RSI Technical Note 037

MicroXM Configuration Files. Creation of .XMP Files Explained

Brett Prairie
William Douglas

2015-05-04

Rockland Scientific International Inc.
520 Dupplin Rd
Victoria, BC, CANADA, V8Z 1C1
www.rocklandscientific.com

1 Introduction

This document specifies the format of microXM configuration files as expected by ODASXMP. Each microXM instrument requires a microXM configuration file generated according to this document.

NOTE: This document is for reference only. The user is not required to generate a microXM configuration file. MicroXM configuration files must be obtained by JFE Advantech Co. upon the purchase of a microXM instrument(s). **Modifying a microXM configuration file increases the risk of the VMP-X recording incorrect data and could result in the loss of your instrumentation at sea.**

2 Specification

Shown in listing 1 is an example microXM configuration file for instrument serial number 8. All unique calibration coefficients required by the instrument are included.

Listing 1: Example MicroXM Configuration File

```
1  [xmp_00008]
2  SN_sn=8
3  T_coef0=290.15
4  T_coef1=3896.4
5  sh1_sens=0.07
6  sh2_sens=0.07
7  P_coef0=-0.5
8  P_coef1=0.349
9  Pitch_coef0=12748
10 Pitch_coef1=-12676
```

1 : Serial number of the microXM. Must be 9 characters long and enclosed within square brackets. Integer value should be prefixed with “0” characters until it is 5 digits in length.

2 : Serial number of the microXM. Used by ODASXMP and the ODAS Matlab Library to verify that correct calibration coefficients were used to generate the file.

3,4 : Thermistor calibration coefficients.

5,6 : Shear sensitivity coefficients.

7,9 : Coefficients from the Keller pressure transducer calibration sheet.

9,10 : Pitch calibration coefficients.

The smallest microXM configuration file would contain calibration values for a single instrument by including the 10 lines from example 1. Multiple instruments are supported by combining calibration values from multiple instruments into a single file.

Note that the resulting microXM configuration file must be saved in ASCII format and have the extension "`.xmp`".

3 Advanced Options

Many parameters required within a microXM configuration file are redundant. Because these parameters are still needed ODASXMP assigns nominal values when not provided within the microXM configuration file. This allows the microXM configuration file to be made much smaller. A complete list of nominal parameters is provided within Appendix A. These parameters can be specified within the microXM configuration file if the nominal values are not correct. ODASXMP will use parameters from the microXM configuration file in place of embedded nominal values.

Listing 2 demonstrates how the nominal differentiator gain in the temperature channel can be replaced with a custom value. In this example, the addition of line #5 results in a differentiator gain of 1.5 being applied when converting observed data into physical units. It is unlikely that custom calibration values would be required unless there are changes made to the microXM design.

Listing 2: Customized MicroXM Configuration File

```
1  [xmp_00008]
2  SN_sn=8
3  T_coef0=290.15
4  T_coef1=3896.4
5  T_diff_gain= 1.5
6  sh1_sens=0.07
7  sh2_sens=0.07
8  P_coef0=-0.5
9  P_coef1=0.349
10 Pitch_coef0=12748
11 Pitch_coef1=-12676
```

4 Resulting ODAS Configuration String

Data files generated by ODASXMP do not contain microXM configuration strings within their headers. This change is new with ODASXMP and changes how the resulting microXM data files are processed. ODASXMP converts the microXM configuration string into a standard ODAS configuration string which is embedded within the resulting data files. The existing ODAS Matlab library functions can then be used to process the file.

The ODAS configuration string generated from the nominal values in appendix A is shown in appendix B. This information is not required to generate microXM configuration files but included to provide a complete description of how microXM data files are generated.

A Nominal MicroXM Calibration Coefficients

```
prefix=xmp
disk=
profile=vertical
rate=1250
recsize=1
no-fast=6
no-slow=2
man_com_rate=3
xmp=1
[matrix]
num_rows=4
row01=255 254 1 2 8 9 5 15
row02=32   13 1 2 8 9 5 15
row03=0     4 1 2 8 9 5 15
row04=10   11 1 2 8 9 5 15
[xmp_00000]
SN_id=254
SN_name=SN
SN_type=xmp_sn
SN_units=
SN_sn=0
gnd_id=0
gnd_name=Gnd
gnd_type=poly
gnd_units=[V]
gnd_coef0=0.0
gnd_coef1=1.0
Ax_id=1
Ax_name=Ax
Ax_type=accel
Ax_units=[]
Ax_coef0=0
Ax_coef1=1
Ay_id=2
Ay_name=Ay
Ay_type=accel
Ay_units=[]
Ay_coef0=0
Ay_coef1=1
T_id=4
T_name=T1
T_type=xmp_therm
T_units=[degC]
T_coef0=290.15
T_coef1=3896.4
T_dT_id=5
T_dT_name=T1_dT1
T_dT_type=xmp_therm
T_dT_units=[degC]
T_dT_diff_gain=1
T_dT_display=false
sh1_id=8
sh1_name=sh1
sh1_type=xmp_shear
sh1_units=[1/s]
sh1_coef0=0
sh1_coef1=1
sh1_diff_gain=1.0
sh1_sens=0.07
sh2_id=9
sh2_name=sh2
sh2_type=xmp_shear
sh2_units=[1/s]
sh2_coef0=0
sh2_coef1=1
sh2_diff_gain=1.0
sh2_sens=0.07
P_id=10
P_name=P
P_type=xmp_pres
P_units=[dBar]
P_coef0=-0.5
P_coef1=0.349
P_dP_id=11
P_dP_name=P_dP
P_dP_type=xmp_pres
P_dP_units=[dBar]
P_dP_diff_gain=20
P_dP_display=false
V_Bat_id=32
V_Bat_name=V_Bat
V_Bat_type=xmp_volt
V_Bat_units=[V]
V_Bat_coef0=4.096
V_Bat_coef1=1.25e-4
Pitch_id=13
Pitch_name=Pitch
Pitch_type=xmp_pitch
Pitch_units=[deg]
Pitch_coef0=12748
```

```
Pitch_coef1=-12676
spare_id=15
spare_name=spare
spare_type=poly
spare_units=[counts]
spare_coef0=0
spare_coef1=1
spare_display=false
```

B Resulting ODAS Configuration String

```
disk=
man_com_rate=3
no-fast=6
no-slow=2
prefix=xmp
profile=vertical
rate=1250
recsize=1
xmp=1

[matrix]
num_rows=4
row01=255 254 1 2 8 9 5 15
row02=32   13 1 2 8 9 5 15
row03=0     4 1 2 8 9 5 15
row04=10   11 1 2 8 9 5 15

[channel]
coef0 = 0
coef1 = 1
id = 1
name = Ax
type = accel
units = []

[channel]
coef0 = 0
coef1 = 1
id = 2
name = Ay
type = accel
units = []

[channel]
coef0 = -0.5
coef1 = 0.349
id = 10
name = P
type = xmp_pres
units = [dBar]

[channel]
diff_gain = 20
display = false
id = 11
name = P_dP
type = xmp_pres
units = [dBar]

[channel]
coef0 = 12748
coef1 = -12676
id = 13
name = Pitch
type = xmp_pitch
units = [deg]

[channel]
id = 254
name = SN
sn = 8
type = xmp_sn
units =

[channel]
coef0 = 290.15
coef1 = 3896.4
id = 4
name = T1
type = xmp_therm
units = [degC]

[channel]
diff_gain = 1
display = false
id = 5
name = T1_dT1
type = xmp_therm
units = [degC]

[channel]
coef0 = 4.096
coef1 = 1.25e-4
id = 32
name = V_Bat
type = xmp_volt
units = [V]

[channel]
coef0 = 0.0
```

```
coef1 = 1.0
id = 0
name = Gnd
type = poly
units = [V]

[channel]
coef0 = 0
coef1 = 1
diff_gain = 1.0
id = 8
name = sh1
sens = 0.07
type = xmp_shear
units = [1/s]

[channel]
coef0 = 0
coef1 = 1
diff_gain = 1.0
id = 9
name = sh2
sens = 0.07
type = xmp_shear
units = [1/s]

[channel]
coef0 = 0
coef1 = 1
display = false
id = 15
name = spare
type = poly
units = [counts]
```