度，如果脉冲是从前面面板接头输入的话。我们也可以在主机背面增加一个固定路径面板，在这种情况下，根据DAQP-FREQ-A模块的滤波器设置，信号将会放大。请参考有关这个主题的用户手册了解更多信息。



2.8 视频采集

在DEWESoft软件中支持四种类型的摄像机：

- 低速 (最高达**30 fps**)网络摄像头或者是手提式摄像机，由**DirectX**支持
- 中速 (最高达**250 fps**) 摄像机 (DeweCAM 或者 Basler)
- 高速摄像机 (最高达**5000 帧每秒**)，这种方式我们需要用后处理来讲数据和视频组合，或者直接利用**DEWESoft** 软件内部的 Photron 驱动(请参考用户手册，了解详细信息)
- **NEC** 热成像仪 摄像机

所以，对于特定应用应该选择哪种摄像机呢？很明显，热成像仪摄像机有自己特殊的用途，高速摄像机是用来在需要非常极端的高速视频摄像来捕捉碰撞，爆炸以及其它快速事件的情况时获取较短触发快照。

不过是选择手持式摄像机还是中速摄像机则不是那么容易了。这两种摄像机和高速摄像机之间的主要差别在于使用中速和低速摄像头我们可以连续将视频信号直接流入计算机硬盘，直到磁盘存满为止。我们还可以在软件中设置视频触发，减少存储的数据量，或者进行视频压缩。

不过，计算机的性能必须比较高，才能保证视频无丢失流盘。我们需要一个高性能的硬盘，一台高性能的计算机系统，即便如此，也很可能达到系统的性能极限。我们必须知道一张典型的VGA尺寸的图片需要**300 kB**存储空间。如果我们每秒采集**100 帧**图像，一台摄像机每秒就需要传送**30 MB**的数据。

很明显，如果我们使用的是高速视频，我们需要采用Dewecam或者Basler摄像机。Basler摄像机的速度更快一些(**100 帧每秒**，VGA，降低分辨率最高到**250 帧每秒**)，不过Dewecam具有更好的图像质量，而且带有自动快门，自动增益以及自动白平衡。这两种类型的摄像机最大的优点在于它们可以通过模拟卡来进行触发，因此数据和视频是完全对齐同步的，不过Dewecam

还是有一点点优势，因为Dewecam摄像机在每帧图像上都增加了图像索引，因此即使触发丢失了，它仍然能够正常工作。

如果每秒**25**或者**30帧图像**就已经满足要求的话，我们最好考虑使用手持式摄像机。不过我还是建议使用逐行扫描摄像机，这样可以避免出现交错的图像。

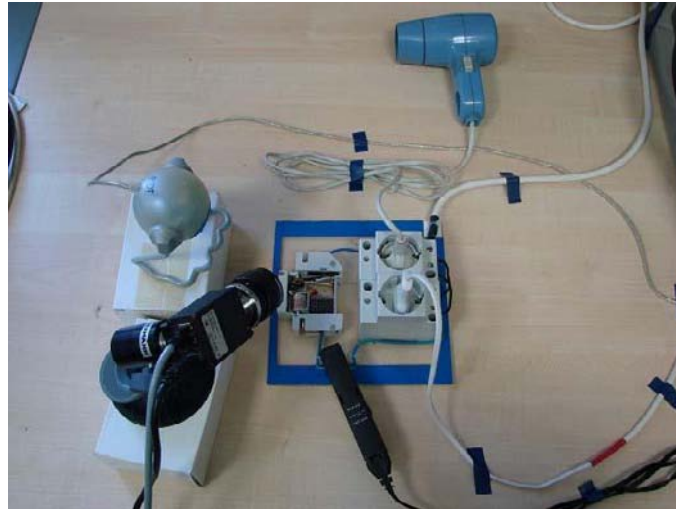
网络摄像头通常价格便宜，速度较低，质量也较低，但是它可以成为非常有帮助的工具，用来记录整个实验过程。我们从用户处得到许多反馈，他们告诉我们，即使是一个简单的甚至可以说是质量较差的图像都能很好的帮助他们更好理解记录的数据。

在以下的测试中，我们看一下网络摄像头和中速摄像机两者之间的差别。

所需硬件	任意 AD 卡，网络摄像头以及Basler/Dewecam摄像机
所需软件	PROF (支持Basler/Dewecam 摄像头)
设置采样率	至少 1 kHz

2.8.1 通道设置

对于这个测试，我们用到了一个**0.5 A** 保险丝，测量保险丝两端输出的电压和电流，用一个电吹风熔断保险丝。Basler摄像机 和网络摄像头用来记录整个测试过程的视频。



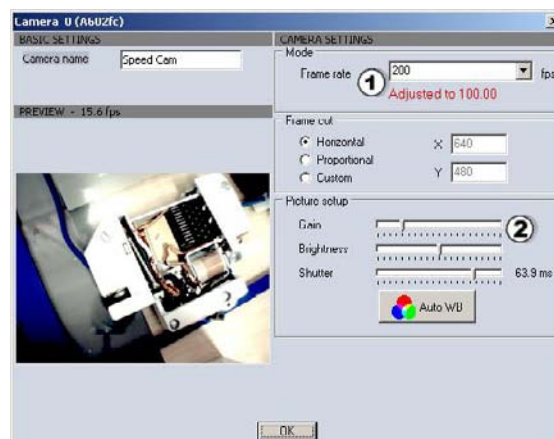
即使有违本教材的出发点，我还是要讲一下如何在**DEWESoft**软件 中进行硬件设置**hardware setting**。如窗口中所示，我们选择了 两种类型的摄像机，**Basler** 以及 **DirectX** (这种形式的支持可以识 别网络摄像头和手提式摄像机)。因此我们将使用中速 **Basler**摄像机 和一个网络摄像头来进行测量过程中的摄像。



在**DEWESoft**软件 的设置屏幕中，我们可以看到两台摄像机。第一台是**Basler A602fc**，第二台是**USB**网络摄像头。两者都选，然后 我们开始设置摄像机。



我们首先来设置Basler摄像机。设置**视频帧率**为**200 fps**①，不过在这个例子中，视频分辨率不能设为 **640 x 480 (VGA)**，而应该降为 **640 x 242**。下一步设置**快门速度**②，快门速度不能超过**5 ms**，这种快门速度可以获取**200 fps**的图像。快门速度块快，获取快速运动物体图像时会降低图像的拖尾效应，但是同时也会降低图像的亮度。因此我们需要把灯光调得更亮，或者增加 图像的**亮度**和**增益**②，不过这样做在另一方面又会增加图像的噪声，降低图像质量。

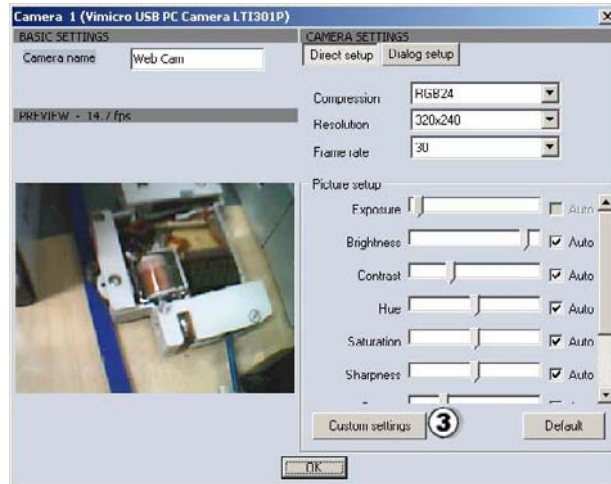


网络摄像头的设置则主要依赖于摄像头的性能。不同种类的网络摄像头之间会有很大的差距，不管是在速度上，图像质量上

还是可以使用的功能上。某些网络摄像头具有自动快门和自动增益，有些甚至有自动聚焦功能。所以这种巨大的差异让我们无法在手册中完全覆盖每种品牌的网络摄像头。Dewetron公司无时无刻都在尝试市场上的新品牌网络摄像头，所以您可以问问他们哪些是最新的，哪些是最好用的。

某些摄像机会不同的压缩比，比如YUV和I420。这表示每个像素并不是用24位(每种颜色占8位)来表示，而是更少。简而言之，使用这种压缩模式会产生更小的图像尺寸，最终减少文件大小，但是图像颜色并不如RGB色彩(未经压缩)那么完美。不过，由于人眼对灰度级的分辨率远高于对彩色色彩的分辨率。这种压缩算法就是利用了这种原理，因此我们几乎很难看出其中的差异。

摄像机通常还会提供客户设置③，这种客户设置会显示所有特定摄像机具有的特殊功能，比如图像剪切或者图像旋转。



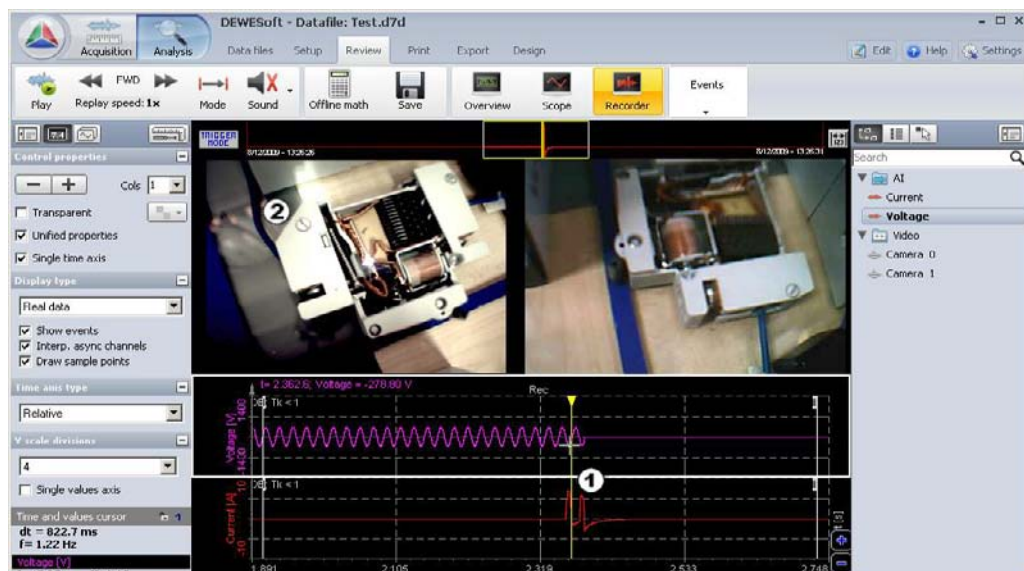
对于模拟数据采集，我们可以按照电压和电流教程中所说的方式设置一个电压和一个电流通道。

SLOT	ON/OFF	C	NAME	AMPLIFIER (003)	PHYSICAL VALUES	CAL	SETUP
0	Used	Shunt	Voltage	DAQP-DMM 400 V / 20 kHz	0 -100 100	-134.9 / 145.3 V Zero	Auto Set ch. 0
1	Used	Shunt	Current	DAQP-V 10 V / 50 kHz	0 -10 10	-6.475 / 6.702 A Zero	Auto Set ch. 1

2.8.2 测量

现在我们来采集一些数据，然后观察一些结果。高速摄像机和模拟数据完全精确同步，所以我们可以把它采集的图形和保险丝熔断过程中的模拟电压和电流进行比较。流经保险丝的电流在被熔断①之前大约持续20毫秒左右，当保险丝熔断瞬间，我们可以清楚看到电火花。还可以清楚看到开关在弹出接触时②所处的位置(在保险丝左边)。

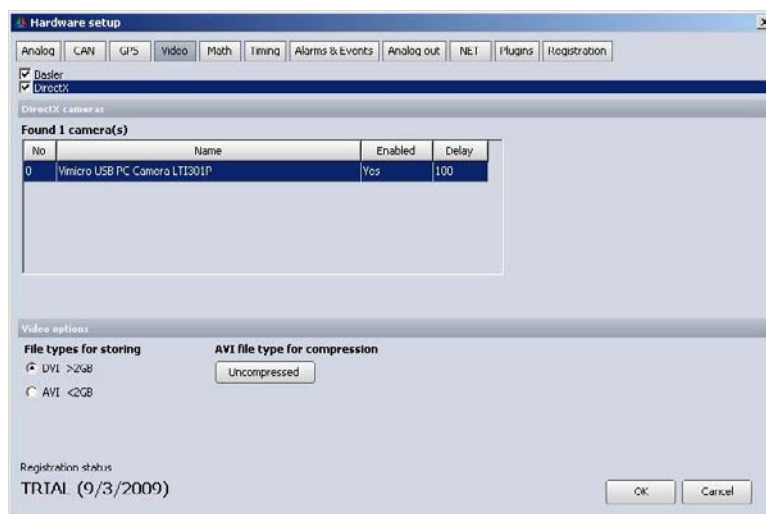
用网络摄像头采集的图像也很好，不过它无法显示上述的东西。



因为网络摄像头没有时钟控制，所以不能和模拟数据完全同步，当采集的图像送入计算机时，计算机给它打上时间戳。因此我们可以发现网络摄像头采集的保险丝熔断图像会有一个大约60 ms^③的延迟。所以我们可以发现网络摄像头有两个局限性：速度和时间精度。



不过，我们有办法通过输入摄像机延迟时间来降低这种时间不准确性。通常这种延迟非常稳定，可以想办法补偿。我们可以在硬件设置Hardware setup窗口中来设置补偿，打开硬件设置窗口，在延迟域中输入延迟值。另外，每帧图像间还存在瞬间的颤动，时间大约在30 ms左右，如果系统性能正好在其性能的边沿，这个值还会稍微大一些，我们无法为其提供补偿。如果需要精确的时间，那么我们就需要使用具有时钟控制的摄像机，比如Dewecam或者Basler摄像机。



现在如果我们重新观察测量过程，两个摄像机都能大致以相同的时间显示熔断事件，不过，在保险丝熔断的瞬间，网络摄像头只显示一帧图像，而Basler摄像机则显示了数张图像，可以更清楚观察电火花的情况，线圈的运动以及开关的运动情况。

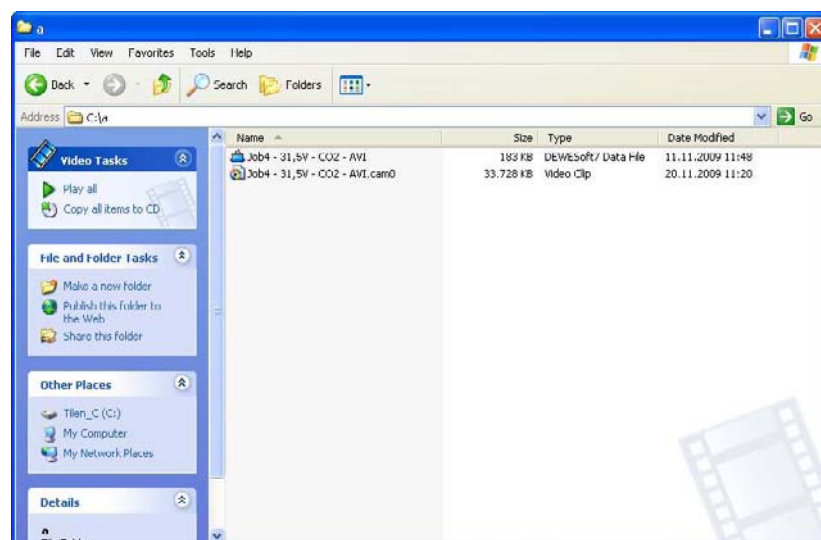


不过这种分辨率也已经达到了250 帧每秒的极限。某些事件记录需要用更高速的摄像机。这种类型的摄像机将图像存储在本身内置的存储器中，然后在导入到视频文件，所以我们需要用后处理来整合视频数据和模拟数据。

2.8.3 视频后同步

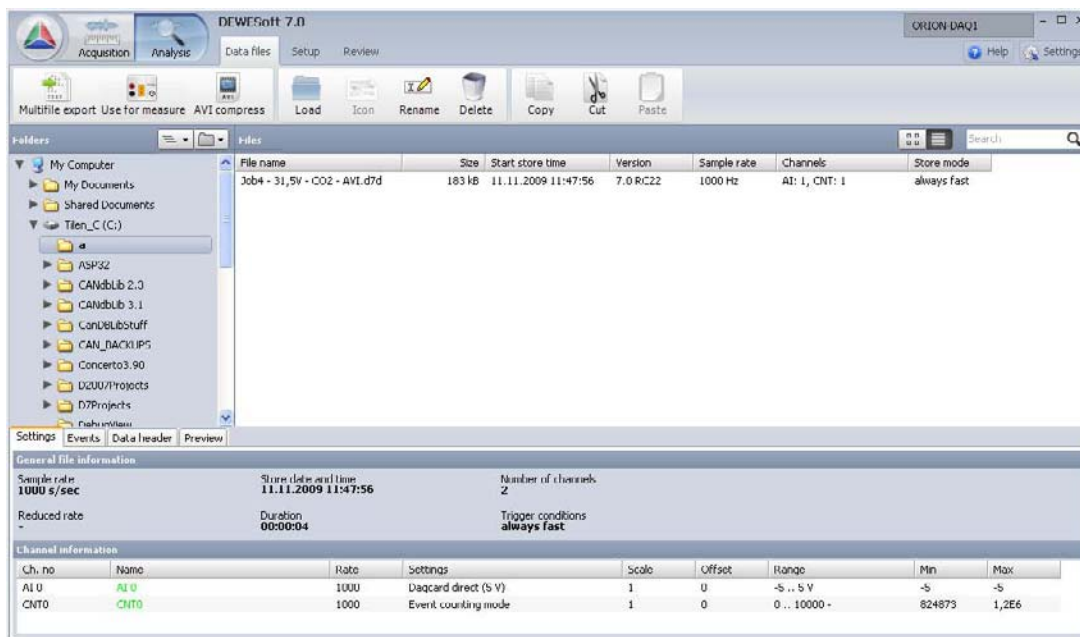
为了将高速视频导入DEWESoft软件，我们需要做以下几步工作。

1. 在采集完视频文件以及DEWESoft数据之后，请将*.avi格式的高速摄像机文件拷贝到需要用来同步的DEWESoft数据文件所在的目录下。
2. 按照xxxxx.cam0.avi格式重命名视频文件 格式中xxxxx指的是需要同步的DEWESoft文件名。如果我们有很多视频文件，我们可以按照xxxx.cam0.avi, xxxx.cam1.avi的格式按顺序为其命名。



上图所示为一些例子。原始的DEWESoft文件是test.dat，因此视频片段的名称为test.cam0.avi。

3. 命名后，打开DEWESoft软件，进入分析Analyse模式，您会发现视频文件已经成为DEWESoft数据文件的一部分了。



双击文件名，打开文件，**DEWESoft**软件会自动识别该文件没有包含同步信息，随后会询问是否用"手工"进行同步。

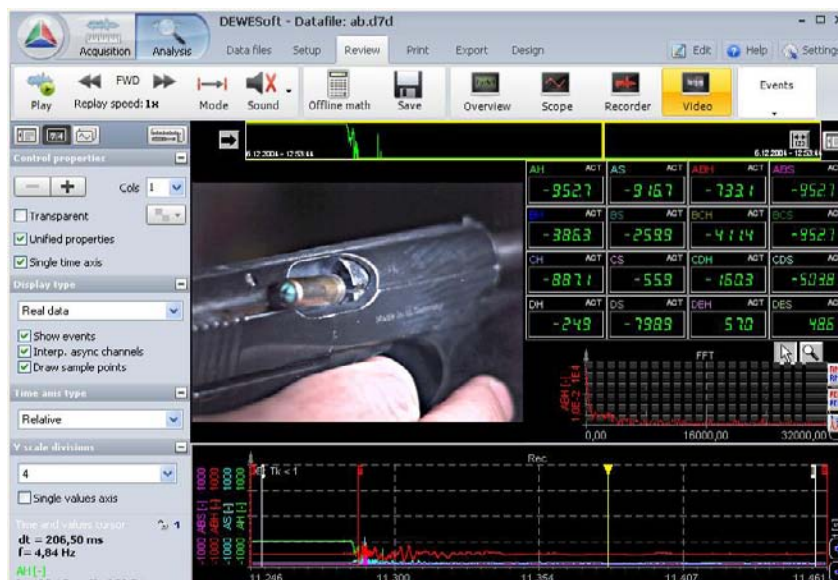


软件会自动识别文件中包含有多少帧图像①。请正确输入摄像机的帧速率**Frame rate**②(有时候`.avi`文件已经包含有正确的帧率值，但是大多数情况下没有)，这时候在信息区域**Video size** 中会显示每个视频大概多长时间，以秒为单位③。

然后在**预触发Pre trigger**④区域中输入在视频触发之前需要抽掉的图像数量。您可以选择**几帧**，**几秒**或者**几毫秒**⑤。**触发时间Trigger time**可以首先选择**从触发器from trigger**触发(假设我们是从相同的触发源来触发采集模拟数据的)，或者选择**从开始from start**测量的相对时间，以秒为单位⑥。

所有设置完成后，**DEWESoft**软件会进入到视频屏幕，观察同步结果，记录仪会显示视频图像和模拟数据对其的情况。

您可以利用黄色光标前进，后退，观察同步的质量和效果。



如果需要重新对其视频文件，请选择 **post synch** 菜单。 在这里我们可以改变参数，另外还可以选择一个其他选项，即从当前位置开始视频。



在本例中，黄色光标将被当做同步的原始数据位置。

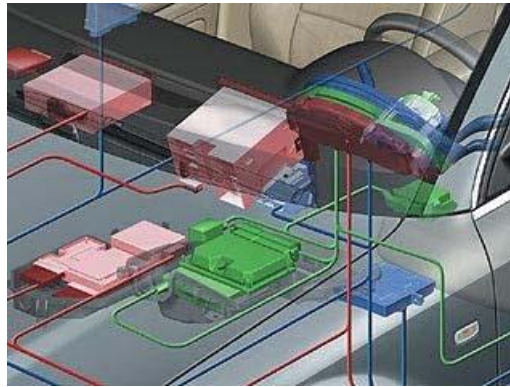
一旦完成后，达到满意效果后，为了存储同步信息，以便将来后处理，不要忘了选择 **"Store frame timestamps to video file"** 菜单。小心！这个操作只能进行一次，因为所有信息都将写入原始视频文件，建议保存一个视频文件的拷贝。

2.9 CAN bus 数据采集

你知道现在的汽车内部平均布线长度达到几公里吗？这个数字是不是让人很惊讶，因此汽车制造商一直在想方设法减少电线布线。随着当前汽车新技术的不断涌现，汽车内部有越来越多的设备需要互相通讯，互相分数据。

因此，在19世纪80年代，博世公司发明了一种串行接口，该接口只需一对双绞线就可以实现传递1Mbit/秒的数据。这样就打打减少了布线的长度(即使这样，车内布线长度仍然以公里计)。不过我们可以这样说，没有这项技术，今天的汽车就不是现在这个样子了。

CAN bus 接口不仅仅用于汽车。它还出现在其他类型的车辆上，比如卡车，火车，坦克，拖拉机等等，同样还用在普通测量设备上。现在已经有大量传感器已经用在 **CAN bus** 技术上。这种技术非常坚固，容错能力强，并且具有很好的碰撞检测算法。



因此，现在我们已经知道CAN bus 是什么东西了。那么我们需要什么东西才能测量CAN bus呢？首先，我们需要一些 **DEWESoft** 软件支持的硬件。

所需硬件	Orion, NI, Softing 或者 Vector CAN
所需软件	LT 或者更高 + CAN 选项或者 EE
设置频率	至少1 kHz

然后，如果我们在车内进行测量，我们需要知道在车内哪个地方进行连接。对于“官方用户”而言，这种工作微不足道，因为他们非常清楚车的布线情况。我们需要注意没有终端的线不会太长，因为汽车车辆连接到测量设备时可能会受到干扰。如果我们已经有一只传感器，或者有传感器阵列，那么就会有详细描述布线方式的接头。在这种情况下，我们最好选择巴士车，巴士车并不像听起来那么难以理解，因为它内部只有一对双绞线，两端电阻值大概120 Ohm，实际上任意连接头都处在其中。

只有正确连接到巴士车上，我们才能够通过CAN总线扫描所有的信息，但是我们只能看到一个信息ID和原始数据。

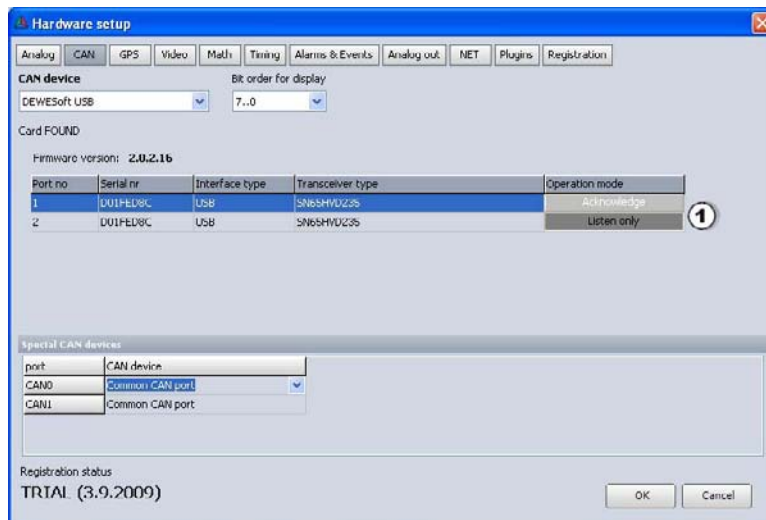
第三件事是我们必须知道如何从CAN总线解码这些信息。要理解这一点，我们需要一个详细的说明书，或者一个数据库。这些数据库由Vector定义，叫做DBC文件。它包含了详细的输出通道信息说明。汽车制造商将这些数据库非常隐蔽，非常得体的藏在车内。对于卡车而言，这些信息是标准的，详细描述详见J1939标准。

2.9.1 通道设置

我们现在按照前面一些章节所描述的相同步骤进行通道设置。首先，进行通用设置。有两种类型的设备，更好的说法是有两种类型的操作。

- 比如，连接到巴士车上时，我们只需 *listen only* 线上发生的所有一切
- 对于专用的传感器而言，通常需要发送收到的数据 *acknowledge*

我们还需要设置硬件，以符合我们现有的应用。这种设置在硬件设置 **Hardware setup** 窗口中完成。

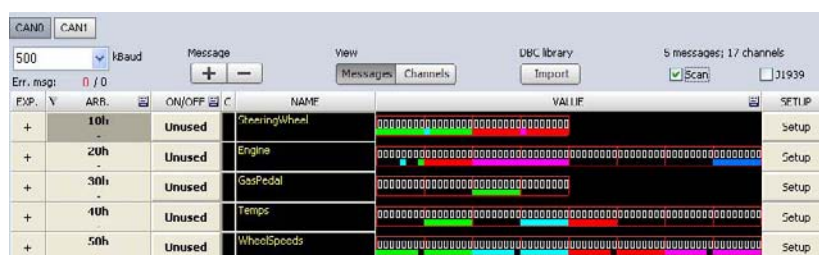


点击操作模式 **Operation mode** 按钮，在 **Listen only** 模式和 **Acknowledge mode** 模式之间选择①。我们需要首先设置这个模式选择，然后再连接到巴士车，以免干扰正常的巴士操作。

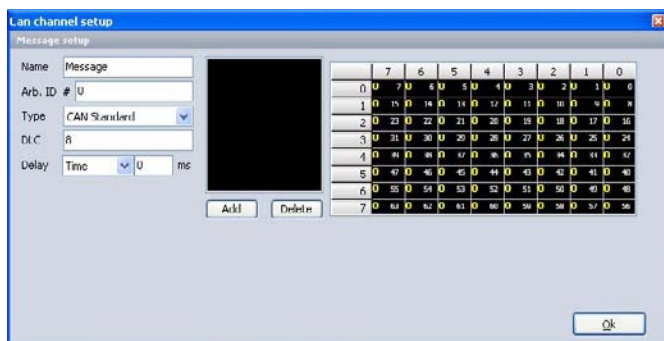
波特率设置②非常重要。实际上，如果我们设置了错误的波特率，连接到巴士车之后，某些汽车的操作可能会受到干扰。在波特率编辑框底下，有一个提示条，显示有多少信息来自总线，其中有多少已经损坏(红色)。只有波特率正确设置，才会显示这些信息，同时还会显示巴士车师是否由于错误链接发生问题，或者巴士车发生过载等信息。



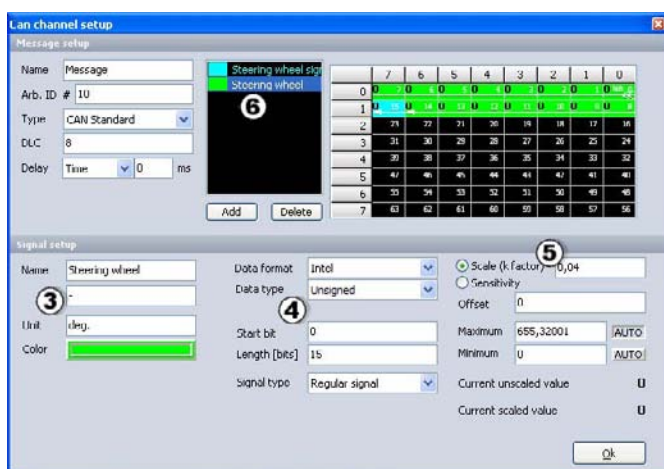
接下来我们要扫描这些信息。一旦我们选取了 **扫描Scan** 选项，来自于巴士车的信息就会显示在窗口上。所以我们就能看到来自巴士车的信息 **ID**，信息速度以及原始二进制值。



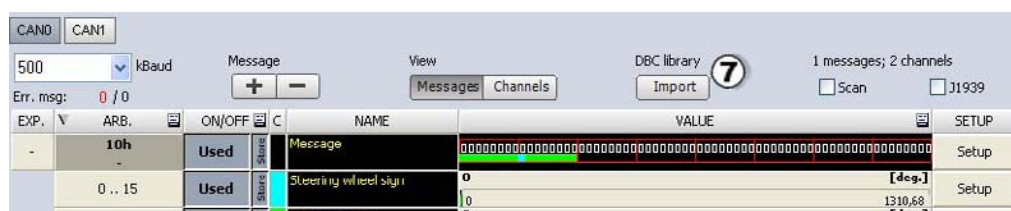
如果我们知道CAN的详情，我们可以从技术指标定义这些通道。选择设置 **Setup** 按钮，会出现一个空白的信息设置屏幕。



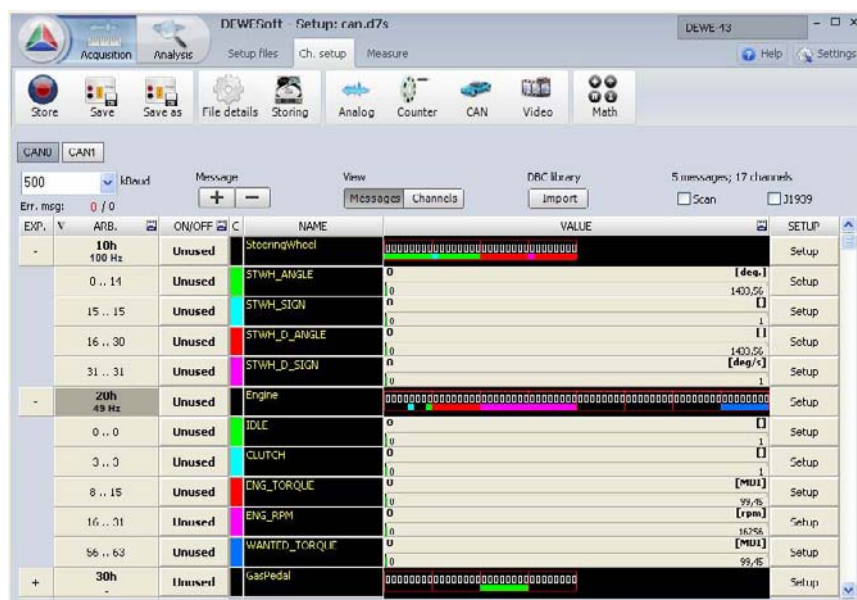
现在...我们必须事先知道...或者去猜。我们知道信息的前15位是方向盘转角。我们需要点击增加**Add**按钮，增加一个通道。然后输入通道名称**Name**，单位**Unit**③，数据格式**Data format**设为**Intel**，数据类型**Data type**设为**Unsigned**，通道长度改成**15位**④，输入量度因子**Factor**⑤。



我们再增加另外一个通道 – 方向盘标记**Steering wheel sign**⑥，只有一位，从第16位开始。到现在为止，我们一定定义了两个通道，这两个通道会在接下来的测量中用到。假设我们离开信息设置窗口，两个刚刚增加的通道会出现在信息栏下面(如果我们拉开信息，可以显示整个通道信息)。



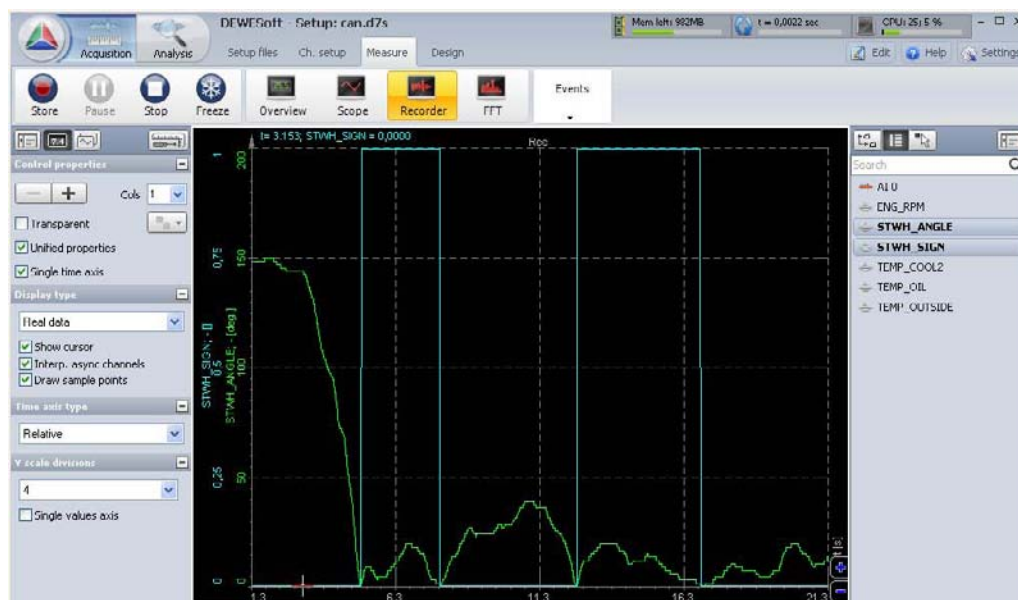
第二种增加信息的方式是简单的装载说明文件即可。说明文件的数据格式是*.dbc。点击**Import**输入按钮⑦，选择输入文件。对于我们的测试来说，说明文件装载了所有通道的信息。



巴士车的信息显示有不同速率 – 我们可以发现越重要的信息，数据传输速度越高。在本例中，方向盘的传输速度达到100 Hz，发动机信息的传输速率为50 Hz，而温度的传输速率只有5 Hz。信息ID优先权的规定迫使汽车制造厂商给最重要的信息通道分配最低的仲裁ID，因此具有最高的优先权。现在我们选择一个通道，看看在实际测量中它们是如何显示的。

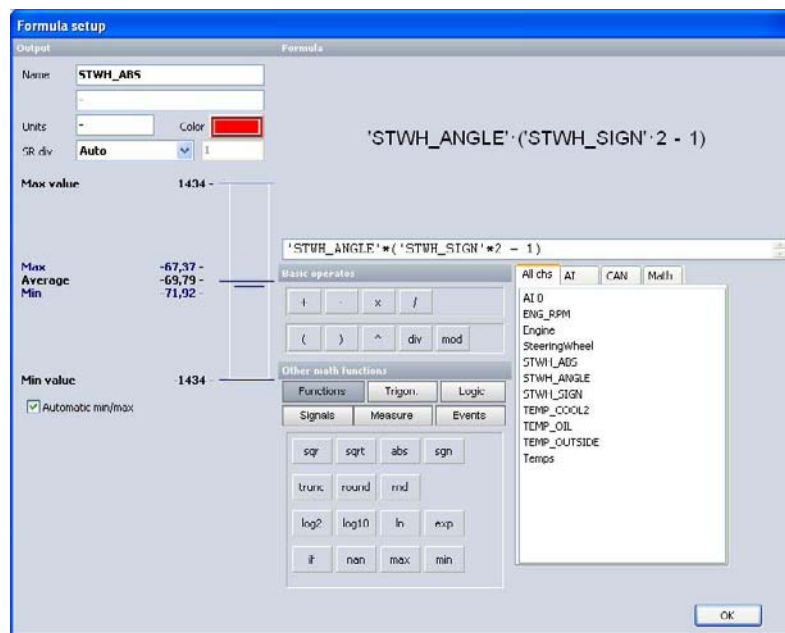
2.9.2 测量

我们先从记录仪recorder上看看这些数据。首先看一下方向盘角度和方向盘标识。这些数据按照规定的时间有规律的从巴士车内传过来。我们可以看到当我们把方向盘从左打到右边，方向盘标识信号马上从0变成1，方向盘角度则显示绝对偏转角。



这里有一个非常好的例子用来观察数学通道能帮上什么忙。我们首先创建一个新的数学通道，考虑一下如何从方向盘标识和方向盘角度信息中创建绝对通道。首先，我们需要一个真正的标识，比如-1表示负值，+1表示正值。因为我们只能得到0和1，所以我们需要对其重新量化。可以通过几何运算的方法来实现，先乘以2(得到0和2)，然后减去1，这样就得到了-1和1，用他们来代表0和1的输入。接下来我们把它和角度相乘。所以，最终公式如下：

$$\text{"STWH_ANGLE"} \{ \text{"STWH_SIGN"} \cdot 2 - 1 \}$$



接下来我们把这个结果通道加入记录仪*recorder*。当蓝色曲线(标识)变低, 角度就是负的, 而当蓝色曲线变高, 角度为正。

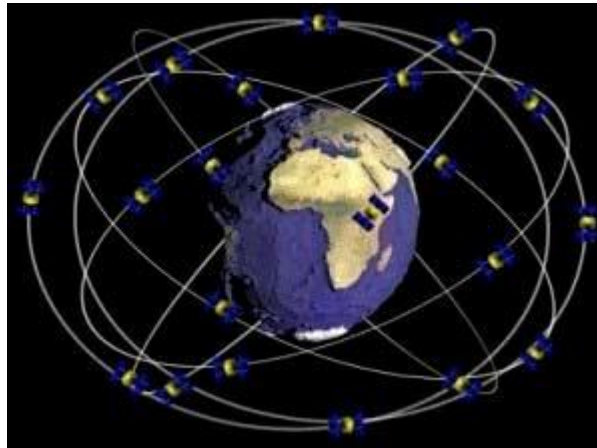
DEWESoft的数学通道有一个非常好的功能, 就是能够监测信号的输入速率。如果信号是不同步的(这表示信号不规则采样), 那么输出通道同样不同步。既然这两个整合通道来自于相同的信息, 所以它们具有相同的时间戳, 所以我们可以将它们整合在一个公式中, 最终可以确认输出通道也具有和输入通道完全一样的速度和时间戳。



2.10 GPS 采集

GPS 表示全球定位系统。实际上这个缩写来自于美国的系统，被称作Navstar系统，这种系统是全球第一套定位系统。俄罗斯则具有另外一套系统，叫做GLONASS。而欧洲的系统叫做伽利略Galileo。

这种系统包括多颗(GPS系统有24颗)运行在地球轨道上的卫星，这些卫星向地面传送卫星的时间信息和位置信息。从这些信息中，用户可以确定所处地球上的确切位置。这种位置信息是通过三角公式来计算的，所以我们至少需要三颗卫星才能确定自己所处的位置。



大家都非常清楚这是最常用的应用，不过我们能用的远不止这些。好的接收器可以利用多普勒效应Doppler effect精确计算速度。速度的测量好于0.1 km/h。因此我们可以把这种接收器，比如VGPS，当做速度传感器。市场上的低价GPS没有这种功能。速度输出的速率在10到100Hz之间变化。

第三种可能的应用来自于GPS系统的工作方式。因为卫星给地面传送精确的绝对时间，所以更好的接收器通常能以小于一毫秒高精度的脉冲输出这个信号，所以我们能够使用这个技术来同步远程系统。实际上，这种应用没有距离的限制 – 我们可以精确同步远在加利福尼亚和印度的两台系统。利用这种技术，我们可以清楚看到旧金山的蝴蝶扇动翅膀和印度地震之间是否存在相互关联(这只是一个笑话，不过这种技术确实能够发现电力网上的错误源)。这种技术我们可以通过GPS-CLOCK和VGPS-200C两种硬件来实现。

来自于GPS系统的信号质量主要依赖于接收器能够找到的卫星数量。卫星的信息无法进入到角落，无法穿越墙体，所以路线上是否有障碍物是非常重要的影响因素。高达建筑物以及数目会阻碍甚至隔离卫星信息。高达建筑物还会产生反射信号，会导致很大的测量错误，即使能找到的卫星数量很多也是如此。

如果能够找到六颗或六颗以上的卫星时，通常能够得到很好的信号；否则会产生速度和位置的抖动。我们可以采用不同的

GPS天线系统，通常来说有非常小的天线和非常大的天线两种。这两种天线之间的差别在于小型天线需要

地面地面挡板防止地球表面的反射。地面挡板不能小于30 x 30 cm大小。汽车车顶，如果是金属的话，就是一个完美的天线地面挡板。对于其他应用，我们需要注意做一个地面挡板，或者选择本身带有地面挡板的一体化天线。

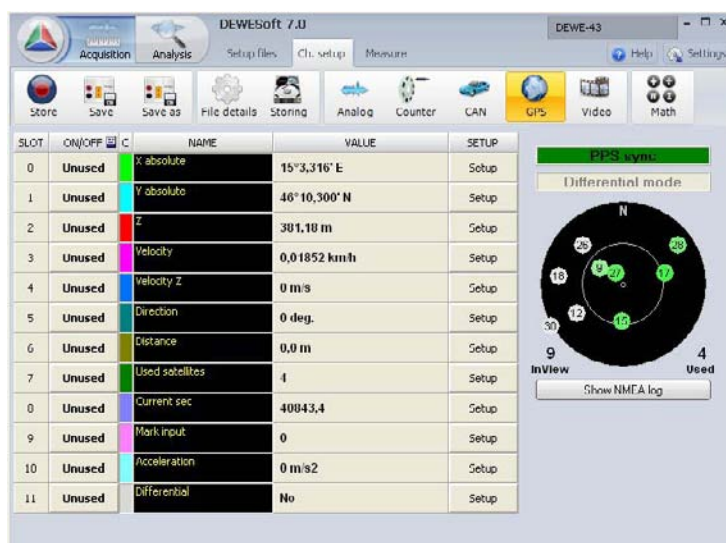
2.10.1 通道设置

所需硬件	Dewetron VGPS, Leane VSAT, Javad, Microsat或者其他任意 NMEA兼容的 GPS
所需软件	任意版本
设置采样率	至少1 kHz

在**硬件设置Hardware setup**屏幕上选择**GPS**后，我们会发现在设置屏幕**setup**中增加了一个专门的**GPS**标签。在该标签上，有几个通道可供我们选择 – x, y, z轴方向上的位置，速度velocity，垂直速度vertical velocity，使用的卫星数used satellites，标记输入(外部事件)以及当天的第几秒。从速度通道还可以计算出加速度和位移。

在标签底部有一个天空图，天空图显示当前天空中的卫星分布。当前正在使用的卫星以**绿色**表示(如果接收器支持**GLONASS**系统，那么当前正在使用的卫星以**红色**表示)，其他颜色则表示信号的强度。**浅绿色**表示信号较弱，**暗绿色**则表示信号非常强。如果接收器支持**PPS 同步PPS sync**的话，我们还能使用时间脉冲信息。这个信息就是接收器通过GPS接口(**RS-232** 或者**USB**)接收的脉冲数每秒信号，这些接口如果有的话，可以大大增强和其他数据源的同步功能。如果窗口中没有显示**PPS sync**选项，我们需要在**DEWESoft Tuner**应用程序中**GPS section**中打开这个功能，以便能够从GPS接收这类数据。通常这类信号都可以用，但是在某些廉价的接收器上可能没有这个功能。

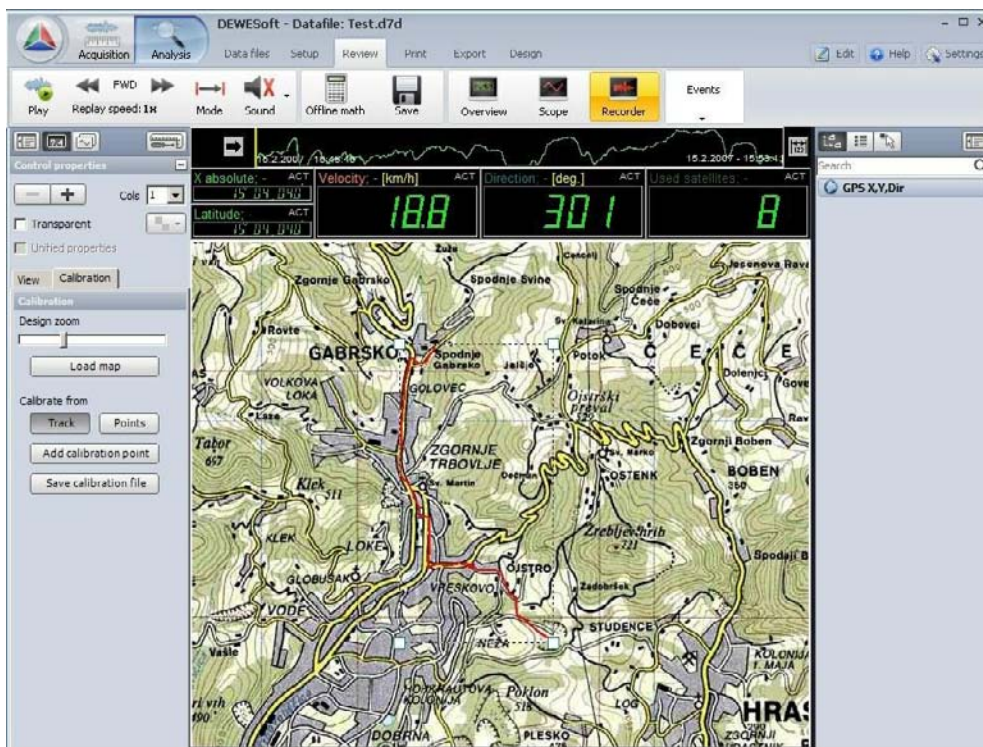
另外，如果接收器支持**SBAS**或者**WAAS**差分模式，并且已经使用了这些信号，在窗口中还会显示一个**差分模式Differential mode**指示器。



当我们选择好通道后，就可以进行一些测试了。**GPS**设置的一部分内容就是给**GPS**显示设置一张背景地图。现在来看看如何在**GPS**行进路线图中增加一张背景地图。假设我们已经有了一张 **bmp**或者**jpg**格式的地区地图，我们就可以将其增加到**GPS**背景中去。

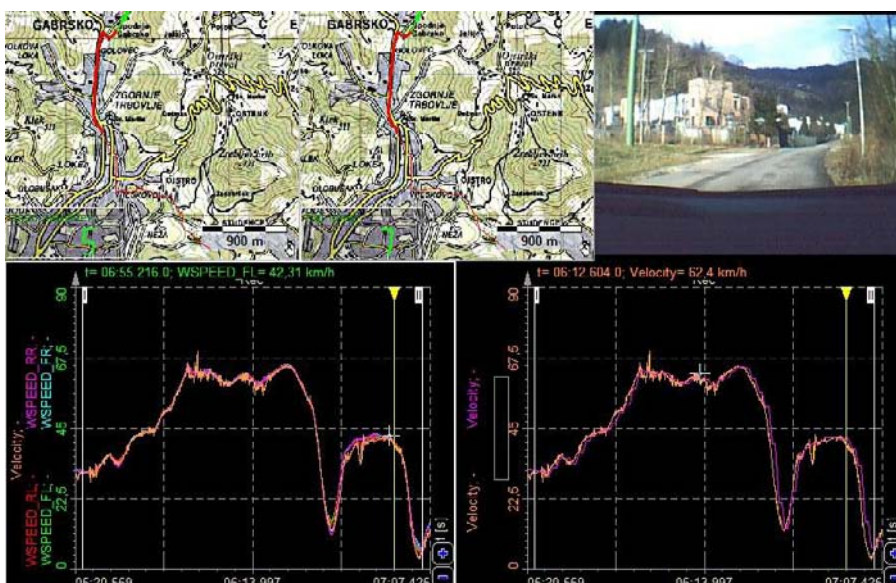
如果我们不知道地图的坐标，需要在这个地方行驶一段路程，然后存储测量。然后回到**GPS**测试界面，找到**校准Calibration**，进入**Load map**菜单，载入地图。地图和行驶路径以背景显示，并且在其四周有四个操控柄。拖拉这几个操控柄，可以调整地图的大小和位置，最后准确对准地图位置即可。

如果我们知道确切的地图坐标，我们可以进入到**Calibrate from – points**模式。一旦地图调整后，我们可以**保存校准文件 Save calibration file**。这个地图将从此点永远显示在每个测量的背后。我们还可以设置几个文件，用于不同的放大显示。我们可以设置类似于这样的概览文件，当放大观察时，地图切换到其它背景图。现在我们可以好好驾驶。看看**GPS**测量到底如何显示。



2.10.2 测量

在我们的测量中，我们用到了两个GPS接收器。一个是低成本的1 Hz接收器，另外一个的高速的20 Hz VGPS。我把两个GPS屏幕显示放置在一起 – 左边的是高速VGPS，右边的是低速GPS。我还连接了**CAN**总线接口，显示轮速(左下方记录仪)。右边的记录仪里面橙色的曲线显示的高速GPS速度信号，而粉红色的曲线显示的是低速GPS的速度信号。网络摄像头仅作为参考记录视频。

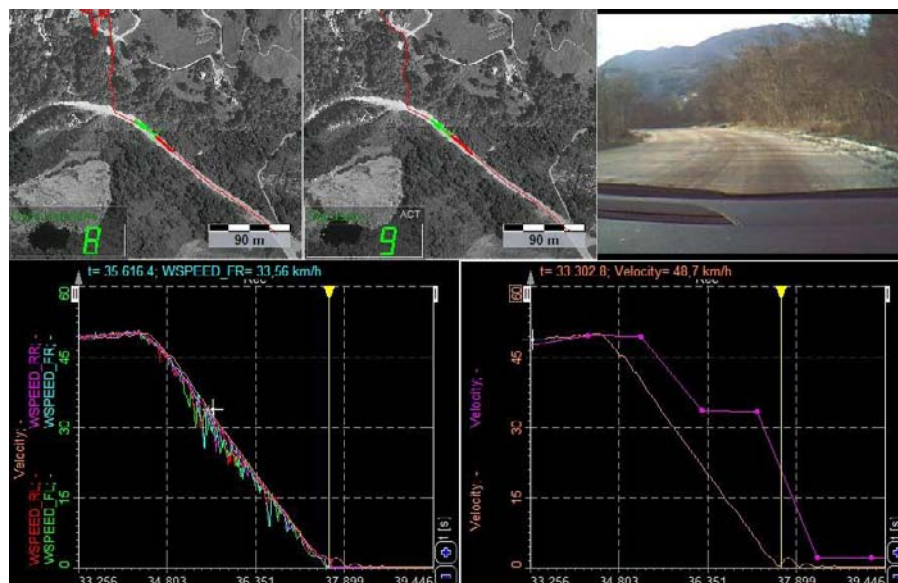


让我们放大行驶路线的某一部分。我们试着在很脏的路上踩刹车，看看车轮抱死的情况。我们可以注意到，当我们放大地图后，GPS图变成了卫星图。当我们在记录仪中放大的时候，一部分道路可以显示在图上，道路显示为一条很粗的蓝色线条，而其他的道路部分则显示为一根细细的红色线。

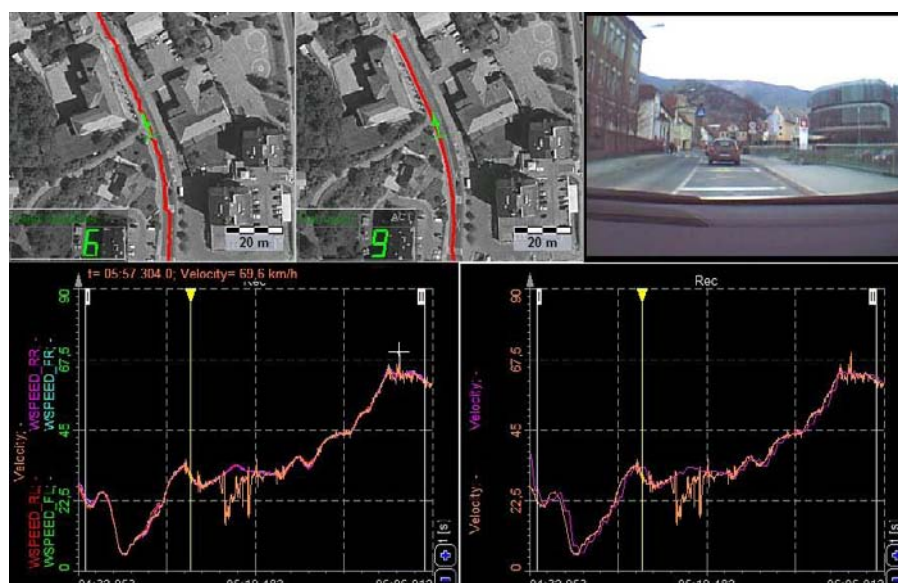
刹车测试显示出高速GPS和低速GPS的不同。右下角的记录仪清楚显示它和实际速度大概有半秒钟的延迟。这种性能对于真正的评价应用毫无用处。

橙色的曲线同样显示在左边的记录仪中，显示结果清楚表明测量到的轮速和GPS速度之间没有明显的延迟。不过，既然轮胎已经被抱死，速度正在下降。但是这时我们就能看到ABS系统起了作用。如果没有ABS系统，轮胎会完全锁死，车辆可能失去控制。在50 km/h速度时，在宽阔的马路上，这可能不是什么问题，但是我们可以想象不同的情况，这种条件根本不像我们想象的那么好。不过，这张图形同样给我们证实了，在这种特殊的情况下，我们不能相信从轮胎得到的测量结果。

第三件指出的事情是车辆停下后，也就是刹车结束的时候。因为车体还在悬挂系统上依靠惯性往前冲，所以我们看到最后又那么一个小小的跳跃。这是因为车辆稍微往后移动了一下，然后才真正停了下来。如果我们测量刹车距离，那么这么一小段距离并不是我们真正想要的，所以需要从计算中剔除掉。



下面一张图很清楚的显示高大建筑物对测量的影响，在建筑物区域内，高速GPS速度明显往下掉，这明显是错的。而低速GPS在这种情况下，反倒有更好的表现，因为他能进行更平滑的计算。同时路径的显示也表明信号受到了干扰。



那么，现在可以得到什么结论呢？当我们行驶速度变化不大的时候，比如火车的行驶，或者我们仅仅把它当做汽车驾驶时候的一个参考时，低速GPS就比较有用了。

当需要高精度速度测量时，就需要选择高速GPS，但是我们需要一张清晰鸟瞰图，能够找到足够多的卫星，以保证测量的可信度。

3 电力模块教程

电力测量模块是**DEWESoft**中**最重要的数学模块**。它可以测量不同配置下的不同频率的电网，甚至可以测量不同的频率源组成的电网，下面的教程将讲解这个模块如何使用



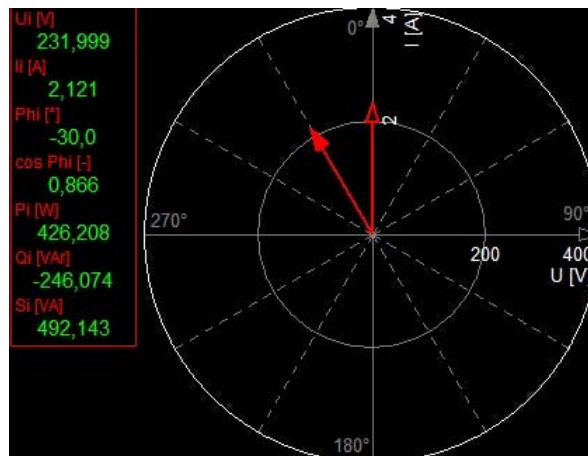
单相电力

测量电压、电流以及它们之间的相位关系；谐波分量，名义功率，有功功率，无功功率

3.1 单相电测量

在 "电压和电流" 章节我们已经知道如何测量电流和电压以及如何通过电压电流输入信号的触发进行数据存储. 当测量这两个参数是, 我们总是会对测试设备电力的消耗或者产生发生兴趣.

如果我们使用 **DEWESoft** 电力测量模块, 就会发现有许多参数需要计算. 事实上, 俩个通道的输入信号通过各种运算可以生成大量的虚拟通道. 首先, 我们看一个电力测量最常见的界面 (如下图). 这个界面叫做矢量示波器 **vector scope**. 这个名字说明除了电压和电流的绝对值, 它们之间的相位关系也是非常重要的. 所以矢量示波器既表示出了电压 (红线黑头的箭头) 和电流的幅值 (红线红头的箭头), 也表示了它们之间的相位关系.



为什么相位的信息这么重要呢? 简单的说机器只能利用电压和电流同相的部分来工作, 因此我们需要测量电压和电流的相位差以角度 **phi** (在我们的例子里是30 度)表示, 也可以用 **cos phi** (也就是这个角度的余弦, 因为这正好就是工作过程中消耗掉的电流). 下面看一下电力的一些定义

$S=U \cdot I$ 这很明显是总功率, 也就是工作所消耗的所有的电力

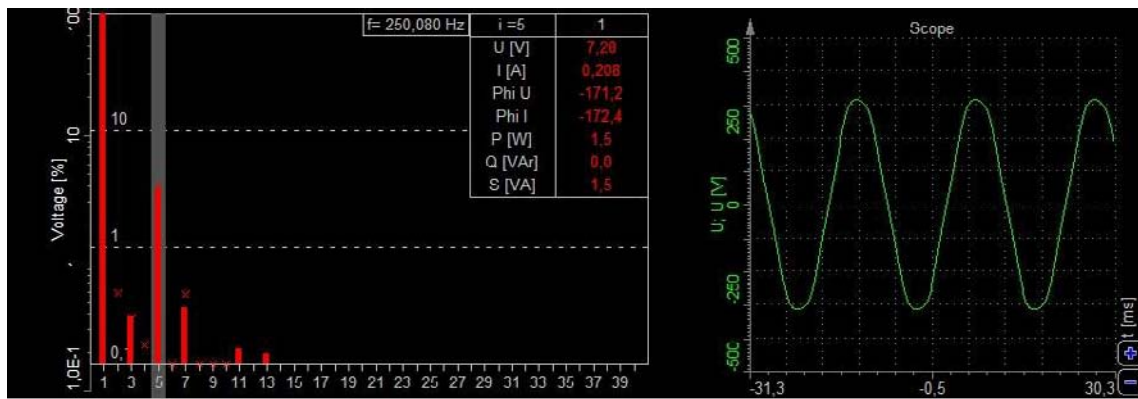
$P=U \cdot I \cos \phi$ 有功功率, 这些才是工作真正使用的电力

$Q_i=U \cdot I \sin \phi$ 无功功率; 这个是被浪费的电力, 因此我们需要尽量降低这个功率

下面我们来计算谐波分量. 理论上, 一个线电压是一个完美的 **50 (或60) Hz 的正弦波**. 既然世界上没有完美的东西, 线电压的波形也会有一些变形, 在频谱分析里就表现为谐波.

下面的图片是线电压信号在示波器里的典型图形. 我们可以看到在示波器里这个电压信号并不是完美的正弦波形, 左侧显示的界面叫做谐波监测仪 **harmonic FFT** 很好的显示出了5个很高的谐波 – 在220V的电网里有7个电压幅值. 那么谐波的作用是什么呢? 试想你有一个交流的马达, 第一个谐波是用来驱动这个马达的, 剩下的谐波则产生了振动和噪声, 而事实上还有的谐波会有更恶劣的影响.

第2, 5, 8... 谐波确实是有害的谐波, 因为它们正在损坏马达. 第 3, 6, 9... 谐波用来驱动或制动马达, 而第 4, 7, 10... 谐波也在驱动马达, 但它们仍然产生很高的噪声和振动. 谐波可以由发电机端或用户端产生, 例如开关电源或变压器的开或关



实际上我们可以计算各个谐波的名义功率，有功功率和无功功率.我们也可以计算次谐波。.

几乎所有的参数都可以计算出来，包括 THD (总谐波失真), 周期值 (每个周期的电压值、电流值、功率值，这对触发有用)和 闪变 (这实际上是一个电力质量的参数，通过测量电压的低频率失真来获得，通俗的讲它可以告诉我们电灯闪了多少次).

3.1.1 通道设置

所需硬件	Any AD card, DAQP-DMM or DAQP-HV, DAQP-V or MDAQ-V or PQL print
所需软件	SE or higher + POWER option or EE
采样率	At least 5 kHz

让我们来测量一个线电压并用电流钳来测量电流,计算功率并加上一些负载, 一个吹风机, 一个普通灯泡, 一个节能灯.



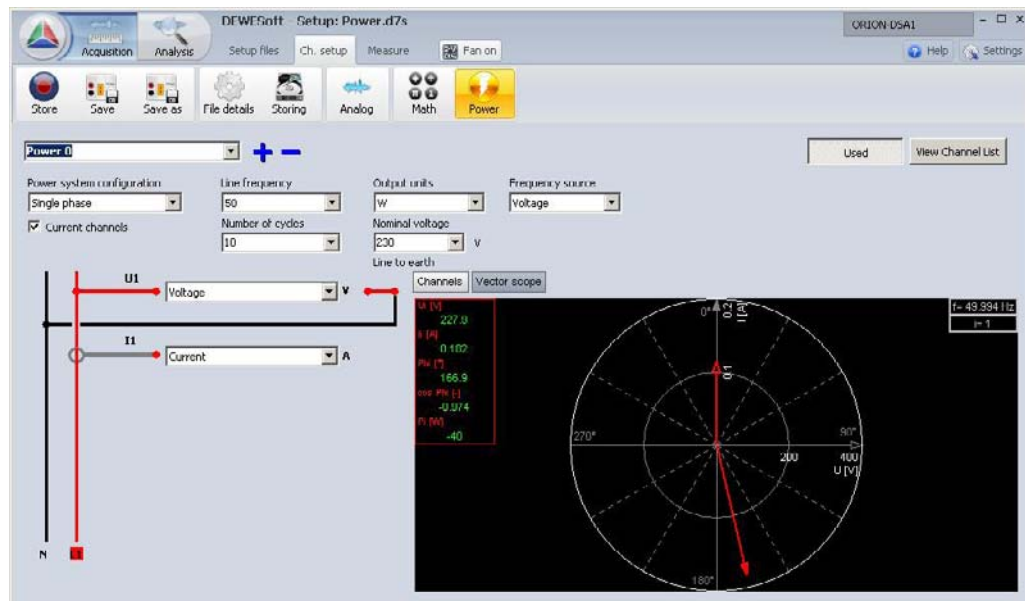
我们在模拟通道 *analog channels* 里象"电压和电流"的章节里的例子一样的设置. 第一个通道是电压, 第二个通道是电流.

SLOT	ON/OFF	C	NAME	AMPLIFIER (007)	PHYSICAL VALUES	CAL	SETUP
0	Used	Voltage	DAQN DMM	400 V	U	-25.0 / 34.9 V	Zero
1	Used	Current	DAQN-V	10 V	I	0.657 / 0.878 A	Set ch. 1

需要注意的是要把电压和电流的单位设置好, 电压的单位可以使 V, kV 和 MV 而电流的单位可以是 A 或 kA.

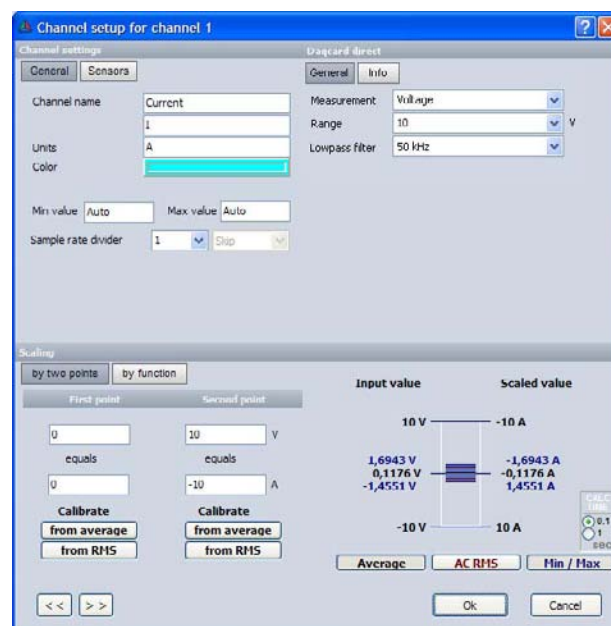
现在我们通过点击 **Power** 标签来到电力模块测试模块, 点击 **+** 按钮产生一个电力测量模块. 首先选择电力系统配置 **Power system configuration** 下拉菜单单相电 "Single phase" 配置, 定义电压和电流的通道. 我们选择 **U** 作为电压通道和 **I** 作为电流通道. 然后选择供电频率 **Line frequency**. 在欧洲通常供电频率为 50 Hz, 在美国 60, 在一些大型车辆和船舶上通常是 400 或 800 Hz. 当选择不同的频率源时 **Frequency source**, 我们也可以测量变频器的频率.

下一步是检查矢量示波器看各个部件是否正确连接. 矢量示波器正好在当前屏幕上. 第一件事情是要看电压和电流的相位是不是正好相反. 因为我们没有发电机, 但是已经连接了吹风机, 我们可以假设我们正在测量一个错误方向的电流.



我们可以通过转换电压插头的极性或者调转电流钳的方向来解决这个问题，但是在有些时候就不会这么简单.如果我们使用不能拆开的电流变压器，最好的办法就是在软件里改变极性. 让我们来到电流的设置界面. 转换极性很简单 – 把 **scaling factor** 由 “+” 改为 “-” 即可.

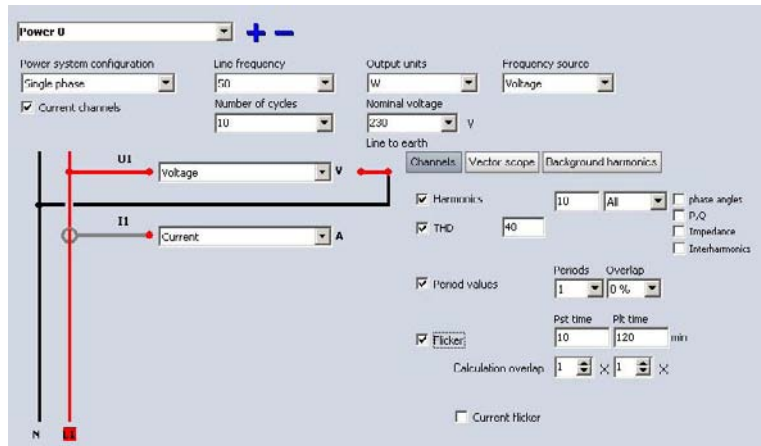
10 V的测量值实际上意味着输入是 -10 A .



现在都准备好了，现在的相位差是10度.

现在看我们有哪些通道可用. 基本的参数象频率 *frequency* 或功率 *power* 通道 (P, Q, S, phi, cos phi) 会自动添加. 所以如果我们只是测量一些基本的电力参数，我们不用选择任何东西.

如果我们想得到更多的参数, 我们可以选择计算谐波 **Harmonics**, 但是即使你没有选择它们, 这些值依然可以在谐波监测仪 *harmonic FFT* 和矢量示波器里 *vector scope* 看到. 唯一的不同，举个例子，就是我们不能在记录仪里显示电压的第三个谐波 我们也可以选择总谐波失真 **THD**, 周期值 **Period values** 和 闪变 **Flicker**.



现在我们将开始测量

3.1.2 测量

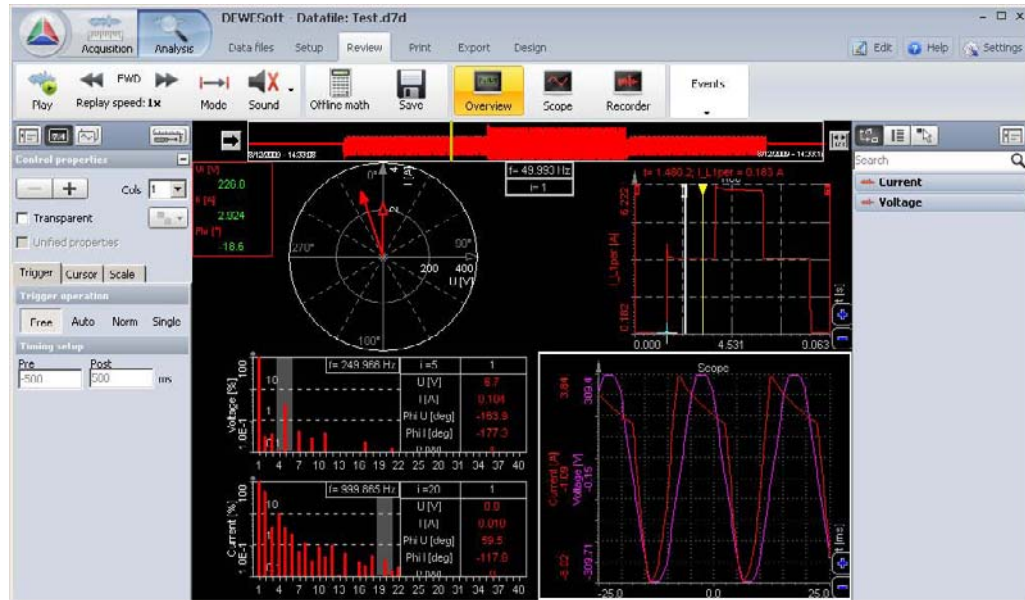
让我们来到 *overview* 界面来设计一个电力参数的测试界面. 如果我们有一个电力测量模块, 那么我们就有了矢量示波器 *vector scope* 和谐波监测仪 *harmonic FFT*.



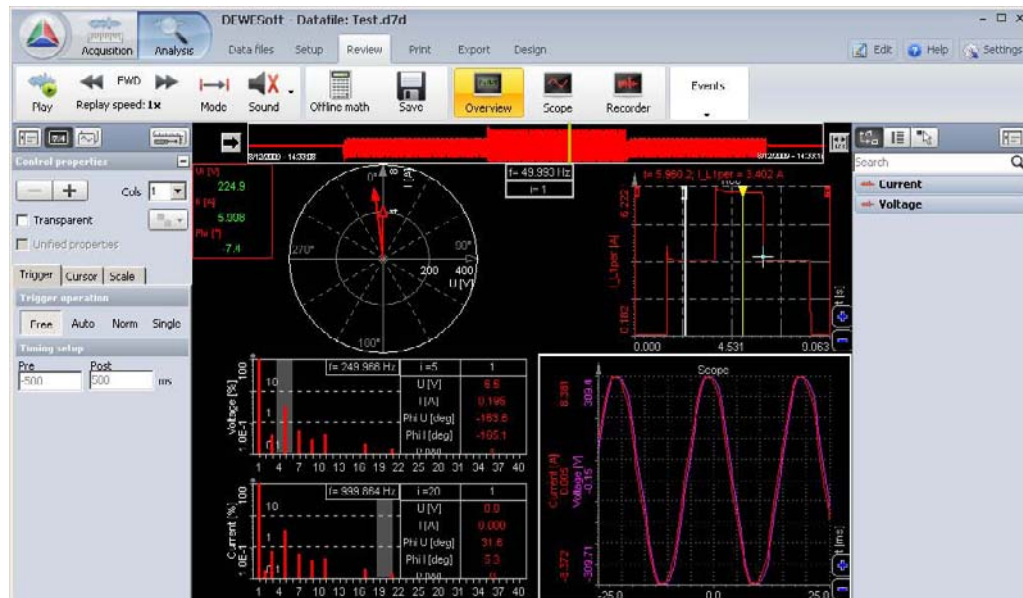
所以我们将矢量示波器 *scope* 放在顶部, 它用来表示电压和电流的相位关系. 我们可以看到电压和电流之间有一个相位角, 这是因为吹风机里的马达. 测试界面上部右侧的记录仪可以显示电流的周期值, 表明吹风机是开还是关.

底部右侧的示波器显示的电压和电流, 我可以清楚的看到吹风机以半功率在工作. 较低部分的电流被截止. 我们可以在一些速度较高的设备例如电钻, 会更好的看到这一点.

在底部左侧有俩个谐波监测仪 *harmonic FFTs*, 上面的那个显示电压谐波, 下面的那个显示电流谐波, 我们可以清楚的看到第5个谐波的点点大概是线电压的7%, 而电流因为有调节电路的原因而有许多谐波.

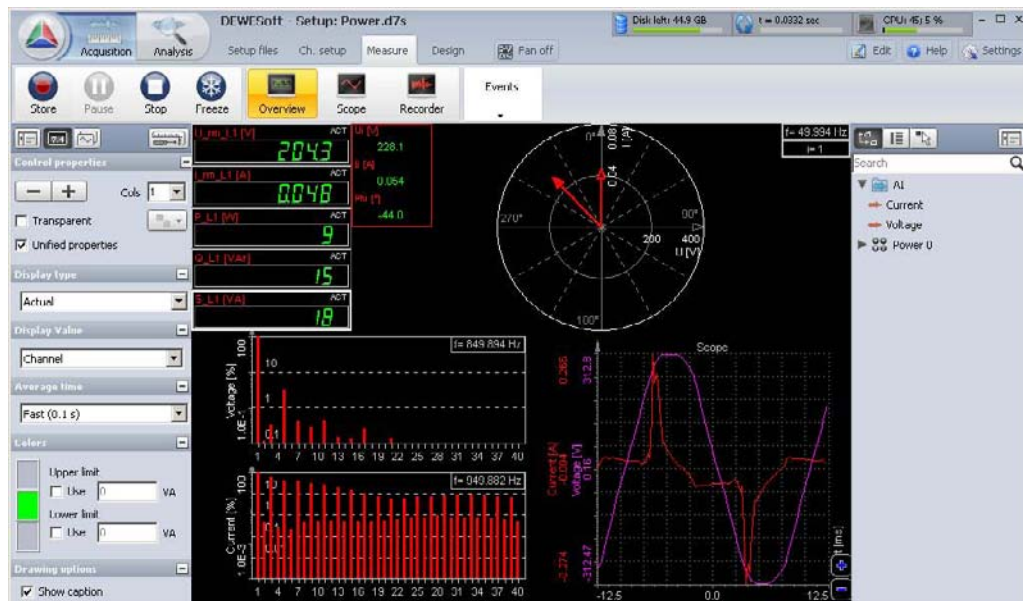


现在我们可以观察整个电力的参数，全循环里的电流和电压电流的相位角 更低了一些，电流谐波也大幅下降基本跟电压谐波保持了一致。

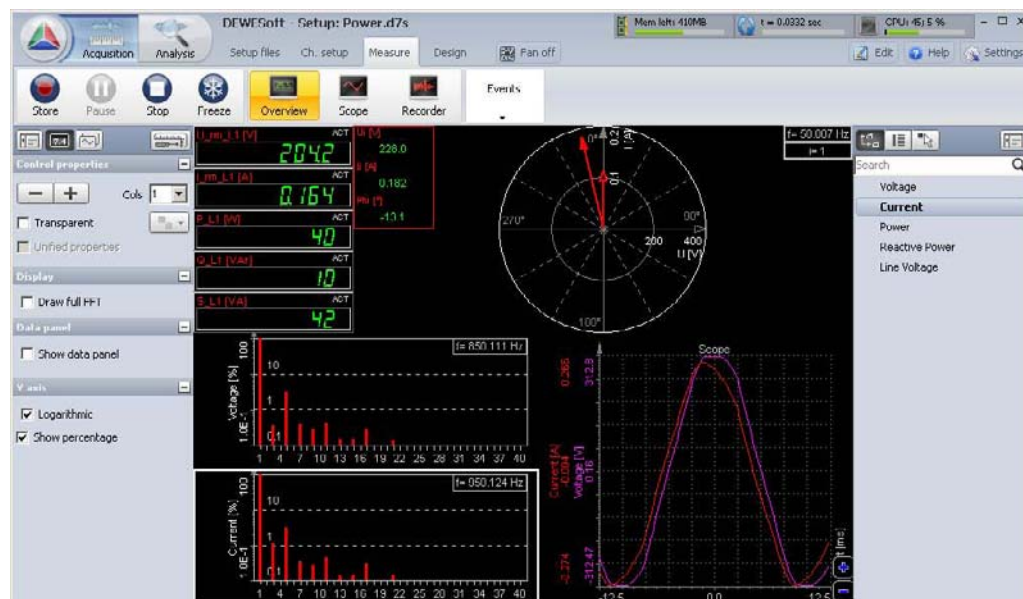


现在我们来做一个测试 – 观察普通灯泡和节能灯泡的区别。

首先我们连接节能灯泡. 我们使用一个11 W 的节能灯泡, 大概相当于 60 W 的普通灯泡. 整个功率事实上非常低, 但是电流的波形不是一个真正的正弦波. 因此会有很多的电流谐波, 这样就会“污染”电网。



现在我们再来看一下传统的普通 40 W 的灯泡. 首先注意到在电网上的负载基本和电网电压是线性的. 测量的功率是 42 W, 但是矢量示波器 *vector scope* 看上去很奇怪. 事实上, 因为普通灯泡基本上是一个纯的电阻负载, 电压和电流应该完全对齐, 但是就像我们看到的, 它们没有对齐, 这是为什么呢? 记得在电压和电流的章节里我们看到了电流钳和精密电阻测电流的区别, 因为我们现在使用的是电流钳, 就有了幅值和相位的误差. 所以电流钳在这个例子了是运算误差的主要原因.

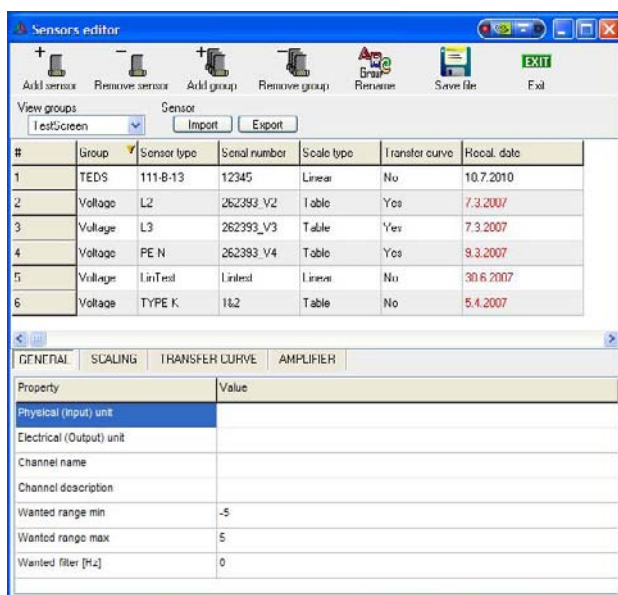


在 **DEWESoft** 里, 我们可以补偿这个误差. 下面我们来看一下.

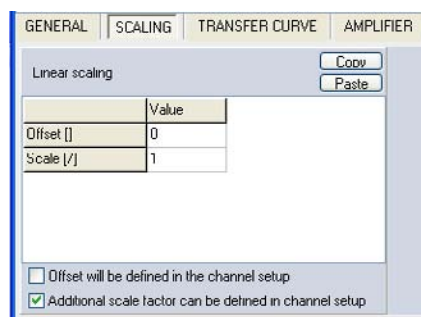
3.1.3 传感器修正

很明显我们要告诉 **DEWESoft** 我们的传感器不完美并且要把传感器的传递曲线输入进去. 传递曲线给出了传感器在特定频率下的相位和幅值的误差, 这样 **DEWESoft** 就可以补偿这些误差.

让我们选择Let's choose the **Settings- Sensor editor...**菜单. 我们得到一个传感器列表. 现在我们输入传感器类型 **Sensor type**和序列号 **Serial number**增加一个传感器. 选择 **General**标签, 输入 **Physical (input) unit**, 在我们的例子里 是**A** (安培) 和 输出**Electrical (Output) unit**, 是 **V** (电压).



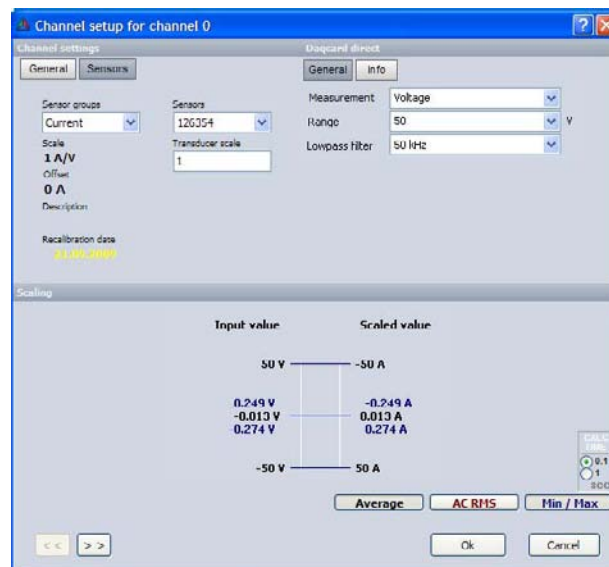
现在让我们来输入换算**Scaling**系数. 因为传感器在幅值上是线性的, 我们只需要输入换算系数, 在我们的例子里就是**(1A=1V)**. 我们也可以在通道设置里再定义一个换算系数来转换极性



现在我们来进行最重要的部分- 传递曲线的定义. 在传递曲线 **Transfer curve** 的表格里选择**Yes** 来表明我们将要定义传递曲线, 现在我们输入曲线的各个点. 我们需要输入**a[dB]** – 以 **dB** 为单位输入幅值偏差和 **fi[deg]** – 以度为单位输入相位误差. 下一个问题是: 到哪得到这个传递曲线? 这里有很多以前使用过的通用的传感器的传递曲线, 所以有时候可以先找一找是否有现成的传递曲线可用. 第二个选择是从传感器的校准证书里拷贝, 如果校准证书里已经有传递曲线的话. 第三个选择是用 **DeweFRF**来测量,但这需要一些设备. 我们得到这些传递曲线后, 我们需要把它输入到表格里, 在我们的例子里我们可以看到电流钳在 **50 Hz** 的时候相位误差是**10 度**, 这样我们就明白刚才试验时所看到的结果了. .

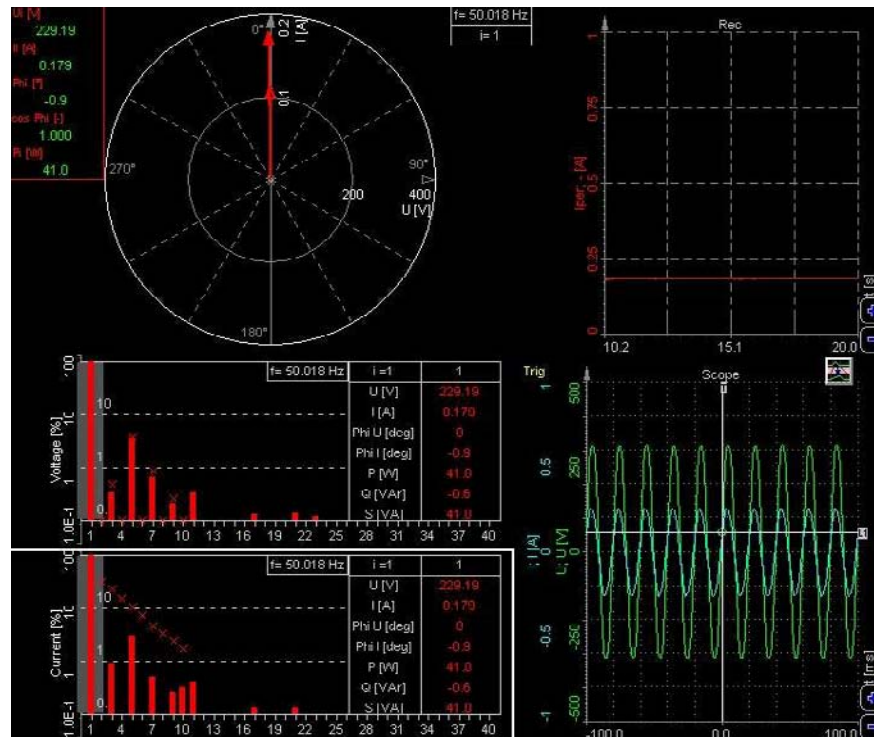


点击 **"Save file"** 按钮存储传感器，然后关闭传感器编辑，退出 **Exit**. 现在我们回到模拟通道设置，选择电流通道的传感器，打开传感器 **Sensors** 标签，选择 之前输入的那个传感器序列号. 现在什么也不用做，我们只需要在换算系数里把极性转换一下，我们只需要在转换系数里输入 **-1** 即可.



就是这样，我们不要定义传感器，只需要从传感器列表里选择即可。现在看看我们的测量结果如何，结果很明显，电压和电流的相位几乎重合在一起，功率的计算也正确了.

从这个例子我们可以看出，如果不考虑传感器的特性会给测试结果带来多大的误差。顺便说一句，放大器和多路复用的AD卡也会产生相位漂移。但是 **DEWESoft** 可以自动修正这些误差。借用这些小技巧我们电力测量的误差可以低于 0.1%。



4 动态信号分析教程

动态信号分析覆盖的测量范围非常广，包括结构动态测量，工业噪声分析以及**m**机器故障诊断。

频率响应函数测量的说明在另外一本操作手册中单独阐述 - [FRF 用户指导](#)。

	声级计	详细解释声级计模块 Sound level module 的工作过程，声级计模块用来计算噪声级，并提供时间计权和频率计权
	扭振测量	详细解释扭振模块的测试工作过程，扭振模块用来测量转动轴的动态和静态弯曲和振动
	人体振动	详细解释人体振动模块的测试工作过程，人体振动模块用来评价振动对人体的影响
	阶次跟踪	详细解释阶次跟踪模块的测试工作过程，阶次跟踪模块用来在机器升速和降速实验过程中抽取谐波

4.1 声级计

声级计数学模块用一支传声器来计算声级测量的典型参数。它允许**DEWEsoft**软件当做一个典型的声级计使用。只要有适当的硬件(比如ORION 1624 AD卡和MDAQ-ACC模块)，就可以非常容易满足一级Class I声级计的要求。

所需硬件	24 位 AD 卡 (Orion), MDAQ-ACC 或者 DAQP-ACC模块
所需软件	SE 或更高 + SNDLVL 选项, DSA 或者 EE
设置采样率	至少 10 kHz

只要选取**声级计Sound level meter**复选框，声级计功能就可以实现(**System→Hardware setup →Math**标签)。选择该选项后，**DEWEsoft**硬件设置**Setup screen**窗口上会出现一个名叫**"Sound levels"**的新标签。

声级计测量的基本程序如下：

- 对使用的硬件进行[通道设置](#)
- [传声器校准](#)
- [测量](#)

4.1.1 通道设置

要使用声级计模块，请首先在[模拟Analog](#)标签中选择一个或者几个模拟通道，使用它们来测量声音。

然后打开[声级计Sound levels](#)标签，点击按钮，增加一个新的声级计**Sound levels**模块。在一个应用窗口中可以加入几个声级计模块。

屏幕上会出现如下图所示的窗口。首先选择输入通道①，这些通道显示在窗口的左上角。我们的例子中，只选择了AI 0。我们当然可以选择多个通道，然后用同样的设置来设置输出通道。

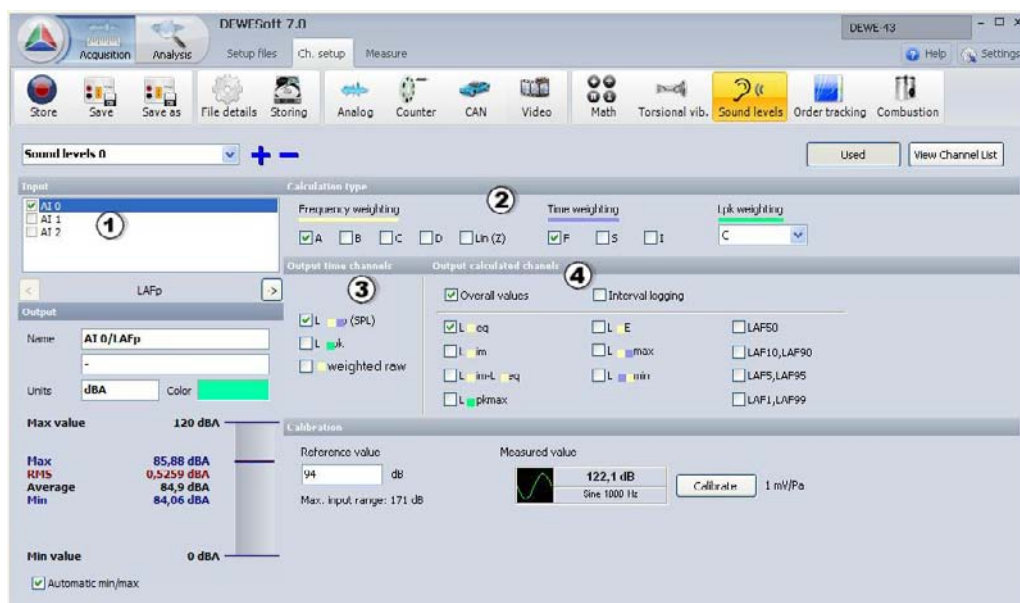
在计算格式Calculation type窗口②中，还有许多选项可供选择。可以以不同的组合形式来选择频率Frequency和时间计权Time weighting方式。对于线性Linear，A以及C计权，我们还能选择特定的Lpk计权。

我们提供三种形式的输出时间通道Output time channels③。首先是L_{FTp} -时间和频率计权声压级，其单位已经设置为dB。其次是L_{pk}值，该值显示的是当前声级的最大值，最后是L_{Fweighted raw}值，它用Pascal帕斯卡为单位显示声音的频率计权时间曲线。

接下来需要选择输出计算的通道Output calculated channels (参数)④。这些参数可以是总值，也就是说在测量结束后只有一个值，要么是间隔式记录。如果我们需要间隔记录Interval logged参数值，连续记录的时间间隔在旁边的输入框中以秒为单位来定义。比如我们选择每隔5秒钟间隔来记录Leq值，那么我们可以每隔5秒钟得到一个新的Leq值。也就是说每隔参数值重新设置一次，计算重新开始。

计算出来的值包括以下参数：

- 频率计权 L_{Feq} 值，即等效噪声级
- L_{Fim} 表示噪声的脉冲级，即脉冲计权噪声级；两者之差按如下公式计算 L_{Fim}-L_{Feq}
- L_{pkmax} 是用 C 或者线性计权噪声最大峰值
- L_{FE} 是 频率计权声能量
- L_{FTmax} 和 L_{FTmin} 是时间和频率计权的最大和最小声压级
- 声音的另外一部分是声级。每个计算出来的声级按分类分组，我们可以从LAF1, 5, 10, 50, 90, 95 或者 99 %等分组中选择



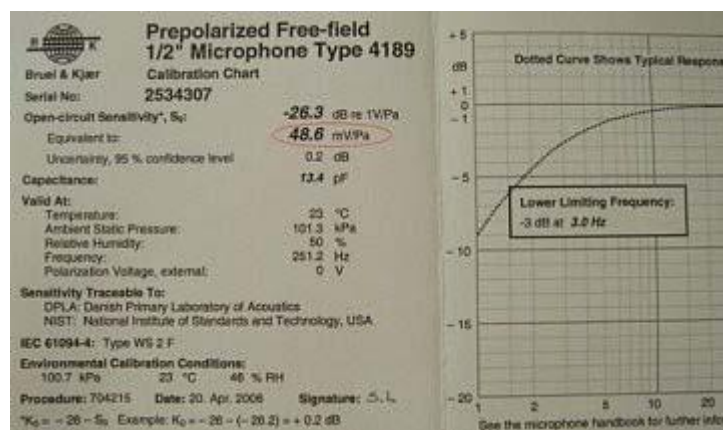
现在来看看如何校准传声器。

4.1.2 传声器校准

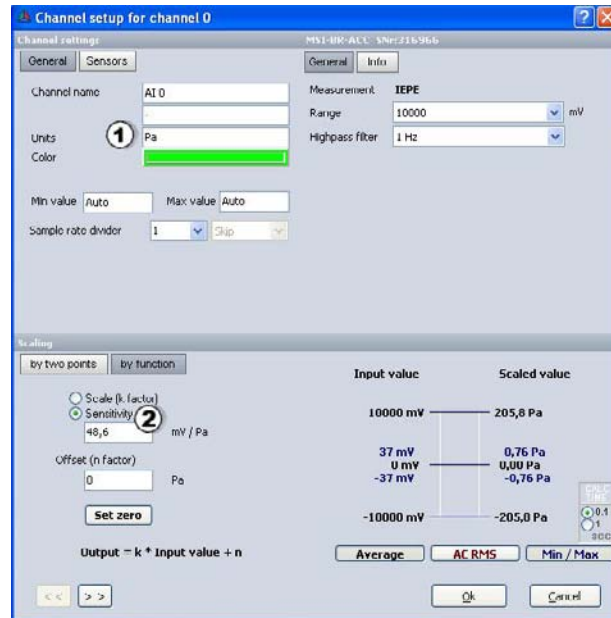
传声器的校准有两种方式。首先我们已经知道从传声器直接测量得到的声级值单位为Pa。因此我们必须将其物理量的单位重新设置。

利用传感器校准表格进行单位设置

如果我们不能使用校准器进行校准，但是我们已经有了传感器的灵敏度值，那么可以直接在通道设置中定义。



首先设置测量的物理单位，即定义测量的实际单位为Pa①。然后进入Scaling by function界面，选择灵敏度Sensitivity，输入单位为V/Pa②的灵敏度值，可以在传感器校准表中找到该值。



用校准器校准传声器

另外一种传声器校准方式是使用校准器，这种方式校准器发生的声级和频率参数是已知的。

我们在此处用的是 94 dB。

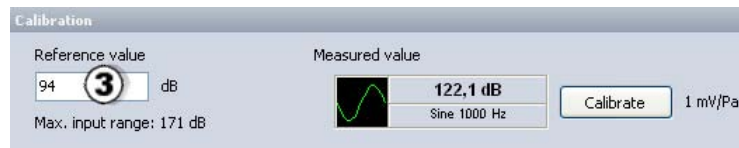


这个值可以在声级计模块窗口中的参考值域Reference value中直接输入即可。然后将传声器套入校准器空腔中，打开校准器开关。我们可以在小窗口中直接观察到校准器的信号。此例中的校准器信号为频率1000 Hz的正弦波。因为所有计权曲线都是以1000 Hz为参考的，即在1000 Hz处计权值为0，所以这个频率用于校准传声器非常有用。

注 为了能够顺利进行声学校准，我们需要在DEWESoft 软件中将采样率最低设置成 5 kHz或者更高。采样率的设置可以在DEWESoft Tuner 中完成



当我们在软件商正确识别到这个1000 Hz的正弦信号，就可以点击校准**Calibrate**按钮，进行校准。声学模块将计算最大FFT幅值，以及传感器的参考灵敏度值。

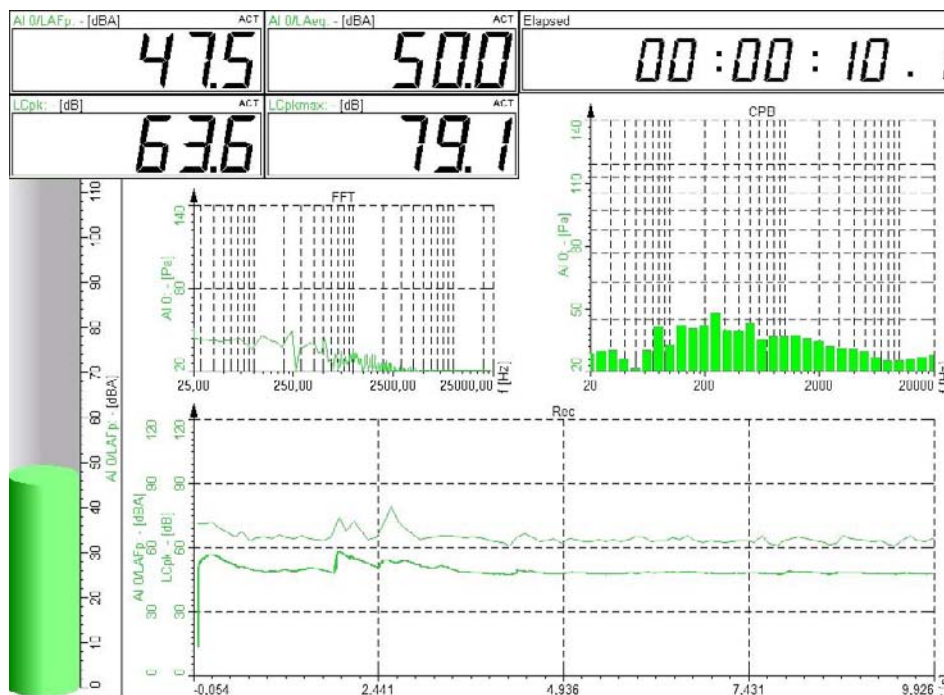


计算得到的灵敏度将直接写入源通道，因此没有必要再进行模拟信号的单位设置。我们还可以直接检查校准后的灵敏度，和传声器提供的校准表上的信息进行对比。

4.1.3 测量

声级计模块不会自动创建任何显示。用户必须自己创建设置，显示声级计通道。下图所示是声学测量的标准显示窗口。我们

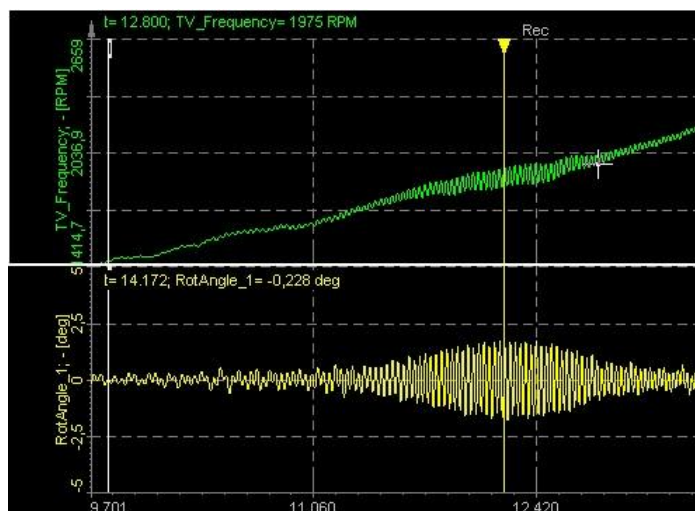
使用典型的A计权来计算Leq和Lp，这两个值可以用数字表，棒状图显示或者记录仪直接记录。另外，同时还计算倍频程和窄带FFT。可以增加一个FFT窗口来显示该分析，在FFT分析仪，可以定义频率计权方式，比如A计权，以及用dB作为显示单位。



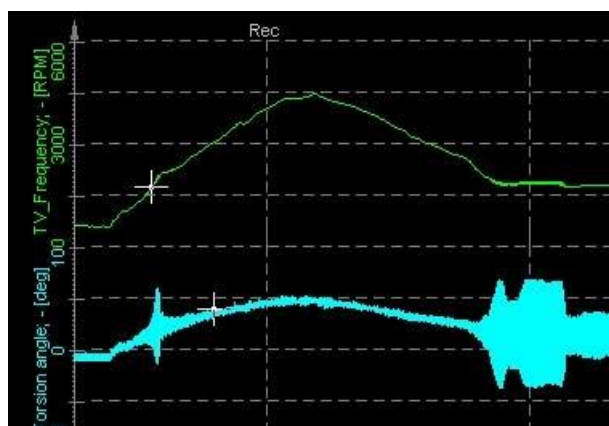
4.2 扭振

首先我们必须清楚了解，用扭振模块我们可以进行两类完全不同的测量：即旋转振动和扭振。那么这两者之间存在什么区别呢？

旋转振动只是简单的测量转动速度的动态差异。如果我们用高精度的转速传感器来测量转子的转动速度，就会发现在升速过程中有某些转速部分出现了明显的转速变化或偏差。这种情况发生的原因是有角度方向上的振动接近了转子的角度固有频率。计算时，需要用滤波器滤掉转动速度或者转角的DC直流成分。从下面升速测试过程中某一部分的放大图所示，我们可以清楚发现在这一频率附近转速差异非常大，黄色曲线显示的是角度差，实际上也就是转动角度振动。



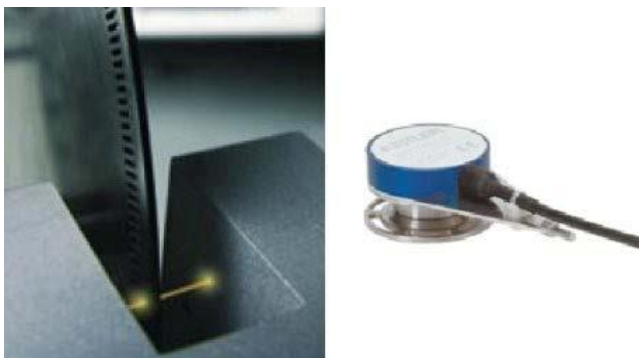
扭振是发生在齿轮箱，曲轴或者是离合器等系统中的旋转部件上的角度域上的振动(扭动)。我们需要两个转角解码器来测量扭振，实际上扭振是两个转角传感器之间的角度差。在高速RPM时，扭振的测量同时也能测量曲轴转子的静态扭动。下图所示为升速和降速的整个过程，从中我们可以发现非常清楚地曲轴转子静态扭动，当振动频率接近曲轴转子的固有频率时，转子的角度方向上的振动变得更加剧烈。



要测量扭振或者旋转振动，我们需要用到带有计数器扩展的Orion卡，因为其它方法都不能满足这种测量的精度要求。

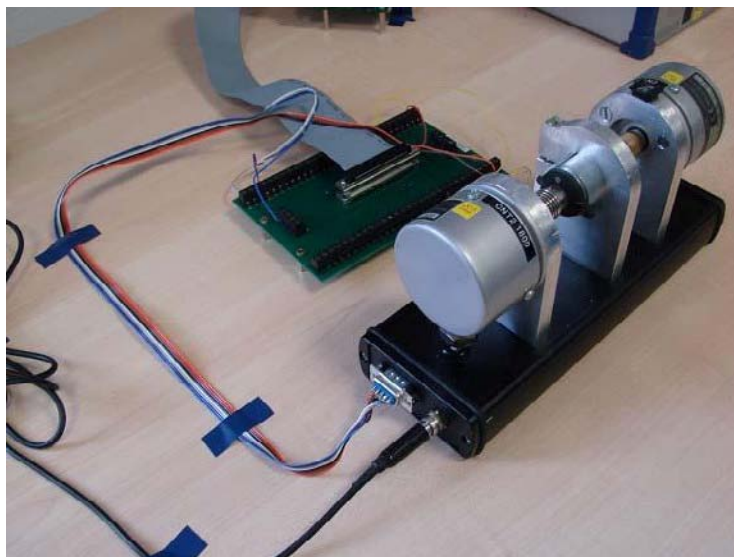
所需硬件	带有计数器扩展的24 位 AD 卡 (Orion)
所需软件	SE 或者更高 + TORVIB 选项 (如果需要抽取阶次，还需要ORDTR 选项)，DSA 或者 EE
设置采样率	至少 10 kHz

扭振和旋转振动可以使用解码器(每转最高达3600个脉冲)或者是分辨率不高(每转最高720个脉冲)但对振动不敏感的特殊转速传感器，在这种条件下标准的解码器很容易损坏。



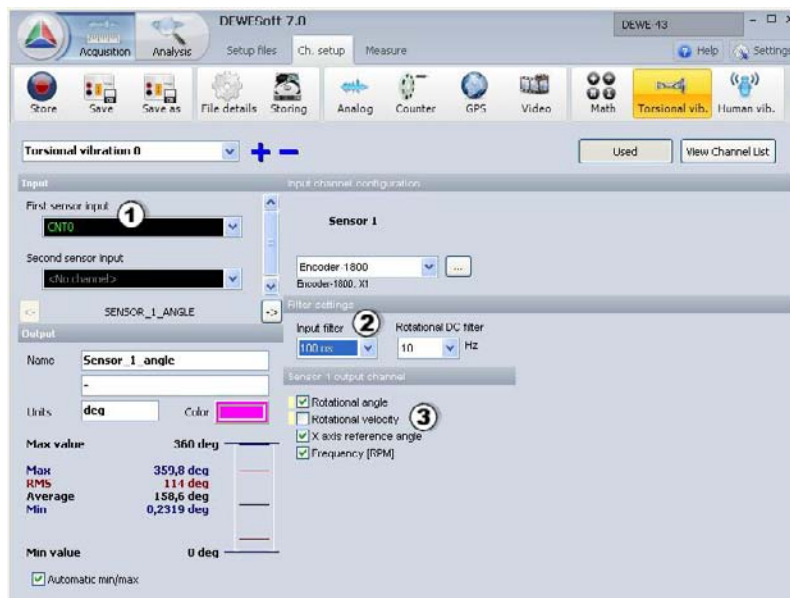
4.2.1 旋转振动设置

对于我们的测试而言，我们在一个小电机两个各安装了一个解码器，中间用弹簧连接，这样可以产生分厂大的扭转。两个解码器都连接到计数器输入上面。首先我们来做旋转振动。



我们无需设置任何模拟或者计数器通道。我们打开扭振Torsional vibration标签，在标签上点击+按钮可以增加一个新的扭振模块。接下来选择第一个传感器输入First sensor input①。因为我们把第一个转速传感器连接到B0_CNT0计数器，那么就需要从列表中选择这个通道。请注意CNT1，CNT3以及CNT5不能使用，因为这些通道是在内部用来计算频率使用的。

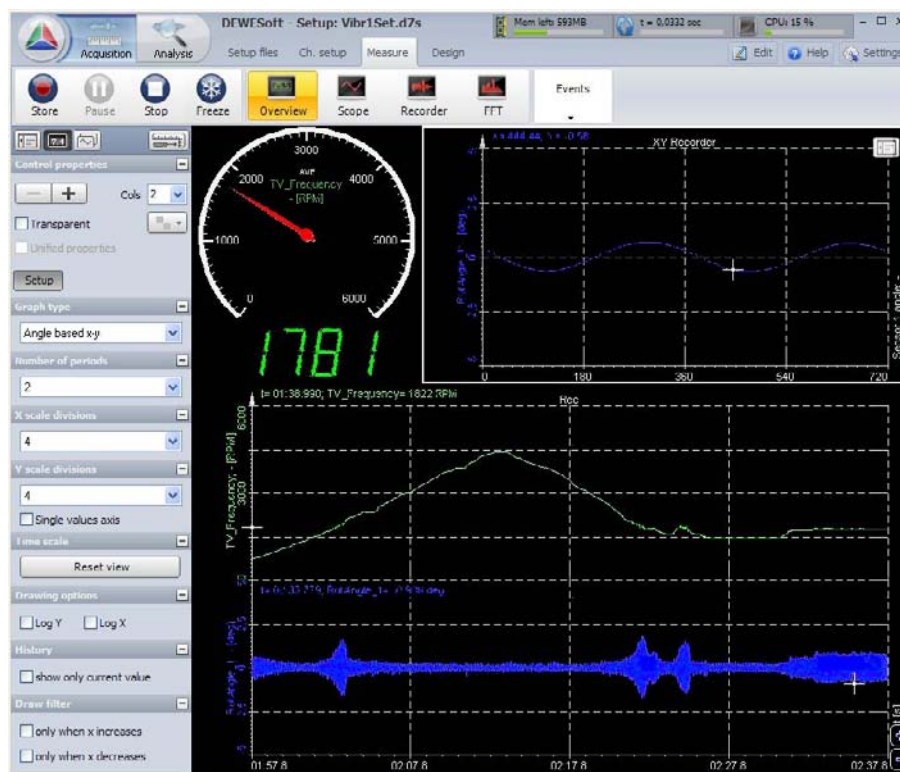
接下来定义**传感器Sensor**。在我们的例子中，我们的传感器是每转提供**512**个脉冲，所以需要选择**Encoder-512**。如果我们使用的传感器目前还没有定义，那么我们就需要在计数器传感器编辑器中进行创建，点击传感器下拉列表旁边的三个椭圆形符号的按钮即可。下一步需要为计数器设置**输入滤波器Input filter**。输入滤波器的目的是为了隔离掉信号中的短时脉冲以及毛刺。**旋转DC滤波器Rotational DC filter**是用来过滤掉RPM信号中的DC直流成分。我们需要设置好滤波器，以便能使所有关注的频率都能通过，但是过低频率则不可以，否则输出信号②上会出现一个静态的DC直流偏差。



输出通道的信号 就是**旋转角Rotational angle** (滤波后的角振动值)，**转动速度Rotational velocity** (滤波后的振动速度值)，X 轴参考角度(参考角度，永远是在0到360°之间，可以在xy图中用作角度的参考)以及以**RPM**为单位③的**频率**。

4.2.2 旋转振动测量

扭振测量模块没有标准的测试界面，所以需要我们自己增加新的测试界面。在这里，我增加了一个模拟表和一个数字表显示 频率，用记录仪来记录频率和转动角度。在分析窗口的右上角，我们还有一个xy记录仪，该记录仪运行再一个特定的模式叫做 "**基于角度的x-y Angle based x-y**"。对于**x轴**，我们需要给它选择Sensor 1 Angle(从0到360°的参考转角，指示当前的转子 旋转角度)，对于**y轴**，则选择旋转角。这种xy记录仪现在显示的就是当前转动的旋转角。类似于示波器，不过这里是用角度作 参考，而不是用时间。



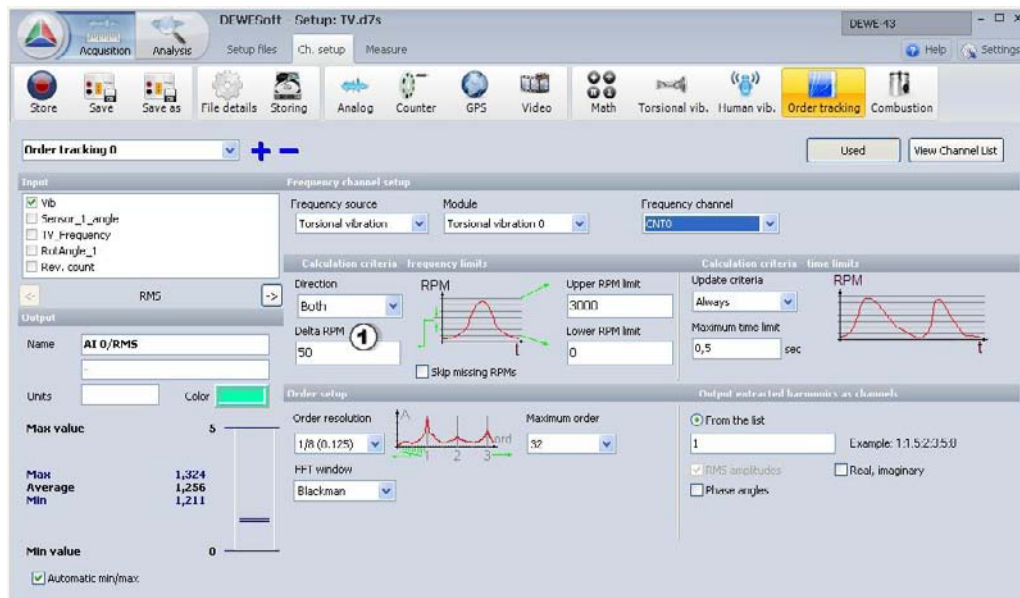
我们可以从下方的记录仪上看出转子转动共振频率大概在1800 RPM左右。也可以从从这些数据观察抽取的阶次。要抽取阶次，我们还要用到阶次抽取order extraction模块。

4.2.3 旋转阶次抽取

要从旋转振动中提取阶次，需要用到阶次跟踪模块。对于频率源，我们需要定义扭振模块，选择模块，然后选择频率通道(如果激活了扭转，就在模块内部)。

下一步需要定义转速RPM的上限Upper和下限Lower limit。这是为了用来给FFT瀑布图预留足够的内存。FFT瀑布图按照"Delta RPM"设定的步进转速①从低速到高速进行绘图。因此，在本例中我们将得到 $(3000-0) / 50 = 60$ 个FFT频谱组成的瀑布图。

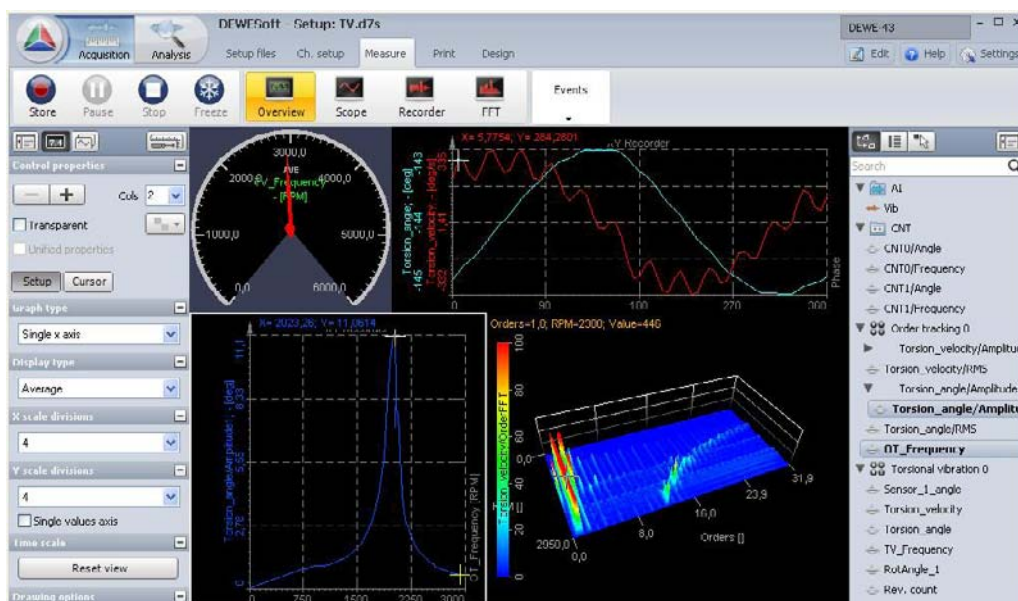
我们选择抽取最前面的三个阶次。



现在我们来做一些测试。我们现在已经有了频率谐波通道，因此可以加入一个x-y分析窗口，显示阶次。OT_Frequency必须用作x轴，作为参考(x-y分析第一个通道选择)，H_1，H_2，H_3三个阶次用作y轴。然后我们可以进行一次降速实验来观察阶次，我们可以非常清楚地看到在1800 RPM处出现的峰值，这个峰值在先前的记录仪中记录的第一阶振动中也看到过。

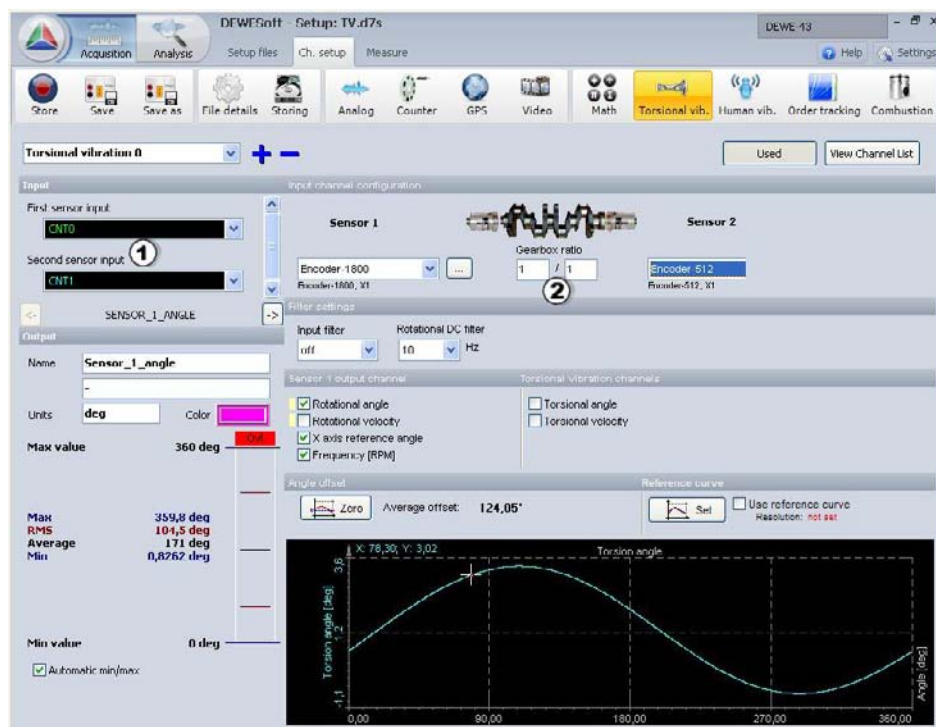


为了取得更好的显示效果，我们可以在此处增加FFT频谱分析仪，在History显示模式线选择History FFT。将History Source设置成阶次跟踪Order tracking②，并选择Rot_Angle 1通道作为数据源。现在我们进行升速和降速实验。因为设定的颜色仅仅在某些特定的频率和转速RPM时才能显示，所以我们必须仔细选择幅值大小，以便能更好显示阶次频谱图。建议使用对数显示来更好显示非常小的幅值。



4.2.4 扭振设置

接下来我们来进行扭振测量。要测量扭振，需要在**Torsional Vibration**扭振设置标签中选择两个输入通道。在第一个通道中选择**B0_CNT0**，在第二个通道中选择**B0_CNT2**①。然后定义**Sensor 1**和**Sensor 2**。如果在两个传感器之间还有变速箱连接，我们还需要输入**变速比Gearbox ratio**②。



输入滤波器**Input filter**的设置范围可以在**100ns**和**5μs**之间。最优的设置值可以从下面的公式中计算得到：

$$\text{InputFilter } r[s] \leq \frac{10}{RPM_{MAX} \times PPR}$$

RPM_{MAX} 每分钟最大转速 [s^{-1}]

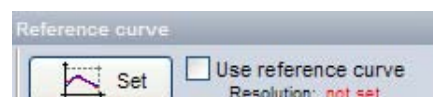
PPR 每转脉冲数 [-]

角度偏移 / 参考曲线配置

点击**Zero**按钮，清除掉两个转速传感器**Sensor**(角度偏移设置为0)之间的角度差(偏移)。



点击**Set**设置按钮，记录某一转当前的扭转角 作为参考。当**Use reference curve**选项被选取时，刚才记录的参考扭转角会从当前测量得到的扭转角中减掉！这样我们就可以克服由于转速传感器以及传感器安装等原因造成的扭转角错误！



这里有两种额外的通道类型：

- **扭转角**，是动态扭转角，也即是Sensor 1和Sensor 2之间的角度差
- **扭转速度**是Sensor 1和Sensor 2之间的角速度差。

4.2.5 扭转测量

现在开始进行测量。在屏幕中我为频率通道增加了模拟和数字表，频率，扭转角，扭转速度则利用记录仪记录，同时还为扭转角和扭转速度增加了一个xy记录仪。

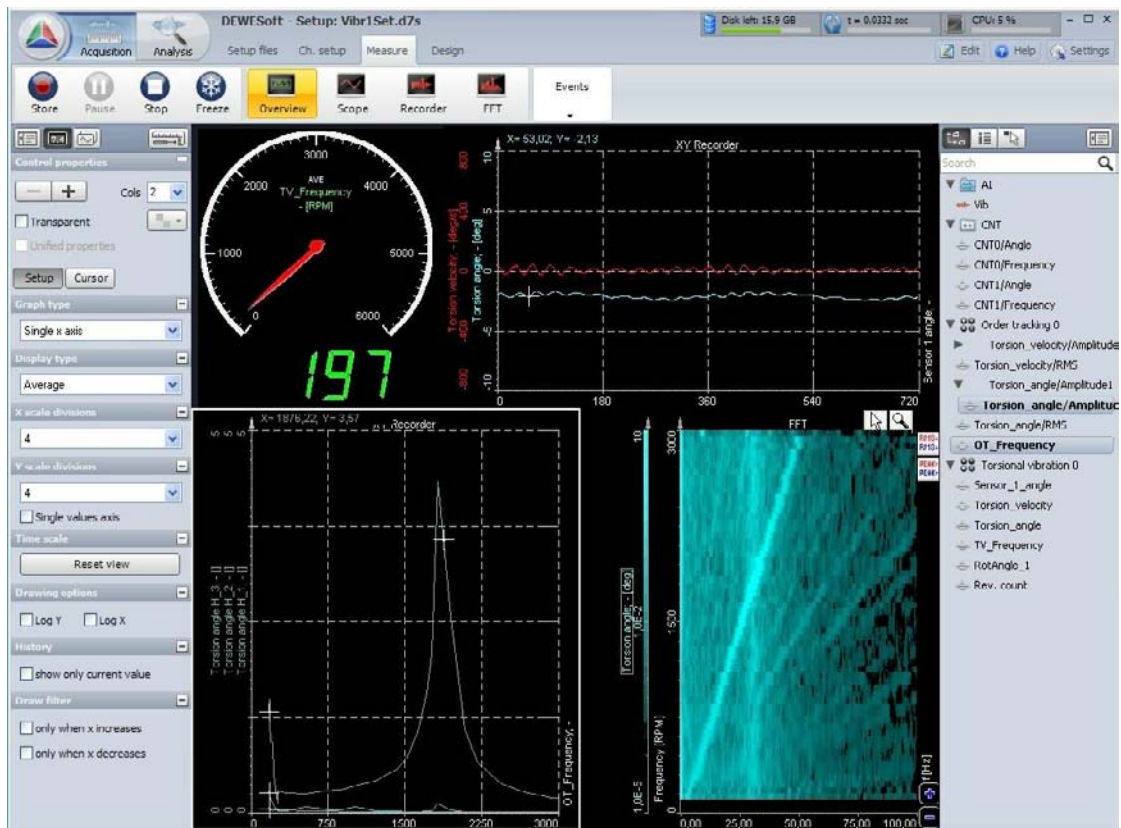
升速实验和降速实验可以清晰显示静态扭转角和转子角度域上的固有频率所处区域，这时的扭振值都非常高。



如果使用阶次跟踪order tracking的话，可以得到更好的结果。

4.2.6 扭振阶次抽取

通常扭振阶次抽取的设置和旋转振动的设置一样，除了这次我们选择的扭转角Torsion angle或者扭转速度Torsion velocity，并且将其分配到阶次跟踪数学模块中的输入通道上。FFT瀑布图很清晰的显示出固有频率，在XY记录仪的阶次中已经得到验证了。



4.3 人体振动

人体振动测量时为了研究振动对人体的影响。尤其是在某些振动非常大的工作场合，可能会对人體某些部位造成永久性的伤害。其中大家比较熟悉的是**Raynaud**病，或者叫做白指病，得这种病的人手指变白，而且伴有疼痛。第二种大家比较熟悉的影响是长期工作在重型机械和车辆(另外一个典型例子是直升机)上的人员容易有腰椎问题。

人体振动模块提供了如何判断这种伤害到达何种程度的一种测试方法。它满足**ISO 2631-1**(发布于1997年)标准，该标准定义了测试的基本步骤，**ISO 8041**(发布于2005年)标准，该标准定义了测量的精确过程，以及**ISO 2631-5**(发布于2005年)标准，该标准定义了如何计算人体脊柱对振动的响应。



人体振动测量主要包括两个方面的内容：全身振动和手臂振动。两者都利用三轴振动传感器(通常使用量程为**50 g**的传感器)进行测量，并且要用到特殊的适配器。对于某些振动量级非常高的场所，(比如冲击钻的测试)，有必要使用**高g值**的振动传感器(**500 g**或者更大)。这类传感器通常也能抗高冲击。

对于这种测试，需要用到几个**ICP**通道，**24位sigma-delta AD卡**(比如Orion1624 或者 DSA41 AD卡)。

全身振动测量

全身振动测量需要用到所谓的坐垫式传感器，在坐垫式传感器的橡胶垫里面安装了一支三轴加速度传感器，所以我们可以坐在上面。因为**z**轴和**x**和**y**轴的计权方式完全不一样，所以垂直方向上的**z**轴更加重要。



手臂振动

第二种应用是测量手臂的振动，这种应用中传感器首先要装在特殊的适配器上，然后再安装在工具或机械的手柄上或者是夹在手指之间。因为这种应用中三个方向的计权方式完全一样，所以传感器的方向就不那么重要了。

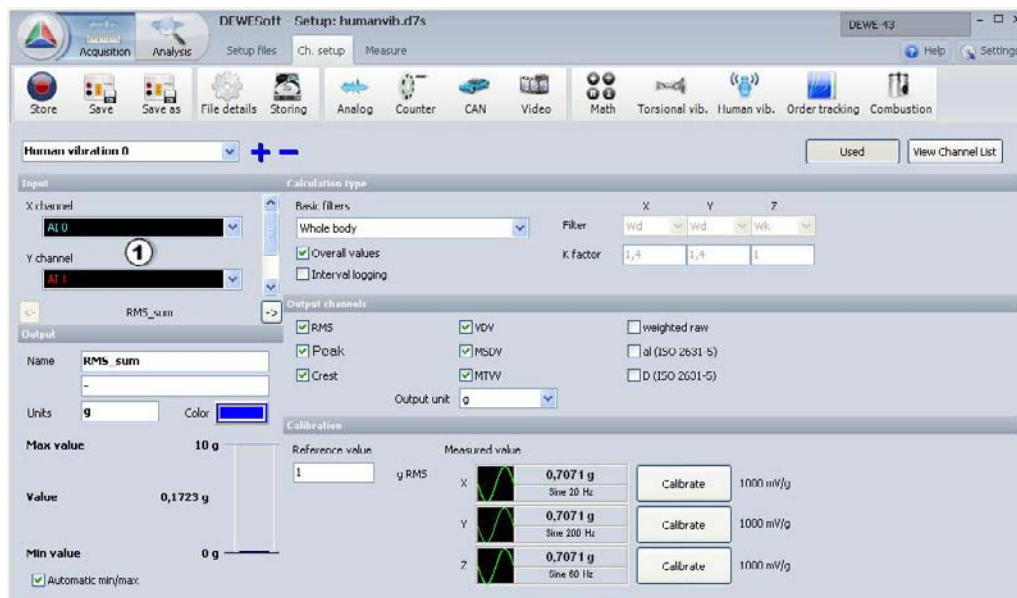


理论上讲，我们需要在所有重要负载的情况下测量一个完整的工作日。但是通常测量的间隔很短，但是我们必须保证在整个测量过程中所有重要的振动类型都已经包含在内了。

4.3.1 输入通道设置

所需硬件	24 位 AD 卡 (Orion 或者 DSA41)
所需软件	SE 或者更高 + HBV 选项, DSA 或者 EE
设置采样率	至少 5 kHz

要使用人体振动模块，要在**模拟Analog**标签上首先至少选择三个振动模拟通道。然后打开**人体振动Human vibration**标签，点击**+**按钮，增加一个新的人体振动**Human vibration**模块。一个设置中可以使用几个人体振动模块，每个模块我们都需要占用三个通道。



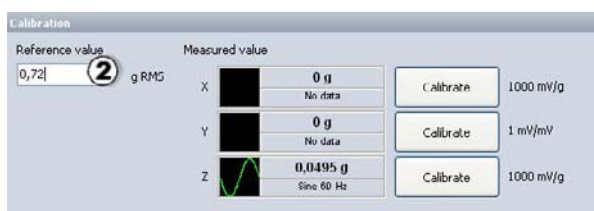
如果我们手上有传感器的校准表格，在模拟输入设置窗口中(请参见[振动测量](#)一章，了解如何在模拟通道设置中进行传感器校准的操作)输入传感器的灵敏度。下一步是在人体振动模块输入窗口②中指定传感器。我们可以在校准窗口看到实时的数据显示。

准窗口看到实时的数据显示。

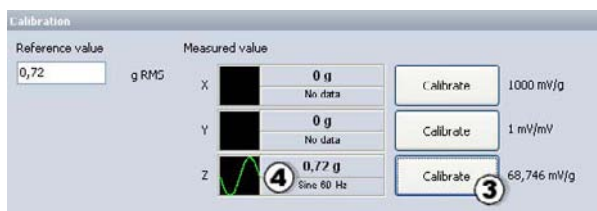
校准

如果我们需要用校准器进行校准，我们可以在这里进行 – 即在人体振动模块内部进行。首先输入振动的参考值Reference value②。这个参考值可以从校准器的校准表中拿到。通常(我们一次为例)，校准器的振动级值为10 m/s²峰值，也就是7,07 m/s² RMS 或者 0,72 g。我们需要在“参考值Reference value”域中输入这个值。

只要我们把传感器连接上校准器，我们就能立即看到信号的幅值和频率。在这个例子中，我们可以看到传感器z轴的值。校准频率一般低于200 Hz(通常为80 或者 160 Hz，我们的例子中位80 Hz)。这也是一种简单检查通道是否完好的方法。



接下来是一个简单的步骤，就是点击靠近各个轴的校准Calibrate按钮③，开始进行校准。一旦点击该按钮，我们就能看到传感器的灵敏度单位显示为mV/g，这个单位也可以从传感器的校准表中查到。微小的差异是可以接受的。



然后我们就能看到窗口中显示的振动(0,7197 g)④校准实时RMS值。

4.3.2 输出通道设置

测量模式

下一步需要定义测量。有两种基本的操作模式。首先是全身振动Whole body模式，第二种是手臂振动Hand arm模式。不

同的测量模式定义用来模拟人体对振动响应所对应的不同基本滤波器Basic filters①。这些滤波器是通过无数次对人体各部分的固有频率的测量才定义出来的。

我们可以使用线性滤波器来Linear filter检查测量链，或者用自定义Custom滤波器也可以。如果我们使用的自定义的滤波器Filter②，每个不同的滤波器都可以从右边的下拉列表中来选择，那么就可以做一些特殊的测试，比如建筑振动，或者是晕船测试。

我想着重指出的是我们需要注意传感器的高通频率限制(频率下限)以及所用放大器的频率下限。对于手臂振动模式，高通频率(频率下限)是6.4 Hz，这种限制对于传感器来说非常容易满足。但是对于全身振动来讲，这个频率限制是0.4 Hz，所以我们需要仔细挑选传感器。如果我们已经知道了在某个更高的频率下面没有任何有用的信息，我们还可以使用更高的滤波器(比如3 Hz)。这样做可以让我们的测量过程更加迅速，发生错误的机会更少(滤波器的频率越低意味着重置的时间越长)。

推荐的测量采样率sampling rate也跟应用的不同而不同。对于手臂振动测量，最小采样率是5 kHz，而对其他所有的测试来讲，1 kHz就足够了。

对晕车，晕船测试时，必须特别注意Wf滤波器，车船的低频下限仅有0.08 Hz，所以此处需要使用非常特殊的传感器。

自定义的滤波器同样需要定义计权因子K③。当计算振动总级值的时候，每个轴都有一个单独的值。



计算的参数

下一步需要我们选择计算的参数。这些参数要么是**总值**，也就是说每次测量过后只能得到一个值；要么是**间隔式记录**。如果我们需要间隔记录参数值，连续记录的时间间隔在旁边的输入框中以**秒**为单位来定义。比如我们选择每隔5秒钟间隔来记录**RMS**值，那么我们可以每隔5秒钟得到一个新的**RMS**值。也就是说每隔参数值重新设置一次，计算重新开始。

我们可以计算许多参数。简而言之，**RMS**值是计权信号的均方根值，**Peak**峰值指的是从零线开始的信号最大偏差，**Crest**峰值因子值则指的是峰值和**RMS**值之比，**VDV**值是第四个能量振动剂量值，**MSDV**是运动晕船车船剂量值，**MTVV**则是指最大传递振动值，每隔一秒计算一个值。

Output channels		
☒ RMS	☒ VDV	☐ weighted raw
☒ Peak	☒ MSDV	☐ al (ISO 2631-5)
☒ Crest	☒ MTVV	☐ D (ISO 2631-5)

每个轴都要单独计算上述所有值，而**RMS**，**MSDV**，**VDV**以及**MTVV**还要计算三个坐标值的总值。用这些参数就足够按照**ISO 2631** 和**ISO 8041**标准来评价人体振动的暴露量。

接下来要输出的是**weighted raw**计权原始通道。这是一种用选定的滤波器进行计权的全速时间信号。我们可以用这些通达来计算FFT或者CPB频谱。

al和**D**是基于**ISO 2631-5**标准计算的值，该标准说明了脊柱对振动响应的计算和限制。该标准的基础是专业客车或者卡车死机在粗糙路面或到处有突起的路面驾驶时所暴露的振动量为基础的。多次的冲击可以导致腰椎脊柱尾骨所承受压力的瞬间改变，常年驾驶可能导致永久性的损伤。

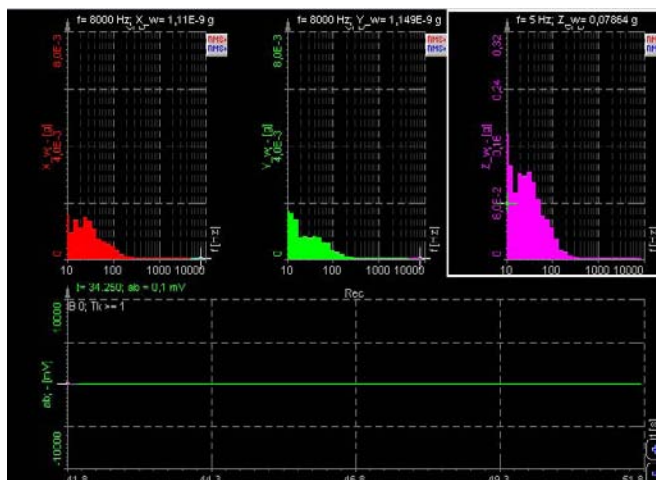
al是脊椎对三个方向测量得到的激励的响应。**D**值则是加速度剂量，从脊椎响应测量得到。这些值已经足够根据**ISO 2631-5**标准来评价人体振动暴露量。

4.3.3 测量

人体振动模块不会自动创建任何显示。基本上我们只需要显示感兴趣的测量统计值。对于长期的监测，最好是关闭掉输入通道的储存功能，只存储输出统计值。这样可以最终减少文件的大小。



某些应用需要测量CPB以及窄带FFT。对于这种应用，我们需要选择计权原始数据通道*weighted raw channels*，然后在FFT或者CPB中选择这些通道。就可以得到信号的计权频谱。CPB频谱变化比较慢，这是因为需要处理较低频率的数据，低频数据通常需要较长的时间来处理。



4.4 阶次跟踪

阶次跟踪分析是用来抽取与机械转动频率相关的谐波成分的一种方法。机械转动形式是激励频率和机械响应函数的混和，激励频率通常和转动速度相关，比如不平衡，偏心，轴承故障等，而机械响应函数则与机械结构和机械安装方式决定的机械固有频率相关。

利用阶次抽取的方法，我们可以发现与特定机械故障相关的特定谐波成分。也就是说 – 第一阶(谐波)通常和机械的不平衡有关，第二阶谐波通常和偏心转动有关，如果我们以9个叶片的转子为例，第9阶谐波一般和叶片故障有关，如果我们以31齿的齿轮为例，那么31阶谐波肯定显示的是齿轮咬合频率的情况。

这些都是引起振动加速度的激励力。激励和系统响应之间的比定义为系统传递曲线。因此最后测量得到的系统振动是激励力和系统传递曲线的混合产物。因为传递曲线是固定的，我们用不同的转动速度激励，就能得到不同的响应。

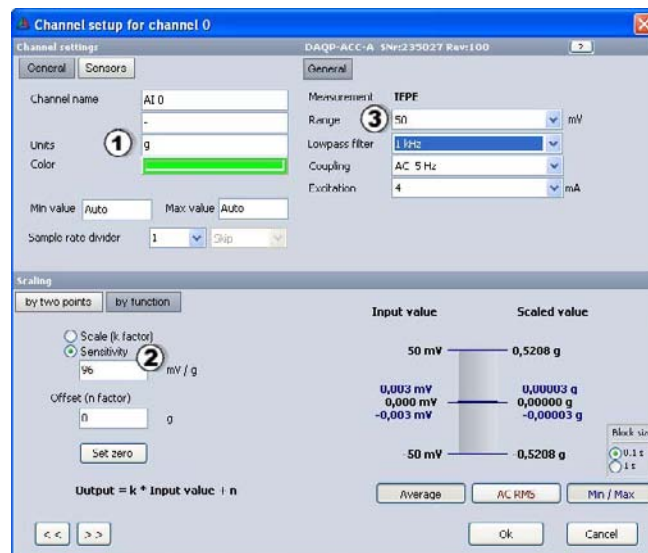
当激励频率接近系统的固有频率时，我们会碰到所谓的共振，共振会大大增加振动的幅值，这可能会对机械造成致命的损害。

所需硬件	Orion
所需软件	SE 或者更高 + ORDTR 选件，DSA 或者 EE
设置采用率	至少10 kHz

4.4.1 通道设置

对于阶次跟踪，我们需要模拟通道来测量振动，噪声，压力以及其它模拟参数，从这些参数中我们要抽取出阶次信息。另外，我们还需要转速传感器，解码器或者光电探头，测量机械的转动速度。我们建议使用ORION计数器来计算RPM，或者利用光电式转速探头直接获取模拟信号。接下来我们需要确认采用速率要足够高，即使在最高转速时也能采集到触发信号。请注意我们还需要内部采样来正确计算阶次跟踪。

在我们的例子中，我们用了一支加速度计来测量振动，一个解码器测量RPM。首先我们来设置振动通道。我们在通道设置窗口中输入测量单位g^①，从传感器的校准表上可以知道传感器的灵敏度为96 mv/g，输入这个灵敏度。接着选择合适的测量量程^②。



第二步设置是增加一个阶次跟踪数学运算模块，选择频率测量的源。我们可以选择任意的计数器传感器。如果我们选择的是解码器，那么我们需要将解码器连接到计数器输入上。请注意在该例子中只能用ORION卡上的计数器。如果我们手上有光电式转速探头(或者其他类型的模拟传感器)，那么我们需要将传感器连接到相应的模拟输入通道上，并且点击出现在**Frequency channel 频率通道**设置窗口右边的**Setup**设置按钮来设置触发电平。

接下来需要定义**最低RPM限制Lower RPM limit**，**最高RPM限制Upper RPM limit**以及**Delta RPM**。这三个参数对于保留以阶次跟踪为基础的FFT瀑布图所需内存非常重要。在我们的例子中，我们希望的RPM值处在0到3000RPM之内，并且希望每50 RPM得到一条FFT谱线。那么可以得到整个RPM范围内的频率分辨率为 $(3000-0) / 50=60$ 条谱线。我们还要定义触发的方向，以便系统既能捕捉**提速事件**，又能捕捉**降速事件**。另外，我们还要定义**Delta时间**间隔。如果RPM不变，我们可以定义时间限制，计算谐波点。在时间间隔之后，它将在升速频谱中计算出一个新的点，用来观察同样速度下的幅值偏差。在这一章节中，我们还需要定义用来计算的**update criteria**更新条件。**Always**表示只要有频率通过，任意时刻的数据都采集下来，而**Only first time**表示只有第一次的频率范围通过时，才把数据采集下来，其它时候不再采集③。

接下来我们定义阶次设置。对于频率历程谱的计算而言，我们需要定义阶次分辨率**Order resolution**，最大阶次**Maxiam Order**和**FFT时间窗FFT Window**。FFT时间窗是用来计算FFT谱的加窗。因为采集的数据是以角度为基准的，如果在信号中只有谐波成分，我们可以使用诸如**Hanning**汉宁等“软”性时间窗。但是如果信号中包含有非谐波成分，那么建议使用**Blackman**等时间窗。最大阶次定义的是频谱的阶次跨度。当数据算出后，数据按照这个最大阶次进行滤波。在Order OT模式中，频谱的结果也受到这个最大阶次的影响。最大阶次**Maxiam Order**的数值取决于我们估计的最大激励频率。阶次分辨率**Order resolution**则定义频谱中的步进程度。我们我们希望非常细致的观察固有频率，建议使用**0.125**阶。

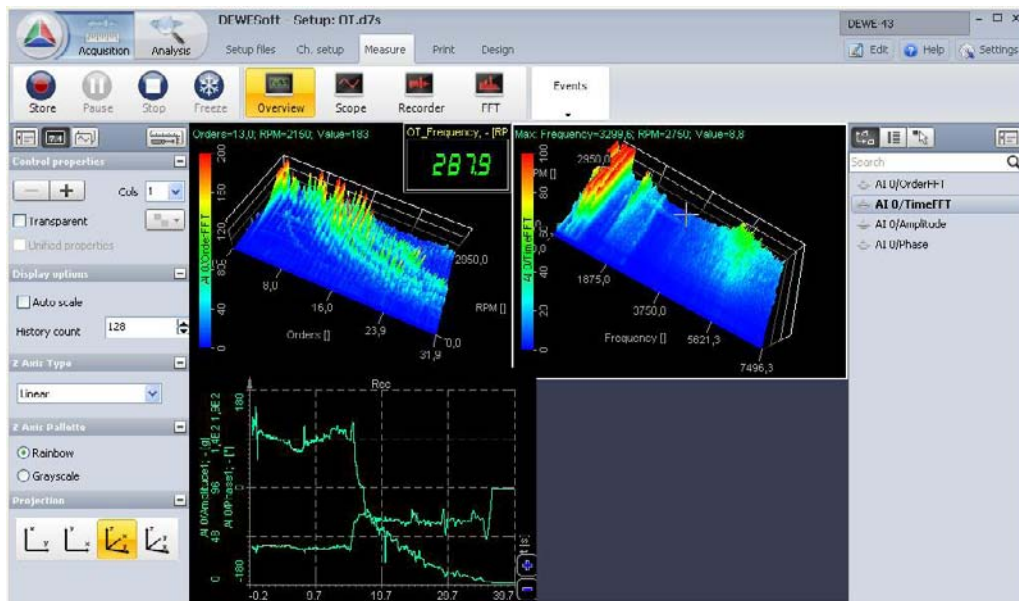
最后我们需要定义**Extracted harmonics**抽取的**谐波**④。我们可以直接输入谐波的阶数(比如1, 3, 8, 12等等)，或者定义抽取所有阶次。软件会为在阶次设置中定义每个阶次创建一个通道，因此根据我们的设置，会有 $8*64=512$ 个通道，这么多通道的数据是难以处理的。因此我们建议只抽取我们真正感兴趣的阶次。我们能抽取的信息包括每个谐波的幅值，实部，虚部以及相位。



4.4.2 测量

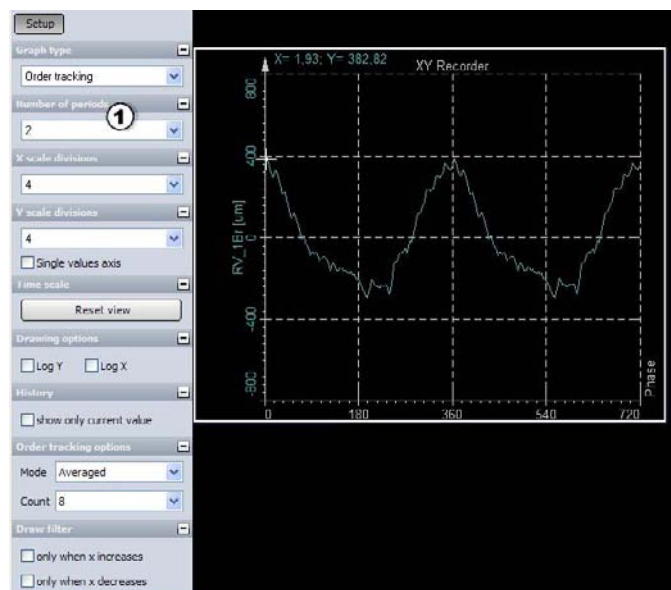
现在我们来进行一些测量。阶次跟踪数学模块并不创建任何独有的显示界面，因此我们需要手动定义显示界面。我在界面上增加了模拟表和数字表，用来显示转速，增加了记录仪记录转速和振动值，这样可以看见升速的整个过程。接下来我们增加xy记录仪，其中x轴必须定义为OT_Frequency，而y轴用来显示谐波成分(在我们的例子中为Vib H_1, Vib H_2等等)。必须选择记录仪的“单一x轴Single X axis”模式，建议使用“实时数据Real data”显示类型，以便在xy记录仪上能够显示升速的每个点。

接下来我增加一个Vib信号的频率图，以便能观察到当前的频率速度，在其下面，又在阶次跟踪模式下增加了一个3D FFT谱图。要做到这一点，需要增加3D FFT谱图，如果想把x轴显示为频率，选择“Time OT”，如果想把x轴显示为阶次，则在History标签①选择“Order OT”。建议选择对数logarithmic坐标，选择两到三位小数显示幅值分辨率(比如我们测量最大1g的振动，那么最小可以显示0.001g的值)，以最适合的颜色显示幅值大小。稍小的动态显示(下图显示的是2位小数)更加突出峰值，而稍大的动态显示则更容易反应出升速中小的成分。

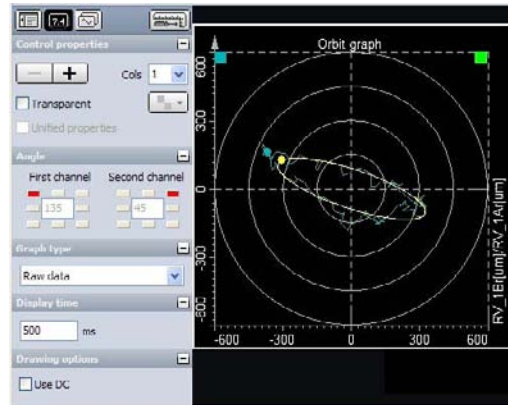


我们可以利用xy图谱显示某个确定阶次的幅值，并和机械频率(转速)相比较。以同样的方式，我们也可以显示相位，创建Bode图。如果我们在x-y记录仪谱图中显示实部和虚部，那么我们就能够得到如下图所示的奈奎斯特Nyquist图。

还有一种在x-y记录仪图中观察当前数据的好方法。我们在x-y记录仪图中选择**阶次跟踪Order tracking**谱图类型，然后选择已经在阶次跟踪中定义的任意通道，我们可以发现周期信号在时间轴上已经和解码器的零脉冲或者是和光电探头的信号对齐了。我们可以选择任意周期数，选择显示当前周期，平均周期数，或者显示连续数据②。

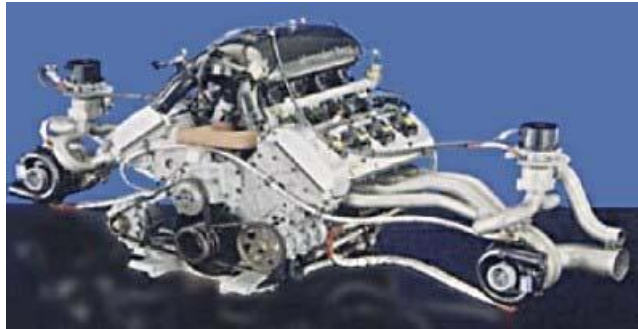


如果我们在转动轴上测量两个位移，就可以用轨迹orbit图显示转动轴的运动情况。选择阶次跟踪模式，显示当前数据，平均数据或者显示连续轨迹。我们还可以显示谐波，如下图所示③。黄色的点显示的是角度传感器的零脉冲的位置。



5 燃烧分析

DEWESoft 燃烧分析仪模块可以对内燃机进行分析.如果我们测量气缸压力和曲轴转角,我们就可以计算发动机研发的主要参数, I象最大压力, 最大压力位置, 放热, 敲缸和其他重要参数.



燃烧分析仪完全集成在 **DEWESoft**里, 这就意味着你还可以使用**DEWESoft** 的其他功能, 包括 *CAN总线*s, *视频*, 其他模拟信号采集等

要进行燃烧分析我们需要:

所需硬件	Dewetron Orion-1616 board and DEWE-CRANKANGLE-CPU for full support (also possible with entry level cards and without CA-CPU, but only in internal clocking mode)
所需软件	DEWESoft PROF with CA option

5.1 通道设置

模拟通道设置

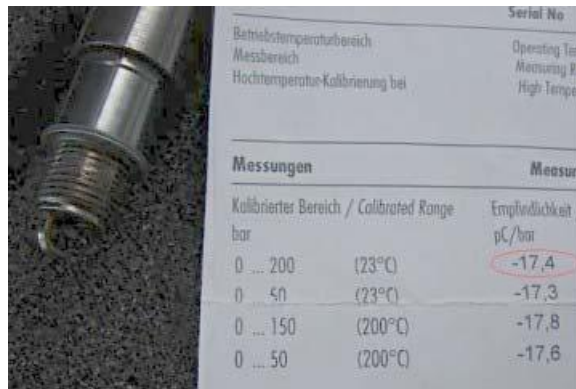
首先选择需要的模拟通道并根据从各传感器的校准文件输入换算系数. 压力传感器一般是电荷型的, 对应我们的放大器模块是 DAQP-CHARGE-B, 这样我们也可以采集一些低频信号 (甚至静态压力).

这里有俩个基本的选择 - **DC** 或 **AC**输入耦合. **DC** 耦合可以测量静态信号这在校准时非常有用, **DC**耦合的特性就象一个重启键, 可以把电荷清零. 现在我们做一个静态压力的测试并校准一个传感器

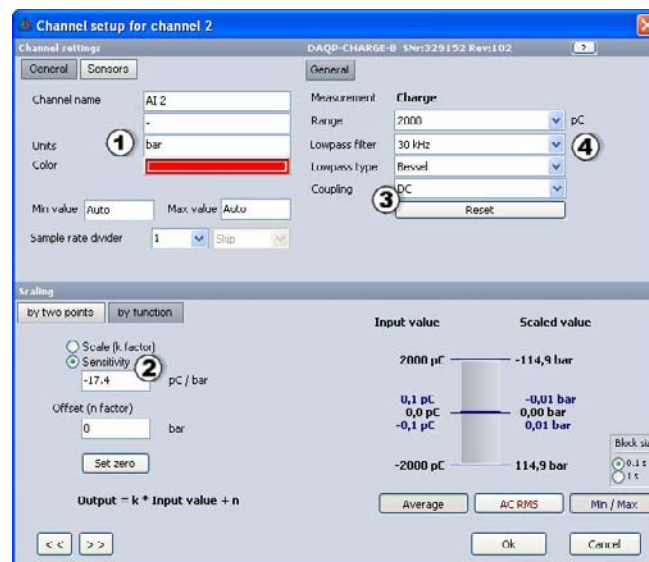
.

请注意正常的 (尤其是长期的) 的测量一般采用 **AC**耦合. **AC** 耦合有一个时间常数从 **2** 到**1000 秒**, 这样发动机的压力信号就可以被正确的测量.

另外一个可能就是使用校准证书. 当我们拿到一个已经经工厂校准的传感器时, 我们就会拿到一个表明灵敏度的证书, 下面就是一个样本.



我们输入"bar" 作为单位, 然后到 "by function" 标签, 选择灵敏度 **Sensitivity**, 在这里输入证书上的灵敏度 (我们的这个例子是-17,4). 一旦我们输入了正确的换算系数, 我们马上就可以看到换算过的量程和当前的压力值。量程可以通过下拉菜单进行选择, 有时候会有一些高频信号的干扰我们可以通过设定低通滤波 **Lowpass filter** (10 or 30 kHz)去除这些高频信号。如果信号是干净的,建议采用最高的滤波(100 kHz).以看到整个动态范围, 在高转速的时候也可以这样做。



转角传感器设置

在燃烧分析中有两个主要的传感器: 角度传感器, 每转360个或更多的麦种并带有零点位置识别, 车载角度传感器一般是轮齿结构, 并会缺少一个或两个齿来确定零点 (典型的如 60-2, 有60个齿, 但是缺少两个齿, 这两个齿的位置就是零点)



传感器通常会连接到 CA-CPU, 整个装置可以产生任意数量的脉冲并且可以从输入信号中识别零点位置. 在一些测试中, 传感器会连接到模拟通道, 但是让我们进一步的详细了解在不同的测试中传感器如何输入

你不用对前两个计数器输入做任何设置,因为在燃烧分析中, 这两个计数器都是在内部工作. 如果你还有其他的计数器, 你可以任意的使用它们.

5.2 燃烧分析设置

点击**Combustion**按钮, 然后点击蓝色的 “+” 按钮增加一个燃烧分析测试模块。We我们可以任意的设置燃烧分析仪, 但是通常我们会像下面的图形那样按5步进行设置。:

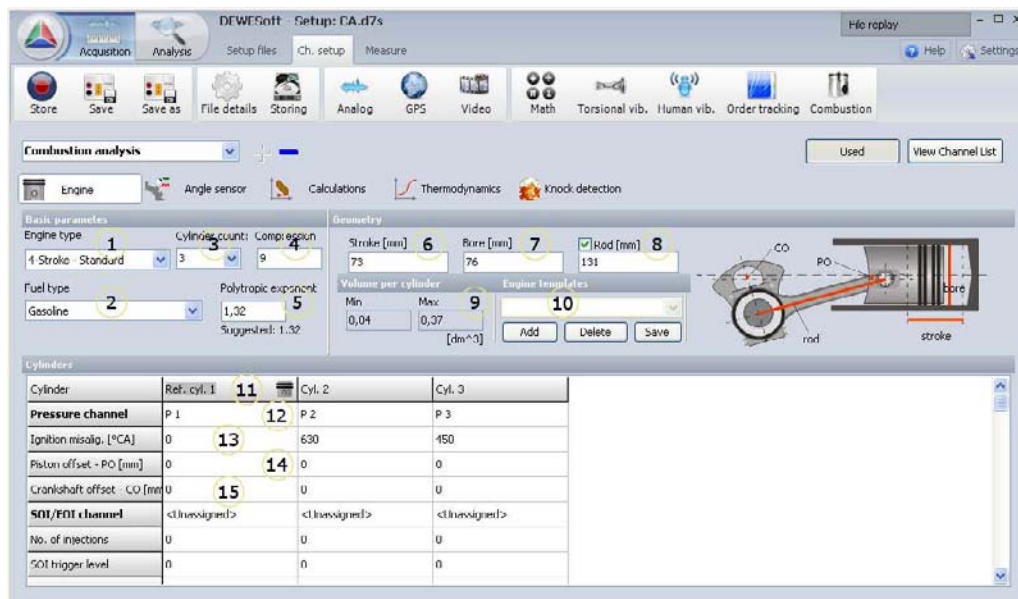


1. 发动机设置

首先我们设定发动机参数 (*几何参数, 发动机类型*). 我们这样定义发动机 (汽油, 柴油[2], 四冲程, 俩冲程 [1]), 汽缸数量 [3] and 发动机几何参数[6-10]用来计算汽缸容量. 对每一个汽缸我们也可以定义活塞piston位置和曲轴偏置crankshaft offset[14, 15], 但是对于大多数发动机来说这些值都是零. 我们还需要定义多变指数polytropic exponent [5], 这是一个跟压缩有关的参数。 (without ignition). 热力学参数和放热也是非常重要的. 如果你不知道你的发动机的多变指数polytropic exponent, 可以选择不同类型发动机的推荐值, 我们后面会告诉你如何找到这些。.

下一步定义对应汽缸压力cylinder pressure [12]的通道, 注意我们可以不为某些汽缸分配通道 (例如下图的第四个汽缸 **Cyl. 4**) 那些通道就不会运算. 我们也可以定义点火错位ignition misalignment, 这定义了发动机的点火顺序. 如果我们拿一个6缸的发动机为例, 点火顺序为 1-4-3-6-2-5, 整个循环是 720 度, 点火错位就是: 0-360-240-600-120-480.

我们也可以定义喷射开始和结束，触发水平和多点喷射. 在测试中我们把喷射角作为额外的通道。.



2.角度传感器设置

角度传感器的设置对于燃烧分析的设置来说是非常重要的.它定义了燃烧分析模块的性能，我们有俩个途径来获得数据：内部和外部时钟。这俩种方法重新定义了数据采集过程. 下面让我们看一下这两种方法。

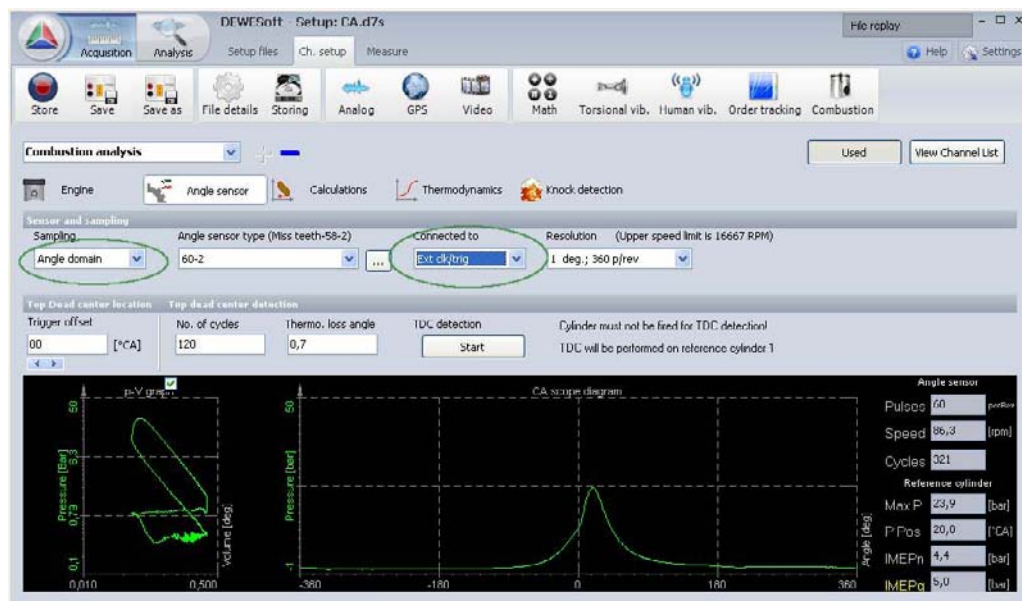
外部时钟

外部时钟意味着将以角域采集数据。换句话说，如果我们选择了每转360个脉冲，无论发动机的转速是多少，它总是每转采集360个点。这个对于高转速或者多汽缸的发动机非常有用，因为它可以直接采集适合燃烧分析运算所需的数据，然而这种方法，我们就不能再做时域的运算，例如燃烧噪音分析。我们也不能通过60-2或类似传感器进行冷启动测试。

在使用外部时钟的情况下，我们需要转角信号调理器 CA CPU 这个硬件来产生任意传感器所需要的脉冲数，下一步我们从列表里选择角度传感器 *angle sensor* (我们例子里是60-2). 如果没有你想要的传感器，我们可以点击右侧的“...”来添加传感器. 转角信号调理器 CA-CPU 支持 解码器 *encoder*, *CDM*, 和缺齿曲轴转角传感器。我们定义传感器跟转角信号调理器 CA-CPU 的输入连接和输出分辨率。

这个分辨率定义了最大速度 (取决于AD板的最大采样率) 和运算量。如果我们选择了0.1 度的分辨率和只用 500 ks/s AD板,我们的最大转速就是 $500000/3600*60=8300$ RPM. 输出分辨率直接定义了所有运算包括显示的每转的点数。

我们一旦正确的设置了发动机和传感器，发动机启动后，我们就可以在显示界面的右侧看到转角信号调理器 CA-CPU追踪的信息和脉冲数，速度，循环数。



内部时钟

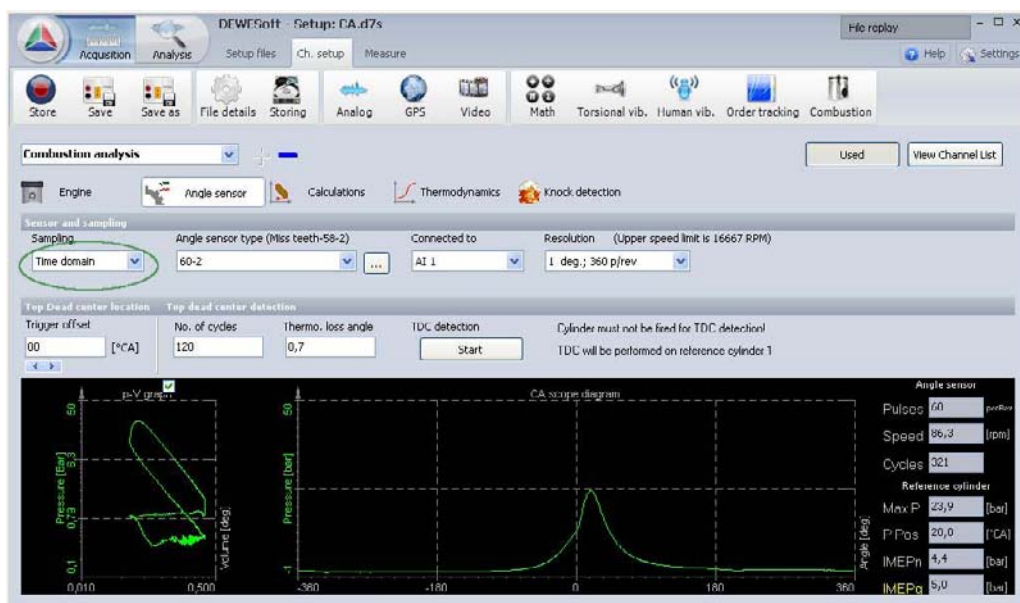
内部时钟是一个完全不同的方法，它采集时域数据并通过软件转换为燃烧分析所需的交域数据。因为数据根据发动机转速进行了低通滤波，因此运算的要求比使用外部时钟时要求更高，但是另一方面，它更容易跟时域关联分析（象 CAN bus, video 等）和对时域参数进行运算例如燃烧噪声。

既然如此，我们就不需要使用转角信号调理器 CA-CPU，而是把 CDM或解码器encoder传感器直接连接到计数器。

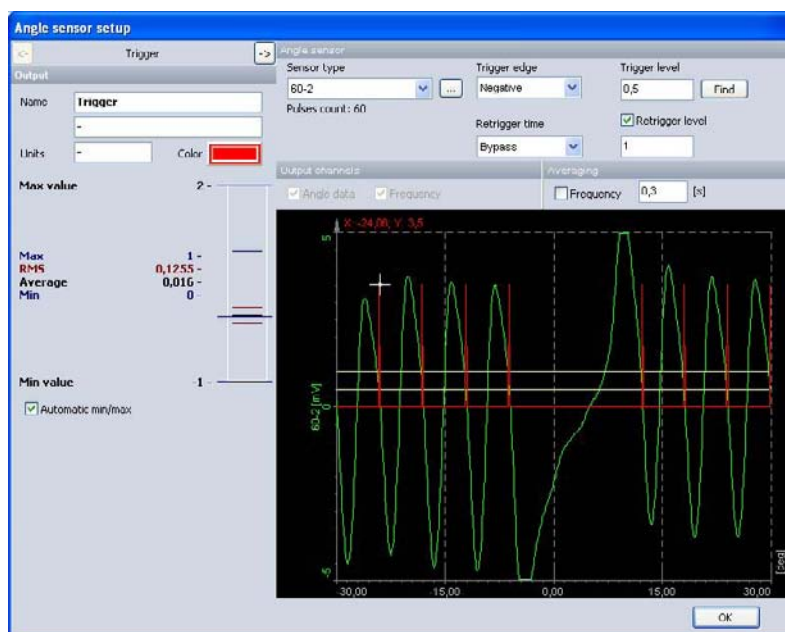


如果我们使用缺齿式的轮齿传感器，就可以直接连接到模拟通道 (如下图). 另外一个好处就是内部时钟的方法可以做冷启动测试 **cold start**. 如果使用外部时钟，转角信号调理器 CA- CPU 会丢失最初的几个循环，这对冷启动测试时非常重要的。

我们点击**Setup**按钮需要设置触发水平和方向。



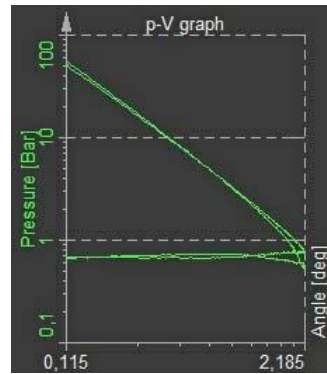
最简便的办法是点击  - 来设置触发的参数，下图我们已经可以看到实时的数据。



上止点检测

无论你是使用内部或外部时钟, 当我们设置结束后, 我们都可以看到在 *CA cycle graph* 上看到一些数据。为了看到正确的 *p-v* 图, 我们需要定义触发偏置. 触发偏置是参考缸触发点和上止点之间的角度差 (注意在发动机设置里的参考缸 *ref. cyl.*).

这个角度可以通过物理测量或者通过上止点**TDC**的检测来定义, 进行这一步的时候注意发动机不能点火. 在台架上这个比较容易获得, 因为我们有选项可以关闭点火; 在车载燃烧分析中, 我们可以通过在某个特定速度下获得, 让汽车在某个档位, 然后松开油门踏板, 让车自然停止, 这时发动机仍然在运转我们称他为空循环, 这时发动机只有压缩和膨胀并没有工作, 下图是典型的引擎空循环, 这里我们可以看到压缩和膨胀的曲线完全重合, 这是因为发动机内部并没有工作。



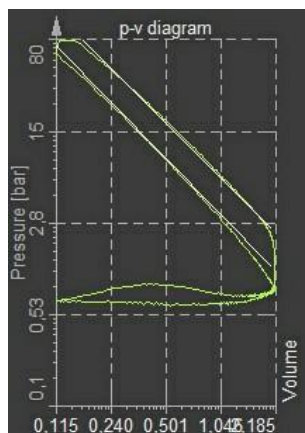
我们输入平均循环数和热动力损耗角. 这是上止点和压力峰值之间的延迟, 压力峰值出现在上止点之后, 我们通过输入损耗角来补偿这个延迟. 然后我们选择开始上止点TDC检测, 你所定义的几个缸开始采集, 触发偏置可以自动计算和设置

3.运算设置

现在我们来定义输出通道和压力修正方法. 最后一件事情是设定零点修正. 如我们之前提到的, 电荷性传感器长时间使用会发生漂移, 所以我们需要计算绝对压力. 这可以有三个方法: 参考热力学零点 **Thermodynamic zero**, 假设汽缸可以无限膨胀, 这个零的绝对压力就是零.



在这个点我们需要好好看一下 $p-v$ 图. 在对数 $p-v$ 图中, 多变压缩和膨胀是一条直线, 它的斜度由多变指数决定. 我们可以在测试中清楚的看到 (如下图). 如果多变指数不合适, 零点的修正和放热率的计算就会是错误的.



回到零点修正zero point correction, 我们需要在一个循环中的压缩部分定义俩个角度, 这个压缩是适合多变指数 **polytropic exponent** 的压缩. 如果我们定义的这两个点是在已经点火或者进气门没有关闭的位置, 运算将会是错误的。

另外一个选择就是输入已知的或者测量过的压力值在某个特定的角度 (一般实在进气门).

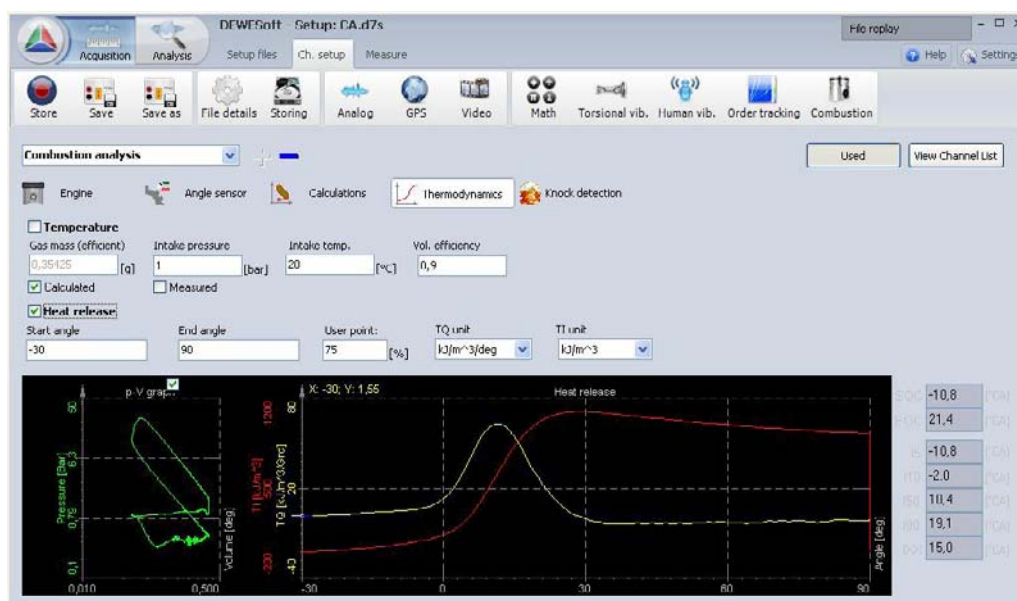
现在真个系统已经设置完毕, 剩下的只是选择我们在测试中需要看到那些参数. 一些常用的统计值象最大压力和最大压力值位置都会在测试中出现. 我们选择压力信号的来源作为选项, 这样我们就可以得到最大压力的位置, **rise**. **MEP** 选项将会计算 **IMEPn** (整个循环), **IMEPg** (循环的做功部分) and **PMEP** (循环的进气过程). **MEP** 值是平均有效压力 **effective pressure**, 事实上是 **p-v** 图表里的一个区域 (发动机能量), 使得汽缸的排量标准化.

另外我们可以通过统计, **MEP**, 放热得到每个汽缸了整个循环的平均值. 这样我们就可以在 **CA-scope** 看到各种平均值, 包括各种统计值.

4. 热力学参数设置

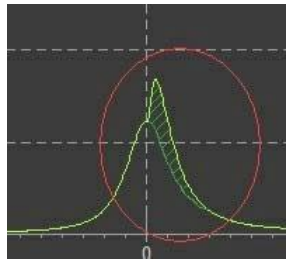
在这部门我们将讨论放热的计算. 放热是一个很精确的计算, 我们需要仔细设置. 我们先这是起止点得计算, 因为放热主要是发生在上止点附近也就是循环中的做功部分. 这个计算的结果是燃烧的起点, 终点, 和某个特定曲线位置的角度

。



但是我们首先看一下什么是放热. 放热表示在多变压缩过程中压力的上升. 我们已经在 **p-v** 里看到了空循环和工作循环的区别,

让我们再看一下 CA 图. 下图空循环管在工作循环的下方，它们中间的区域就是发动机做的功两条曲线之间不重合的部分就是发动机的放热I.



在这个部分我们选择温度计算，通过选择进气量或定义进气压力, 进气温度和容积效率，汽缸内部的温度就可以计算出来.

5. 敲缸检测设置

敲缸检测是通过高通，低通滤波后，比较压缩和膨胀冲程的一个过程结果是敲缸系数Knocking factor,它表明了发动机内部总的敲缸次数.

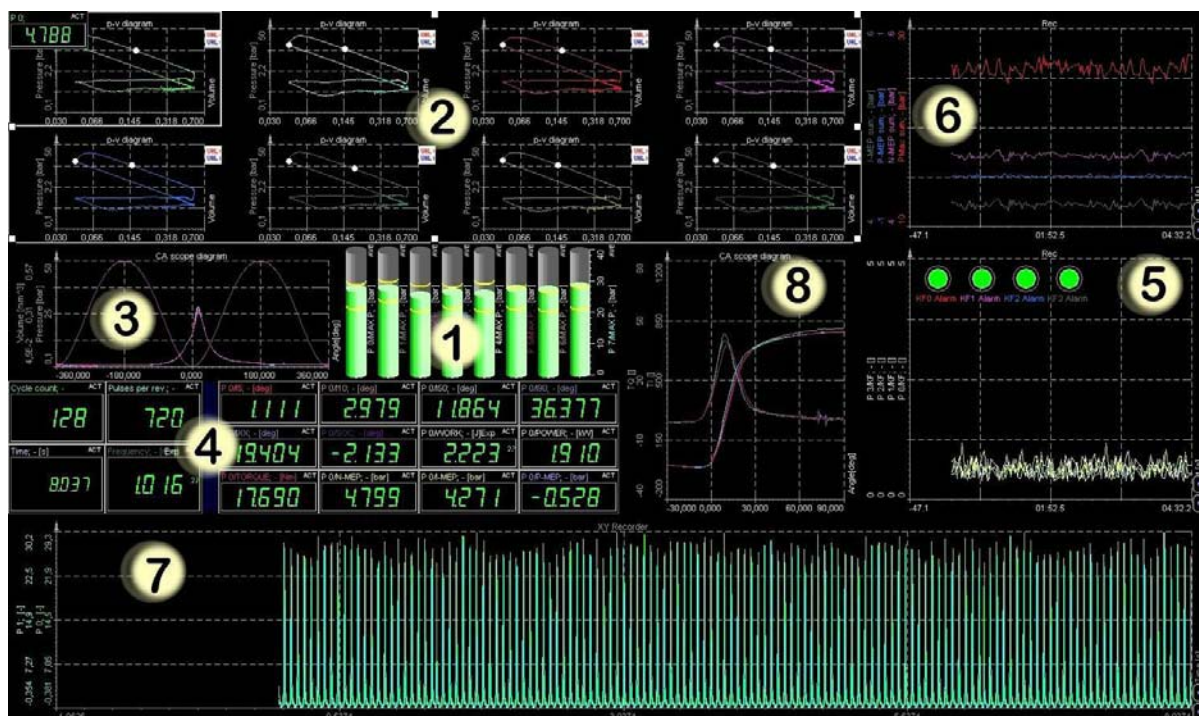
☐ Knock detection (Method Mannesmann VDO AG)						
Lowpass filter:	40	[taps]	Shift reference window:			
Highpass filter:	6	[taps]	Speed	Correction		
Noise threshold:	0.05	[bar]	1000	[Rpm]	0	[°CA]
Reference signal window width:	30	[°CA]	6000	[Rpm]	0	[°CA]
Knock signal window width:	30	[°CA]				

5.3 测量

现在我们已经这只好各个参数, 我们可以进行测量了. 燃烧分析没有标准的测试界面, 需要我们自己设计一个.

我们可以在下图看到典型的 CA 测试设置:

- 所有汽缸的最大压力值①
- Pressure versus volume (pV图) – 白色的点代表燃烧的起点②
- 最后一个循环的汽缸容积和压力 (CA Scope) ③
- MEP 值 (IMEPn, IMEPg, PMEP), 转速频率, 循环计数.. ④⑥
- 敲缸系数 ⑤
- 放热 (TI, TQ, burn angles) ⑧
- 时域数据 (记录仪) ⑦

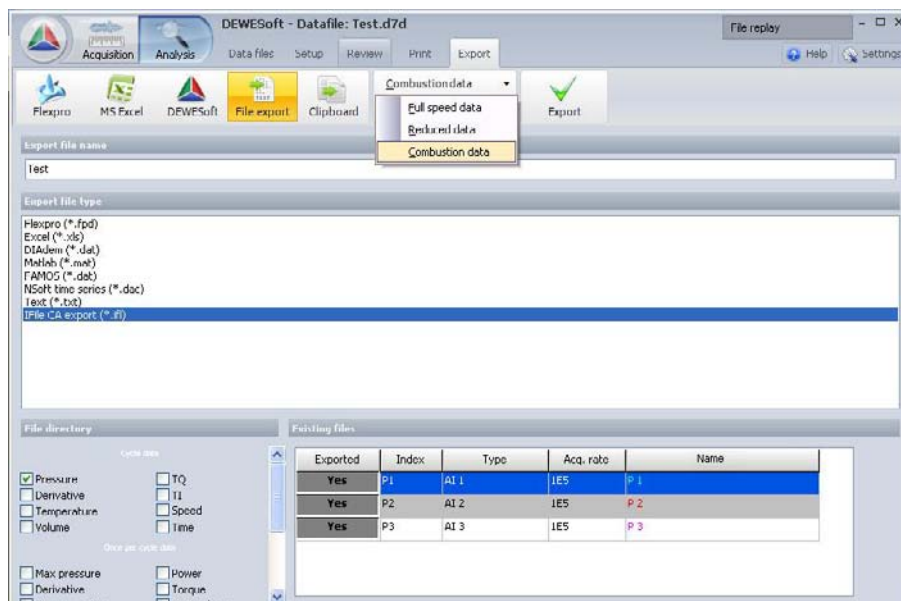


当我们使用 **CA math**, 我们有俩个显示界面: *p-v* and *CA scope*. 如果我们使用外部时钟我们可以用 *x-y* 记录仪: 来显示时域数据. **CA math** 可以产生一个叫 *Time* 的通道用于 *x-y* 记录仪的 *x* 轴.

5.4 分析

回放燃烧分析的数据，我们可以使用分析 **analysis** 的所有功能. 当在数据中移动光标时，我们可以在 *p-v* 或 *CA scope* 看到当前循环的数据。

一个重要的选项就是以不同格式输出角域数据，我们可以在输出 **Export** 功能下选择燃烧分析数据 **Combustion data** . 我们可以输出为一些支持实时数据的标准格式例如 Flexpro或Diadem, 也可以输出为专业的燃烧分析数据格式 IFile, 可以被大多数的燃烧分析或后处理的软件使用.



我们几个输出选项:

选择 **Cycle data** , 所得的数据是角域值或者压力值以及其他参数，都是显示在 *p-v* 图里的数据.

选择 **Once per cycle data** , 数据都是统计值 (**MEP**, 最大压力 ...) 每个有效的循环运算一次. 它们是燃烧分析运算的结果.

选择 **Cycle average** , 将会输出一个循环的数据包括平均压力 **average pressure**, 放热 **heat release**, 温度曲线 **temperature curve**. 数据是基于角域所有循环数据的平均值.

6 网络数据采集系统教程

所需硬件	任何 AD 卡
所需软件	SE 或更高版本 + NET 选件, EE
设置采样率	取决于使用的数学模块

DEWESoft NET 应用程序模块允许一台或者多台Dewetron 测试单元 (简称 **MU**) 能够通过其他计算机, 也叫**客户端CLIENTS** 进行控制。测试单元**MU**和 客户端**CLIENT** 之间必须通过TCP/IP 连接。运行最好的连接方式是高速直接的以太网网络拓扑技术, 虽然无线以太网同样能够工作, 但是速度慢一些。也可以混合使用硬线连接和无线连接系统。

用**DEWESoft-NET** 进行测试需要进行下面 三步基本操作(步骤):

NET 设置 网络配置, 选择合适的硬件以及**DEWESoft-NET** 设置:

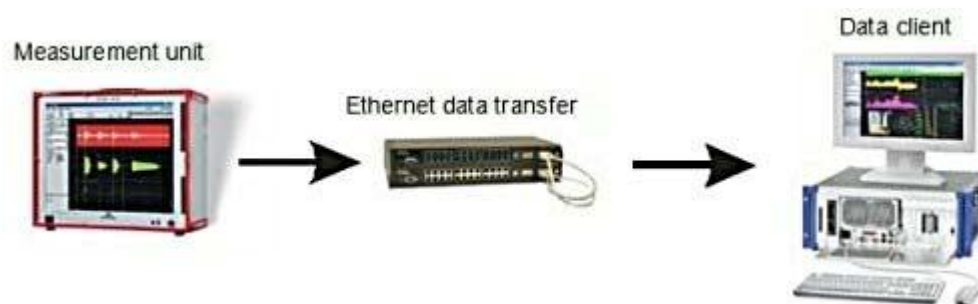
- 设置客户端**Client**和测试单元**MU**
- 远程控制某个被控制测试单元**MU**

测试 创建显示界面, 测量-采集数据, 将数据存储在网上 分

析 分析存储的数据, 将数据存储在网上, 输出测量数据

DEWESoft NET 应用模块运行原理

需要着重注意的是, 客户端**CLIENT**能够观察那些通道的数据, 实际数据存储**在测试单元MU上**。这点非常重要, 它可以在网络出现故障死机时, 或者数据传递过程收到干扰情况下保证数据不丢失。即使上述两种情况正的发生了, 所有数据仍然**安全存储在测试单元MU上**。等到网络连接重新建立, 系统会自动重新连接。



通常情况下, 即使用的是千兆以上以太网接口, 也不可能实时传递所有通道的数据给客户端。

假设只有一套32通道的DEWE系统, 每通道用24位AD采样, 采样率200 kB/s每通道。需要传输的数据就已经达到了25.6 MB/秒, 也就是说已经超过了200 Mb每秒 (1 Byte = 8 bits)。不需要太长时间, 网络数据流就会完全过载, 最终无法避免会导致数据包丢失。

采集停止后, 控制客户端**client**界面马上出现一个新的按钮, 提示您是否立即从**测试单元MU**上将数据文件上载到客户端**client**电脑上, 并在客户端**client**电脑上**观察数据**。

6.1 设置测量单元和客户端

客户端和测量MU配置

有几种可能的配置，因此也就有几种可能的设置，包括：

- **独立运行单元** – 不通过网络连接 (DEWESoft安装后的默认设置)
- **从动测量单元** – 即可通过本地控制也可通过主控客户端client来测量数据
- **主控测量单元** – 即可直接测量数据，也可控制 其它测量单元MU(选项)
- **观察客户端** – 可以观察记录在测量单元MU上的数据，但是 不能控制控制测量单元MU
- **主控客户端** – 既能控制测量单元MU，又能观察测量单元MU上存储的数据

每个测量单元MU必须设置成从动测量单元，这样才能使用 DEWESoft NET 软件对其进行控制 – 如果您要用其中一台测量单元MU 作为主控单元来控制其他测量单元MU，那么需要把这台测量单元MU 设置成主控测量单元。一个DEWESoft NET 网络系统中，不可能 有超过一台"主控单元"，这是为了避免产生混淆和相互冲突。

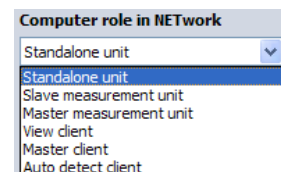
硬件设置

为了在DEWESoft NET 中给每个系统设置合适的配置模式，首先必须运行DEWESoft软件。然后点击System menu，选择 **Hardware setup**。 当出现对话框时，点击对话框上面的**NET**

按钮，会出现如下屏幕：

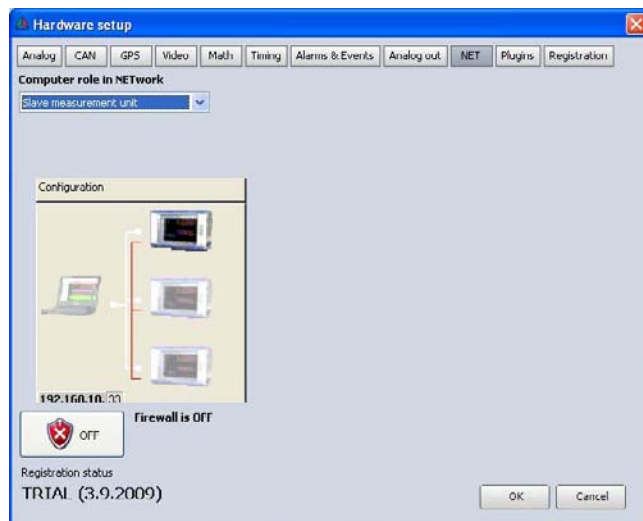


在选择列表中，可以选择需要的配置类型。



设置从动测量单元

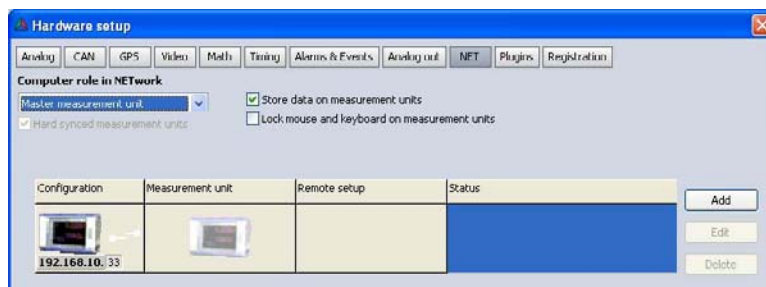
使用选择列表为该系统指定任务 (上图所示选择的是 "Standalone unit")。假设这是我们第一台测量单元，如下图所示将其设置为从动测量单元 **Slave Measurement Unit**:



注意这时会出现一个叫做 **Configuration** 的图形，图形中显示系统网络拓扑关系，图形中的黑色系统即代表本系统 ... 还有其他 Dewetron 数采系统 (该图形只是作为一个参考 – 并不表示所看到的图形就是实际连接的系统型号)。白线代表 **TCP/IP 网络连接**，连接所有测量单元 MU 和客户端 client (图中用一台笔记本电脑表示，同样也仅是作为参考)。红线代表同步线，只要有超过一台测量单元 MU，而且想同时在客户端 client 电脑上显示来自不同测量单元 MU 的通道的情況下，必须使用同步。也确保了记录在每台测量单元 MU 上的数据文件是真正完全同步的。除非你有硬件同步，就像 [DEWESoft Manual → Data Synchronization](#) 一章中提到的一样，否则是不可能达到完全精确的数据文件同步。

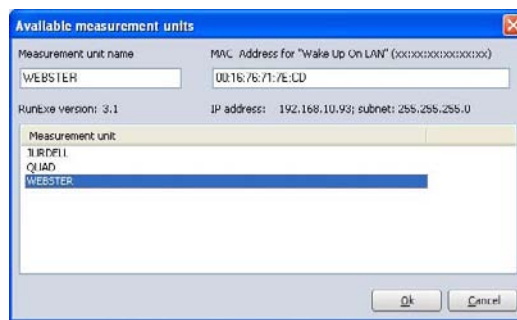
设置主控测量单元 (选项)

如前所述，如果想要使用一台测量单元 MU 作为控制单元，控制一个或多个 Dewetron 测量单元 MU，同时又要单独记录数据，那么必须将本台系统指定为主控测量单元 **Master Measurement Unit**:

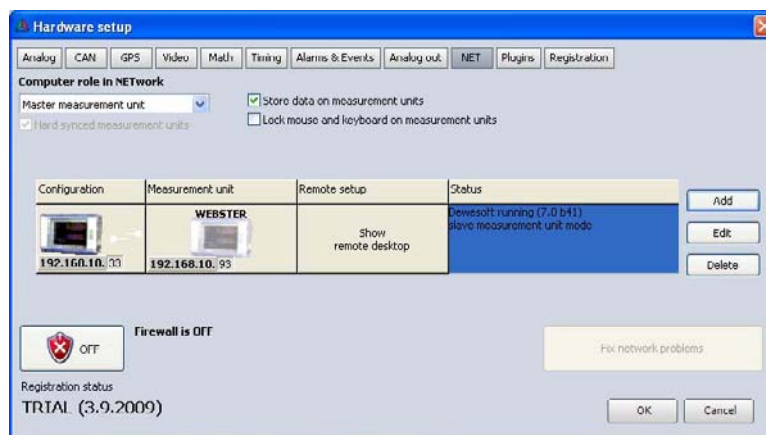


当把本台测量系统指定为主控测量单元 **Master MU**，这时屏幕会出现图形，图中黑色图形代表的是本测量单元 MU，如上图所示。

假设您已经将一台或几台测量系统配置成从动测量单元MU，这时可以点击**Add** 按钮，软件会自动检测已经连接到网络上的所有测量系统：



上例中，软件在网络上只找到了一台测量单元MU，名称为 **WEBSTER**。选择该系统，点击对话框底部的**OK**按钮：



现在图上显示，这台仪器成为第一台从动测量单元MU。如果还有其他测量单元需要添加，点击**Add** 按钮，按照上述步骤加入即可，加入后这些设备的名称同样会显示在上图列表中。注意 –**DEWESoft**软件会自动存储这些信息，所以下次使用时无需重复这些操作，除非需要改变这些设置。

注意恰当选取对话框中的下述几个复选框：

- 将数据储存在测量单元上 **Store data on measurement units** (默认设置，强烈推荐!)
- 锁定测量单元上的鼠标和键盘 **Lock mouse and keyboard on measurement units**

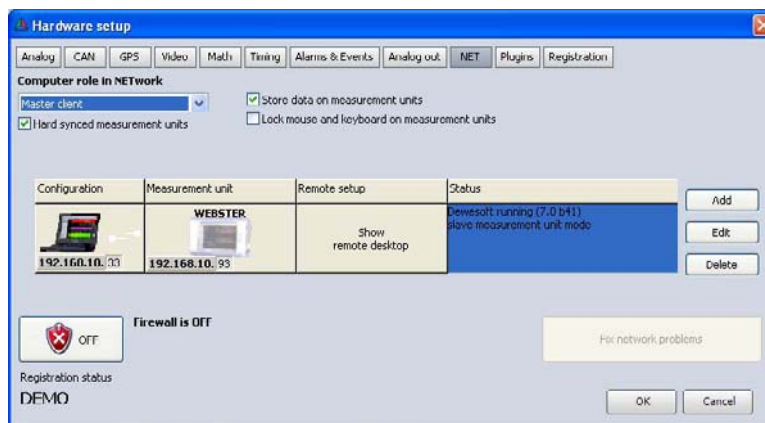
如果您想禁止现场的操作工程师擅自改变测量单元MU上的任何设置，或者防止其干扰测试过程，请选取 “锁定测量单元上的鼠标和键盘lock mouse and keyboard” 复选框。选取该复选框后，测量单元MU就无法从现场对其操作 – 所有的操作通过客户端client计算机完全控制。

重要提示 如果打算使用另外一台独立的计算机作为主控客户端**MASTER CLIENT**，那么就千万不要指定这台测量单元作为主控测量单元 **MASTER MEASURING UNIT**。在本例中，所有测量单元MU必须设定成 从动测量单元**SLAVE MEASUREMENT UNITS**。

设置主控客户端

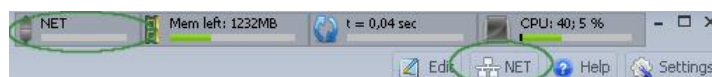
在将所有Dewetron 测试系统全部设置成从动测量单元slave measurement units情况下，就需要单独指定一台主控客户端Master client计算机来控制它们。我们首先假设用我们手边的这台计算机作为客户端，而且计算机已经正确安装了**DEWESoft** 软件。那么点击系统菜单**System menu**，然后选择硬件设置**Hardware setup**，再点击**NET** 标签。

现在通过下拉菜单选择，将这台计算机指定为**主控客户端Master client**，如下图所示：

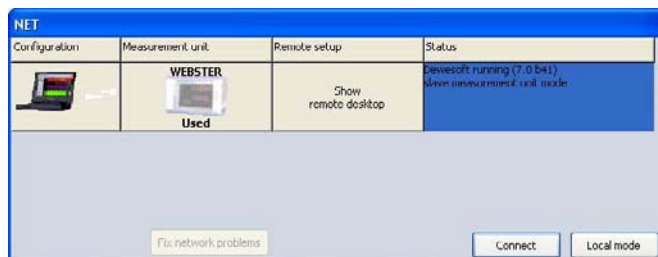


注意那台名叫**WEBSTER**的从动测量单元**slave MU**已经显示在图中，因为我们前面已经将其加入网络系统了。不过，如果以前没做这些工作，即没有配置并连接任何**测量单元MU**到系统中的话，那么就需要点击**Add**按钮，然后选择一台或多台**测量单元MU**，将它们加入到系统中。

现在点击 **OK**，关闭本对话框，你会发现**DEWESoft** 软件屏幕上菜单栏中增加一个新的功能图标 – **NET** 图标：

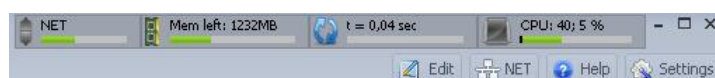


必须点击这个按钮，然后选择连接**connect**，将其和**测量单元MU**相连：



点击 **连接Connect** 按钮，**DEWESoft** 软件将计算机和**测量单元 MU**连接，然后这个对话框会自动关闭。

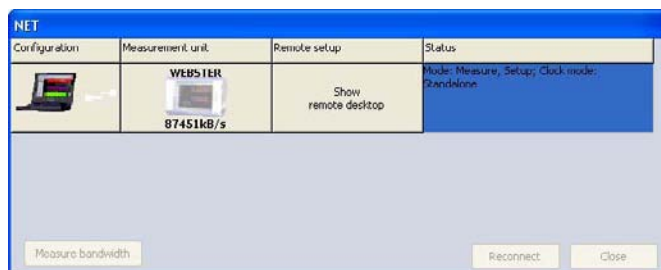
NET 图标此时会稍稍改变：



红色小圆点表示传输的速度。

我们还必须有真正传输文件，因此这些点式红色的。

点击**NET**图标，打开连接屏幕。点击"**测量传输带宽Measure bandwidth**" 按钮，系统会自动检测网络传输性能。如果使用的是**100 Mbit**网络，传输速度大概为**10 MB/秒**，如果使用的**千兆LAN网**，传输速度大概能到**100 MB/秒**。不过还需注意到真实的带宽同样受到系统性能的影响。



注意**NET** 菜单 - **NET** 菜单还提供如下几个有用的功能选项：

- 断开测量单元MU连接 **Disconnect from MU's** – 释放所有测量单元MU的连接
- 关闭测量单元MU DEWESoft上的软件 **Close DEWESoft on MU's** – 关闭t所有测量单元MU上的**DEWESoft**软件
- 重启测量单元MU **Reboot MU's** – 重启 测量单元MU计算机(当测量单元计算机死机或挂起时，该功能非常有用)
- 关闭测量单元MU **Shut down MU's** – 关闭测量单元MU (需要在测量单元MU上安装ACPI电源管理系统)
- 唤醒测量单元MU **Wakeup MU's** – 需要在测量单元MU上激活 "局域网唤醒Wake-up on LAN" 功能

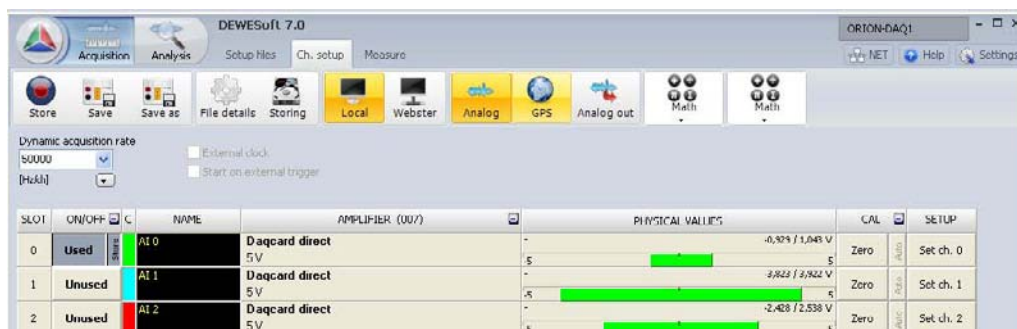
6.2 远程控制从动测量单元MU

在本节中，我们仅仅通过主控客户端**master client** 计算机来实现 远程配置以及 远程控制从动测量单元MU。Dewetron 测量单元MU 设置成从动测量单元MU模式，而且我们已经在上面一节中一步一步将它们连接起来。所有的步骤都在客户端计算机上进行！

我们从来没有触碰过**测量单元MU**。测量单元MU可能在几米以外，也有可能你在另外一座建筑内，甚至几英里以外。只要他们之间保持可靠的网络连接，我们就可以从这台**客户端client**计算机上完全控制各个测量单元！

DEWESoft NET 设置

现在点击 **设置Setup** 按钮。如果你稍微了解一点 **DEWESoft Setup** 屏幕原来的样子，你就会发现一点点但是非常重要的差别：



注意一下名叫**本地LOCAL**和**WEBSTER**的标签。如果已经连接超过一台**测量单元MU**，各个**测量单元MU**的名称t也会显示在这个菜单栏上。本地计算机就是我们的**主控客户端**，因此它本身没有任何测量通道。不过，他还是具有一个**数学Math**标签。非常有意思的是，你可以在本客户端上实时对任意**测量单元**的任意通道数据进行数 学运算！你甚至可以把不同**测量单元MU**的不同通道组合在一个数学运算当中— 只要所有**测量单元MU**是同步的！

点击**WEBSTER** 标签，这样就可以进一步设置这台**测量单元 MU**：



我们可以看到 **WEBSTER** 有四个通道被设置成实时传送数据给**客户端**。当然，这些通道同样可以设置成将数据存储在本地**测量单元MU**上，这种设置是由**hardware setup**屏幕， **NET**页面上默认设置的。

注意一下WEBSTER的三个标签：



· 通道设置 **CHANNEL SETUP**

设置，激活，标定真正需要存储的通道



· 显示设置 **DISPLAY SETUP**

设置测量单元MU本地屏幕显示



· 数据传输设置 **TRANSFER SETUP**

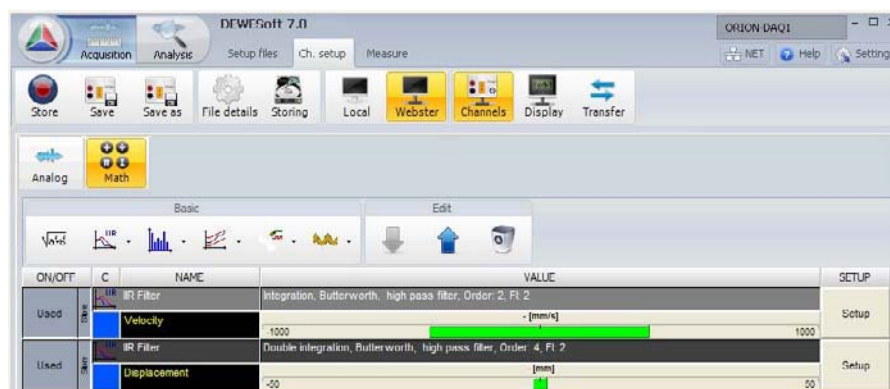
设置测量单元数据传输

6.2.1 测量单元MU通道设置

这可能是这一章节中三个步骤之中最重要的步骤，因为这一步骤需要设置，激活以及标定实际真正需要保存在WEBSTER本地计算机上的通道。因此，我们从这里开始。点击WEBSTER下面的通道设置 **Channel setup**。



在这个屏幕您可以进行的操作和把WEBSTER当做独立运行单元MU时的操作完全一致...使用 **Used / Unused** 按钮激活通道，使用 **Set ch. #** 按钮标定通道等等。你还可以设置动态采集速率 **DYNAMIC ACQUISITION RATE** 以及减少采样率 **reduced sample rates**，输入文件名，以及更多其他功能。在我们的例子中，WEBSTER i 是一台带有2个模拟输入通道的计算机。另外，还创建了两个数学通道：



因此我们可以将这四个通道的数据传送给主控客户端。

6.2.2 测量单元的显示设置

这种设置仅仅是在本地测量单元MU的显示屏上显示某些数据的情况才比较重要。如果没人需要在本地测量单元MU上观察数据的话(比如这是一台远程遥控的设备, 当前正处在无人值守的状态), 那么我们可以跳过这一步。

不过, 如果你想在本地测量单元MU的显示器上显示一些数据以便现场观察, 点击显示设置Display setup标签, 然后随意设置你所需的屏幕显示, 只要使用通常的DEWESoft 屏幕设置方法即可。



还有一点非常重要, 客户端计算机的显示分辨率必须超过测量单元显示器的显示分辨率! 比如, 你的测量单元MU显示器分辨率为1024x768, 那么客户端的分辨率要比这更大。否则, 你可能会碰到一些显示问题, 当远程控制测量单元MU的显示屏幕时, 屏幕底部的一些部分就会看不见。

上面的屏幕画面是显示在远程测量单元MU WEBSTER显示器上的屏幕显示! 而不是你在客户端屏幕上看到的屏幕显示。

6.2.3 测量单元MU传输设置

最后我们需要设置的是测量单元MU传输设置。这是什么意思呢? 这表示我们需要把“哪些通道”设定为在记录的同时通过网络传递给客户端, 并在客户端屏幕上显示。

这就是我们传递什么东西的全部意义。传递数据的同时, 不能对客户端上实际的数据存储造成任何影响 (假设已经激活本地存储功能 – 该功能是默认设置, 强烈建议使用该模式)。

因此, 我们可以有几台测量单元MU, 每个测试单元又有几十个甚至几百个通道, 但是我们只选择几个通道实时传递数据 – 甚至不传递任何通道的数据 – 给客户端。这样设置不会对测量单元MU的存储所有通道的数据产生任何影响! 理解这点非常重要。

因为任何网络都存在带宽限制, 我们建议大家谨慎选择需要实时传输数据的通道 – 时刻记住网络带宽, 只选择那些真正需要在客户端屏幕上监控、显示和控制整个测试的通道来传输。

在本例中，我们选择传输四个通道的数据。注意**Transfer setup** 传输设置屏幕上显示的是所有通道都设置成 **Yes**：



6.3 测量

在本章中，我们将讨论一下从客户端如何控制系统采集数据的操作过程：

- 在客户端上创建显示 **Creating a Display on the CLIENT**
- 在测量单元 **MU** 上存储数据 **Storing Data on the MU(s)**
- 将数据上传到客户端 **Uploading Stored Data to the Client**

在客户端上创建显示

开始存储之前，我们应该设置本地屏幕显示。前面所讲的一些步骤，我们已经在所有测量单元 **MU** 屏幕上配置好了显示的内容，但是我们当然还想在客户端电脑上观察数据！

其实我们已经很清楚该怎么做。在客户端软件屏幕工具栏中选择使用总览 **Overview**，示波器 **Scope**，记录仪 **Recorder** (等等) 按钮创建屏幕显示，可以选择所有测试单元 **MU** 的任意通道的组合！

正如前面所述，所有的测量单元 **MU** 必须有同步手段保证同步，这样可以确保下述三件事：

- 多台测量单元 **MU** 测量的数据文件真正同步
- 在客户端上显示来自不同测量单元 **MU** 的不同通道数据的能力
- 在客户端计算机上利用来自不同测量单元 **MU** 不同通道创建数学通道的能力



注意通道CHANNELS 列表中显示的通道名称都带有测量单元MU的名称，软件自动获取该名称。这样我们就可以很方便的确切了解每个通道所处的实际位置。

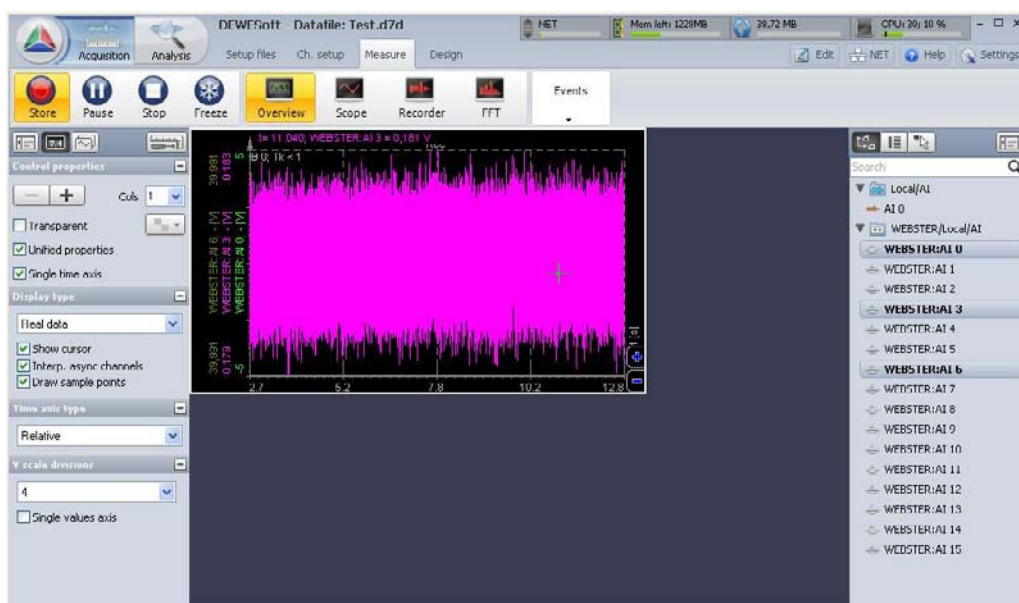
所以在我们的例子中，测量单元MU的名称，"WEBSTER" 已经加到默认测量单元MU的传输通道名前面了，正如我们所看到的一样。

每个测量单元MU的通道f都会以这种方式自动显示。这也是唯一和独立运行单元设备DEWESoft软件运行屏幕不一样的地方。

如果需要进一步有关显示设置的帮助，请参考最详细的[显示设置教程](#)。

在测量单元MU上存储数据

合理配置好客户端和测量单元MU 后，我们就可以开始存储数据了。只需按照正常模式，点击工具栏上的存储Store f按钮即可。



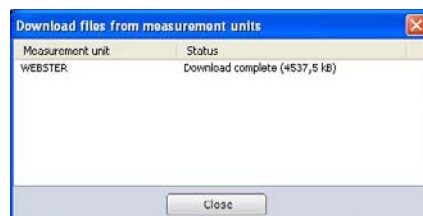
我们可以从上面显示的事件日志中看到，我们已经开始存储数据，记录过程中按下空格键增加一个事件标记，然后停止记录。

将数据上载到客户端

一旦存储过程被终止，屏幕上就会自动出现一个重要按钮，叫做**传输Transfer**：



点击该按钮，我们刚刚存储在各个**测量单元MU**上的数据文件，将自动下载到客户端电脑上。并在屏幕上出现一个**传输窗口**，显示传输过程，并最终完成整个下载过程。



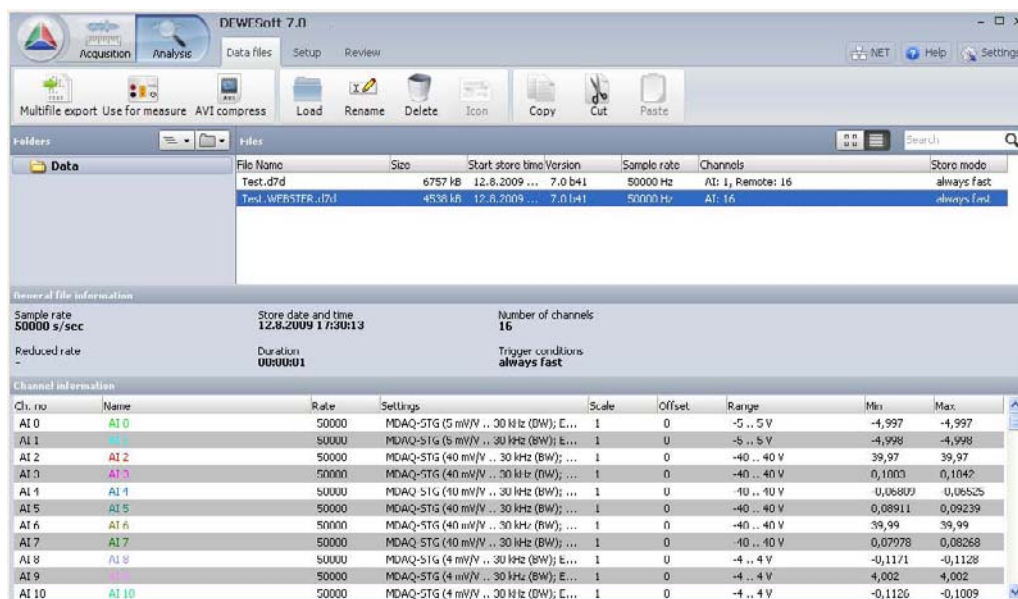
我们的例子中，只有一台**测量单元MU**，因此只有一个文件需要下载。

6.4 分析

回放采集的数据

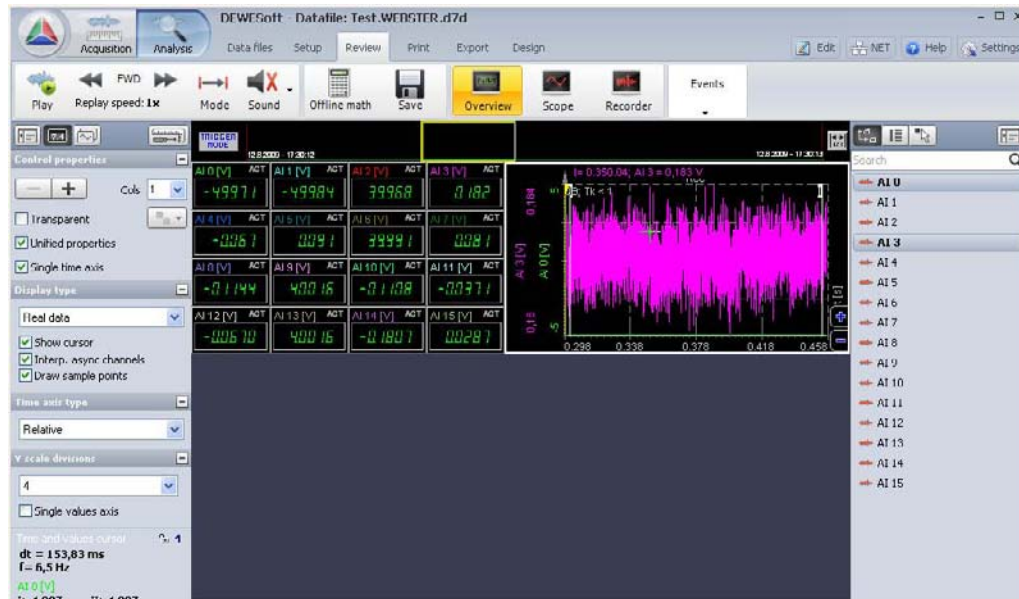
只要采集的数据文件完全下载到**客户端电脑**上，我们就可以在**客户端电脑**上回放数据。

点击**分析ANALYSE**按钮，找到已有的任何传输过来的数据文件。注意，软件同样默认在文件名中加入**测量单元MU**的名称，这样的话，我们就可知道这个文件来自于名叫**WEBSTER**的**测量单元MU**。



文件名默认设置为以**NET_test**开头，我们已经激活了**Multifile**选项，我们做了许多采集，得到了很多数据文件，最后的号码是**14**。因此这里显示的名字就是：**NET_test_0014.WEBSTER.dsd** (***.dsd** = **DEWESoft datafile**)。

双击文件，打开数据文件，然后使用通常的工具对其进行分析，浏览，打印等等。



Index

- B -

Basic tutorials 2
display settings 11
simple measurement 2
storing strategies 15

- C -

CAN bus acquisition 114
channel setup 115
measurement 117
Combustion analysis 158
analysis 168 channel
setup 158
combustion setup 160
measurement 167
Counter measurement 74
counters in automotive 96
encoder 80
event counting 74
frequency / super-counter 91
period and pulsewidth 85
sensor mode 94
two pulse edge separation 89

- D -

Display settings 11
Dynamic signal analysis tutorials 134
human vibration 147
order tracking 153
sound level 134
torsional vibration 139

- E -

Encoder 80
X1, X2, X4 modes 82
zero pulse 84
Event counting 74
gated event counting 77
simple event counting 75
up/down counting 79
Exporting recorded data 9

- F -

Frequency measurement 101
channel setup 101
encoder measurement 103
tacho probe measurement 105

- G -

GPS acquisition 119
channel setup 120
measurement 122

- H -

Human vibration 147
input channel setup 148
measurement 152
output channel setup 150

- M -

Measurement tutorials 24
CAN bus acquisition 114
counter 74
frequency measurement 101
GPS acquisition 119
strain gage 60
temperature 41
vibration 46

Measurement tutorials 24
 video 107
 voltage and current 25
 voltage/current/digital output sensors 36

Microphone calibration 136

- N -

Networked data acquisition tutorial 169
 analyse 180
 measurement 178
 remotly cotroling slave MU 174
 setup measurement units and client 170

- O -

Order tracking 153
 channel setup 153
 measurement 155

- P -

Period and pulsewidth 85
 Period and pulsewidth measurement
 duty cycle 88
 period 86
 pulsewidth 87
 Power module tutorials 124
 single phase power measurement 124

- R -

Remotly cotroling slave MU 174
 cotroling slave MU 176
 display Setup on the MU 177
 transfer Setup of a MU 177

Reporting recorded data 9

- S -

Sensor correction 131

Simple measurement 2
 analyse 7
 channel setup 3
 exporting data 9
 first measurement 5
 making a nice screen 7
 reporting data 9
 video setup 4

Single phase power measurement
 channel setup 126
 measurement 128
 sensor correction 131

Sound level 134
 channel setup 135
 measurement 138
 microphone calibration 136

Storing data strategies 15

Storing strategies
 always fast 16
 always slow 17
 fast on trigger 19
 fast on trigger, slow otherwise 22

Strain measurement 60, 63
 full bridge measurement 71
 full bridge setup 69
 load cell setup 73
 quarter bridge measurement 66
 quarter bridge setup 64

- T -

Temperature 41
 channel setup 42
 measurement 45
 PAD channel setup 43
 Torsional vibration 139
 rotational order extraction 142
 rotational vibration measurement 141
 rotational vibration setup 140
 torsional order extraction 146
 torsional vibration measurement 145

Torsional vibration 139
 torsional vibration setup 144

Triggered storing 33
 fast on trigger 19
 fast on trigger, slow otherwise 22

Triggering
 glitch triggering 30
 triggered storing 33

Tutorials 1
 basic tutorials 2
 combustion analysis 158
 dynamic signal analysis tutorials 134
 measurement tutorials 24
 networked data acquisition tutorials 169
 power module tutorials 124

- V -

Vibration
 analyse 58
 channel setup 49
 Math setup 54
 measurement 56
 video setup 56

Video acquisition 107
 channel setup 108
 measurement 110
 video post synchronization 112

Voltage and current
 channel setup 26
 glitch triggering 30
 measurement 28
 triggered storing 33

Voltage/current/digital output sensors 36
 analyse 39
 channel setup 37
 measurement 39