

Aquadopp “小阔龙” 感应猫选项

可能感兴趣的点：

- 带有感应猫选项的小阔龙在有无IMM的情况下都可以运行
- 最少的配置即是带有感应猫的小阔龙，感应猫连接到系统缆顶部末端浮标上的控制器
- 如果小阔龙开启了IMM，睡眠模式中的功耗会增大大约35微安，或者是RS232下的功耗的两倍。



- 小阔龙感应猫可以通过带有外壳的锚系统缆通信
- 小阔龙可以设置成将数据以ASCII码或者二进制格式存储在感应猫模块(IMM)
- 小阔龙可以通过Aquadopp软件进行设置和布放

小阔龙可以通过感应调制解调器模块(IMM)进行感应通信，该模块位于压力舱内小阔龙电子元件旁边。IMM连接感应铜块，铜块夹持在带有绝缘外壳的锚系统缆上，与海水一同形成回路。

在开启感应猫选项以前，必须开启IMM并设置其发射功率和ID。这必须作为设置过程的一部分在布放之前完成。使用小阔龙软件设置仪器并准备布放。在标准布放时，为了可以与感应猫一同工作，仪器必须处于自容布放模式。

注意：如果在布放设置菜单中开启诊断数据采集，诊断数据将不会通过IMM传递，而且只有当常规数据通过IMM发送出去以后才能进行采集。因此，如果小阔龙在等待IMM发送数据，将会延迟诊断数据的采集。如果IMM和SIM之间正在通信，诊断数据的采集也会延迟。

缩写：

- IMM：感应调制解调器模块(感应猫模块)
SIM：表面感应模块

只有关闭IMM选项，小阔龙才可以进行在线测量。小阔龙在自容模式下测量的数据保存在内存中，而且还可以发送至IMM电子模块。然后IMM将数据传递至水面感应模块SIM。数据传输的过程即是IMM和SIM之间感应通信的一部分。

可能感兴趣的点：

- 小阔龙可以通过末端的常规通信插口和IMM铜接口进行设置和布放。
- 小阔龙内部电池可以直接为IMM供电，这就确保小阔龙处于睡眠状态下，IMM的通信正常。
- 开启IMM选项后，小阔龙的通信波特率必须设置为9600。固件升级和内存数据下载的波特率可以设置更高。
- SIM在与小阔龙IMM通信时会经常在末尾出现<BUSY/>

数据传输

在布放之前，软件通常会估算IM和通过感应连接传递数据需要的额外电能。估算是在假定表面模块仅与一个小阔龙通信的前提下进行的。功耗估算考虑了SIM不使用最大功率效率方法读取IMM的数据的情况，从而加大了对功耗的估算值。

在运行过程中，小阔龙数据传输到IMM中的具有40组采样能力的循环缓冲区。SIM可以选择读取数据的频率（例如感应连接传输以后），例如一个小时内每10分钟读取一次平均平均流速。SIM负责从缓冲区读取数据并在循环缓冲区存满以前删除数据。

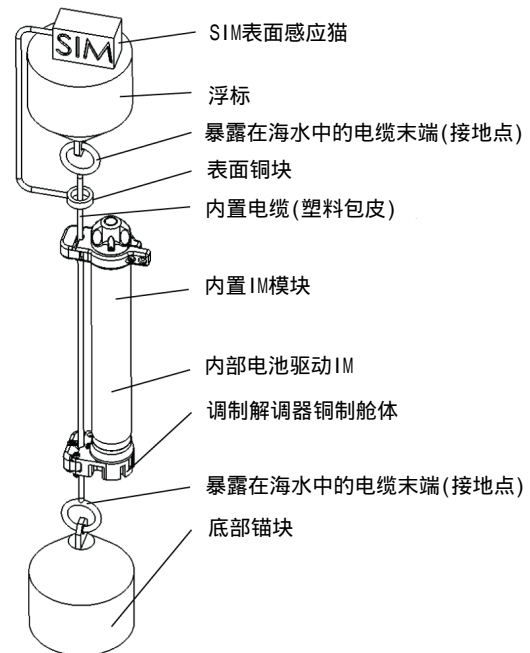
如果循环缓冲区存满，IMM会自动在加入新一组数据之前删除最先存储的一组数据。IMM中循环缓冲区写入运算的过程中，小阔龙的硬件每隔一秒会询问一次IMM，直到获得允许将

数据存储于缓冲区。SIM必须准备重试所有发送给IMM的命令，以应对IMM没有空闲时间处理小阔龙请求的情况。

如果沿着通信电缆有多台小阔龙，我们有信心错开SIM中的感应猫通信。

开启IMM后，测量间隔最小为60s，且必须比平均间隔和诊断采样时间之和要大。因为小阔龙可能会在放弃连接之前等待45s，以便IMM获得许可。

在控制表面感应猫SIM的软件设计上，小阔龙硬件获得授权许可可是非常重要，以便可以在设定的间隔内将数据传输到IMM。



小阔龙IM原理

可能感兴趣的点：

- 在布放过程中，软件会测试与IMM铜座的内部连接和铜座的阻抗值。阻抗值存储在配置日志文件内。
- Nortek系统集成手册内有关于IMM内存存储的数据格式，手册可以从www.nortek-as.com进行下载。
- 因为ASCII码数据格式与dat文件格式相同，当小阔龙文件转化为ASCII格式数据并输出时，也会输出hdr文件且其中有关于ASCII码数据格式的介绍。

感应猫的集成

下面布放设置选项在小阔龙软件中会出现：

- 开启IMM
- 设置IMM中仪器的ID
- 设置发射功率水平
- 选择ASCII码后者二进制格式

在开始布放仪器上述参数都不能进行设置。

小阔龙软件会将布放过程中IMM的设置存储在配置日志文件中。小阔龙全部的设置存储在IMM主文件中，这使得表面感应猫(SIM/IMM)可以通过命令HostFileGetData接受小阔龙的设置。

这些数据只能以二进制格式存储，因此如果SIM使用的命令必须设置成二进制。与此情况相同，为了在IMM中存储数据，小阔龙也选择二进制格式。

为了从小阔龙回收数据，下面列举了SIM中最相关的命令：

- !iiSampleGetList
- !iiSampleGetData:
- !iiSampleGetLast
- !iiSampleEraseAll
- !iiHostFileGetData

下一页中列举的example.log 是SIM传输的部分二进制数据。

与之相对应的data01.log文件中列举了ASCII码数据。阔龙文件data01.dat显示了从小阔龙内存中转化而来的数据。



Vangkroken 2
NO-1351 Rud
Norway

example.log

```
IMM>captureline  
<Executing/>  
!15HostFileGetData  
<RemoteReply><Executing/>  
<HostData Len='784' CRC='0x43FFA2E2'>
```

Binary data returned:

```
    a5 05 18 20 41 51 44 20 35 35 30 35 20  
20 20 20 20 20 0c 20 7a 22 20 20 03 20 90 20 2c  
20 20 20 41 fa 20 20 7a 22 41 fa 20 20 39 69 33  
33 40 ce a5 04 70 20 0f 20 d0 07 20 20 41 51 44  
20 30 36 39 38 20 20 20 20 20 20 20 20 20 20  
20 50 0b 50 0b 20 20 b0 f4 50 0b 20 20 b0 f4 b0  
f4 a0 16 01 80 a8 20 c0 34 b3 fe 01 80 77 06 39  
34 f7 fe 20 20 20 20 20 20 ff ff 20 20 01 20 20  
20 01 20 20 20 20 20 20 20 20 20 ff ff 20 20 ff  
ff 20 20 ff ff 20 20 20 20 20 20 ff ff 01 20 20  
20 20 20 ff ff ff ff 20 20 20 20 01 20 20 10 20  
20 c9 05 03 01 d8 1b 6d 2a 01 80 13 01 68 2e 38  
20 01 80 c9 02 69 2e 78 ff ff 7f d9 01 53 01 d9  
01 af 7e e0 f7 7a fc c9 fb ae 73 67 20 e0 ff 88  
ff 20 20 20 20 20 20 20 20 20 20 20 20 20 20  
20 55 15 10 0e 10 0e 10 27 64 20 20 20 20 20 20  
20 20 20 20 20 20 20 20 20 20 20 20 20 20 03  
20 7e a4 a5 20 20 01 7d 20 31 20 20 20 b5 01 20  
02 01 20 3c 20 03 20 40 20 20 20 20 20 20 20 20  
20 02 20 20 20 01 20 20 20 2c 01 69 6d 6d 74 73  
74 20 20 30 20 07 12 09 04 10 0e 20 20 02 20 11  
41 14 20 01 20 14 20 04 06 21 20 2e 33 5e 01 02  
3d 1e 3d 39 3d 53 3d 6e 3d 88 3d a2 3d bb 3d d4  
3d ed 3d 06 3e 1e 3e 36 3e 4e 3e 65 3e 7d 3e 93  
3e aa 3e c0 3e d6 3e ec 3e 02 3f 17 3f 2c 3f 41  
3f 55 3f 69 3f 7d 3f 91 3f a4 3f b8 3f ca 3f dd  
3f f0 3f 02 40 14 40 26 40 37 40 49 40 5a 40 6b  
40 7c 40 8c 40 9c 40 ac 40 bc 40 cc 40 db 40 ea  
40 f9 40 08 41 17 41 25 41 33 41 42 41 4f 41 5d  
41 6a 41 78 41 85 41 92 41 9e 41 ab 41 b7 41 c3  
41 cf 41 db 41 e7 41 f2 41 fd 41 08 42 13 42 1e  
42 28 42 33 42 3d 42 47 42 51 42 5b 42 64 42 6e  
42 77 42 80 42 89 42 91 42 9a 42 a2 42 aa 42 b2  
42 ba 42 20 20 20 20 20 20 20 20 20 20 20 20 20  
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20  
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20  
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20  
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20  
20 20 20 20 20 20 20 20 20 20 20 20 20 20 2a  
20 0f 20 20 20 20 20 20 20 20 20 20 20 20 20 20  
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20  
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20  
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20  
20 06 20 03 20 20 20 20 20 20 20 20 20 20 20  
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20  
20 0a ff cd ff 8b 20 e5 20 ee 20 0b 20 84 ff 3d  
ff 4b 5a
```

```
</HostData>  
<Executed/>  
</RemoteReply>
```

example.log (cont.)

```
<Executed/>  
IMM>releaseline  
<Executing/>  
<Executed/>  
IMM>
```

```

<TIMEOUT msg='HostService 2 min timeout'/>
IMM>

IMM>
<Executed/>
IMM>captureline
<Executing/>
<Executed/>
IMM>!15samplegetsummary
<RemoteReply><Executing/>
<SampleDataSummary NumSamples='8' TotalLen='336' FreeMem='16006'/>
<Executed/>
</RemoteReply>
<Executed/>
IMM>!15samplegetlist
<RemoteReply><Executing/>
<SampleList>
<Sample ID='0x0000024c' Len='42' CRC='0xC7589EDD' />
<Sample ID='0x0000024b' Len='42' CRC='0x4C593F18' />
<Sample ID='0x0000024a' Len='42' CRC='0x457FCB04' />
<Sample ID='0x00000249' Len='42' CRC='0xD6C60EE2' />
<Sample ID='0x00000248' Len='42' CRC='0x8AFE5C1A' />
<Sample ID='0x00000247' Len='42' CRC='0x2301B4DB' />
<Sample ID='0x00000246' Len='42' CRC='0xB3B4B2D4' />
<Sample ID='0x00000245' Len='42' CRC='0x0BCBBDE2' />
</SampleList>
<Executed/>
</RemoteReply>
<Executed/>
IMM>!15SAMPLEGETDATA:245
<RemoteReply><Executing/>
<SampleData ID='0x245' LEN='42' CRC='0xbcbbbe2'>
Binary data returned:
      a5 01 15 20 35 20 07 12 09 04 20
20 ff ff 7d 20 20 20 33 0d ed ff e1 ff 20 b1 b3
37 48 fd cc ff 8e 04 23 03 0e 0e 0e 20 9c d6
<Executed/>

```

example.log (cont.)

```

</RemoteReply>
<Executed/>
IMM>!15SAMPLEGETDATA:246
<RemoteReply><Executing/>
<SampleData ID='0x246' LEN='42' CRC='0xb3b4b2d4'>
Binary data returned:
      a5 01
15 20 40 20 07 12 09 04 20 20 ff ff 7d 20 20 20
3c 0d ed ff e1 ff 20 b1 92 38 48 fd 16 fd 02 05
3f 03 0e 0e 0e 20 69 d5
<Executed/>
</RemoteReply>

```

```
<Executed/>
IMM>!15SAMPLEGETDATA:247
<RemoteReply><Executing/>
<SampleData ID='0x247' LEN='42' CRC='0x2301b4db'>
Binary data returned:
      a5 01 15 20 45 20 07 12 09 04
20 20 ff ff 7d 20 20 20 33 0d ed ff e1 ff 20 b1
99 38 48 fd e0 ff 74 06 bb 06 0e 0e 0e 20 24 dd
```

```
<Executed/>
</RemoteReply>
```

```
<Executed/>
IMM>!15SAMPLEGETDATA:248
<RemoteReply><Executing/>
<SampleData ID='0x248' LEN='42' CRC='0x8afe5c1a'>
Binary data returned:
```

```
      a5 01
15 20 50 20 07 12 09 04 20 20 ff ff 7d 20 20 20
37 0d ed ff e1 ff 20 b1 4b 38 48 fd 3a fd 7b 05
d8 04 0e 0e 0e 20 63 d7
```

```
<Executed/>
```

```
<Executed/>
IMM>!15SAMPLEGETDATA:249
<RemoteReply><Executing/>
```

```
<SampleData ID='0x249' LEN='42' CRC='0xd6c60ee2'>
Binary data returned:
```

```
      a5 01 15 20 55 20 07 12 09 04
20 20 ff ff 7d 20 20 20 33 0d ed ff e1 ff 20 b1
7f 39 48 fd 9c fe 18 05 76 09 0e 0e 0e 20 35 de
```

```
<Executed/>
```

```
</RemoteReply>
```

```
<Executed/>
IMM>!15SAMPLEGETDATA:24A
```

example.log (cont.)

```
<RemoteReply><Executing/>
<SampleData ID='0x24a' LEN='42' CRC='0x457fcb04'>
```

```
Binary data returned:
      a5 01 15 20 20 20 07 13 09 04 20 20 ff ff 7d 20
20 20 33 0d ed ff e1 ff 20 b1 ab 38 48 fd 29 01
22 03 ca 04 0e 0e 0e 20 f7 d9
```

```
<Executed/>
```

```
</RemoteReply>
```

```
<Executed/>
IMM>!15SAMPLEGETDATA:24B
<RemoteReply><Executing/>
```

```
<SampleData ID='0x24b' LEN='42' CRC='0x4c593f18'>
Binary data returned:
```

```
      a5 01 15 20 05 20 07 13
09 04 20 20 ff ff 7d 20 20 20 33 0d ed ff e1 ff
20 b1 81 38 48 fd 79 ff 33 07 bd 04 0e 0e 0e 20
26 dc
```

```
<Executed/>
```

```
</RemoteReply>
```

```
<Executed/>
IMM>!15SAMPLEGETDATA:24C
<RemoteReply><Executing/>
<SampleData ID='0x24c' LEN='42' CRC='0xc7589edd'>
Binary data returned:
  a5 01 15 20 10 20 07 13 09 04 20 20 ff ff 7d 20
  20 20 3c 0d ed ff e1 ff 20 b1 14 38 48 fd 7e fd
  0d 01 b3 05 0e 0e 0e 20 a2 d4
<Executed/>
</RemoteReply>
<Executed/>
IMM>!15SAMPLEERASEALL
<RemoteReply><Executing/>
<Executed/>
</RemoteReply>
<Executed/>
IMM>releaseline
<Executing/>
<Executed/>
IMM>
```

data01.log

```
IMM>
<Executed/>
IMM>CaptureLine
<Executing/>
<Executed/>
IMM>!12SampleGetSummary
<RemoteReply><Executing/>
<SampleDataSummary NumSamples='5' TotalLen='550' FreeMem='15792' />
<Executed/>
</RemoteReply>
<Executed/>
IMM>!12SampleGetList
<RemoteReply><Executing/>
<SampleList>
<Sample ID='0x00000254' Len='109' CRC='0x45326FC3' />
<Sample ID='0x00000253' Len='110' CRC='0x74E6ED95' />
<Sample ID='0x00000252' Len='109' CRC='0x0D7F6C29' />
<Sample ID='0x00000251' Len='111' CRC='0x9175F508' />
<Sample ID='0x00000250' Len='111' CRC='0x42E806B9' />
</SampleList>
<Executed/>
</RemoteReply>
<Executed/>
IMM>!12SAMPLEGETDATA:250
<RemoteReply><Executing/>
<SampleData ID='0x250' LEN='111' CRC='0x42e806b9'> 4 7 2009 13 35 0 0 177 -0.708 0.799
0.154 14 14 14 12.5 0.0 338.3 -1.9 -3.1 14.403 -6.96 65535 0 1.067 318.5
</SampleData>
<Executed/>
</RemoteReply>
<Executed/>
IMM>!12SAMPLEGETDATA:251
<RemoteReply><Executing/>
<SampleData ID='0x251' LEN='111' CRC='0x9175f508'> 4 7 2009 13 45 0 0 177 -0.744 1.434
0.975 14 14 14 12.5 0.0 338.3 -1.9 -3.1 14.412 -6.96 65535 0 1.615 332.6
</SampleData>
<Executed/>
</RemoteReply>
<Executed/>
IMM>!12SAMPLEGETDATA:252
<RemoteReply><Executing/>
<SampleData ID='0x252' LEN='109' CRC='0xd7f6c29'> 4 7 2009 13 55 0 0 177 0.292 0.669
0.810 14 14 14 12.5 0.0 338.3 -1.9 -3.1 14.371 -6.96 65535 0 0.730 23.6
</SampleData>
<Executed/>
</RemoteReply>
```

```

<Executed/>
IMM>!12SAMPLEGETDATA:253
<RemoteReply><Executing/>
<SampleData ID='0x253' LEN='110' CRC='0x74e6ed95'> 4 7 2009 14 5 0 0 177 -0.732 0.830
0.805 14 14 14 12.5 0.0 338.3 -1.9 -3.1 14.603 -6.96 65535 0 1.107 318.6
</SampleData>
<Executed/>
</RemoteReply>
<Executed/>
IMM>!12SAMPLEGETDATA:254
<RemoteReply><Executing/>
<SampleData ID='0x254' LEN='109' CRC='0x45326fc3'> 4 7 2009 14 15 0 0 177 0.343 0.964
0.796 14 14 14 12.5 0.0 337.9 -1.9 -3.1 14.493 -6.96 65535 0 1.023 19.6
</SampleData>
<Executed/>
</RemoteReply>
<Executed/>
IMM>!12SampleEraseAll
<RemoteReply><Executing/>
<Executed/>
</RemoteReply>
<Executed/>
IMM>releaseline
<Executing/>
<Executed/>

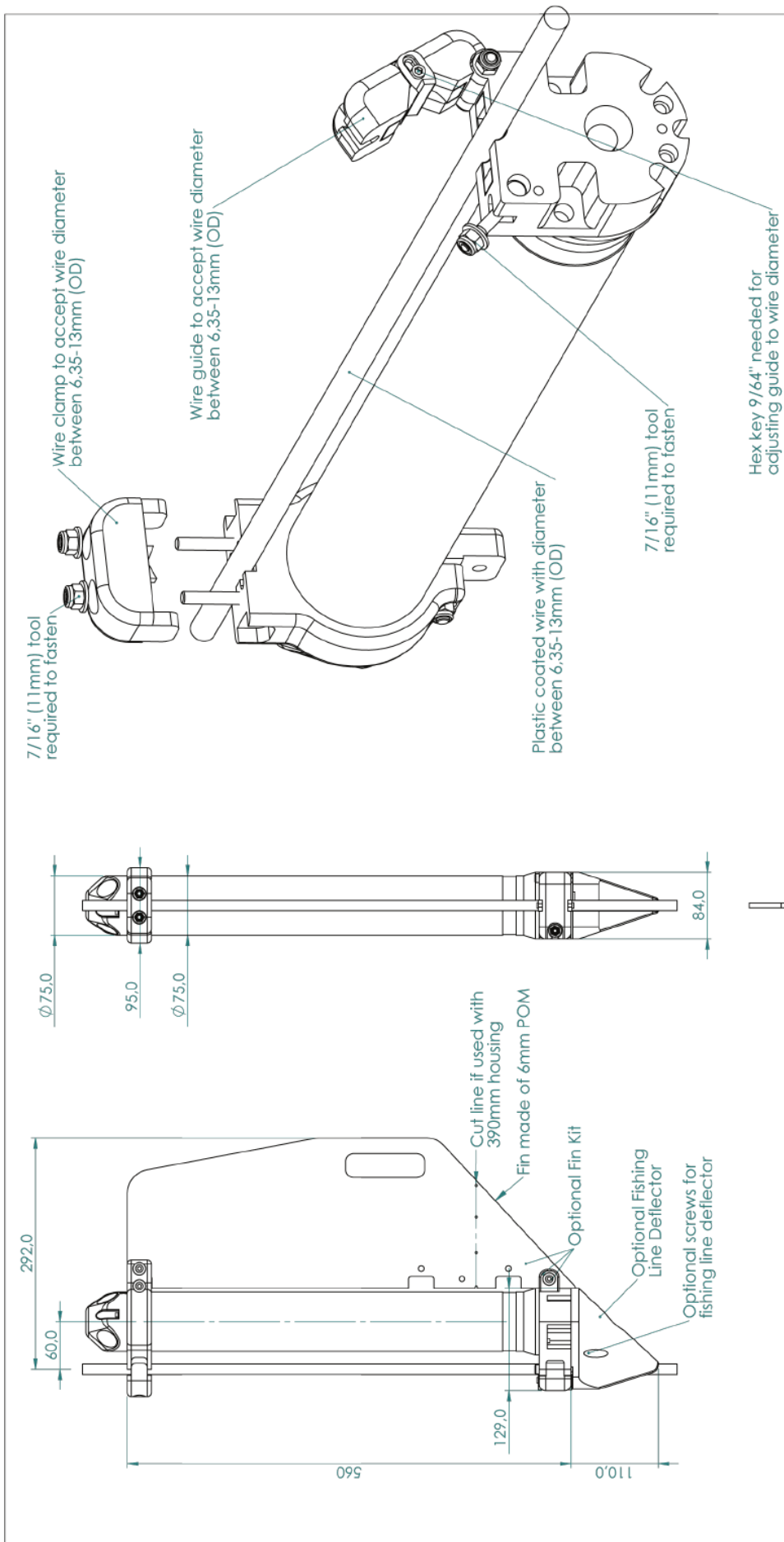
```

data01.dat

```

04 07 2009 13 35 00 00000000 10110001 -0.708 0.799 0.154 14 14 14 12.5
1414.6 338.3 -1.9 -3.1 14.403 -6.96 65535 0 1.068 318.46
04 07 2009 13 45 00 00000000 10110001 -0.744 1.434 0.975 14 14 14 12.5
1414.6 338.3 -1.9 -3.1 14.412 -6.96 65535 0 1.616 332.58
04 07 2009 13 55 00 00000000 10110001 0.292 0.669 0.810 14 14 14 12.5
1414.6 338.3 -1.9 -3.1 14.371 -6.96 65535 0 0.730 23.58
04 07 2009 14 05 00 00000000 10110001 -0.732 0.830 0.805 14 14 14 12.5
1414.6 338.3 -1.9 -3.1 14.603 -6.96 65535 0 1.107 318.59
04 07 2009 14 15 00 00000000 10110001 0.343 0.964 0.796 14 14 14 12.5
1414.6 337.9 -1.9 -3.1 14.493 -6.96 65535 0 1.023 19.59

```



7/16" (11mm) tool required to fasten

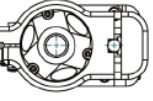
Wire clamp to accept wire diameter between 6.35-13mm (OD)

Wire guide to accept wire diameter between 6.35-13mm (OD)

7/16" (11mm) tool required to fasten

Hex key 9/64" needed for adjusting guide to wire diameter

REV.	C/O No	DESCRIPTION	DATE	APPROVED
		NORTEK AS Aquadopp IM (STD)		Vangkroken 2 1351 Rud, Norway
<small>Unless otherwise specified, dimensions are in mm Tolerances are: Angles: ± 0.5 degrees Decimals: x = ± 0.05mm 0.6 = ± 0.2mm 0.06 = ± 0.02mm Nortek Standard N1000-008 SS 2785-X Material</small>				
	A-3	Size	Rev.	-
		Drawing No	Sheet	1 of 1
	BL	Date:	Date:	08.06.2009
		Controlled by:	Scale:	1:5



Optional parts

Nortek Part	Description	Quantity needed
N2011-509	Fin kit AQD IM complete w/screws	1
N2011-494	Fishing line deflector (IM head cone)-without screws	1
	Screws for fishing line deflector (Soc.cop 1/4-20x1.25" if fan)	3